TESIS DOCTORAL EN INGENIERÍA ELECTRÓNICA
POR
GONZALO MATEO GARCÍA

ESCOLA TÈCNICA SUPERIOR D'ENGINYERIA

TRANSFER LEARNING OF DEEP LEARNING MODELS FOR CLOUD MASKING IN OPTICAL SATELLITE IMAGES

Director:
LUIS GÓMEZ-CHOVA

ENERO 2022

Vniversitat de València

Vniversitat de València
ETSE-UV
Escola Tècnica Superior d'Enginyeria

Prof. Dr. Luis Gómez Chova, Doctor en Ingeniería Electrónica, Catedrático de Universidad del Departamento de Ingeniería Electrónica de la Escuela Técnica Superior de Ingeniería de la Universidad de Valencia.

Hace constar que:

Gonzalo Mateo García, Licenciado en Matemáticas, Ingeniero en Informática y MSc en Tratamiento Estadístico Computacional de la Información, ha realizado bajo su dirección el trabajo titulado *Transfer Learning of Deep Learning Models for Cloud Masking in Optical Satellite Images*, que se presenta para optar al grado de Doctor por la Universidad de Valencia.

Y para que así conste a efectos oportunos, y dando el visto bueno para la presentación de este trabajo ante el Tribunal de Tesis que corresponda, firma el presente certificado en Valencia en Enero de 2022.

Luis Gómez Chova

TESIS DOCTORAL:

Transfer Learning of Deep Learning Models for Cloud Masking in Optical Satellite Images

AUTOR:

Gonzalo Mateo García

DIRECTOR:

Dr. Luis Gómez Chova

El tribunal nombrado para juzgar la Tesis Doctoral citada anteriormente, compuesto por:

Presidente: ____________________

Vocal: ____________________

Secretario: ____________________

acuerda otorgarle la calificación de ____________________

Y para que así conste a efectos oportunos, firmamos el presente certificado.

En Valencia, el de de 2022

NOTE TO THE READER

According to the University of Valencia Doctorate Regulation[1] this PhD thesis is presented as a compendium of at least three publications in international journals containing the results of the conducted work. This Thesis describes four published methods, their results and the context within they were developed. Those publications are included as an annex of this Thesis (Appendix). Furthermore, in accordance with the aforementioned regulation, the Thesis also includes an extended abstract in Spanish (Chapter 7).

[1] Reglamento sobre el depósito, evaluación y defensa de la tesis doctoral aprovado por el Consell de Govern de 28 de Juny de 2016. ACGUV 172/2016.

Agradecimientos

Aunque nos empeñemos en valorar una Tesis desde el punto de vista individual, la realidad de la investigación en el siglo XXI nos muestra que ésta es un trabajo colectivo en el que muchas personas están involucradas. La aquí presente lo es, estoy seguro que la mayoría de las buenas ideas que aquí se incluyen no son mías y para su ejecución me he apoyado en mucha gente que me ha ayudado y dado consejo. Quiero remarcar por tanto que hago uso de la primera persona del plural en la mayor parte de la Tesis de manera premeditada.

Con la presentación de esta Tesis concluye una etapa (quizá más larga de lo esperada) que comencé hace cinco años mudándome de Madrid a Valencia. Fue una decisión difícil y cargada de incertidumbre pero mirando atrás fue una decisión muy acertada. Laboralmente, el doctorado me ha dado muchos aprendizajes, capacidad de explorar nuevos temas con gran libertad y un montón de gente distinta con la que he tenido la suerte de trabajar. Personalmente, he tenido la suerte de caer en un grupo de gente buena, cercana, abierta y sencilla donde he hecho muchos amigos y que tiene una cultura donde se prima la colaboración y no hay competición interna.

Desde mi llegada al IPL me he sentido plenamente integrado, gracias a todas las personas del IPL por haberme hecho sentir así. En particular, es un placer trabajar con la gente del grupo ISP del que he formado parte estos años. Gracias a Gustau, por fomentar esa cultura de grupo de la que es fácil sentirse parte. Del *faculty* del ISP gracias primero a Jordi por mantener funcionando todo el chiringuito computacional; cada vez que me creo que sé algo de ordenadores hablo contigo y me doy cuenta de lo mucho que tengo todavía que aprender. Gracias a Ana y Julia con las que he compartido despacho, a Jesús por sus cursos de visión, a Adrián, Jose Adsuara y Álvaro. De mis primeros años en el ISP aprendí mucho de Luca Martino; gracias por las *stupid things* a las 5 en punto y por las infinitas discusiones sobre los GPs y kernels (a ver si algún día hacemos el paper definitivo explicando todo "bien" sin mentar a Hilbert). Mención especial también para Valero, gracias a él al menos dos papers de la Tesis se pueden leer, gracias por organizar los *explorers groups*, por esas cervezas en tú local que nunca hemos pagado y por la infinidad de consejos y discusiones de la vida durante estos años.

Muchas gracias al grupo de doctorandos (y algún post-doc infiltrado) del IPL, gracias por esos momentos de desconexión a la hora del café, por las cenas y por los viajes; espero que sigamos yendo a Calpe muchos años en verano. En particular gracias a Laura, a Jordi, a Nieves, a Diego, a Roberto y a Dan (que también han sido parte del *team clouds*) y a Shari, Irene, Eatidal y Miguel Ángel. Dejo para el final a las personas más importantes que me llevo de esta etapa: a Anna la organizadora y mente pensante detrás de todas esas quedadas (aunque se vaya de los grupos) y a Emiliano, Daniel y Emmanuel con los que he compartido cenas, cervezas, desayunos en la playa, comidas durante la pandemia, *summer schools* y IGARSS todo aderezado de discusiones de la vida y de *machine learning*. Espero seguir teniendoos en mi vida aunque la distancia lo haga más complicado.

Que esta Tesis haya llegado a buen puerto es fundamentalmente gracias a mi director, Luis. He tenido la suerte de tener un director que me ha dado la confianza y libertad para llevar mi trabajo por donde más me ha llamado la atención, me ha buscado financiación para ir a las conferencias, *summer schools* y reuniones de grupo que me resultaban interesantes, a peleado para que me dieran docencia (además la que me gustase ¡y que fuese en castellano!) y me ha permitido hacer las estancias que he querido aunque no siempre a él beneficiara. Me llevo muchos aprendizajes de él y espero poder seguir trabajando juntos en el futuro.

La parte final de esta Tesis ha estado muy influenciada por el Frontier Development Lab. Me siento muy afortunado de haber podido participar varios años. Del sprint de 2019 agradezco especialmente a Josh, Lewis, Dietmar y Gunes por el trabajo juntos. Del de 2020, aunque virtual, me dio la oportunidad de conocer a Dolo y a Freddie de los que he podido aprender mucho y con los que sigo trabajando y teniendo relación. Me gustaría también agradecer al equipo de Trillium por acogerme, confiar en mí y darme nuevas responsabilidades; en particular a James por la confianza y a Jodie por estar siempre pendiente de que todo salga bien.

Haber llegado hasta este punto no hubiera sido posible sin el apoyo y soporte de mi familia. Quitando los primeros 24 años de mi vida en los que me dieron absolutanmente todo, también me apoyaron cuando decidí venir a Valencia dejando un trabajo mejor pagado; me hace feliz que se sientan orgullosos aunque no estén muy puestos en *machine learning* ni en teledetección (tanto mis padres como mi hermana son abogados).

Finalmente, gracias a Mercedes, ella ha sido la causante principal de que esté aquí en este momento. La verdad es que tenías ventaja porque ya sabes lo que es vivir una Tesis (Cervera-Alamar, 2019), no obstante me has aguantado (y aguantas) no sé muy bien cómo ni por qué. Gracias por tragarte mis días malos y por escuchar mis rollos que no le interesan a nadie, gracias también por exigirme y por hacer que me plantee mis prioridades y no viva con el piloto automático. Gracias, en definitiva, por hacer mi vida más feliz.

Contents

Acronyms

3DEP 3D Elevation Program
ACCA automatic cloud cover assessment

AoI area of interest

ASE Autonomous Sciencecraft Experiment

AVHRR Advanced Very High Resolution Radiometer

CCD charge-coupled device

CD cloud detection

CMIX Cloud Masking Inter-comparison eXercise

CNN Convolutional Neural Networks

CyCADA Cycle-Consistent Adversarial Domain Adaptation

CV computer vision

DA Domain adaptation

DL Deep learning

EO Earth observation

ESA European Space Agency

FCNN Fully Convolutional Neural Networks

FDL Frontier Development Lab

GEE Google Earth Engine

GSD ground sampling distance

MERIS Medium Resolution Imaging Spectrometer

MIR medium infrared

MISR Multi-image super-resolution

ML Machine Learning

MLP Multilayer perceptron

MTF Modulation transfer function

NASA National Aeronautics and Space Administration

NDSI Normalized difference snow index

NDVI Normalized difference vegetation index

NIR near infrared

NLL negative log likelihood

NN Neural Networks

OLI Operational land imager

PSF point-spread function

PV72 Proba-V cloud detection manually labeled dataset

PVQWG Proba-V Quality Working Group

ROC Receiver operator curve

RS Remote Sensing

SGD Stochastic gradient descent

SRF spectral response function
TL Transfer learning
TOA top of atmosphere
SWIR short-wave infrared
USGS United States Geological Survey
VHR Very High Resolution
VIS visible

Abstract

Remote sensing sensors onboard Earth observation satellites provide a great opportunity to monitor our planet at high spatial and temporal resolutions. Nevertheless, to process all this ever-growing amount of data, we need to develop fast and accurate models adapted to the specific characteristics of the data acquired by each sensor. For optical sensors, detecting the clouds present in the image is an unavoidable first step for most of the land and ocean applications. Although detecting bright and opaque clouds is relatively easy, automatically identifying thin semi-transparent clouds or distinguishing clouds from snow or bright surfaces is much more challenging. In addition, in the current scenario where the number of sensors in orbit is constantly growing, developing methodologies to transfer models across different satellite data is a pressing need.

Henceforth, the overreaching goal of this Thesis is to develop accurate cloud detection models that exploit the different properties of the satellite images, and to develop methodologies to transfer those models across different sensors. The four contributions of this Thesis are stepping stones in that direction. In the first contribution, *"Multitemporal cloud masking in the Google Earth Engine"*, we implemented a lightweight multitemporal cloud detection model that runs on the Google Earth Engine platform and which outperforms the operational models for Landsat-8. The second contribution, *"Transferring deep learning models for Cloud Detection between Landsat-8 and Proba-V"*, is a case-study of transferring a deep learning based cloud detection algorithm from Landsat-8 (30 m resolution, 12 spectral bands and very good radiometric quality) to Proba-V, which has a lower 333 m resolution, only four bands and a less accurate radiometric quality. The third paper, *"Cross sensor adversarial domain adaptation of Landsat-8 and Proba-V images for cloud detection"*, proposes a learning-based domain adaptation transformation of Proba-V images to resemble those taken by Landsat-8, with the objective of transferring products designed on Landsat-8 to Proba-V. Finally, the fourth contribution, *"Towards global flood mapping onboard low cost satellites with machine learning"*, tackles simultaneously cloud and flood water detection with a single deep learning model, which was implemented to run onboard a CubeSat (ΦSat-1) with an AI accelerator chip. In this case, the model is trained on Sentinel-2 and transferred to the ΦSat-1 camera. This model was launched in June 2021 onboard the Wild Ride D-Orbit mission in order to test its performance in space.

Resumen

Los satélites de observación de la Tierra proporcionan una oportunidad sin precedentes para monitorizar nuestro planeta a alta resolución tanto espacial como temporal. Sin embargo, para procesar toda esta cantidad creciente de datos, necesitamos desarrollar modelos rápidos y precisos adaptados a las características específicas de los datos de cada sensor. Para los sensores ópticos, detectar las nubes en la imagen es un primer paso inevitable en la mayoría de aplicaciones tanto terrestres como oceánicas. Aunque detectar nubes brillantes y opacas es relativamente fácil, identificar automáticamente nubes delgadas semitransparentes o diferenciar nubes de nieve o superficies brillantes es mucho más difícil. Además, en el escenario actual, donde el número de sensores en el espacio crece constantemente, desarrollar metodologías para transferir modelos que funcionen con datos de nuevos satélites es una necesidad urgente.

Por tanto, los objetivos de esta tesis son desarrollar modelos precisos de detección de nubes que exploten las diferentes propiedades de las imágenes de satélite y desarrollar metodologías para transferir esos modelos a otros sensores. La tesis está basada en cuatro trabajos los cuales proponen soluciones a estos problemas. En la primera contribución, *"Multitemporal cloud masking in the Google Earth Engine"*, implementamos un modelo de detección de nubes multitemporal que se ejecuta en la plataforma Google Earth Engine y que supera los modelos operativos de Landsat-8. La segunda contribución, *"Transferring deep learning models for Cloud Detection between Landsat-8 and Proba-V"*, es un caso de estudio de transferencia de un algoritmo de detección de nubes basado en aprendizaje profundo de Landsat-8 (resolución 30 m, 12 bandas espectrales y muy buena calidad radiométrica) a Proba-V, que tiene una resolución de 333 m, solo cuatro bandas y una calidad radiométrica peor. El tercer artículo, *"Cross sensor adversarial domain adaptation of Landsat-8 and Proba-V images for cloud detection"*, propone aprender una transformación de adaptación de dominios que haga que las imágenes de Proba-V se parezcan a las tomadas por Landsat-8 con el objetivo de transferir productos diseñados con datos de Landsat-8 a Proba-V. Finalmente, la cuarta contribución, *"Towards global flood mapping onboard low cost satellites with machine learning"*, aborda simultáneamente la detección de inundaciones y nubes con un único modelo de aprendizaje profundo, implementado para que pueda ejecutarse a bordo de un CubeSat (ΦSat-1) con un chip acelerador de aplicaciones de inteligencia artificial. El modelo está entrenado en imágenes Sentinel-2 y demostramos cómo transferir este modelo a la cámara del ΦSat-1. Este modelo se lanzó en junio de 2021 a bordo de la misión WildRide de D-Orbit para probar su funcionamiento en el espacio.

Resum

L'observació de la Terra amb sensors de satèl·lits ens proporciona una capacitat sense precedents per monitoritzar el nostre planeta a alta resolució tant espacial com temporal. Tot i això, per processar tota aquesta quantitat creixent de dades, necessitem desenvolupar models ràpids i precisos adaptats a les característiques específiques de les dades de cada sensor. Per als sensors òptics, detectar els núvols a la imatge és un primer pas inevitable per a la majoria de aplicacions tant terrestres com oceàniques. Encara que detectar núvols brillants i opaques és relativament fàcil, identificar automàticament núvols semitransparents o diferenciar núvols de neu o superfícies brillants és molt més difícil. A més, a l'escenari actual on el nombre de sensors a l'espai creix constantment, desenvolupar metodologies per transferir models que funcionen amb dades de nous satèl·lits és una necessitat urgent.

Per tant, els objectius d'aquesta tesi són desenvolupar models precisos de detecció de núvols que exploten les diferents propietats de les imatges de satèl·lit i desenvolupar metodologies per transferir aquests models a altres sensors. La tesi està composta de quatre treballs que proposen solucions a aquests problemes. A la primera contribució, *"Multitemporal cloud masking in the Google Earth Engine"*, implementem un model de detecció de núvols multitemporal que s'executa a la plataforma Google Earth Engine i que supera els models operatius de Landsat-8. La segona contribució, *"Transferring deep learning models per Cloud Detection between Landsat-8 and Proba-V"*, és un cas d'estudi de transferència d'un algorisme de detecció de núvols basat en aprenentatge profund de Landsat-8 (resolució 30 m, 12 bandes espectrals i molt bona qualitat radiométrica) a Proba-V que té una resolució de 333 m, només quatre bandes i una qualitat radiométrica pitjor. El tercer article *"Cross sensor adversarial amb domini adaptat de Landsat-8 and Proba-V images for cloud detection"*, proposa aprendre una transformació d'adaptació de dominis que faci que les imatges de Proba-V s'assemblin a les preses per Landsat -8 amb l'objectiu de transferir productes dissenyats amb dades de Landsat-8 a Proba-V. Finalment, la quarta contribució, *"Towards global flood mapping onboard low cost satellites with machine learning"*, aborda simultàniament la detecció d'inundacions i núvols amb un únic model d'aprenentatge profund; en aquest cas, el model s'implementa per executar-se a bord d'un CubeSat (ΦSat-1) amb un xip accelerador d'aplicacions d'intel·ligència artificial. El model està entrenat en imatges Sentinel-2 i es transferit a la càmera del ΦSat-1 que es va llançar en juny de 2021 a bord de la missió WildRide de D-Orbit per a provar el seu funcionament a l'espai.

1. Introduction

This Thesis is concerned with the development of algorithms for the automatic detection of clouds in optical satellite images. In this chapter, we introduce the context and motivation of this Thesis to the reader: we briefly introduce remote sensing with optical sensors; the problem of cloud detection together with the existing cloud detection algorithms; the machine learning and deep learning methodologies for cloud screening; and the rationale of transfer learning and domain adaptation. Finally, we describe the research objectives and outline the contributions addressed in the remaining chapters.

1.1 Observing the Earth with satellites

Remote Sensing (RS) is the research field that estimates the properties of objects by measuring its reflected and emitted radiation at a distance. Remote sensing sensors studied in this Thesis are observing the Earth onboard orbiting satellite platforms. Remote sensing sensors are divided in *active* –when the instrument in the satellite emits radiation that is reflected by the surface and gets measured back on the sensor– and *passive* – when the sensor measures the reflectance emitted by other object which in most cases is the sun[1]. All the contributions of this Thesis focus on passive sensors looking at the Earth surface and measuring the reflected sun light in different wavelengths.

First Earth observation (EO) satellites were launched in the 60s to monitor weather patterns (Tatem et al., 2008). In 1972, the iconic Landsat-1 took off to monitor land cover at *high* resolution, kicking off the Earth observation era. Since then, the number of satellites observing the Earth has grown remarkably: as of 2021, the number of active EO satellites is estimated to be 4,550 according to the UCS Satellite Database (2021). This number has been significantly boosted by constellations of CubeSats –small inexpensive lightweight satellites– which are increasing the amount of data obtained from space to unprecedented rates (2,520 out of the 4,550 active satellites were launched between 2020 and 2021). In the last fifteen years we have also witnessed the benefits of open satellite

[1] There are other examples such as sensors measuring the radiation emitted by the Earth or by night lights.

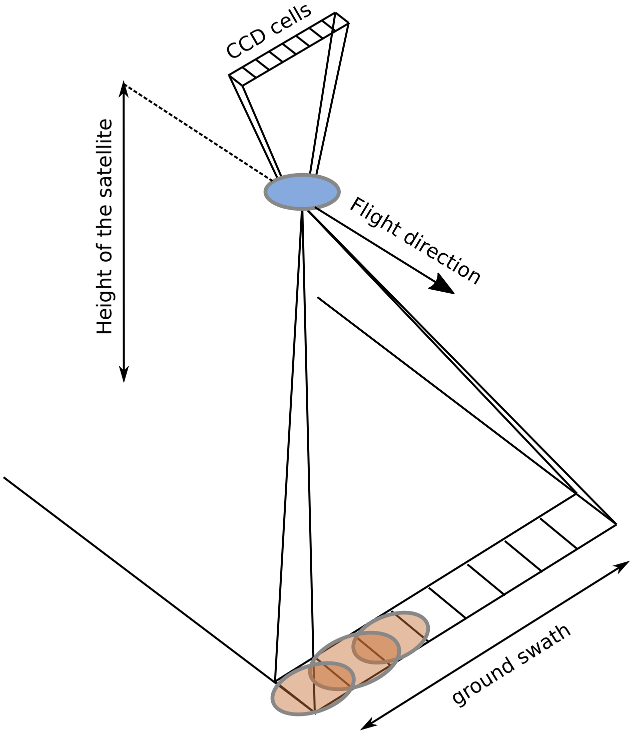


Figure 1.1: Image acquisition diagram of a pushbroom scanner.

data archives: in December 2008, the United States Geological Survey (USGS) started to provide all Landsat scenes at no charge to all users. This led to other major players such as the European Space Agency (ESA) to embrace the concept of free and open-access data. Nowadays, free access to satellite imagery is the norm rather than the exception[2] and there are platforms, such as the Google Earth Engine (GEE) (Gorelick et al., 2017), to facilitate the access and aggregation of different data sources. This Thesis exploits this context of open multi-sensor data to develop new data-driven cloud detection products for remote sensing.

There are three main properties, or *dimensions*, that characterize Earth observation sensors. These properties determine the time series of images acquired by the satellite and play a major role in the Thesis. These properties are the **spatial resolution**, the **spectral resolution** characterized by the number of spectral bands acquired by the satellite, and the **temporal resolution**, which is determined by the revisit time of the satellite to the same location. Understanding the trade-offs between these three dimensions is key to develop methods that exploit all the available information in remote sensing archives and to propose methodologies for transferring models across different instruments.

1.1.1 The spatial dimension

The spatial dimension of a sensor is characterized by its ground sampling distance (GSD). The GSD is the size in meters between two consecutive pixels in an image. Since pixels are squared, the GSD is also the length of the edge of that square; hence, in a 30 m resolution image, each pixel covers 900 m^2 in the surface, theoretically. However, the effective spatial

[2] High resolution (less than 10 m) image archives remain closed yet; still access for research purposes and prepared datasets are common.

resolution of a sensor depends on the optics, type of sensing instrument onboard, and the height of the satellite. Among the different optical instruments, we will focus on *pushbroom* scanners since those are the main sensor of Landsat-8, Proba-V and Sentinel-2 satellites. Pushbroom scanners consist of an array of sensors, usually a charge-coupled device (CCD), arranged perpendicularly to the flight direction of the spacecraft. Each of those CCD cells, during a time of exposition, integrates the reflected solar irradiance over an area on the ground. These quantities are converted to each of the pixel values in the acquired image. Figure 1.1 shows a simplified diagram of the scanning process: as the satellite flights over the Earth, one line of the image is captured at a time. In the figure, we can see that the size of each pixel in the ground depends on the size of the CCD cells in the cross-track direction and the exposition time in the along-track or flight direction. Additionally, if the satellite increases its altitude, the ground swath –area imaged on the surface in the cross-track direction (see Fig. 1.1)– will increase augmenting also the area in the ground where each CCD detector integrates radiance (depicted in orange in the figure); this will turn out into a higher GSD (i.e. worse spatial resolution).

The GSD of a sensor determines the size of the smallest object that can be detected on an image. Figure 1.2 shows Landsat-8 and Proba-V images over the same date and location. Landsat-8 GSD is 30 m and Proba-V is 333 m. We can see that in the Landsat-8 image of the top, smaller objects such as the ponds and the cultivated areas in green can be resolved, whereas in the Proba-V images at the bottom this cannot be reliably observed.

Nevertheless, GSD is not the only factor that influences the resolving quality of the imaging system: the radiometric resolution, limitations of the optical system and the effect of diffraction play also an important role in the effective spatial resolution of the instrument. In order to account to all these effects, the point-spread function (PSF) of the instrument describes the end-to-end response of the imaging system to a pinpoint source of light. The PSF and the Modulation transfer function (MTF) –its analogous in the frequency domain– are used in Chapter 3 for the Thesis contributions Mateo-García et al. (2020b) and Mateo-García et al. (2020a) to simulate Proba-V images from Landsat-8 ones.

1.1.2 The temporal dimension

The temporal resolution, aka revisit time, is the time difference between two consecutive images acquired over the same location. The satellites studied in this Thesis are polar orbiting sun-synchronous satellites; this means that (a) the satellite passes over the poles at every revolution around the Earth, and (b) when it passes over a given point of the Earth's surface, it does it at the same local solar time. For these satellites, temporal resolution depends on the orbit followed by the satellite and on the ground swath. The later can be seen in Figure 1.1: the wider the swath the more likely a given location is more frequently imaged. We can understand now that there is a trade-off between the spatial and the temporal resolution of a satellite: if the satellite is flying on a higher orbit, the swath will we larger but so will be the size of each pixel. Hence, for a given satellite imaging system, if temporal resolution is high, spatial resolution is low and the other way around. Figure 1.2 shows this trade-off for Proba-V and Landsat-8 images: Proba-V images with high temporal resolution (1 day revisit over this location) contrast with its low spatial resolution (333 m) and Landsat-8 higher spatial resolution (30 m) has lower revisit frequency (7 to 14 days).

In order to overcome this limitation, i.e. to have high temporal and spatial resolutions,

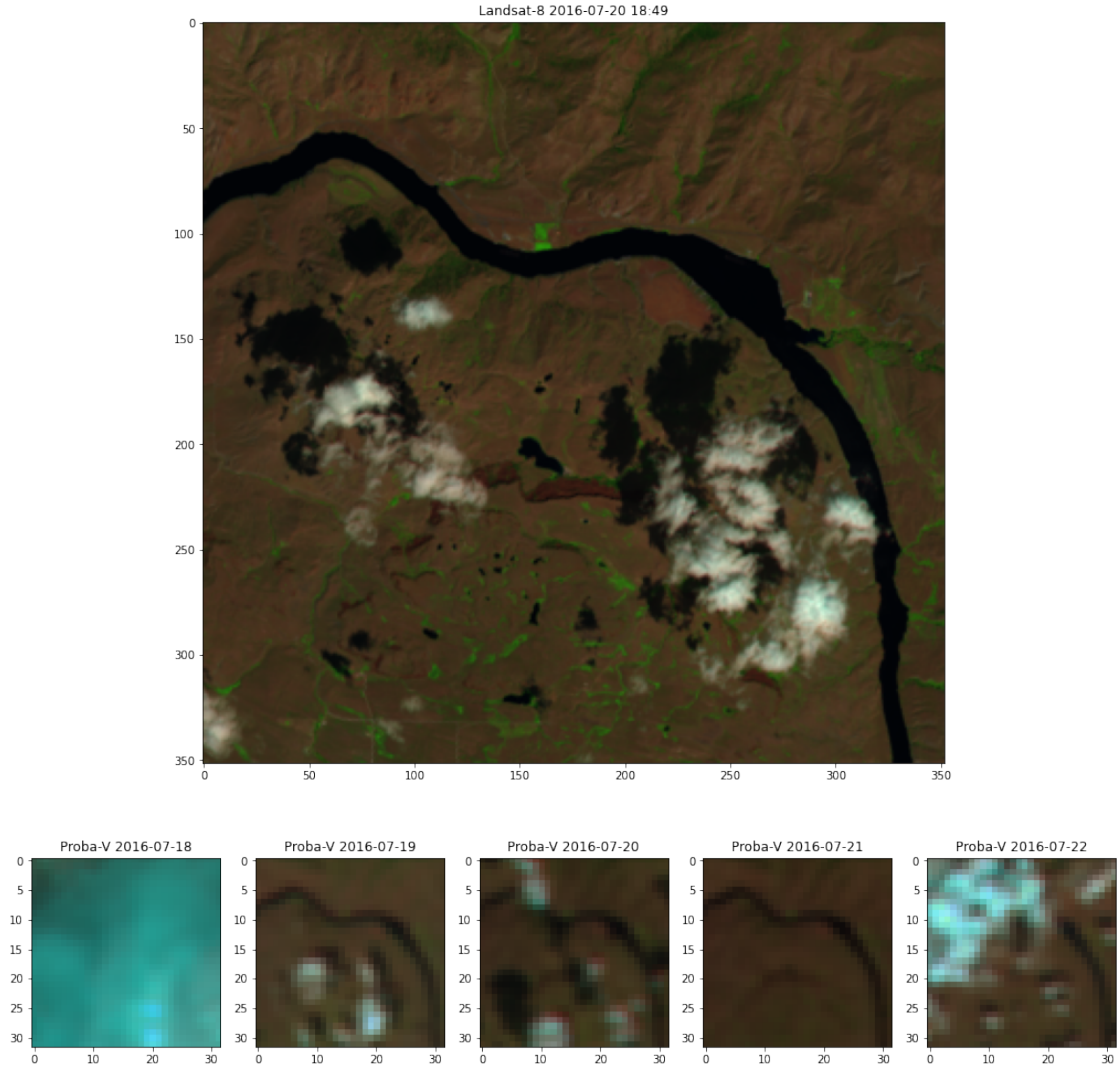


Figure 1.2: Landsat-8 (top) and Proba-V images (bottom) over the same 10.5 km^2 × 10.5 km^2 location. Landsat-8 image acquired on 20th July 2016 and Proba-V images acquired between 18th and 22nd July 2016. On one hand, Proba-V has high temporal resolution of 1 day over this location which contrast with its low spatial resolution (333 m GSD). On the other hand, Landsat-8 higher spatial resolution (30 m GSD), has a lower temporal resolution: between 7 to 14 days. False color composites: SWIR, NIR and Red bands of Proba-V and bands B6, B5, B4 of Landsat-8.

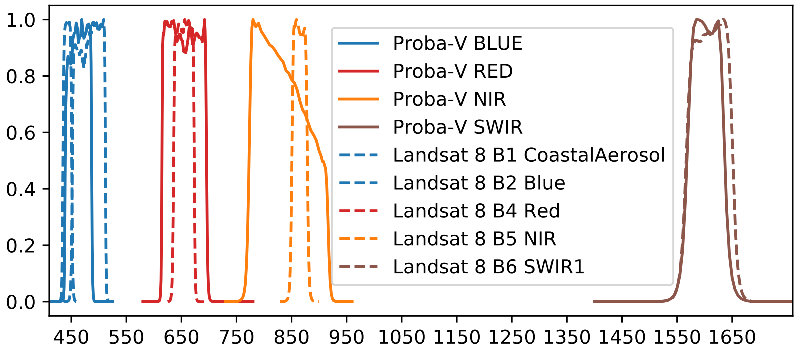


Figure 1.3: Spectral response function of all four Proba-V bands (solid) and B1, B2, B4, B5 and B6 bands of Landsat-8 (dashed).

the current trend is to launch several identical instruments flying in complementary orbits. This is the case of the European Copernicus missions Sentinel-1, Sentinel-2 and Sentinel-3 which each of them deployed two satellites with similar instruments. Another example is the Planet Dove constellation which currently has more than 180 nano-satellites providing daily 3m resolution imagery. In the first contribution of this Thesis (Mateo-García et al., 2018) we exploit the temporal dimension to develop multi-temporal cloud detection models and in the last one (Mateo-Garcia et al., 2021) we discuss the potential of large constellations of CubeSats with low revisit times to speed-up disaster response after flooding events.

1.1.3 The spectral dimension

The spectral dimension of the satellite images are the different wavelengths in the electromagnetic spectrum where the sensor integrates radiances. Optical sensors measuring the reflected solar light are sensitive to wavelengths from 430 nm to 2300 nm comprising the visible (VIS), the NIR, the medium infrared (MIR), and the SWIR spectral ranges. Measuring radiance (TOA reflectance) in different parts of the spectrum is used to characterize the different materials that we observe; this is because different materials have different spectral signatures: they absorb, reflect and emit electromagnetic radiation at different wavelengths depending on their composition and structure.

Most of the sensors that we studied in this Thesis are multi-spectral sensors. These sensors have *few* pre-defined regions in the electromagnetic spectrum, called *bands*, where they integrate the electromagnetic radiation. For instance, the Operational land imager (OLI) pushbroom sensor onboard the Landsat-8 satellite has 9 different *bands*. Since all the bands are measured simultaneously while the satellite is scanning the Earth, for each of these bands we will have one 2D image that consists of the radiance measured by the sensor on that *band*. The spectral response function (SRF) of a band is the function that indicates which wavelengths, and with which weight, are integrated in the measurement. Figure 1.3 shows the SRF of the four bands of Proba-V and the corresponding overlapping bands of Landsat-8 (bands B1, B2, B4, B5 and B6). Landsat-8 bands are narrower than Proba-V ones, specially the NIR band. The SRF of Landsat-8 and Proba-V is used in contributions Mateo-García et al. (2020b,a) to simulate the reflectance that would be acquired by Proba-V from a Landsat-8 image.

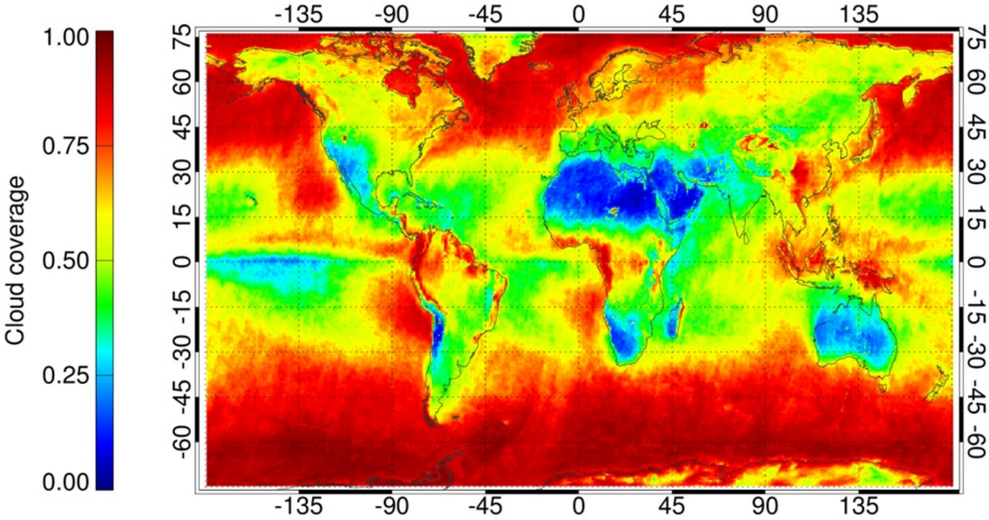


Figure 1.4: Global annual mean cloud cover derived from three years (2007–09) of Envisat data. The map shows areas with little to no cloud coverage (blue) as well as areas that are almost always cloudy (red). Credits: ESA (https://www.esa.int/ESA_Multimedia/Images/2013/09/Cloud_cover).

1.2 Cloud detection

Clouds are masses of condensed water vapour or ice particles suspended in the atmosphere of the Earth (Lohmann et al., 2016). Cloud cover climatological studies estimate that around 60-70% of the Earth's surface is covered with clouds and that the cloud cover is 5%-15% higher over oceans than over land (Stubenrauch et al., 2013). Nevertheless, cloud cover is not equally distributed over all land locations: Figure 1.4 shows the annual global mean cloud cover derived from Envisat between 2007 and 2009. We can see that, for instance, the north of Africa and most parts of Australia have very few clouds around the year whereas equatorial South America and western Africa is almost always cloud covered.

For remote sensing optical applications, clouds can either be seen as a source of contamination for applications looking at the Earth surface or a source of information for applications seeking to understand the atmosphere. In both cases, cloud detection (CD) is an unavoidable first step in their processing chains: from crop detection (Wolanin et al., 2019), ocean color analysis (Ruescas et al., 2018), to cloud type classification (Zantedeschi et al., 2019), identifying the cloudy pixels is needed to further proceed in their analysis. Cloud detection errors in those applications lead to biased analysis in the case of false negatives (undetected cloudy pixels) (Bulgin et al., 2018) or to discard valid meaningful data (clear pixels classified as clouds) (Pipia et al., 2019). Given the necessity of cloud masking, most of the operational satellite missions distribute a cloud mask together with each image acquired by the satellite. Hence, each optical sensor usually has an official operational cloud detection algorithm for computing a cloud mask for each acquired image.

1.2.1 Threshold-based cloud detection algorithms

One of the simplest and most common approaches for cloud masking are the so-called threshold-based. These models are knowledge-based systems based on physically grounded heuristics (clouds are bright, clouds are cold, high cirrus clouds reflect radiation in the 1.36-

1.39 nm region of the spectrum, etc). These models afterwards set some static or dynamic thresholds in one or several spectral bands of the image, or band combinations, that exploit these heuristics to discriminate clouds (i.e. pixels are identified as *cloud* or *clear* if the values on those bands are bellow/above these thresholds). Most optical missions still rely on threshold-based methods for their operational cloud detection approaches. Although some threshold-based algorithms perform well when they are highly tuned and the sensor has several bands (e.g. the FMask algorithm for Landsat-8 (Zhu & Woodcock, 2012)), these models still missclassify clouds in several critical situations and it is not possible or difficult to transfer them to other sensors.

In the first contribution of this Thesis (Mateo-García et al., 2018), we explore threshold based algorithms using multi-temporal information (i.e. using previous images over the same location). Multi-temporal cloud detection is a fundamentally easier problem than single-scene cloud detection since the Earth surface usually varies slowly with time and hence, abrupt changes in reflectance are mostly caused by clouds. Nevertheless, multitemporal algorithms are computationally expensive since they require co-located information of previous images; thus, all operational CD algorithms are single scene (i.e. they produce the cloud mask based only on the reflectance in the current image acquisition). In the rest of this chapter, we focus on single scene CD approaches, which are also the object of study of the rest of the contributions of this Thesis.

1.2.2 Is really cloud detection a big issue?

Improving the accuracy of cloud detection models is not a theoretical problem but a pressing need for some operational satellite missions. In the course of this Thesis, we have been involved in several projects aiming to improve the operational cloud detection algorithm of some of those missions. The Proba-V cloud detection Round Robin (Iannone et al., 2017), organized in 2016, asked to six different institutions to provide a cloud detection algorithm for Proba-V since their current operational algorithm based on thresholds (Toté et al., 2018) was failing in several critical situations. In that project, our proposed solution based on neural networks (Gómez-Chova et al., 2017b) was selected for its operational implementation. This model will replace the threshold-based algorithm in the C2 reprocessing of the Proba-V archive. The Cloud Masking Inter-comparison eXercise (CMIX) organized in partnership between the ESA and the National Aeronautics and Space Administration (NASA) benchmarked ten different cloud detection algorithms for Sentinel-2 and Landsat-8 (European Space Agency, 2019). Again, the goal of this study, which is submitted for publication (Skakun et al., Submitted), is to find out if current operational threshold-based models for those satellites could be improved or replaced by more accurate methodologies.

On the other hand, transferring threshold-based cloud detection algorithms to other sensors might seem easy if both sensors share similar bands. However, these algorithms are very tailored to the specific characteristics of each sensor (spectral bands and radiometric values) and they are usually highly sensitive to their input data. To see this, the first Proba-V operational CD algorithm (collection CO) was a static threshold technique using the Blue and the SWIR bands transferred from SPOT-Vegetation (Lisens et al., 2000). This algorithm was the first CD operational model for Proba-V and their cloud masks were released together with every Proba-V image. However, when users started to test the operational Proba-V products, they faced many errors related to the quality of the cloud mask, which were eventually reported to the Proba-V Quality Working Group. This situation led to the development of a completely new dynamic-threshold algorithm for collection C1 (Toté et al., 2018) that has been recently replaced by the neural networks approach of Gómez-Chova et al. (2017b) in collection C2 (Toté et al., 2021).

Additionally, the constant growth of satellite launches every year will make developing a cloud detection model for a new sensor a more and more common task. Thus, contributions of this Thesis Mateo-García et al. (2020b) and Mateo-García et al. (2020a) show how to transfer cloud detection models between sensors with high accuracy. These models could be used at the beginning of satellite missions when little or no data is available. In Chapter 5 we discuss also a use case of transfer learning for cloud and water detection which has been deployed onboard the WildRide mission launched in June 2021 (see section 6.3.7 in Chapter 6).

1.3 Machine learning methods for cloud detection

One of the methodologies to boost cloud detection accuracy is Machine Learning (ML). Three contributions of this Thesis propose ML models for cloud detection (Mateo-García et al., 2020b,a; Mateo-Garcia et al., 2021). Thus, in this section, we set the foundations of the ML approach for CD. ML for cloud detection provides a principled paradigm to build more complex and sophisticated cloud detection algorithms. On its simplest setting, ML cloud detection is framed as a supervised binary classification problem where a labeled dataset of pixels is required; i.e. pixels, $i = 1,\dots,N$, on some images, must be classified as *cloudy*, coded with 1, or *clear*, coded with 0 (we will denote this as $y_i \in \{0,1\}$). For each of these pixels some informative *features* $\boldsymbol{x}_i$ are extracted; these features must be chosen to be useful for the task of discriminating clouds. Examples of these features include the pixel values of the image, pixel values in the surrounding of the pixel or pixel value combinations such as spectral indices, e.g. the Normalized difference vegetation index (NDVI), the Normalized difference snow index (NDSI) or whiteness (Gomez-Chova et al., 2007).

Once we have a dataset of features and target values $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$, we could use one of the different supervised ML algorithms to *learn* the mapping function f from $\boldsymbol{x}$ to y. This *training* process in most cases is an optimization process where some free parameters θ of the mapping function, f_θ, are optimized to minimize a given criteria called *loss* ($\mathcal{L}$). In the case of binary classification one of the most common losses is the sum or average[3] of the cross-entropy (CE) of the estimated cloud probabilities, $f_\theta(\boldsymbol{x}_i) \in [0,1]$, for all the labeled pixels in the dataset:

$$\mathcal{L} = \mathrm{NLL}(\theta;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{CE}\big(f_\theta(\boldsymbol{x}_i), y_i\big) = \frac{1}{N}\sum_{i=1}^{N} -y_i \log(f_\theta(\boldsymbol{x}_i)) - (1-y_i)\log(1-f_\theta(\boldsymbol{x}_i)) \tag{1.1}$$

This equation follows the maximum likelihood principle (the negative log likelihood (NLL) loss) for independent binary outputs y_i given the inputs $\boldsymbol{x}_i$ (Murphy, 2013)[4]. When the

[3] Both sum and average yield the same minimum over θ.

[4] Assuming $Y_i \mid \boldsymbol{X}_i$ follows a Bernoulli distribution: $Y_i \mid X_i \sim \mathcal{B}(f_\theta(\boldsymbol{X}_i))$ and $Y_i \mid \boldsymbol{X}_i$ is independent of $Y_j \mid \boldsymbol{X}_j$ $\forall i \neq j$.

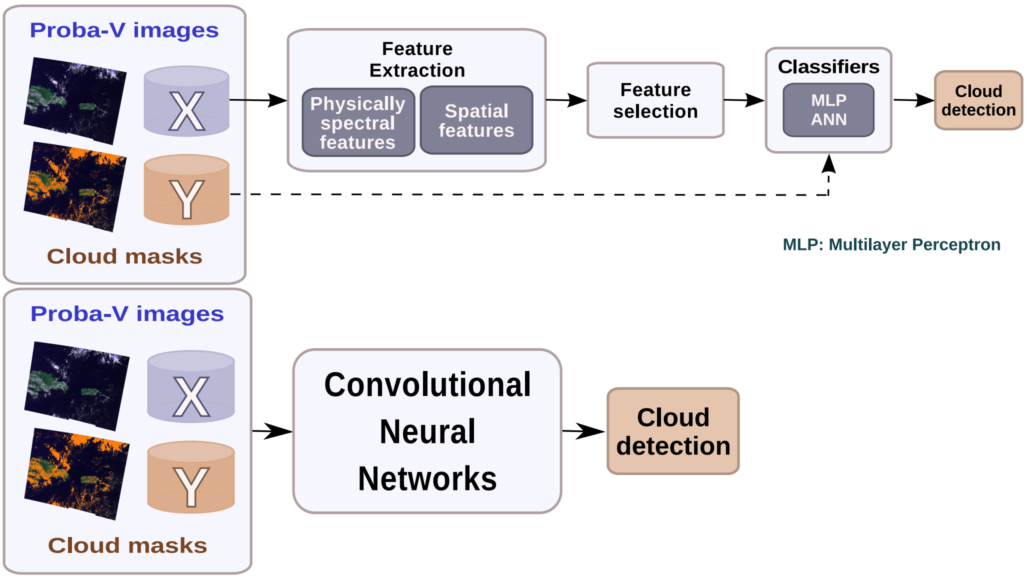


Figure 1.5: ML Classical (top) and deep learning (bottom) methodologies to develop ML based CD models. The classical approach creates a pixelwise model from a set of manually selected features for each pixel whereas in the deep learning approach inputs are raw images and the feature extraction step is learned end to end.

training process ends, the best found weights $\theta^\star$, together with the function f, are used to estimate cloud probabilities of the pixels from new images.

There are tons of ML approaches for cloud masking. The earliest paper found in the literature is Lee et al. (1990), which proposes neural networks to classify cloud types in single channel Landsat MSS image subscenes. Also, in the early nineties, Slawinski et al. (1991) and Yhann & Simpson (1995) already proposed neural networks to mask clear versus cloud pixels for the Advanced Very High Resolution Radiometer (AVHRR) sensor and highlighted the benefits of ML compared with threshold based approaches. Since then, other ML models have been proposed such as kernel methods (Ishida et al., 2018; Gómez-Chova et al., 2010), random forest (Hollstein et al., 2016), or gradient boosting machines (Sentinel Hub team, 2017). In this Thesis, we will focus mainly on neural networks since they are the most predominant ML models nowadays due to their capability to scale to arbitrarily large datasets.

1.3.1 Classical and deep learning approaches for cloud detection

ML approaches to cloud detection can be further divided into *classical* and *deep learning* approaches. The so called ML classical approach was just described before: briefly, a set of manually selected spatial and spectral features are extracted for each pixel in the training set, afterwards a supervised classifier is trained to distinguish the label of those pixels based on the provided features. Figure 1.5 shows a diagram of the classical approach. As an example of the *classical* approach, the CD model proposed in Gómez-Chova et al. (2017b) first selects forty different spatio-spectral features extracted from Proba-V images. Afterwards, an Multilayer perceptron (MLP) is trained on a dataset $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}$ of extracted features ($\boldsymbol{x}_i \in \mathbb{R}^{40}$) and labeled pixels y_i. Classical ML approaches are normally pixelwise, in the sense that the trained classifier f_θ can be applied independently to each pixel in the test image after the feature extraction step. In the Proba-V example, this means that, for prediction, we first extract the forty spatio-spectral features to create a 40 channel image. Afterwards the trained MLP is applied independently to every pixel in the image.

Classical Neural Networks (NN) approaches have been used for operational CD; for instance, as mentioned before, for the Collection C2 Proba-V reprocessing (Toté et al., 2021) or also for MERIS and AATSR sensors onboard the Envisat satellite. For these two sensors, the developed NN was implemented in the ESA BEAM/SNAP software. Gómez-Chova et al. (2013) describes the methodology which was the basis of the work for Proba-V.

On the other hand, the *deep learning* approach for cloud masking trains end-to-end models where the input is the raw image and the output is the cloud mask. These models seek to learn the feature extraction step directly from the raw data. There are thus two main differences between the *classical* and *deep learning* approach: firstly, in the former, a set of useful features are manually selected whereas in the later the input is the raw multi-spectral image; secondly, the *classical* approach produces a pixelwise model whereas deep learning models for CD are trained in patches of the images and therefore require subimages as inputs. The contributions of this Thesis, Mateo-García et al. (2020b,a); Mateo-Garcia et al. (2021), propose models that follow the deep learning approach; henceforth, section 1.5.3 explains neural networks and deep learning in depth and describes *fully convolutional neural networks* which are models that we propose in those contributions for segmenting satellite images.

1.4 Labeled data for cloud detection

Before delving into neural networks formulation, it is worth highlighting what perhaps is the biggest shortcoming of ML for cloud detection: The need of a *representative enough, accurately labeled* dataset $\mathcal{D}$ for training the model. However, the need of labeled data is not exclusive of ML models: labeled datasets are required not only for training but more importantly for testing and benchmarking cloud detection models. Indeed, not only ML based algorithms but also threshold based CD models require independent and representative manually labeled datasets for the validation of their methodologies. This is because, the most reliable approach to estimate the error of a prediction method is the so called holdout method. In the holdout method, we create a fresh dataset of labeled data $\mathcal{D}_S$ and compute the error in that subset (empirical error). Learning theory shows that this error is an unbiased estimator of the out-of-sample error (error over any possible data) and the Hoeffding's inequality guarantees that the gap between this error and the empirical error computed in $\mathcal{D}_S$ is no larger than $\mathcal{O}\left(1/\sqrt{S}\right)$; where S is the number of elements in $\mathcal{D}_S$ (see the generalization chapter of Hardt & Recht (2021) for a modern approach to generalization theory for deep learning models).

However, the assumptions needed to prove the Hoeffding inequality require that the prediction method (the cloud detection method in our case) is fixed *before* we generate the dataset $\mathcal{D}_S$; in other words, re-using of testing data invalidates the statistical guarantees of the holdout method (see Hardt & Recht (2021) or Abu-Mostafa et al. (2012)). The violation of this criteria is what is known as overfitting: the estimation of the error in the sampled labeled data is no longer a good estimate of the out of sample error. From the practical point of view, this means that, to construct ML models, we need two samples of labeled data: one for training the model and another one to later validate that model (that we should *hold out* until model is chosen). Hence, the main difference between ML and threshold based approaches is that the former doubles (at least) the amount of data needed. In this Thesis, one of the recurring topics that we will tackle is about alleviating the data requirements for training ML models. Contributions Mateo-García et al. (2020b) and Mateo-García et al. (2020a) propose to use data from a similar satellite for training (transfer learning) whereas in contribution Mateo-Garcia et al. (2021) the *WorldFloods* training dataset is mostly built from ground truths gathered from other satellites at slightly different acquisition times.

Training and testing splits

As we explained before, splitting the labeled data in different subsets for training and testing is a required condition to demonstrate generalization of ML models. Yet, carefully creating this split is one of the most critical points to attain the generalization we are looking for (in our case, we seek that our cloud detection models generalize to new images acquired by the sensor). The rule of thumb to create a train/test split suggested in Ng (2017) is that labeled test data must be as similar as possible to the data where you plan to deploy your model; in particular, they suggest that correlations between training and testing data must be the same as correlations between training data and data seen when the model is deployed. For our application (global cloud detection models), deployment data are new images acquired by the sensor. Hence, one sensible split in this case is to use pixels from different image acquisitions for training and testing. Otherwise the models that we develop might overfit to the particularities of a given acquisition. There is currently a big concern on the rigour of validation of ML models in RS; for instance, Ploton et al. (2020) suggested that ignoring the spatial auto correlation of the data in the train/test split leads to overestimated validation metrics that might invalidate previous published results. In cloud detection, a common practice in some studies is to divide satellite images in tiles and later splitting these tiles in training and testing. Although pixels in this split are not used at the same time for training and testing, correlation between training and testing pixels could be high for tiles from the same image acquisition. Therefore, in order to avoid to over-estimate of the accuracy of our methods, in all the contributions of this Thesis we followed the approach of splitting the data at the satellite image acquisition level; that is, we always used images from different acquisitions for training and for testing.

1.4.1 Labeling clouds

Labeled datasets that can be used as *ground truth* for cloud detection are not usually real 'ground' measurements as in other RS applications. This is because obtaining co-located measurements of cloud presence by ground stations is difficult or not feasible given the complex nature of clouds. A very recent study, Skakun et al. (2021), showed that co-locating simultaneous ground based and satellite based images is possible; although, to develop a global dataset, this system should be extended and tested at a dense network of global observatories. Therefore, as of now, most of the manually labeled ground truth datasets for cloud detection are derived by photo-interpretation; that is, an operator manually identifies the pixels on the image as cloudy or clear, normally using some dedicated software.

In the course of this Thesis, my collaborators and I have curated two big datasets for training and testing ML models: the Proba-V manually labeled cloud mask dataset used in contributions Mateo-García et al. (2020b) and Mateo-García et al. (2020a), and the *WorldFloods* dataset used in Mateo-Garcia et al. (2021). We will delve into the details of the creation process of the former to illustrate the time and dedication that is needed to build those datasets.

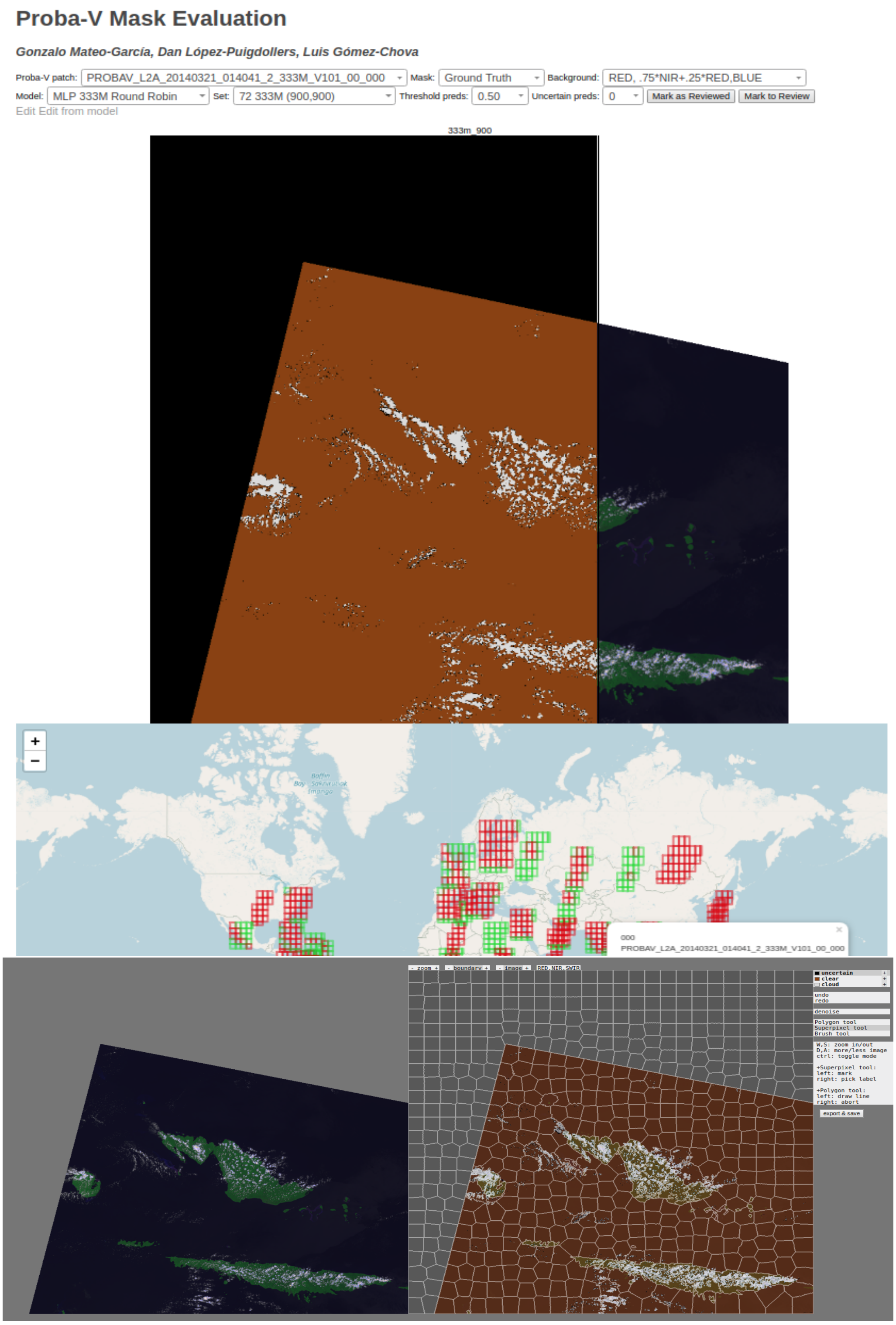
The Proba-V manually labeled dataset

The Proba-V cloud detection manually labeled dataset (PV72) was developed in two phases: in the first phase, used for the Proba-V Round Robin exercise (Iannone et al., 2017), we adapted a semi automatic cloud labeling methodology developed for the Medium Resolution Imaging Spectrometer (MERIS) sensor proposed by Gomez-Chova et al. (2007). This methodology consists of labeling clusters of pixels extracted by an expectation-maximization algorithm. Afterwards, a post-processing of the labeled clusters together with an unmixing algorithm is applied to obtain a cloud abundance product for every pixel in the image. This cloud abundance is used as ground truth to train the neural networks that we presented in Iannone et al. (2017) and Gómez-Chova et al. (2017b). For the Proba-V ground truth used in contributions Mateo-García et al. (2020b), Mateo-García et al. (2020a) and for the development of collection C2 operational Proba-V algorithm, we improved the aforementioned labels by manually checking and refining them using a custom software developed by us for this purpose. This was required to remove systematic errors of the ground truth in thin clouds and over bright surfaces. Figure 1.6 shows three screenshots of this labeling application: at the top, we show an overlay of the manual cloud mask and the false RGB composite of Proba-V; in the middle, the map with locations already reviewed (red) and locations pending to review (green); at the bottom, we display the labeling refinement tool that we adapted from Tangseng et al. (2017). Using these tools we labeled 72 Proba-V acquisitions with more than 10^9 pixels. It took us approximately one month to label the clusters and two extra months to refine the labels by two persons. We measured the agreement of this dataset with a completely independent dataset gathered by Brockmann Consult for the development of the Proba-V C2 cloud mask; the agreement over 950 pixels in 12 different images was over 93%. Our Proba-V manually labeled dataset can be seen in this web page: https://isp.uv.es/projects/cdc/probav_dataset.html.

Caveats of labeling clouds

Although manually labeled datasets are the golden standard to evaluate CD algorithms and therefore to measure progress in the field, the process of manually labeling clouds is not exempt of errors. In Scaramuzza et al. (2012), authors reported a mean overall disagreement of 7% when creating the Irish dataset in 11 Landsat-7 scenes labeled by three different experts (following the same labeling methodology). For the PV72 dataset, we estimated a similar figure (6.62%) over 950 pixels in 12 different Proba-V acquisitions, in this case, the labeling teams were different and followed different labeling procedures. These errors are higher than the errors reported in similar computer vision datasets such as CityScapes (Cordts et al., 2016) (4% error).

Figure 1.7 shows some of those (possible) errors in images from the manually labeled Biome dataset (Foga et al., 2017) created for the validation of Landsat-8 cloud detection methods and used in contributions Mateo-García et al. (2018, 2020b,a) of this Thesis. The images at the top show the Landsat-8 RGB composite and the images at the bottom the same content with labeled cloud pixels in black. We can see that several thin and



Proba-V Mask Evaluation

Gonzalo Mateo-García, Dan López-Puigdollers, Luis Gómez-Chova

Proba-V patch: PROBAV_L2A_20140321_014041_2_333M_V101_00_000

Mask: Ground Truth

Background: RED, .75*NIR+.25*RED,BLUE

Model: MLP 333M Round Robin

Set: 72 333M (900,900)

Threshold preds: 0.50

Uncertain preds: 0

000
PROBAV_L2A_20140321_014041_2_333M_V101_00_000

■ uncertain
🟫 clear
□ cloud

W,S: zoom in/out
D,A: more/less image
ctrl: toggle mode

+Superpixel tool:
left: mark
right: pick label

+Polygon tool:
left: draw line
right: abort

Figure 1.6: Labeling application used to refine the cloud labels of Proba-V images. At the top, we show an overlay of the manual cloud mask (brown clear, white cloud) and the false RGB composite of Proba-V (red, NIR and blue bands). In the middle, the map with locations reviewed (red) and pending to review (green) is shown. When a rectangle in the map is clicked, the corresponding overlay is shown at the top. At the bottom, we display the labeling refinement tool that we adapted from Tangseng et al. (2017).

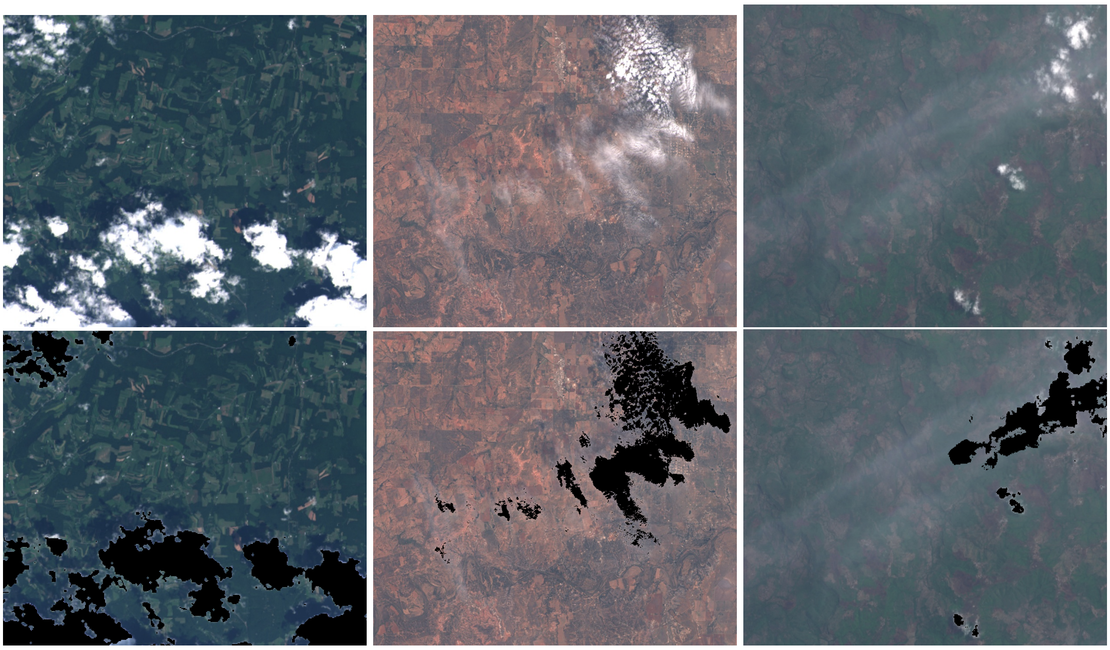


Figure 1.7: Landsat-8 images from the Biome dataset (Foga et al., 2017). Images in the second row show in black pixels marked as clouds in the ground truth of the Biome dataset. In the three images we can see thin clouds, small clouds and cloud borders that are not marked as clouds. The Biome dataset can be explored using our Google Earth Engine script: https://code.earthengine.google.com/f5ff4b932dbfcdbe242b74938694a9c1.

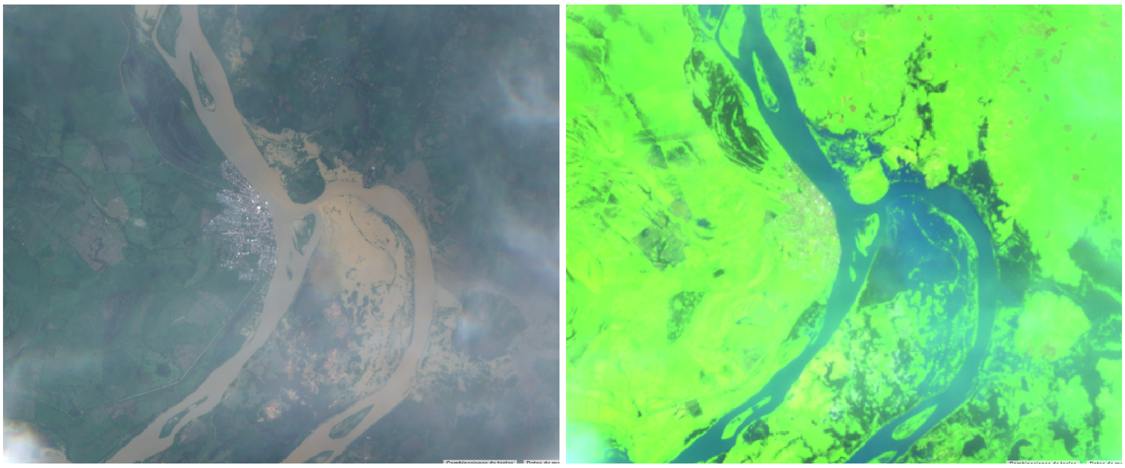


Figure 1.8: Sentinel-2 RGB (left) and SWIR, NIR, Red (right) composites. This image is covered by thin clouds, still we can see the devastating effects of a riverine flood of the Cauca river in Nechi, Colombia in May 2018. The s2cloudless CD method identifies all the pixels in this image as clouds; nevertheless this image can still be used to estimate the extent of the flood. This image was not included in the *WorldFloods* dataset of contribution Mateo-Garcia et al. (2021) for the problems with the cloud mask.

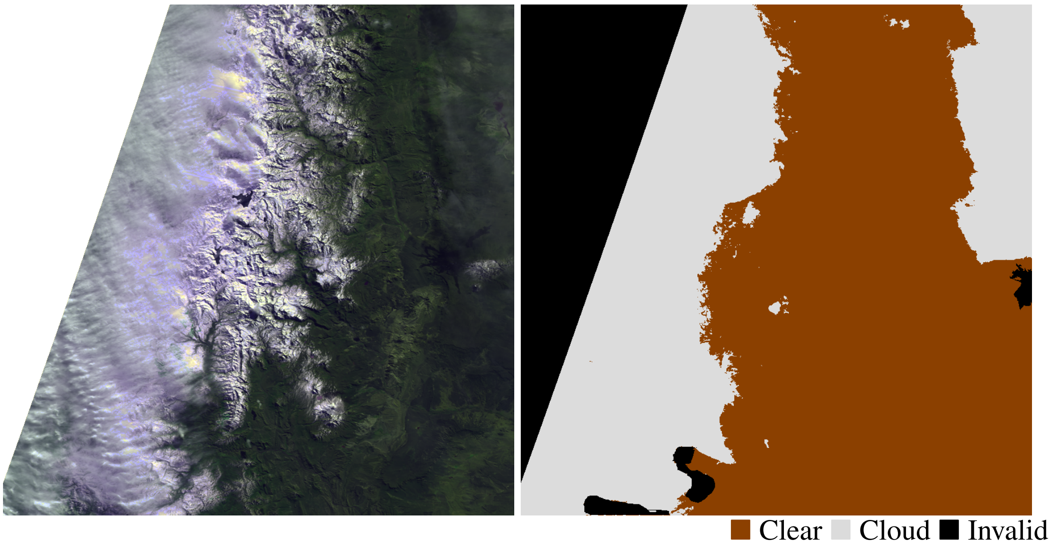


Figure 1.9: Proba-V image (left) and manually labeled cloud mask (right) over the Andes mountains. In the right image, pixels in black are areas where the analyst could not decide whether those pixels are cloud contaminated or not. These pixels are excluded when computing the metrics and losses for training the models. Image from the PV-72 dataset.

semi-transparent clouds are not labeled in the ground truth. At this point, it is worth to highlight that, at the end of the day, ML models are only as good as the data they are trained on and that they inherit the biases of the training data. The slogan "garbage in, garbage out", or more recently "bias in, bias out", synthesizes very well this situation. In particular, in the case of the Biome dataset, we have observed that several thin clouds are unlabeled. Hence, models using this data for training will exhibit the same kind of biases as the dataset, i.e. omission errors in thin and semi-transparent clouds (see our latest paper with models trained on this data (López-Puigdollers et al., 2021)).

The problems with thin and semitransparent clouds happen, to some extent, because cloud detection is an ill-posed problem. Indeed, the cloud definition at the beginning of section 1.2 is rather vague: how many suspended particles of water vapour are needed to *be* a cloud? In Lohmann et al. (2016), authors refine later that definition using the concept of *optical depth*, which is the amount of radiation removed from a Sun's light beam by scattering and absorption. Still, this definition does not help either to label semi-transparent clouds where we have mixed radiation from the cloud and from the surface (which is difficult or not possible to quantify). Additionally, for land applications, thin clouds might or might not be an issue. Figure 1.8 shows the RGB channels of a Sentinel-2 image covered by thin clouds. For biophysical parameters retrieval, this image should be masked, however, to estimate the flood extent, this image is perfectly valid. Additionally, in this figure we can see another problem with thin clouds, which is that they affect different spectral bands in a different way. Indeed, the RGB composite on the left is much more affected than the SWIR, NIR, Red composite on the right.

Finally, when manually labeling by photo-interpretation there are some errors caused by the inability of the labeler to identify certain pixels. Figure 1.9 shows an example of a partially cloud covered Proba-V image over a snowy area in the Andes mountains. This is a particularly challenging scene where clouds and snow are difficult to distinguish. For these scenes, one practice that we undertake in the PV-72 dataset is to leave undefined pixels in the ground truth when the labeler is unsure of the class (black pixels in the ground truth mask of the left). Those pixels are then masked (not used) for training and for validating the models.

1.5 Deep learning

Deep learning (DL) models are at the core of the works of this Thesis: Mateo-García et al. (2020b,a); Mateo-Garcia et al. (2021). In particular, in all of these, we propose *fully convolutional neural networks*, which is a type of deep learning model, to segment clouds (and floods) in satellite imagery. Why do we choose deep learning? in a nutshell, the main benefit of DL models is that their accuracy scale with data; in other words, they can grow to accommodate large amounts of (labeled) data. That means that these models a) are able to exploit large labeled datasets such as the Biome or the PV-72 datasets mentioned in the previous section, and b) if we provide them with even more labeled data, their accuracy will increase. The reader should be warned that this is an empirically based statement: actually what has been shown is that for some problems with very large labeled datasets, the most accurate models by a large margin are based on DL. Henceforth, the hypothesis of this Thesis is that, since some of these problems are very similar to cloud detection in satellite images, DL models should also outperform other approaches for this task *provided a large and accurately labeled dataset*[5].

Therefore, in this section, we delve into the details of deep learning to expedite how these models of the so called *deep learning approach for cloud detection* are constructed (see sec. 1.3.1). Deep learning is just a term to refer to neural networks with many intermediate layers; hence, in the next subsections we explain neural networks incrementally starting from the basics (subsec. 1.5.1), going through the multilayer-perceptron (subsec. 1.5.2) and convolutional neural networks (subsec. 1.5.3), to finally reach fully convolutional neural networks which is our proposal for cloud detection (subsec. 1.5.4).

1.5.1 Neural Networks building blocks

NN trained with the back-propagation algorithm became popular in the late eighties after the highly influential paper of Rumelhart et al. (1986). NN models consist of an stack of differentiable operations applied to an input $\boldsymbol{x}$ and some parameters (aka weights that we will denote by θ). The training procedure of NN consists of optimizing those weights to minimize a differentiable training loss (such as the loss of equation (1.1) in section 1.3). The reason why differentiability is required is because NN use *gradient based* optimization algorithms. These algorithms work by iteratively optimizing θ using at each iteration the gradient of the loss with respect to those weights. Algorithm 1 shows the pseudocode of gradient descent which is the simplest gradient based optimization method; we use the notation of section 1.3 and equation (1.1); in particular, we denote with $\text{NLL}(\theta;\mathcal{D})$ the scalar negative log likelihood (NLL) loss function that we seek to minimize,

Using this simple algorithm, NN models produce powerful prediction functions $f_{\theta^\star}$ that we use for detecting clouds in images. Nevertheless, there are two concepts that are behind the success of NN. The first one is the **back-propagation** algorithm which gives a procedure to evaluate gradients of arbitrarily multi-layered complex functions. The second one is using the **stochastic gradient**, which allows to scale the optimization procedure of

[5] Ways to go around this requirement are also tackled in this Thesis (see sec. 1.6).

Algorithm 1 Gradient descent

```
$\theta_1 \leftarrow$ random
for $s \in 1...K$ do
    $\theta_{s+1} \leftarrow \theta_s - \gamma\frac{\partial}{\partial\theta}\mathrm{NLL}(\theta_s;\mathcal{D})$
end for
$\theta^\star \leftarrow \theta_K$
```

algorithm 1 to arbitrarily large datasets. We delve into the details of both of them in this section.

Back-propagation: the magic behind computing gradients

The back-propagation algorithm proposed in Rumelhart et al. (1986) is an efficient algorithm to do backwards differentiation; that is, to evaluate the gradient of a differentiable but arbitrarily complex function, such as NLL of equation (1.1), w.r.t. its inputs. It assumes that this function is an stack of simple functions with known gradients ($NLL(\theta) = h_1 \circ h_2 \ldots \circ h_K(\theta)$) and that the function is an *scalar* function[6]. We call a function *scalar* when it has a vector input but a single scalar output ($\mathrm{NLL} : \mathbb{R}^d \to \mathbb{R}$). The back-propagation algorithm is just the algorithm to evaluate the gradient of such a function following a computationally inexpensive procedure (see e.g. chapter 6 of Goodfellow et al. (2016) for a detailed explanation).

Nowadays, back-propagation is implemented at the core of most machine learning libraries such as TensorFlow (Abadi et al., 2015) and PyTorch (Paszke et al., 2019). This has the advantage that NN users only have to implement the error function, $NLL(\theta)$ (referred as forward pass of the network) and simply calling a method of the library to obtain the evaluation of the gradient of that function in that θ (The `.backward()` method in Pytorch or `.compute_gradients()` in TensorFlow). Although this is extremely handy, understanding what goes under-the-hood of the back-propagation is still highly recommended to design effective learning algorithms (see Karpathy (2016) for a good set of reasons).

Stochastic gradient: the secret of the scalability of NN

Stochastic gradient descent (SGD) (Robbins & Monro, 1951) is a simple, yet very powerful procedure to reduce the computational cost to compute gradients of standard machine learning losses. Using previous notation of section 1.3, we denote with f the differentiable function with inputs $\boldsymbol{x}$ and θ, and $f_\theta(\boldsymbol{x})$ the output of that function. The (full) gradient of the negative log likelihood (NLL) loss of equation (1.1) with respect to its weights θ is:

$$\frac{\partial}{\partial\theta}\mathrm{NLL}(\theta;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial\theta}\mathrm{CE}\big(f_\theta(\boldsymbol{x}_i), y_i\big) \tag{1.2}$$

The stochastic gradient used in NN consists simply on sampling randomly B elements of $\mathcal{D}$ and computing the gradient of the loss function on that subset. Under very mild conditions, it can be shown that such gradient is an unbiased estimator of the gradient of equation (1.5.1) (see e.g. Bottou (1998)). Formally, this just means that, if we denote $\mathcal{D}_B$

[6] Although it can be easily extended to vector functions

as a random subset of size B from $\mathcal{D}$

$$\mathbb{E}_{\mathcal{D}_B\sim\mathcal{D}}\left[\frac{\partial}{\partial\theta}\text{NLL}(\theta;\mathcal{D}_B)\right]=\mathbb{E}_{\mathcal{D}_B\sim\mathcal{D}}\left[\frac{1}{B}\sum_{i=1}^{B}\frac{\partial}{\partial\theta}\text{CE}\big(f_\theta(\boldsymbol{x}_i),y_i\big)\right]=\frac{\partial}{\partial\theta}\text{NLL}(\theta;\mathcal{D})\quad(1.3)$$

Using the stochastic gradient instead of the full gradient massively reduces the amount of computation to obtain a sufficiently good estimation of the gradient: in eq. (1.5.1) we can see that with SGD we reduce the number of gradient evaluations from N to B. In modern ML problems, this is a huge gain since the dataset $\mathcal{D}$ could have in the order of $N=10^6$ samples, however, practitioners estimate gradients using at each step subsamples (aka batches) of size as little as $B=32$ (B is known as *batch size*). Algorithm 2 shows the pseudo-code of stochastic gradient descent:

Algorithm 2 Stochastic Gradient descent

```
$\theta_1 \leftarrow$ random
**for** $s \in 1...K$ **do**
    $\mathcal{D}_B \leftarrow \text{sample}(\mathcal{D},S)$        ▷ Sample $S$ elements from $\mathcal{D}$
    $\theta_{s+1} \leftarrow \theta_s - \gamma\frac{\partial}{\partial\theta}\text{NLL}(\theta_s;\mathcal{D}_B)$
**end for**
$\theta^\star \leftarrow \theta_K$
```

1.5.2 Multilayer perceptron

Multi-layer perceptrons (MLP) are the simplest NN models that we can encounter. Mathematically, they are functions that receive a d dimensional vector and apply a series of linear transformations intertwined with non-linear functions to produce an l dimensional output vector. The building blocks of MLP are fully connected layers (FullyConnected). FullyConnected layers consist of a matrix of weights $\boldsymbol{w}\in\mathbb{R}^{l\times d}$ and biases $\boldsymbol{b}\in\mathbb{R}^l$ followed by a non-linear per-item function σ.

$$\text{FullyConnected}(\boldsymbol{x})=\sigma\left(\boldsymbol{w}\boldsymbol{x}+b\right)$$

Before the deep learning revolution, the non-linear function used to be the tanh or sigmoid functions; however, nowadays it is the reLU function or its derivations (PreLU, LeakyReLU,..). The reLU function for an input $x\in\mathbb{R}$ is just $\text{reLU}(x)=\max(x,0)$. Following this notation, an MLP is thus a composition of FullyConnected layers:

$$\text{MLP}(\boldsymbol{x})=\left(\text{FullyConnected}_K\circ\ldots\circ\text{FullyConnected}_1\right)(\boldsymbol{x})\qquad(1.4)$$

In the MLP model we seek to learn all the weight and biases matrices of all the FullyConnected layers using back-propagation and the SGD algorithm. To be more explicit, in the cloud detection case formalized in section 1.3, the f_θ function of equations (1.1) and (1.5.1) is the MLP function defined before in equation (1.4)[7] and the θ parameters are all the weights and biases of all the neurons of the FullyConnected layers: $\theta=\{\boldsymbol{w}_1,\boldsymbol{b}_1,..,\boldsymbol{w}_K,\boldsymbol{b}_K\}$.

[7] Composed with a sigmoid function to output probabilities; i.e. $f_\theta(x)=(\text{sigmoid}\circ\text{MLP})(\boldsymbol{x})\in[0,1]$.

MLP are the models that we used in the so called ML classical approach of section 1.3.1, in particular, as described in that section, MLP are implemented operationally for detecting clouds in Proba-V (Gómez-Chova et al., 2017b) and in MERIS/AATSR (Gómez-Chova et al., 2013).

1.5.3 Convolutional neural networks

Convolutional Neural Networks (CNN) were proposed by LeCun et al. (1989) in the late eighties. However it wasn't until 2012 when the breakthrough of Krizhevsky et al. (2012) in the ImageNet image classification challenge (Russakovsky et al., 2015) kicked off the deep learning revolution and popularized them. The building blocks of CNN are *convolutional layers* which work in the image domain (i.e. its inputs are images) and learn filters that exploit the correlations of the spatial dimensions of images. The effectiveness of CNN in ML image problems is attributed to the hierarchical feature representations imposed by CNN architectures. These architectures seem to be good priors for vision systems. For instance, it is well known that the filters learned by convolutional layers are similar to those found in biological vision systems: LeCun et al. (1989) already pointed out that these filters resemble those described in Hubel & Wiesel (1962). Therefore, given their success with natural images, the RS community has also adopted CNN models to tackle ML problems with satellite images as we also do in this Thesis. Henceforth, to get some insights on the mechanics of these models, in the rest of this section we delve into the details of CNN.

The most important layer of CNN are convolutions. The convolution operation is actually a discrete spatial cross-correlation between an image of height H, width W, and C channels, $\boldsymbol{x} \in \mathbb{R}^{C\times H\times W}$, and some weights $\boldsymbol{\omega} \in \mathbb{R}^{C\times K\times K\times L}$ that produce an output image $\boldsymbol{z}$ with L channels. The value of this image for a pixel in (a,b) spatial position and channel l is given by the following equation:

$$\boldsymbol{z}[l,a,b] = \sum_{c=0}^{C-1}\sum_{i=0}^{K-1}\sum_{j=0}^{K-1} \boldsymbol{\omega}[c,i,j,l]\boldsymbol{x}[c,a+i,b+j] \tag{1.5}$$

We denote this operation as $\boldsymbol{z} = \boldsymbol{x} * \boldsymbol{\omega}$. An standard convolutional layer, Conv2d, consists of the convolutional operation plus a bias followed by a pixelwise non-linear function:

$$\text{Conv2d}(\boldsymbol{\omega}, b; \boldsymbol{x}) = \sigma\left(\boldsymbol{x} * \boldsymbol{\omega} + b\right)$$

The building blocks of CNN are these layers where $\boldsymbol{\omega}$ and b are learned weights. These weights are optimized following the stochastic gradient descent algorithm explained before. Modern deep learning libraries have highly optimized implementations of this operation and the derivative of this operation w.r.t. its inputs ($\boldsymbol{\omega}$, b and $\boldsymbol{x}$) (these derivatives are needed for running the back-propagation algorithm explained before). These fast implementations are one of the reasons of the success of deep learning; actually, part of the breakthrough of Krizhevsky et al. (2012) was due to the use of graphic cards for the implementation of convolutional layers.

In computer vision (CV), the image classification problem consists of assigning a class, or category, to an input image. The ImageNet (Russakovsky et al., 2015) challenge is perhaps the most famous image classification problem. In ImageNet, we have a training set of $1M$ images each of them with a given category among $1,000$ possible classes. CNN architectures used to solve this problem stack repeatedly Conv2d operations and pooling operations until they produce an output value. Specifically, if we call $\boldsymbol{x} \in \mathbb{R}^{C\times H\times W}$ to the input image, the output in an image classification CNN with L classes is an L dimensional vector of probabilities $\mathrm{CNN}_\theta(\boldsymbol{x}) \in [0,1]^L$; i.e. a vector where the l output is the probability of the image to belong to class l (with θ we denote the weights of all the Conv2d layers of the network). Note that, since the output of Conv2d layers and pooling layers are also images, at some point a global pooling operation is needed to get rid of the spatial dimensions (i.e. to convert a 3-D tensor $C_z \times H_z \times W_z$ to a 1-D tensor). In the latest state-of-the-art CNN architectures this step is tackled by a global average pooling layer:

$$\mathrm{GlobalAveragePooling}(\boldsymbol{z}) = \frac{1}{HW}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1} \boldsymbol{z}[:,i,j] \quad \in \mathbb{R}^C_z \tag{1.6}$$

After the GlobalAveragePooling step, the CNN becomes a MLP with fully connected layers which will eventually output the L dimensional vector with the estimated probabilities for each of the L classes. Hence, conceptually, CNN models could be described as an arbitrary concatenation of Conv2d and MaxPool operations followed by a GlobalAveragePooling and a MLP:

$$\mathrm{CNN}(\boldsymbol{x}) = (\mathrm{MLP}\circ\mathrm{GlobalAveragePooling}\circ\mathrm{Conv2d}\circ\mathrm{MaxPool}\circ\ldots\circ\mathrm{Conv2d}\circ\mathrm{MaxPool})\,(\boldsymbol{x})$$

By now, a couple of considerations remain for designing effective CNN. These are the number and order of the stack of operations of the network (aka architecture of the network) and the hyperparameters of the operations (e.g. number and size of the convolutional filters ω). In practice, in the field of RS, neural networks architectures are mostly inherited from successful computer vision applications; that is, architectures that obtain high accuracy in computer vision problems are used later on to solve RS problems. Indeed, most of the networks that we propose in this Thesis are adaptations of networks proposed for computer vision problems.

1.5.4 Fully Convolutional Neural Networks

In computer vision, semantic segmentation is the problem that seeks to classify every pixel in an input image. Note that, this is a fundamentally more difficult problem than image classification since the later only outputs a class for the whole image whereas in image segmentation we have to predict a class for every single pixel of the image. We can see that CD is a semantic segmentation problem where the number of output classes is two (clear and cloud)[8]. Therefore, the techniques used to address semantic segmentation can be used for cloud detection.

The success of CNN for image classification, followed its adaptation to other computer vision problems such as semantic segmentation. First naive adaptations of CNN to semantic segmentation consists of classifying the center pixel of the image. Specifically, first, we build a dataset of sampled patches from the image and assign the class of its center pixel as the category of that image. Afterwards, the image classification CNN is trained on

[8] It could be more than two if we want to e.g. discriminate between thin and thick clouds or also detect cloud shadows but in most parts of this Thesis we only consider two classes: clear and cloud.

this data. Finally, in order to obtain predictions for the whole image, the CNN model is slided over all the input image to obtain a dense per-pixel prediction. The work of Farabet et al. (2013) followed this approach and the CNN that we presented in Mateo-García et al. (2017) conference paper too. It is worth to mention that to our knowledge, Mateo-García et al. (2017) is the first approach of CNN to cloud detection in the literature. That work was the seed of the contribution of this Thesis Mateo-García et al. (2020b).

The problem of the so called naive approach described before is that the prediction step is computationally expensive. This is because we need to run the CNN model every time over every single pixel of the image. To make the point, for a $5,000 \times 5,000$ Proba-V image, the models that we trained in Mateo-García et al. (2017) had to be run on the $25M$ of pixels of the image; which took in the order of several hours to produce the cloud mask.

Fortunately, the works of Long et al. (2015) and Chen et al. (2015) came up with a simple solution to this problem. The key insight of their approach is to realize that the features that are extracted by a convolutional layer for computing the center pixel prediction can be reused to predict the nearby pixels. Hence, they suggest to remove the GlobalAveragePooling layer of a CNN and replace the fully connected layers of the MLP part with Conv2d layers with kernel size 1 ($K = 1$ in equation (1.5) which boils down to a pixelwise scalar product on the channels of the image). With those changes, the output of the network will be a tensor of $L \times H_{\boldsymbol{z}} \times W_{\boldsymbol{z}}$ where L is the number of outputs of the MLP part and $H_{\boldsymbol{z}}$ and $W_{\boldsymbol{z}}$ are the spatial sizes of the feature map $\boldsymbol{z}$ used as input to the GlobalAveragePooling layer (see equation (1.6)). Since all the operations in this new architecture are convolutions, this model is called Fully Convolutional Neural Networks (FCNN).

The first advantage of this model is that it is much faster for inference than the center pixel approach since with the same convolutional features $\boldsymbol{z}$ we obtain $H_{\boldsymbol{z}} \times W_{\boldsymbol{z}}$ predictions instead of just one. Additionally, since all operations are convolutions, the trained network could be applied to images of arbitrary sizes. Furthermore, another advantage of this model is that it can be trained *end-to-end* with the full ground truth mask y without the sampling described in the so called naive approach. To see this, let FCNN_θ be the FCNN model with the aforementioned replacements, if the spatial dimensions of $\boldsymbol{z}$ are the same as the dimensions of the input image $\boldsymbol{x}$ (i.e. $\boldsymbol{z} \in \mathbb{R}^{K \times H \times W}$), we could use the ground truth mask y to compute the loss in all those output pixels. For the binary classification task where $L = 1$, using the notation of eq. (1.1), this loss boils down to:

$$\text{NLL}(\text{FCNN}_\theta; \mathcal{D}_B) = \frac{1}{BHW} \sum_{b=0}^{B-1} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \text{CE}\big(\text{FCNN}_\theta(\boldsymbol{x}_b)[i,j], y_b[i,j]\big)$$

Note that, in this equation, the size of the tensors are $\boldsymbol{x}_b \in \mathbb{R}^{C \times H \times W}$, $\text{FCNN}_\theta(\boldsymbol{x}_b) \in [0,1]^{H \times W}$ and $y_b \in \{0,1\}^{H \times W}$. Unfortunately, the CNN used for image classification, the spatial sizes of the feature map $\boldsymbol{z}$ before the GlobalAveragePooling layer is usually lower than the size of the input image $\boldsymbol{x}$ because CNN networks use convolutions with *strides* and/or pooling operations to reduce the spatial size of the feature maps of the network (to save computational time and to learn long range dependencies between the pixels of the image). Therefore, in order to train with the ground truth mask y (of size $H \times W$), we should upsample these feature maps back to the $H \times W$ shape. For upsampling, Long et al. (2015) and Chen et al. (2015) propose to use fractionally strided convolutions (also named

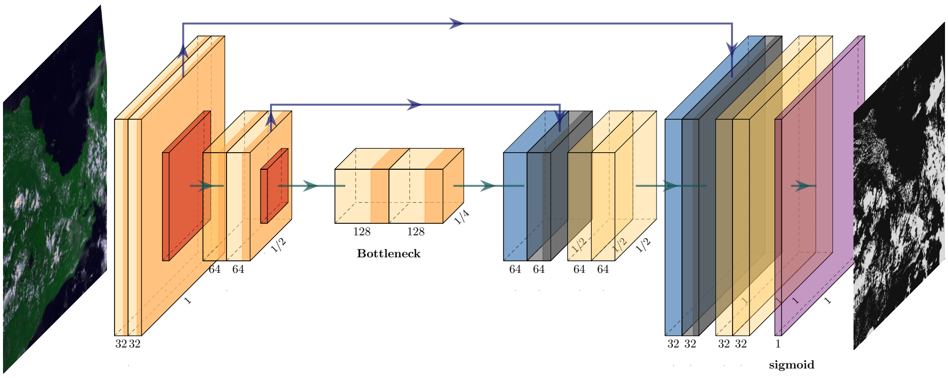


Figure 1.10: U-Net architecture used in works Mateo-García et al. (2020b,a)

deconvolutions or transpose convolutions). Other works propose simple bilinear or bicubic interpolation or pixel-shuffling (i.e. transposing channels to spatial dimensions).

Arguably, the most famous fully convolutional architecture for semantic segmentation is the U-Net. The U-Net architecture, originally proposed for medical imaging segmentation (Ronneberger et al., 2015), has been widely used in a plethora of RS applications (e.g. Schuegraf & Bittner (2019); Drönner et al. (2018); Kruitwagen et al. (2021)). In particular, after our works, it has also been used for cloud detection in Jeppesen et al. (2019) and Wieland et al. (2019) for Landsat-8. It has 5 pooling/unpooling stages and it adds skip connections between feature maps of the same resolution. Overall, the U-Net is conceptually simple yet accurate. In this Thesis we extensively used the U-Net architecture in works Mateo-García et al. (2020b,a); Mateo-Garcia et al. (2021). In particular, in contributions Mateo-García et al. (2020b,a) we used a simplified architecture shown in Figure 1.10 that reduces the number of pooling steps from five to two and which uses separable convolutions layers (Chollet, 2017). In Mateo-Garcia et al. (2021), we used the original U-Net but changing from the transpose convolutions to bilinear interpolation.

1.6 Transfer learning and Domain Adaptation

At this point, we have explained the models (FCNN) that we apply for cloud detection and we have also described or mentioned some of the large manually annotated datasets that we used to train and validate our models: the Biome dataset of Foga et al. (2017), the PV72 dataset explained in section 1.4.1, or the *WorldFloods* dataset of contribution Mateo-Garcia et al. (2021). With this data and these models very accurate cloud detection models can be trained that generalize to new image acquisitions of Proba-V (Mateo-García et al., 2020b), Landsat-8 (Jeppesen et al., 2019; López-Puigdollers et al., 2021), or Sentinel-2 (Mateo-Garcia et al., 2021), respectively. In our view, these models work well because they operate in the regime where DL models succeed: that is, when large amounts of accurate data are available for training the very complex models (with hundred of thousands of free parameters to estimate). However, the price to pay to build them is high: we have to label clouds in several different images which is hard, time-consuming, and it is not exempt of errors (see sec. 1.4.1). Additionally, every time a new sensor is launched, we will have to gather images and label manually their clouds again. Henceforth, one of the main goals of this Thesis is to explore strategies to transfer deep learning models to other sensors and to explore methodologies to alleviate their data requirements.

To this end, in this section, we briefly introduce the concepts of Transfer learning (TL) and Domain adaptation (DA) and their subtle differences. These concepts are very intertwined and sometimes they are used to refer similar things depending on the context; hence, in this Thesis, as we did in contribution Mateo-García et al. (2020b), we will follow the definitions of Pan & Yang (2010).

Let S be the source domain and T the target domain. In TL, we are interested in learning a model for T; however, data in T is scarce, hence, TL schemes seek to exploit data in a similar domain S to learn a better model for T. In the context of the contributions of this Thesis, the model is a CD model and the domains could be the images and manual labels of Landsat-8 (source domain S) and the images and labels of Proba-V (target domain T). Using mathematical notation similar to rest of this chapter, we use $\mathcal{D}_S = \{\boldsymbol{x}_i^S, y_i^S\}_{i=1}^N$ to denote the dataset of the source domain, $\boldsymbol{X}^S = \{\boldsymbol{x}_i^S\}_{i=1}^N$ to denote the dataset of only the inputs in the source domain, and $Y_S = \{y_i^S\}_{i=1}^N$ to denote the dataset of only the labels. Similarly, for the target domain $\mathcal{D}_T = \{\boldsymbol{x}_i^T, y_i^T\}_{i=1}^M$ represents the dataset of inputs and labels in the target domain and $\boldsymbol{X}^T = \{\boldsymbol{x}_i^T\}_{i=1}^M$ and $Y_T = \{y_i^T\}_{i=1}^M$ to only the inputs and the labels, respectively. In this setting, we assume that data in the source domain is much more abundant than in the target domain (i.e. $N >> M$); hence, the approach in TL is to train a model in the source data $\mathcal{D}_S$ in a manner that *works well* in T. TL violates one main assumption of ML which is that the distribution of the training data should be the same as the distribution of the test data. It does so in order to circumvent other limitation that is that training data must be a large enough representative sample of the distribution of the data. Hence, there are little theoretical guarantees of TL and, in practice, its effectiveness relies on the assumed similarity between the source and target domains.

The work of Pan & Yang (2010) categorizes transfer learning depending on different factors of the source and the target domains. One of these factors is the task aimed to solve; a transfer learning scheme is called *multi-task* if the problem to address in the source domain is different than in the target domain. One example of multi-task transfer learning is to use the weights of the models trained in ImageNet as starting point for other tasks. In RS instead of using ImageNet, the work of Neumann et al. (2020) suggests to use BigEarthNet Sumbul et al. (2019) or more recently its multilabeled version (Sumbul et al., 2021) as initial weights for multi-task TL. Conversely, in *single-task* TL the problem in the source and target domains is the same. The contributions of this Thesis, Mateo-García et al. (2020b) and Mateo-García et al. (2020a), are *single-task* TL where the task addressed is cloud detection in both domains.

A second categorization in Pan & Yang (2010) is based on the type of data available in the target domain. Here TL schemes are divided in *transductive transfer learning* and *inductive transfer learning*. In *transductive transfer learning*, we assume that there is no labeled data in the target domain at training time (i.e. $Y_T = \varnothing$). Transductive TL is called in other works *unsupervised domain adaptation* (Tuia et al., 2016; Ganin et al., 2016). In *inductive transfer learning*, it is assumed to have, in addition to the source data $\mathcal{D}_S$, some labeled data in the target domain. To highlight the differences, in *transductive transfer learning* we only have $\mathcal{D}_S$ and X^T to train a model whereas in *inductive transfer learning* we have $\mathcal{D}_S$ and $\mathcal{D}_T$.

In this context, the term Domain adaptation (DA) is used in this work to refer to the transformation functions applied mainly to the inputs in the source and target domains to

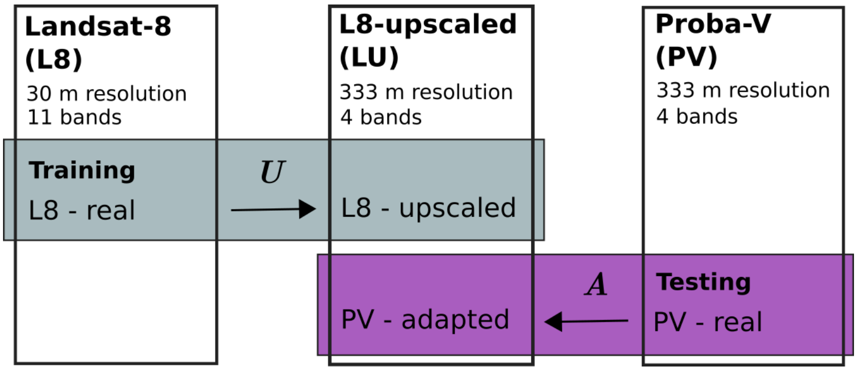


Figure 1.11: High level overview of the domain adaptation scheme of Mateo-García et al. (2020a): we use the physically based domain adaptation transformation (U, corresponding to the transformation of Fig. 3.1) to transform Landsat-8 images to the Landsat-8 Upscaled (LU) domain where images have the same spatial resolution and spectral bands as Proba-V. Afterwards, the learned transformation between Proba-V and Landsat-8 Upscaled (LU) $A = G_{\text{PV}\to\text{LU}}$ is applied to obtain Proba-V adapted images. These images have a spectral distribution and radiometry statistically similar to images in the LU domain.

transform them to the opposite domain. These transformations are needed for different reasons: one could be just to match the spatial resolution and the number of spectral bands between the domains. As an example, suppose that we have a classifier trained in the source domain; images in that domain have certain bands an spatial resolution, if we wish to apply such classifier to an image from another target sensor with other bands and spatial resolution, we need to select the spectral bands in the target image that correspond to the bands in the source domain and reproject the image to the source domain resolution. Notice that we assume here that the spectral bands used by the classifier in the source domain are compatible with the spectral bands of the target sensor (i.e. for each band used by the classifier we have a compatible band, or set of bands, in the target image). In Chapter 3, Fig. 3.1, we show a domain adaptation transformation based on the known properties of Proba-V and Landsat-8 that is used to match the spatial and spectral characteristics of Proba-V from a Landsat-8 image. Another reason to apply a domain adaptation transformation to an input image is to make it look *similar* to images in the other domain. Using the same example, when we have a classifier trained in data of the source domain, if images in the target domain are statistically different from those in the source training data, it is expected to have a drop in performance since we are no longer in the assumptions of learning theory to work (see section 1.4.1). This is often called data-shift problem (Torralba & Efros, 2011), i.e. training and testing distributions are different. In order to mitigate this, domain adaptation transformations are applied to (statistically) align source and target input distributions. The contribution of this Thesis Mateo-García et al. (2020a) proposes to learn such transformations using generative adversarial networks. In that work, we propose to use two domain adaptation transformations described in Fig. 1.11: one used for training, to match the spatio-spectral characteristics of Proba-V from Landsat (U transformation in Fig. 1.11 to go to the L8-upscaled domain) and, at test phase, A transformation is used to make Proba-V images statistically similar to L8-upscaled ones.

1.7 Research objectives

We propound that novel cloud detection models based on deep learning are significantly more accurate predictors of clouds specially in challenging situations. Nevertheless, these models, in order to truly excel, require large amounts of labeled training data; thus, transfer learning and domain adaptation methodologies are needed to overcome this limitation.

> *"The overarching goal of this Thesis is to improve cloud detection models by exploiting the spatio-temporal dimensions of the data and to propose methodologies to transfer those models to images acquired by other sensors."*

This goal is pursued by proposing novel multitemporal models that exploit computing platforms resources such as the GEE; proposing FCNN models that exploit the spatial and spectral correlations in satellite images to boost cloud detection accuracy; developing new methodologies to run these models in images from other satellite instruments; learning domain adaptation transformations to improve the performance of the trained models; and showing that these models can be deployed even onboard the target satellites.

Why is the topic important?

The constant growth in the number of optical sensors orbiting the Earth makes deploying a new CD mask for a particular sensor a more and more common task. Threshold-based CD approaches are highly tailored to the specifics of each sensor; this makes those algorithms brittle to small changes in input reflectance. This is a real necessity; for instance, the errors in cloud detection of the pre-launching CD mask of Proba-V hampered the usability of its data and the new ML-based CD model has not been ready until almost the end of the mission. Unfortunately, errors in threshold-based CD models developed in the commissioning phase of satellite missions are not new: with Envisat MERIS the ESA had also to provide an alternative to the CD mask with an ML-based one (Gómez-Chova et al., 2013) which also took several years to deploy. Similarly, the USGS changed the automatic cloud cover assessment (ACCA) (Scaramuzza et al., 2012) algorithm with FMask (Foga et al., 2017) two years after the launching of Landsat-8. Hence, we argue that, in order to expedite the development of CD models for future missions, we need to develop methodologies to reliably transfer models across similar instruments. Additionally, clear protocols must be established to fine-tune these models once they start retrieving images. We believe that this is a pressing need since now, with CubeSats, we are witnessing an explosion in deployed Earth observation orbiting devices, which needs reliable algorithms to produce useful Earth observation products.

How do we plan to address it?

In order to develop accurate cloud detection models, we propose data-driven solutions that take advantage of the open-access large archives of remote sensing optical images. We first propose (Chapter 2) multi-temporal models to exploit the temporal dimension of remote sensing data. We also propose FCNN models, which have shown to excel in remote sensing and computer vision applications. One advantage of FCNN is that they are patch-based models, i.e. its input is an image patch and its output is a patch with the same spatial size. This allows the model to exploit the spatial auto-correlations of images. We propose to transfer FCNN models across satellites. For this task, as a representative example, we use labeled data from Landsat-8 and Proba-V which have different spatial resolution, different radiometric quality and compatible spectral responses –as well as representative and large enough labeled datasets to validate the quality of the transferred models. We show how to transfer those models (Chapter 3) and how to improve the quality of transfer learning approaches using learning-based domain adaptation transformations (Chapter 4). Finally, we show that FCNN can also be used in specific RS applications for joint flood and cloud detection, and that these models can be deployed onboard satellites to reduce the latency to obtain RS products required by emergency response systems (Chapter 5).

1.8 Outline

The remainder of the Thesis is organized as follows:

Chapter 2 we present a simple yet powerful multi-temporal cloud detection model that is implemented in the Google Earth Engine platform. The model compares favorably against state-of-the-art single-scene threshold based approaches in Landsat-8 scenes.

Chapter 3 describes transfer learning of FCNN cloud detection models between Landsat-8 and Proba-V. It showcases results on transferring models in both directions (from Proba-V to Landsat-8 and from Landsat-8 to Proba-V).

Chapter 4 presents a learning-based domain adaptation transformation to statistically align Proba-V and Landsat-8 images. We show that this transformation produces Proba-V images with improved radiometry and produces a boost in cloud detection performance of the models presented in the previous chapter.

Chapter 5 covers a recent on onboard cloud and water segmentation in a real use-case of transductive transfer learning, proposed in Chapter 3, and highlights the potential of FCNN for onboard applications.

Chapter 6 summarizes the contributions of this Thesis, discusses the main conclusions, and provides a set of related projects, outcomes and other publications that resulted from the work performed during this PhD Thesis.

Chapter 7 provides a summary of the Thesis in Spanish.

2. Multitemporal Cloud Masking in the Google Earth Engine

2.1 Motivation of the work

As we introduced in the previous chapter, currently, most of operational missions rely on (over-simplistic) single-scene threshold-based methods to mask clouds. In the previous chapter, we showed some of the limitations of these algorithms and the necessity to improve them. In this first contribution of the Thesis, Mateo-García et al. (2018) [Appendix I], the goal was to understand the cloud detection problem and their limitations, the existing single-scene and multi-temporal cloud detection models and the methodologies and datasets to validate them. As a hands-on experiment, we decided to build on previous work of Gómez-Chova et al. (2017a) on multi-temporal cloud masking and implementing and validating their methodology on global, recently published, manually labeled cloud detection datasets for Landsat-8 (Foga et al., 2017). In the course of this work, we improved and expanded the methodology of Gómez-Chova et al. (2017a), implemented the model in the Google Earth Engine and demonstrated on the validation data state-of-the-art cloud detection performance. All these results were presented in what is now the journal publication Mateo-García et al. (2018)[Appendix I].

The choice to implement the multi-temporal cloud detection scheme (Fig. 2.1) in the Google Earth Engine platform was initially to fulfill the requirements of the Google Earth Engine Award project *Cloud detection in the cloud* granted to Prof. Luis Gómez-Chova. Nevertheless, this proved to be an excellent choice since one of the main limitations of multi-temporal models is the access to the catalog of images over a given location: without the Google Earth Engine the existing approaches when we did this work were limited to download full Landsat-8 acquisitions from the USGS Earth Explorer catalog. On the other hand, using the Google Earth Engine, no data downloading is needed and the processing to get the cloud mask is limited to the area of interest; additionally, the process runs on the Google servers. The result is a multi-temporal CD algorithm that could be used operationally without the computational constrains of other multi-temporal methods.

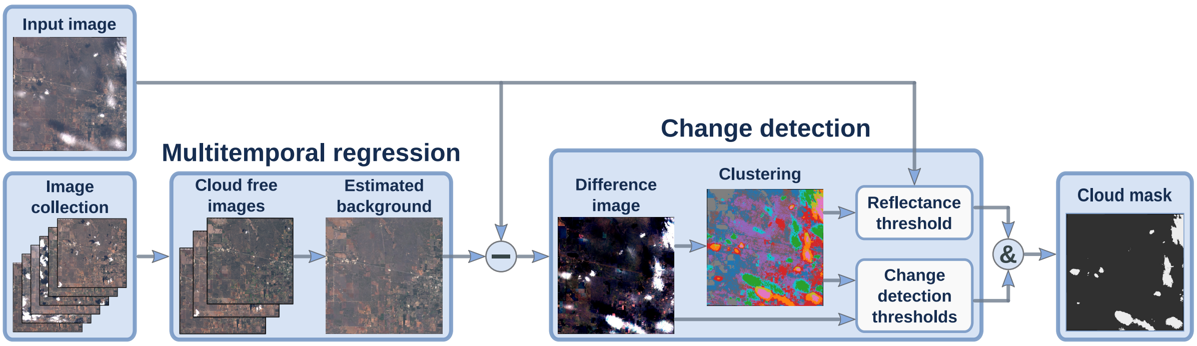


Figure 2.1: Proposed multi-temporal cloud detection scheme: a set of images with low cloud cover is filtered to estimate the expected reflectance of the surface (estimated background). A pixelwise difference between the input image and the background is used as input to the change detection module. In this module, we cluster the pixels of the difference image and apply two thresholds to the clustered values. Cluster values above these two thresholds are marked as clouds.

2.2 Summary and main results

In the contribution of this Thesis, Mateo-García et al. (2018), we propound a lightweight multi-temporal cloud detection algorithm for Landsat-8 that is validated in the Google Earth Engine using a large dataset of manually labeled clouds. This algorithm is based on two steps (see Fig. 2.1). In the first step, called multi-temporal regression, we estimate the reflectance of the surface using previous (almost) cloud free images over the area of interest (AoI). These images are filtered using a simple single-scene CD algorithm over the AoI [1]. In order to estimate the reflectance of the surface, we tested four different methods: on the one hand, linear and kernel ridge regressions as proposed in Gómez-Chova et al. (2017a), on the other hand, simpler approaches such as the pixelwise percentile of the cloud free scenes or the previous cloud free image (persistence). For the second step, once we have an estimation of the background (see Fig. 2.1), the cloudy input image and the difference in reflectance of the image with the background is fed to the change detection module. This module first clusters the difference image and then, for each cluster three simple values or features are calculated. These features are: α, the averaged difference in norm over the RGB channels of the Landsat-8 image, β the average difference over the same channels, and γ the mean brightness of them. Finally, we used a simple decision rule based on thresholds over those features to produce a binary cloud mask.

The rationale of the proposed method is that the reflectance of the surface usually changes slowly over time and abrupt changes are mostly caused by clouds specially if the brightness of those pixels is high. It is worth noting that the proposed methodology is very similar to many other works, e.g. Hagolle et al. (2010); Zhu & Woodcock (2014); Frantz et al. (2015); Candra et al. (2017), which are also based on a very similar two-step methodology: estimation of surface reflectance plus thresholding of the differences. Nevertheless, the differentiating factors of our approach are that the models are adapted to run operationally using the GEE platform and that the models are validated over a large amount of different scenes (see Figure 2.2). None of the previous multi-temporal approaches were shown to work over such a large variety of locations and biomes.

[1] In this work we used the ACCA algorithm (Scaramuzza et al., 2012) which was formerly present in the BQA mask of Landsat-8 TOA scenes, but it can be replaced by any other single-scene cloud mask.

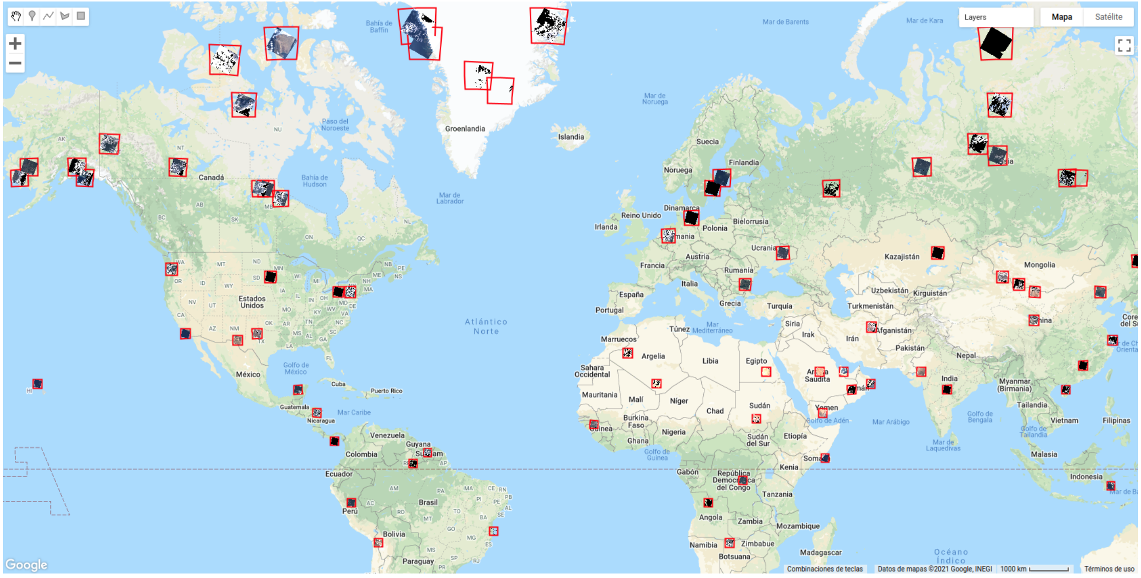


Figure 2.2: Biome dataset of Foga et al. (2017) ingested in the Google Earth Engine to validate the proposed multi-temporal cloud detection algorithm. This dataset can be browsed at: https://code.earthengine.google.com/f5ff4b932dbfcdbe242b74938694a9c1

-

After a thoughtful reflection, one of the most illuminating results of this work is shown in Figure 2.3. Here we show in the left and in the center the Receiver operator curve (ROC) which shows the trade-off in commission and omission errors as we change the threshold in the difference on reflectance (α). We show with a cross the commission and omission errors of our methods with the selected threshold and of the methods used as a baseline: FMask (Zhu & Woodcock, 2012) and ACCA (Scaramuzza et al., 2012). On the right of Figure 2.3, we show the same data on a different dimension, here we show the difference in radiance (α in x-axis) and the brightness (γ in y-axis) for each of the clusters in all the validation database. We show in blue the clusters where most of the pixels are clear and in orange the cloudy ones. In this figure, we can see how both dimensions help to discriminate cloudy pixels and how the selected thresholds are well aligned with the task aiming to solve (the dotted blue lines). For further details the journal publication Mateo-García et al. (2018) is included in Appendix I of this Thesis.

2.3 Reproducibility

The proposed implementation as well as the benchmarking code is open-source and reasonably well documented in a GitHub repository https://github.com/IPL-UV/ee_ipl_uv. In addition to the package with the multi-temporal cloud detection models, the aforementioned repository contains a set of notebook tutorials that can be run directly on Google Colab. In this tutorials, we cover different use-cases and details of the methodology:

- There are ready-to-use examples of the proposed cloud detection scheme for a given Landsat-8 scene.
- There are also ready-to-use examples for Sentinel-2. Although the original publication did not cover Sentinel-2, the method can be easily extended to this sensor. This extension has been implemented and it can be checked in one of the tutorials.

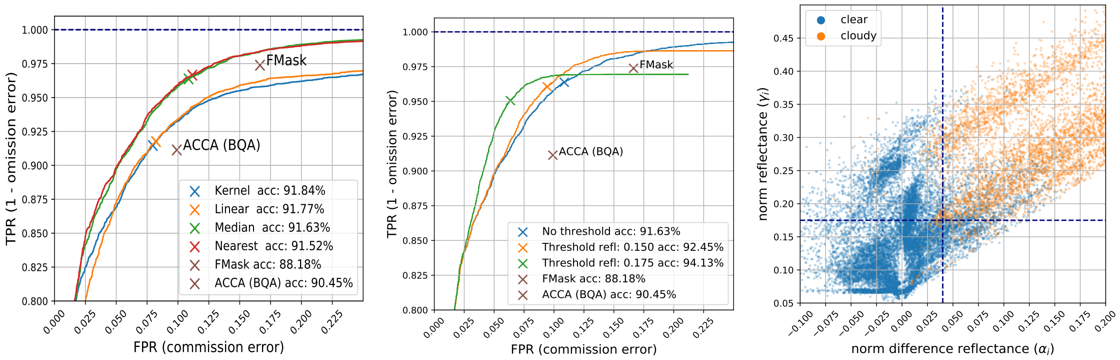


Figure 2.3: Left and center: Receiver operator curve of the proposed detection models. This curve shows the trade-offs in commission and omission errors when we change the threshold in the difference of reflectance (α) for different background estimation methods (left) and for different thresholds in brightness (γ) (right). The crosses correspond to the selected thresholds for each of the methods and for the single scene baselines (FMask and ACCA). Right: Scatter plot of the clusters over all the validation scenes. In the y-axis we show the norm of TOA reflectance of the visible bands (γ) and on the x-axis the norm of difference in reflectance (α) (see Mateo-García et al. (2018) for further details).

- There are detailed explanations of the different background estimation methods and the different configurations for the clustering and thresholding procedure.

The Biome dataset was ingested on the GEE platform and can be used by other users, e.g. in the script https://code.earthengine.google.com/f5ff4b932dbfcdbe242b74938694a9c1. We also developed a viewer to show the output of our model compared with FMask in the considered locations: https://isp.uv.es/projects/cdc/viewer_l8_GEE.html.

We believe that the effort carried out to document and maintain all these sources is one of the main reasons of the relatively good metrics of the article (69 citations according to Google Scholar, 2022/01/10). Thanks to this we have received several queries for users which show interest in the work and highlight the necessity of more accurate cloud detection models for Landsat-8 and Sentinel-2.

3. Transferring deep learning models for CD between Landsat-8 and Proba-V

3.1 Motivation of the work

Proba-V is a small one cubic meter satellite launched in May 2013 which main goal has been to provide continuity to Envisat/MERIS and SPOT-5 observations until the launch of Sentinel-3. Proba-V has three cameras, one pointing at nadir and the other two on its sides. Each of these instruments is a pushbroom sensor (see sec 1.1) which measures radiance in four spectral bands whose wavelengths are depicted in Fig. 1.3. The limited amount of spectral information of Proba-V makes detecting clouds specially challenging. As we covered in the introduction, the errors in Proba-V cloud detection led to the development of a dedicated study funded by ESA, the Proba-V Round Robin experiment, with the aim of improving cloud detection for Proba-V. We participated in this study and in its continuation for the operational implementation of the proposed cloud detection algorithm (Proba-V Collection C2). In order to build an accurate ML based model for Proba-V, we created the PV72 dataset, which has manually labeled cloud masks for 72 Proba-V acquisitions (see sec. 1.4.1). Creating this dataset made us aware of the huge amount of time and dedication that is needed to produce an accurately labeled and diverse dataset (we estimated at least three months of dedicated work see sec. 1.4.1). Henceforth, in this second contribution of the Thesis, Mateo-García et al. (2020b) [Appendix II], we decided to explore transfer learning to find out if models trained on data from other satellites would also work in Proba-V and to understand the trade-offs of using such models. Demonstrating that models trained on data from other sensors work well for a new instrument defines a clear path to develop ML based models for those new instruments.

3.2 Summary and main results

The main goal of this contribution is to demonstrate inductive and transductive transfer learning for detecting clouds in Proba-V (see sec. 1.6 for definition of inductive and transductive TL). To this end, we chose Landsat-8 as a suitable source domain since Landsat-8 has higher spatial resolution and radiometric quality than Proba-V and, additionally, there

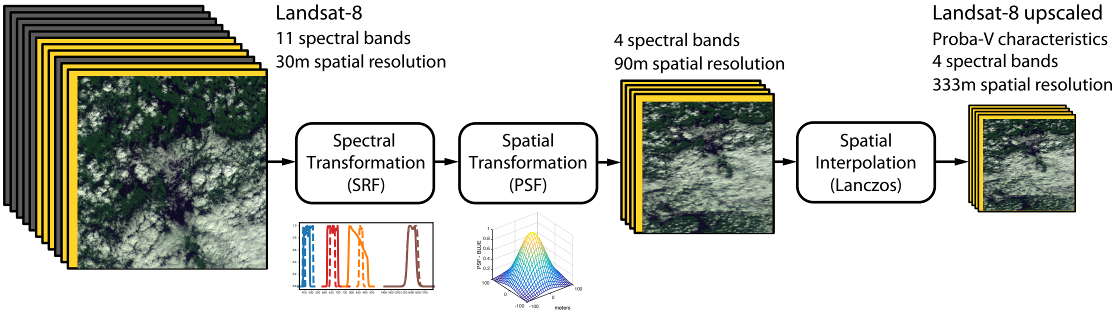


Figure 3.1: Sensor-based domain adaptation transformation applied to Landsat-8 images to resemble the Proba-V instrument characteristics.

exist several manually labeled CD datasets for Landsat-8. In this work, we used the L8Biome (Foga et al., 2017) and L8SPARCS (Hughes & Hayes, 2014) datasets. In order to adapt Landsat-8 images to the spatio-spectral characteristics of Proba-V, we used a transformation based on the specifications of the sensors (see Fig. 3.1). Afterwards, we designed a set of experiments to benchmark inductive and transductive transfer learning of FCNN for CD. Although the original goal was to investigate TL using Landsat-8 and Proba-V as the source and target domains, respectively, we ended up testing TL in both directions: from Proba-V to Landsat-8 and from Landsat-8 to Proba-V. For training the FCNN models, we used the L8Biome and the training split of the PV72 dataset (PV48). For testing the models, we used the L8SPARCS dataset and the L8Biome for the Landsat-8 domain and the testing split of the PV72 (PV24) for the Proba-V. We compared these models with the operational cloud detection models of Landsat-8 (FMask (Zhu & Woodcock, 2012)) and Proba-V (Toté et al., 2018), and with other deep learning based models for Landsat-8 (Jeppesen et al. (2019) and Li et al. (2019)). For this Thesis, we also included the comparison with our new operational Proba-V algorithm of collection C2.

Most of the experiments of this article are focused on transductive transfer learning. Specifically, we tested transductive transfer learning from Proba-V to Landsat-8 and from Landsat-8 to Proba-V. As explained in the introduction (Chapter 1.6), in transductive transfer learning, we assume there is no labeled data available from the target domain during the training. Hence, the transductive TL experiments of this work use data only from Landsat-8 (Proba-V) for training and labelled Proba-V (Landsat-8) data is only used to evaluate the models (resp.). Table 3.1 shows a comprehensive set of results of the proposed transfer learning models compared with the same models trained with data of the same domain and many other different models of the literature. Results in this table are grouped based on the test set where models are evaluated. The first group shows the results of the models tested on the Proba-V domain. We compare the FCNN models trained on the L8Biome dataset using the proposed spatio-spectral transformation $\text{TL}_{\text{L8},333}$ against the models using only the spectral transform $\text{TL}_{\text{L8},30}$ (i.e. without upsampling to the Proba-V spatial resolution). We see that the former ($\text{TL}_{\text{L8},333}$) performs significantly better than the later (much higher accuracy and lower omission errors). Comparing this model ($\text{TL}_{\text{L8},333}$) with the same architecture but trained on the Proba-V domain ($\text{TL}_{\text{PV},333}$), we see that there is a significant boost in accuracy for the models trained with real Proba-V data (4-5 points). This shows that training in Landsat-8 adapted images is still not optimal; i.e. Proba-V and Landsat-8 adapted images are still different even though we transformed Landsat-8 images according to the physical specifications of the sensors. In the third contribution of this Thesis, Mateo-García et al. (2020a), we propose a learning based domain adaptation to reduce this gap. It is worth noting that the ranges shown in the table correspond to models trained with different random seeds. This dependency is often neglected in most deep learning studies; nevertheless we see that there are significant discrepancies specially when we test on a different domain (2 points) and around 0.3 if testing in the same domain. Comparing with the operational model of Collection 1 (Toté et al., 2018), the model transferred from Landsat-8 is significantly better (+5 points in accuracy and half commission errors). This demonstrates that transferred models could be more accurate than threshold based approaches and that they can be safely used at the commissioning phase of the mission when no data from the satellite is available. Additionally, in this Thesis, we have included the results of the model presented in the Round Robin experiment (Gómez-Chova et al., 2017b) and the model developed for collection C2. Those models follow the ML classical approach (see sec.1.3.1 of introduction chapter). In this case, we see that deep learning models trained in Proba-V ($\text{TL}_{\text{PV},333}$) still perform slightly better than the highly tuned models that we developed with the ML classical approach for collection C2. The inference time of the proposed architecture (shown in Figure 1.10) is very similar to the C2 model and even faster if using a GPU.

Table 3.1: Table with results over the different test sets of the transductive transfer learning models and selected models of the literature. Ranges show minimum and maximum values obtained in 10 runs changing the random seed value for the training of the network.

Model	Train Set	Test Set	Commission Error%	Omission Error%	Overall Accuracy%	F_1 score%
$\text{TL}_{\text{L8},333}$	L8Biome	PV24	5.10 - 12.18	8.42 - 14.63	88.84 - 91.87	87.20 - 90.69
$\text{TL}_{\text{L8},30}$	L8Biome	PV24	5.00	38.23	80.37	73.48
$\text{TL}_{\text{PV},333}$	PV48	PV24	4.32 - 5.61	4.66 - 6.01	94.81 - 95.10	94.14 - 94.43
Oper. PV C1 (Toté et al., 2018)	-	PV24	25.86	5.70	83.01	83.00
Round Robin (Gómez-Chova et al., 2017b)	PV48	PV24	5.57	8.69	93.08	91.96
Oper. PV C2	PV48	PV24	5.72	4.77	94.69	93.95
$\text{TL}_{\text{L8},333}$	L8Biome (73)	L8Biome (19)	6.78	7.67	92.90	93.11
$\text{TL}_{\text{L8},30}$	L8Biome (73)	L8Biome (19)	6.63	5.58	93.92	94.17
$\text{TL}_{\text{PV},333}$	PV48	L8Biome (19)	7.32 - 10.5	6.83 - 9.79	90.85 - 91.89	91.11 - 92.22
FMask (Foga et al., 2017)	-	L8Biome (19)	13.18	6.99	89.59	89.3
MSCFF (Li et al., 2019) (all bands)	L8Biome (73)	L8Biome (19)	4.16	6.07	94.96	94.5
MSCFF (Li et al., 2019) (NRGB)	L8Biome (73)	L8Biome (19)	6.35	5.48	93.94	92.6
$\text{TL}_{\text{PV},333}$	PV48	L8Biome	10.99 - 17.13	6.01 - 10.55	87.79 - 89.77	87.95 - 89.71
FMask (Foga et al., 2017)	-	L8Biome	-	9.69	88.48	85.03
RS-Net (Jeppesen et al., 2019)	L8SPARCS	L8Biome	-	5.51	91.59	91.52
$\text{TL}_{\text{L8},333}$	L8Biome	L8SPARCS	1.16 - 1.86	36.34 - 37.82	91.25 - 91.81	73.48 - 74.73
$\text{TL}_{\text{L8},30}$	L8Biome	L8SPARCS	1.24	29.91	93.20	79.98
$\text{TL}_{\text{PV},333}$	PV48	L8SPARCS	1.05 - 3.26	33.08 - 40.84	90.93 - 92.14	71.68 - 76.27
FMask (Foga et al., 2017)	-	L8SPARCS	6.03	13.79	92.47	81.61
RS-Net (Jeppesen et al., 2019)	L8Biome	L8SPARCS	2.19	27.66	93.26	80.62

The rest of the groups of Table 3.1 show the results of the models tested in the Landsat-8 domain. The first of those groups show models tested on the 19 scenes of the L8Biome used in the split proposed in the work of Li et al. (2019). These show the model trained in Proba-V ($\mathrm{TL}_{\mathrm{PV},333}$) and evaluated in the Landsat-8 domain using this split. To perform the comparison in the Landsat-8 domain, we need to resample the predicted cloud mask to the Landsat-8 30m resolution (see the journal paper Mateo-García et al. (2020b) included in Appendix II for further details on the training and testing schemes). In this case, we also see that the accuracy of the models transferred from Proba-V have high accuracy, which is slightly higher than threshold based approaches for Landsat-8 (FMask). In this case, we see a smaller gap between the transfer model and the models trained on data of the same domain ($\mathrm{TL}_{\mathrm{L8},333}$ and $\mathrm{TL}_{\mathrm{L8},30}$; 2-3 points difference respectively); additionally, we show the results of the models trained by Li et al. (2019) (called MSCFF) which use all the Landsat-8 bands at its nominal spatial resolution. The last two groups show models tested on the full L8Biome and L8SPARCS datasets, respectively. Again, we reach similar conclusions: the model transferred from Proba-V ($\mathrm{TL}_{\mathrm{PV},333}$) performs as good as FMask (slightly better in L8Biome and slightly worse in L8SPARCS) and deep learning approaches trained on data from the same domain obtain the highest accuracy. For further details, the reader is recommended to go over the published contribution Mateo-García et al. (2020b), which is included in Appendix II.

The main conclusion extracted from the transductive transfer learning results is that FCNN CD models trained on data from a different (but related) sensor work better or on-par with highly-tuned threshold based models. In the case of Proba-V, this is expected since threshold-based cloud detection for Proba-V is very challenging due to the limited amount of spectral information. In the case of Landsat-8, the results are surprising, since FMask (Zhu & Woodcock, 2012) is a very-well established algorithm with several improvements proposed over the years to adapt to the characteristics of the Landsat-8 imagery. The takeaway message, in our opinion, is the same in both cases: FCNN CD models could be trained on data from a similar sensor and deployed at the commissioning phase of the mission.

In this contribution we also explored inductive transfer learning, in this case using Landsat-8 as the source domain and Proba-V as the target. In inductive TL, we want to demonstrate that using few labeled data from the target domain together with data from the source domain we can boost the accuracy of transductive transfer learning models. For this, we made 8 balanced subsets of the training data each of those with 6 images. For each of those subsets, we trained models with consecutive numbers of Proba-V images. We trained models using only the target data (Proba-V images) and jointly on the source (Landsat-8 adapted) and target data. Figure 3.2 shows the results of this experiment. For each number of Proba-V images used for training (x-axis), we show in the y-axis the accuracy over the PV24 dataset. For each element in the x-axis, we show on the left the accuracy of models trained only on Proba-V data and on the right the models trained on that data together with adapted data from the L8Biome dataset. We see that models trained jointly consistently outperform models trained only in Proba-V. Additionally, we see that by using only three or four labeled images from Proba-V we already outperform transductive models (models trained only on Landsat-8). With 6 images or more we observed that the effect of joint training is negligible and it is sufficient with training with only data from the target domain (Proba-V).

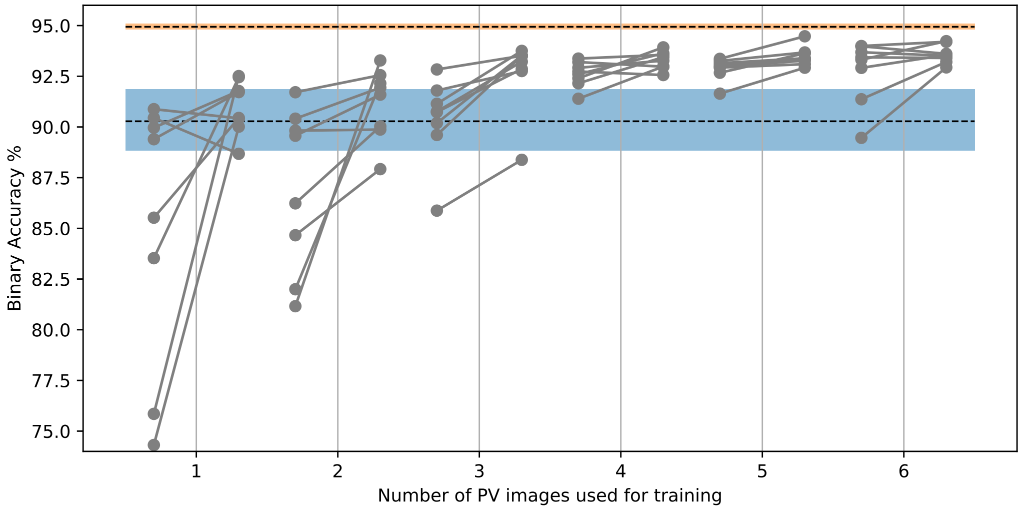


Figure 3.2: Test accuracy of models trained using different number of Proba-V images. For each value, on the left, only Proba-V data is used; on the right, models trained jointly on Landsat-8 and Proba-V data. Blue shaded area depicts the accuracy of the models trained only in the L8Biome dataset. Orange area depicts the accuracy of models trained on all Proba-V images (PV48).

To conclude, in this work we demonstrated inductive and transductive transfer learning of cloud detection models between Landsat-8 and Proba-V. We show for the first time that it is possible to transfer CD models between sensors of significantly different spatial resolutions (30 m Landsat-8 and 333 m Proba-V) and compatible spectral responses. Moreover, such models are as good as state-of-the-art threshold-based CD models. We believe this work paves the way for developing CD for forthcoming satellites; in particular, in Chapter 5 we will show an example of transfer learning that has been deployed onboard a new CubeSat satellite.

4. Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images

4.1 Motivation of the work

The transfer learning results of the previous chapter show a good performance of the models trained only on data from the source domain (transductive transfer learning). Nevertheless, in the case of using Landsat-8 as the source and Proba-V as the target, if we look back at Table 3.1, we see that there is still a significant gap between models trained on target data (Proba-V data from the PV48 split) and the transferred models. This gap is around almost 5 points in accuracy and between 4 and 10 points in omission error. A closer look at transformed Landsat-8 images comparing them with real Proba-V ones show that colors and texture on those images are different; even when images are retrieved on the same location and with less than one hour between acquisitions (see two first columns of Fig. 4.3 or first image of Mateo-García et al. (2020a)). In particular, we see that Proba-V images are more blueish and more noisy (lower radiometric quality) than Landsat-8 ones.

In order to inspect further this discrepancy, we downloaded the closest in time Proba-V image overlapping each scene in the L8Biome dataset. We found 65 Proba-V images from the same date as the Landsat-8 images in the L8Biome dataset. For those, the time difference between Landsat-8 and Proba-V acquisitions is on average 50 minutes with a minimum and maximum difference of 2 and 200 minutes, respectively. Using those images, we computed the histograms of reflectances of the four overlapping bands of Proba-V and Landsat-8. We found those histograms significantly different even when we used the upscaling transformation of Fig. 3.1, which takes into account the PSF and SRF of Landsat-8 and Proba-V (Mateo-García et al., 2019).

Henceforth, in this third contribution of the Thesis, Mateo-García et al. (2020a) [Appendix III], we propose to learn a domain adaptation transformation to make Proba-V images similar to upscaled Landsat-8 ones. With this, we seek to bridge the gap in performance of the cloud detection models transferred from Landsat-8 to Proba-V. Unfortunately, learning such transformation cannot be done in a paired supervised way. If we look again at the images in Fig. 4.3, we see that even though those images are from the same location and close in time, clouds have moved significantly. In next section, we detail how to

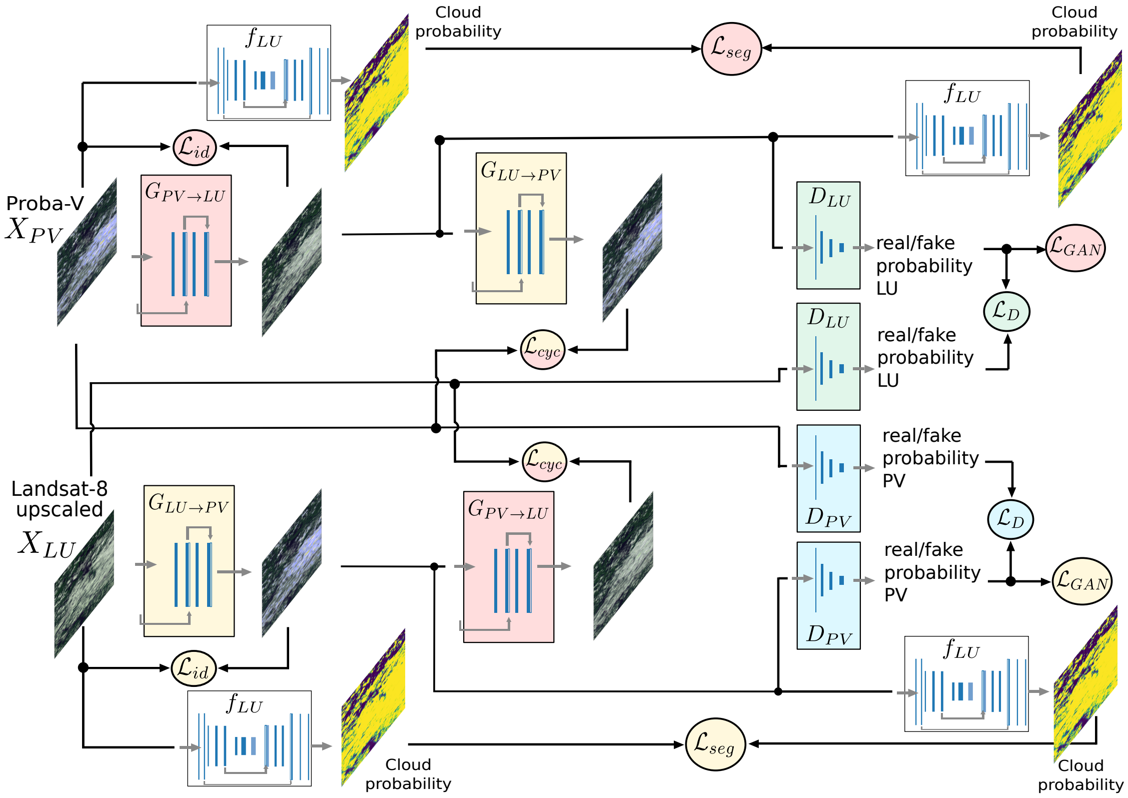


Figure 4.1: Scheme of the forward passes for the training procedure of the proposed cycle consistent adversarial domain adaptation method. The four networks ($G_{\mathrm{PV}\to\mathrm{LU}}$, $G_{\mathrm{LU}\to\mathrm{PV}}$, D_{PV}, D_{LU}) have a different color. Losses are depicted with circles and their fill color corresponds to the color of the network that they penalize.

tackle this problematic scenario with cycle consistent generative adversarial networks (Cycle-GANs).

4.2 Summary and main results

Obtaining simultaneous co-located images over the same area it is not always possible for most optical sensors onboard different satellites. For Proba-V and Landsat-8, images acquired the same day are common since Proba-V revisit time is between 1 to 2 days due to the high swath obtained when combining its three cameras. Nevertheless, the differences in time between those same-day Landsat-8 and Proba-V images is still high enough to observe large displacements in clouds between the two acquisitions. In this work, we propose Cycle-consistent generative adversarial networks to learn DA transformations between Proba-V images and Landsat-8 images upscaled to the Proba-V resolution (denoted as Landsat Upscaled, LU), which are obtained with the transformation shown in Fig. 3.1. The proposed scheme is *unpaired*, i.e. it does not require simultaneous co-located images to learn such transformation. Additionally, we propose some specific penalties to the loss function in order to preserve the calibrated input values of input images, i.e. to avoid hallucination artifacts common in GANs.

The procedure used to train the DA transformations (denoted as $G_{\mathrm{PV}\to\mathrm{LU}}$ $G_{\mathrm{LU}\to\mathrm{PV}}$) is shown in Figure 4.1. This scheme is based on CyCADA (Hoffman et al., 2018) which

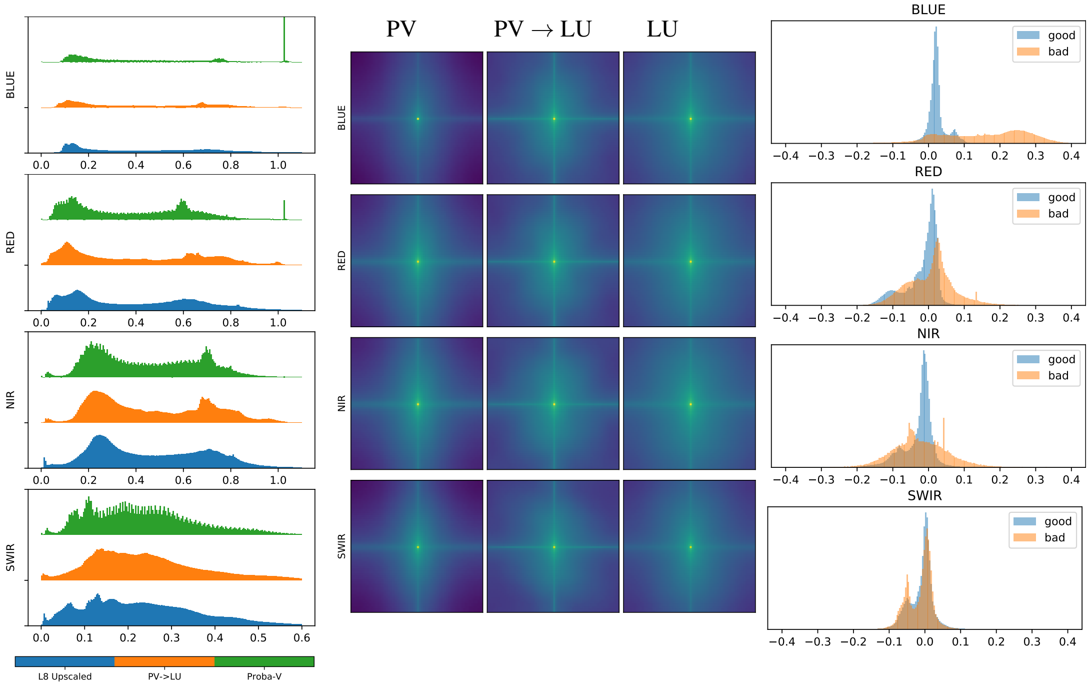


Figure 4.2: Left: TOA refectance distribution on each of the spectral bands for Proba-V (green), Proba-V images transformed using the proposed DA method (orange) and pseudo-simultaneous Landsat-8 Upscaled images (blue). Center: 2D Fourier transform in dB for each of the four spectral channels averaged across all patches of 64 × 64 pixels. Right: Differences in TOA reflectance for Proba-V images before and after applying the proposed DA transformation ($X_{\mathrm{PV}} - G_{\mathrm{PV}\to\mathrm{LU}}(X_{\mathrm{PV}})$) stratified by the quality indicator of Proba-V QA band. Values in all figures measured across all image patches in the *38-Clouds pseudo-simultaneous dataset*

proposes a simultaneous adaptation between both domains with different losses. In our approach, which can be explored in greater detail in the presented publication in Appendix III, we include adversarial losses ($\mathcal{L}_{GAN}$) learned through domain discriminators (D_{LU} and D_{PV}) which are trained simultaneously and we also include the cycle-consistency losses ($\mathcal{L}_{cyc}$) as in CyCADA (Hoffman et al., 2018). Additionally, we include an *identity consistency loss* $\mathcal{L}_{id}$ which seeks to avoid large changes in input values. This is used, as mentioned before, to preserve the calibrated top of atmosphere (TOA) values of Proba-V and Landsat-8 images since those adapted images might be further used for other remote sensing applications that require calibrated values. Additionally, as an optional loss, specific for our CD application, we include a *segmentation consistency loss* as in CyCADA. This loss slightly improves the performance on the cloud detection task although it could be excluded if we prefer to be task agnostic (see Table III of the publication for an ablation study on the different losses).

Using the aforementioned scheme, we trained the DA transformations $G_{\mathrm{PV}\to\mathrm{LU}}$ and $G_{\mathrm{LU}\to\mathrm{PV}}$ using images from the *L8Biome pseudo-simultaneous dataset* which contains around 38,000 pairs of close in time subimages of Proba-V and Landsat Upscaled of size 64 × 64. Afterwards, we test the performance of the models over the *38-Clouds pseudo-simultaneous dataset* which contains Landsat Upscaled and Proba-V images over 35 different acquisitions not used for training (images are from different years and different

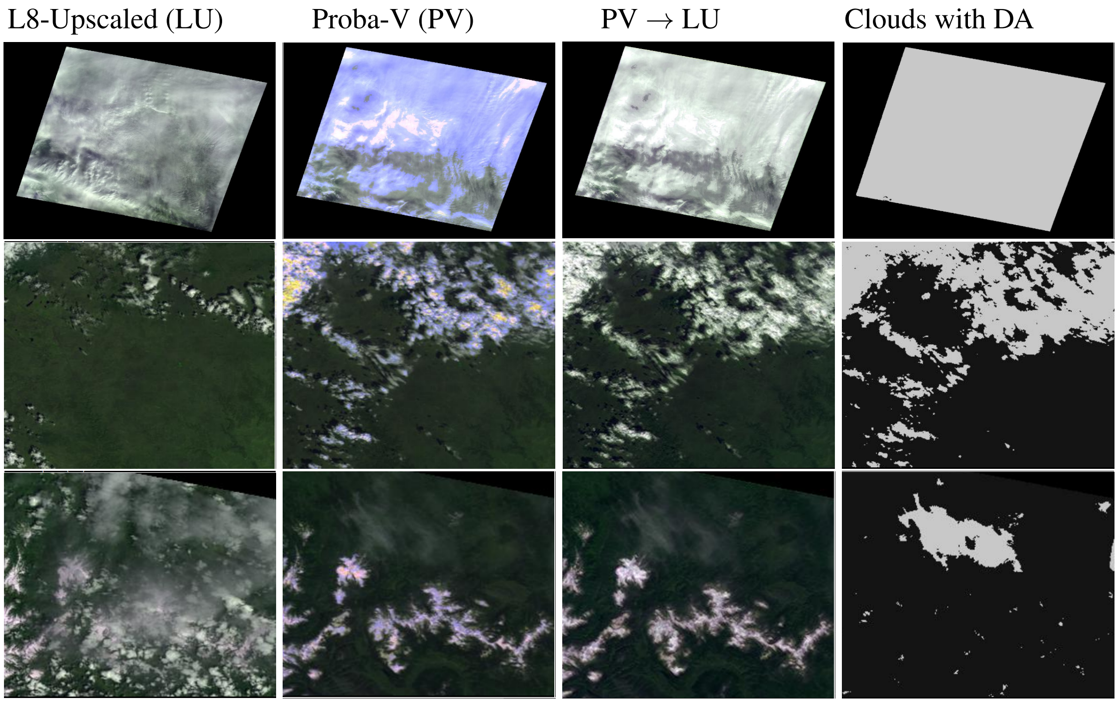

L8-Upscaled (LU) Proba-V (PV) PV → LU Clouds with DA

Figure 4.3: From left to right: Pseudo-simultaneous Landsat-8 Upscaled (LU) image, Proba-V image, Proba-V as LU, Clouds from Proba-V as Landsat-8 upscaled. See https://isp.uv.es/projects/cloudsat/pvl8dagans for more examples.

locations). Figure 4.2 shows the distribution of the reflectances in this dataset with and without the proposed DA transformation (Proba-V to Landsat upscaled, $G_{\text{PV}\to\text{LU}}$). The figure in the right shows in green the distribution of Proba-V reflectances for each of the four common bands in Proba-V and Landsat-8, in blue we show the distribution of Landsat-8 Upscaled images, and in orange the distribution after applying the DA transformation to Proba-V images. Here, we see that the DA transformation is successful at its task of making the adapted images (orange) statistically similar to the Landsat radiance (blue); in particular, it is quite remarkable that the transformation is able to smooth out the saturated values of Proba-V (the spikes in the largest TOA reflectances which are quite high in the blue and red channels). In the middle, we show the 2D Fourier transform for each of the four bands. In this case, it is also clear that Landsat-8 Upscaled images and the adapted images (PV → LU center column) have more high frequency components than Proba-V images (Proba-V values in the corner are darker). This shows that the transformation is adding high-frequency components in order to make Proba-V images more similar to Landsat-8 Upscaled ones. Finally, in the right hand side of Fig. 4.2, we show the difference in radiance between Proba-V and DA adapted data. This difference is the one that is penalized with the proposed identity loss ($\mathcal{L}_{id}$, which uses the ℓ_1 norm of this difference). Pixel differences used to compute these histograms are stratified by the quality indicator included in the Proba-V quality assessment band (BQA). In this case, what we see is that even though the network did not use the BQA information as input, in general, it learned to maintain the TOA values of quality ”good” pixels. This is quite encouraging since we see that most changes in pixels values are very small and in many cases within the Proba-V error sensitivity.

The results in Figure 4.2 show cues that highlight that the proposed transformation is making Proba-V images more statistically similar to Landsat-8 Upscaled ones without modifying significantly the TOA reflectance of Proba-V. Nevertheless, having good aggregate statistics is not enough to demonstrate that the transformation has been successful. Figure 4.3 shows some examples of the proposed transformation (the full dataset can be inspected at: https://isp.uv.es/projects/cloudsat/pvl8dagans). In these examples, what we see is that the images after the proposed transformation (column PV $\rightarrow$ LU) maintain the content of Proba-V images with the colors and texture of LU. It is remarkable to see here that the adapted images indeed reduced the amount of saturation specially in the blue channel (adapted images are less blueish compared to Proba-V ones). Finally, in the last column, we show the cloud mask calculated with the PV $\rightarrow$ LU image. The CD model that we used to compute this cloud masks is the model trained in Landsat Upscaled images of the L8Biome dataset (model $\mathrm{TL}_{\mathrm{L8,333}}$ of previous chapter in Table 3.1). In these cases, we see that the cloud masks are very accurate. Table II of the paper in the Appendix III shows the results of cloud detection accuracy in the PV24 dataset. The takeaway message of this table is that the overall accuracy of the models after the DA transformation is 91.87-93.10 which is around 2 points higher in accuracy than models without DA (88.84-91.87 in Table 3.1).

4.3 Reproducibility

We published the data and code to reproduce our results; in particular, we published the *L8Biome pseudo-simultaneous dataset* and the code to train and run inference with the implemented DA transformations (https://github.com/IPL-UV/pvl8dagans). We also included the Landsat to Proba-V physically-based domain adaptation transformation, the CD model trained on the L8Biome Upscaled dataset ($\mathrm{TL}_{\mathrm{L8,333}}$). Additionally, we included checkpoints with the trained DA transformation and the cloud detection model and a script to produce Proba-V corrected images with the proposed cloud mask. Last but not least, the PV72 dataset can be browsed at https://isp.uv.es/projects/cdc/probav_dataset.html and it is available upon request.

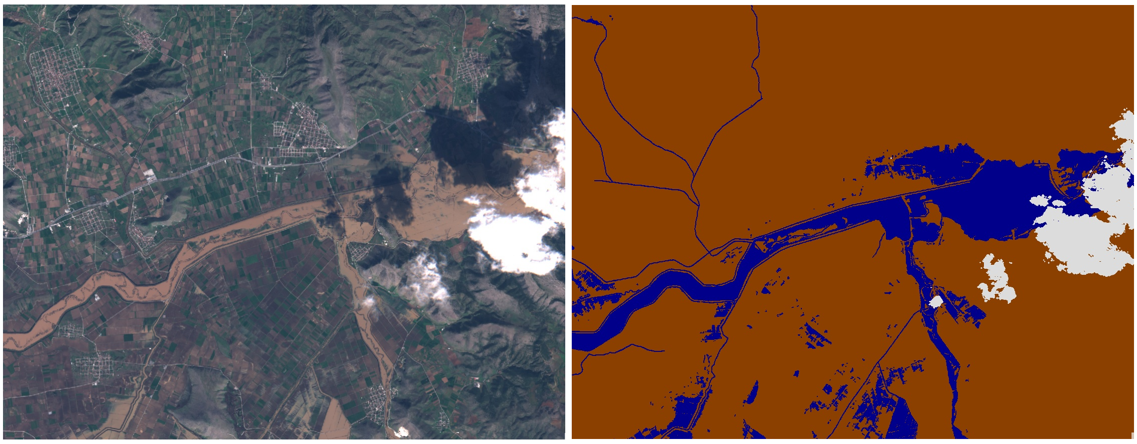
5. Towards global flood mapping onboard low cost satellites with machine learning

5.1 Motivation of the work

The Frontier Development Lab (FDL) is a research program organized in partnership with ESA in Europe and NASA in the United States[1]. It takes place during summer in a fully-funded eight weeks research sprint. During that time a set of researchers are selected and paired with domain and ML experts in small groups of 5-8 people. Each group is assigned to one pre-defined *challenge* in the Earth or Space Sciences to be tackled with ML. I participated in the 2019 Europe research sprint in the 'disaster prevention, progress and response' team, where we worked on onboard flood segmentation. The current research publication Mateo-Garcia et al. (2021) was originated during this sprint and was consolidated over the follow-up months.

The overreaching goal of this fourth contribution of the Thesis, Mateo-Garcia et al. (2021) [Appendix IV], is to demonstrate an end-to-end flood segmentation application to be run onboard small CubeSats to produce fast maps for disaster response applications. The long-term vision is that with a constellation of CubeSats we could significantly reduce the time to obtain detailed maps after an emergency event. This response time is currently bounded by the revisiting time of the satellite. Since CubeSats are relatively cheap, launching such constellation would be feasible for countries and organizations which will reduce revisit time massively: we estimated that with the cost of a mission such a Sentinel-2 we could launch 30 CubeSats which would reduce the revisit time from 5 days to 8 hours. Nevertheless, large constellations of satellites require a higher amount of data transfer between the satellite and ground stations, which is expensive and introduce other bottlenecks. In this contribution, we argue that onboard processing is a potential solution to overcome some of these limitations. With onboard processing we can offset part of the product generation to the hardware in order to get lighter products that are cheaper to download from the satellite to the ground stations (in this case flood binary maps). An oversimplistic metaphor of this vision is to have satellites with 'eyes' and 'brains' instead of having the 'eyes' in orbit and the 'brains' on Earth.

[1] ESA-FDL Europe (https://fdleurope.org). NASA-FDL USA (https://frontierdevelopmentlab.org).



(a) (b)

Figure 5.1: (a) Sentinel 2 RGB bands and (b) associated labelled map (🟫 land ⬜ cloud 🟦 water) over Farkadhon (Greece) derived from Copernicus EMS 271 activation. Base image and reference labels are included in the *WorldFloods* database.

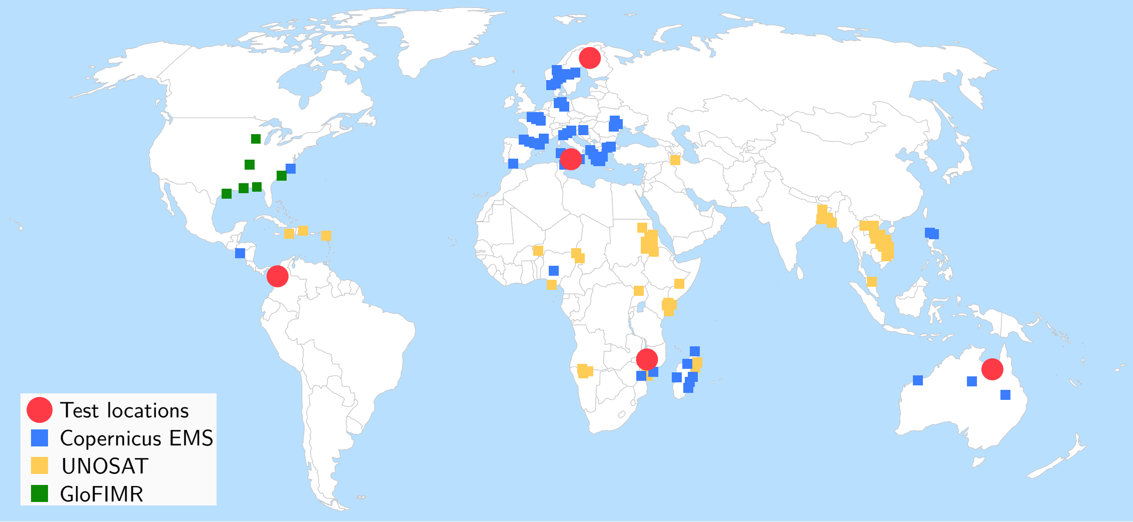
The main motivation is therefore to develop a flood detection model using optical sensors; however, a direct consequence of deploying the model onboard is that, in addition to the flooded areas, it has to simultaneously detect the clouds present on acquired images.

5.2 Summary and main results

The ESA's ΦSat-1 satellite is a 6U CubeSat part of FSSCat, a technology demonstrator mission launched in September 2020. It contains a 49 band hyper-spectral camera (HyperScout-2) which integrates a Intel Movidius Myriad2 GPU to accelerate computer vision applications. The goal of ΦSat-1 is to demonstrate onboard ML applications accelerated with dedicated hardware and assessing the robustness of the payload to ionizing radiation. The goals of the mission have been recently attained by the onboard cloud detection application of Giuffrida et al. (2020) and the onboard results have been published in Giuffrida et al. (2021). In this work, we targeted ΦSat-1 in order to take advantage of the opportunity window opened with this demonstrator platform: at the time of developing this work the satellite was not launched and the PhiLab was looking for other onboard applications. Although at the end this work could not be deployed at ΦSat-1, we had a second chance with D-Orbit WildRide mission, and a similar payload to the proposed in our publication was launched in June 2021. We further describe this derived outcome of the Thesis in section 6.3.7.

Hence, in the contribution Mateo-Garcia et al. (2021), we develop a flood&cloud segmentation model based on FCNN for the HyperScout-2 hyper-spectral camera aboard ΦSat-1. Since ΦSat-1 was not launched at the moment of developing this work, no data was available of the HyperScout-2 camera to develop an ML-based flood segmentation model. Therefore, in this work we propose also a *transductive transfer learning*. In particular, in this case we propose to develop a model in Sentinel-2 data and ground truths and transfer such model to the HyperScout-2 sensor.

The proposed strategy is therefore similar to the described in the second contribution of this Thesis: train a model in a proxy dataset and transfer such model to the target sensor.



Dataset	Flood events	Flood maps	256x256 patches	Water pixels (%)		Land pixels (%)	Cloud pixels (%)	Invalid pixels (%)
				Flood	Permanent†			
Training	108	407	182,413	1.45	1.25	43.24	50.25	3.81
Validation	6	6	1132	3.14	5.19	76.72	13.27	1.68
Test	5	11	2029	20.23	1.16	59.05	16.21	3.34

† Permanent water obtained from the yearly water classification product of Pekel et al. (2016) available at the Google Earth Engine.

Figure 5.2: *WorldFloods* dataset. On the top training floodmaps colored by source and test floodmaps shown with red circles. At the bottom, statistics of number of pixels on each subset and percentage of flood and permanent water pixels, land and cloud.

Nevertheless, there is an extra caveat here which is that, at the moment of carrying out this work, there were no publicly available global dataset of labeled flooding data for optical sensors. Therefore, one of the major contributions of this work is to develop and curate such dataset which we called *WorldFloods*.

WorldFloods is an ML-ready dataset which contains floodmaps coming from three different organizations that produce these maps to monitor flooding events for disaster response. A floodmap is a vector product which indicates presence of water; these products are derived manually or semi-automatically by an operator from a satellite image. For each of these floodmaps, we obtained the first Sentinel-2 image after the event and we rasterised the floodmap with a cloudmask obtained from s2cloudless (Sentinel Hub team, 2017) in a 3-class ground truth as shown in Figure 5.1 (left Sentinel-2 image, right rasterised ground truth). Figure 5.2 shows the location and statistics of the images included in the dataset. We show with different colors the original source of the floodmap which corresponds to the Copernicus Emergency Management Service (Copernicus EMS), the United Nations Satellite Center (UNOSAT), and the Global Flood Inundation Map Repository of the University of Alabama (GloFIMR). Additionally, we show in red the location of the floodmaps used for testing the models: for testing we manually chose locations where the ground truth data was generated from Sentinel-2, which were geographically diverse and where the quality of the ground truth was sufficiently good. This is highlighted in the

Table 5.1: IoU and recall results for models trained on *WorldFloods*.

	Model	IoU total water	Recall total water	Recall flood water	Recall permanent water
10m	NDWI (thres -0.22)	65.12	**95.75**	95.53	**99.70**
	NDWI (thres 0)	39.99	44.84	42.43	86.65
	Linear	64.87	95.55	**95.82**	90.75
	SCNN	71.12	94.09	93.98	95.93
	U-Net	**72.42**	95.42	95.40	95.83
80m	NDWI (thres -0.22)	64.10	94.76	94.57	98.15
	NDWI (thres 0)	39.07	44.01	41.69	84.55
	Linear	60.90	95.00	94.79	**98.58**
	SCNN	68.87	**96.03**	**96.11**	94.76
	U-Net	**70.22**	94.78	94.85	93.50
80m HyperScout-2 overlapping bands	NDWI (thres -0.22)	64.10	**94.76**	94.57	**98.15**
	NDWI (thres 0)	39.07	44.01	41.69	84.55
	Linear	50.27	80.47	79.69	94.03
	SCNN	**65.82**	94.62	**95.17**	84.99
	U-Net	65.43	94.59	**95.17**	84.44

statistics of Figure 5.2 which shows lower amount of clouds an higher amount of flood pixels in the test set; this is done deliberately to obtain more meaningful metrics.

The *WorldFloods* dataset allows to train and validate flood segmentation models for Sentinel-2. In this contribution, we used this dataset to develop FCNN models to be transferred to the HyperScout-2 camera. Nevertheless this dataset can be used for other tasks: in section 6.3.8, we describe a follow up work which is being tested for deployment at UNOSAT.

For the HyperScout-2 sensor, we selected the bands in Sentinel-2 that overlap the region of the spectrum sampled by HyperScout-2 (from 390nm to 990nm). This corresponds to bands B1 to B9 of Sentinel-2 (see figure 2 of the publication). We upscale the resolution of the images from the Sentinel-2 resolution (10 m) to the HyperScout-2 resolution (80 m). Additionally, we simulate the noise expected in a smaller platform such as a CubeSat with data augmentation for training; in particular, we include degradation such as Gaussian noise, motion blur and channel jitter.

In the current contribution Mateo-Garcia et al. (2021), we trained different FCNN models for the task of land, water and cloud segmentation. Since the model was designed to run onboard we decided to tackle simultaneously water and cloud segmentation to optimize the processing pipeline. We tested two architectures which shown a relatively similar performance in the segmentation task (see Table 5.1). These architectures are the U-Net architecture originally proposed in Ronneberger et al. (2015) and a lightweight 5-layer FCNN (SCNN in Table 5.1). In Table 5.1, we provide results for different models trained at different resolutions, i.e. with and without the domain adaptation transformation mentioned before. We see that there is a significant drop in performance from models using all the bands of Sentinel-2 compared with models only using the overlapping bands with HyperScout-2 (bands B1 to B9 of S2). This is because those models do not have the short-wave infrared bands, which are strong indicators of the presence of water.

For implementation, the SCNN model was chosen due to its lower computational cost. We tested this model in a Raspberry-Pi with an attached Intel Movidius Myriad2 chip similar to the one available in ΦSat-1. We used the OpenVINO software to transfer the model and run it on the Raspberry-Pi. This model shows a processing rate approximately of 12MP per minute.

For further details of the current contribution we refer the reader to the original publication included in Appendix IV.

5.3 Reproducibility

As mentioned earlier, one of the most important contributions of this work is the *WorldFloods* dataset which contains a curated collection of flood extent maps and Sentinel-2 images over a variety of locations. We published this dataset alongside the publication Mateo-Garcia et al. (2021). Additionally we also published the codebase to reproduce all the experiments in this work in https://gitlab.com/frontierdevelopmentlab/disaster-prevention/cubesatfloods. In section 6.3.8, we describe the ml4floods package which is an extended version of the published codebase to do end-to-end flood extent segmentation with Sentinel-2. This package is currently being tested by UNOSAT for deployment in their rapid response platform for flooding.

6. Discussion and Conclusion

6.1 Discussion

Earth observation with optical satellite sensors is a key technology to monitor our Planet. In the last years, we are witnessing an exponential increase in the number of optical instruments launched in orbit. These new instruments, accounted together, provide an unprecedented data stream with high spatial and temporal resolutions all over the globe. A proper exploitation of this data is improving our understanding of the biosphere (Wolanin et al., 2019), the oceans (Sauzède et al., 2020), our capacity to respond rapidly to natural disasters (Rudner et al., 2019), and ultimately it is helping us to adapt to climate change. Nevertheless, this data abundance also implies challenges since the raw data provided by these sensors is not sufficient to address these problems. Henceforth, in order to truly exploit this data, useful remote sensing products need to be developed at a fast pace for each of these new sensors.

The current scenario of optical remote sensing has two defining properties. Firstly, it is **heterogeneous**: there are many different sensors with different spatio-spectral characteristics that provide different views of the Earth. Secondly, remote sensing data is **abundant** and in many cases **freely available**: since the USGS opened in 2008 the Landsat archive other major players in the Space community have made also their products open. Nowadays, platforms such as the GEE provide access to hundreds of different remote sensing products freely to the Science community.

These defining properties are bringing a **paradigm shift** on how remote sensing products are developed. Traditional, **knowledge-based**, remote sensing products are based on a very deep understanding of spectroscopy and optical physics and rely on understanding very well the characteristics of the instruments and orbits. Over the course of this Thesis, we have witnessed how this approach is being boosted by a new **data-driven** trend which seeks to exploit the aforementioned abundance of data. This new approach, as we understand it, has the potential to go over the cases where physics models are computationally expensive or not well resolved to produce more accurate remote sensing products. Additionally, since there are open data archives of images of many different optical sensors, new products can be built using this data for the new upcoming sensors. In this Thesis, we call this process *transferring a model across sensors*. Some of the contributions of this Thesis are devoted to explore methodologies to transfer data-driven products across similar optical sensors.

One product that is required by virtually all optical sensors observing the Earth (from the visible at 390nm to the SWIR at 2500nm) are cloud masks. Clouds are suspended in the Earth's atmosphere reflecting the sun light captured by our optical satellites and preventing us from observing the surface of the Earth. We know by these sensors that clouds are pervasive in the atmosphere and that they cover on average almost 70% of the surface of our planet. Therefore, distinguishing between cloud pixels and surface pixels is the first step on most applications, either those observing the surface (e.g. Álvaro Moreno-Martínez et al. (2018)) or the atmosphere (e.g. Zantedeschi et al. (2019)). This is needed because, before starting any further analysis, these applications need to know whether they are looking at the surface or not, to discard (or use) those pixels. Obtaining accurate cloud masks is therefore vital for many downstream applications. Additionally, since most of these applications exploit large amounts of data, they require that these cloud masks are produced automatically without human intervention; for instance in the work of Wolanin et al. (2020) or Mateo-Sanchis et al. (2019) authors propose to exploit year-long time series of images to estimate crop yields over large regions; manual filtering of cloudy images in 5-day image time series is therefore not feasible.

Current operational knowledge-based algorithms for cloud masking (aka threshold-based) do not produce accurate cloud masks in several situations. Over the course of this Thesis, we have been involved in several projects aiming to improve the cloud detection accuracy of operational models for Proba-V in the 'Proba-V Cloud Detection Round Robin' (Iannone et al., 2017) and for Landsat-8 and Sentinel-2 in the 'Cloud Masking Inter-comparison eXercise (CMIX)' (European Space Agency, 2019). These projects highlight the current shortcomings of threshold-based algorithms and the need of novel, more accurate, cloud detection products. The contributions of this Thesis are devoted to provide an answer to these problems and to improve cloud detection for optical remote sensing satellites.

The data-driven models that we propose for cloud detection and for transfer learning in most of the contributions of this Thesis are based on deep learning. Among data-driven methodologies, we choose deep learning because: (a) It has the capacity to scale to arbitrarily large datasets without a plateau in performance, i.e. its accuracy keeps increasing with larger volumes of data. This makes them particularly well suited for remote sensing where satellite image archives are in the order of the petabytes. (b) They have shown outstanding performance in natural images tasks by using convolutional networks. In this Thesis, the models that we propose are Fully Convolutional Neural Networks (FCNN), these models are specially designed to obtain per-pixel predictions (one category assigned to each pixel of the image). In particular, the contributions of this Thesis include some of the first works using FCNN for cloud detection (Mateo-García et al., 2017) and domain adaptation across images of different sensors (Mateo-García et al., 2020a).

In the following sections, we will briefly discuss the contributions of each of the publications of the Thesis and their impact.

6.1.1 Multi-temporal cloud masking in the Google Earth Engine

The first contribution of this Thesis, Mateo-García et al. (2018) [Appendix I], proposes multi-temporal cloud detection models for Landsat-8 images. With this contribution, first I got introduced to remote sensing in general and to the cloud detection problem in particular. In this work, we focused on Landsat-8 which is arguably the satellite with a higher number of proposed cloud detection schemes. We extended the work of Gómez-Chova et al. (2017a) and proposed a multi-temporal cloud detection algorithm that is based on two stages: *surface (or background) estimation* and *change detection*. In the first stage, we use previous cloud free images to estimate the reflectance of the surface, testing different methods. In the second stage, we compare the estimated background against the current image and fit a set of global thresholds that are used to mask the clouds. The proposed approach is a simplified method very similar to other multi-temporal methods proposed in the literature (e.g. Zhu & Woodcock (2014) or Hagolle et al. (2010)). The novelty of the work is that the proposed algorithm can be implemented efficiently in a platform where multi-temporal models could be run operationally (the Google Earth Engine (GEE)). All previous works of multi-temporal cloud detection in the literature require downloading previous cloud free images for the background estimation step. This strongly limits the applicability of the CD model. In our work, we are able to run the CD model in any new location directly in the GEE platform without any data downlinking. Another significant contribution of this work is its validation in a large corpus of manually labeled images from the L8Biome dataset. The aforementioned issues of running previous cloud detection models on new locations make those methods poorly validated. It is common to see that previous works are only validated in a small set of images and in many cases this validation only includes visual inspection of the masks. The results of our validation show significantly better performance than single-scene operational threshold-based methods such as FMask (Zhu & Woodcock, 2012) or ACCA (Scaramuzza et al., 2012). Finally, we made a significant effort in open-sourcing the models and ensuring the reproducibility of our results. The code repository https://github.com/IPL-UV/ee_ipl_uv contains the implementation and several tutorials with use-cases. Additionally, the web page https://isp.uv.es/projects/cdc/viewer_l8_GEE.html shows the masks compared with the ground truth in all the acquisitions used for validation.

6.1.2 Transferring deep learning models between Landsat-8 and Proba-V

The second contribution of this Thesis, Mateo-García et al. (2020b) [Appendix II], demonstrates transfer learning of FCNN models across two different optical instruments (Proba-V and Landsat-8). Transfer learning of ML-based CD models was one of the main goals of the PhD Thesis. On the one hand, it has not been demonstrated before and its trade-offs have not been studied. On the other hand, transfer learning could enable the development of ML-based CD models for upcoming sensors since it could significantly reduce the amount of training data required to create such models.

In this work, we make a very comprehensive study of transductive and inductive transfer learning using several labeled datasets of Proba-V and Landsat-8. In particular, we first introduce a domain adaptation transformation based on the physical properties of the sensors, which we apply to Landsat-8 images to make them similar to Proba-V acquisitions (i.e. to have similar spectral bands and the same nominal spatial resolution). We called the transformed images *Landsat-Upscaled images*. Using this transformation we carry out three different types of transfer learning experiments: (a) *from Landsat-8 to Proba-V*, where we show that models trained only with Landsat-Upscaled images produce cloud masks 5 points more accurate than the current threshold-based operational Proba-V method; (b) *from Proba-V to Landsat-8*, where models that use only Proba-V data for training have a similar accuracy to the Landsat-8 operational FMask in the L8Biome dataset (87.79–89.77% vs 88.48%); and (c) *jointly from Proba-V and Landsat-8 to Proba-V*, where we demonstrate inductive transfer learning using Landsat-Upscaled images and few labeled Proba-V images jointly. In this case, the accuracy increases from 1–5 points compared with using only the Landsat-8 labeled dataset.

With this work, we reached most of the goals of the PhD Thesis since we were able to show in the same publication that FCNN models produce very accurate cloud masks and that these models can be transferred to a compatible sensor with small drops in accuracy. We would also like to point out that we carried out the comparisons of the proposed models in a very comprehensive and rigorous manner: we trained different copies of the same network with different random seeds to account for the stochastic behaviour of neural networks and we compared with other works of the literature that use FCNN (most notably Jeppesen et al. (2019) and Li et al. (2019)), and with the official operational CD methods of the considered satellites.

Finally, I would also like to highlight that, in order to carry out this work and the development of the Proba-V collection C2 algorithm (Toté et al., 2021), we manually labeled a large dataset of Proba-V images (the PV72 dataset described in section 1.4.1). Without this data none of those works could not have been tackled. This dataset can be browsed at https://isp.uv.es/projects/cdc/probav_dataset.html and which is available upon request.

6.1.3 Cross sensor adversarial domain adaptation of Landsat-8 and Proba-V Images for Cloud Detection

The third contribution of this Thesis, Mateo-García et al. (2020a) [Appendix III], explores learning-based domain adaptation also between Landsat-8 and Proba-V. The purpose of domain adaptation transformations is to make images in one *source* domain similar to images in a *target* domain.

This work was motivated by the observed discrepancies between the Landsat-Upscaled images (introduced in the previous section) and the real Proba-V acquisitions. In particular, we observed that the distribution of the colors of the bands and its real spatial resolution measured by its Fourier transform is significantly different even in images acquired with time differences of minutes. We hypothesize that the drop in cloud detection accuracy in Proba-V images of models trained in Landsat-Upscaled data might be due to this difference, which is called *data-shift* in the ML literature.

Hence, in this work, we adapt one of the latest most successful learning-based domain adaptation methods in computer vision to bridge the gap between Proba-V and Landsat-Upscaled images. In particular, we propose a variant of Cycle-Consistent Adversarial Domain Adaptation (CyCADA) (Hoffman et al., 2018) with a custom loss specific to the remote sensing use-case which aims to maintain the calibrated reflectance values of the satellite images (called *identity loss* in the paper). One of the main advantages of CyCADA is that it is *unpaired*; that is, it does not require of simultaneous co-located images for training. This is because the supervision (learning feedback) comes through a discriminator model trained simultaneously to the DA models. This is crucial for the intended use of the DA application, which is cloud masking, since clouds move fast and even in images acquired with differences of minutes it can be observed a displacement of the clouds.

For this work, we collected a diverse dataset of close-in-time co-located Landsat-Upscaled and Proba-V images. Those images are split in training and testing sets by image acquisition to maintain the independence of the test set. Over this set, we observe that the global statistics of the Proba-V images adapted with the trained generator are similar to Landsat-Upscaled ones. Additionally, we see that the content of the Proba-V images is maintained whereas the colors are more similar to the Landsat-Upscaled images; moreover, the adapted images are slightly sharper than the real Proba-V images and with significantly less saturated values. The complete test set can be inspected at https://isp.uv.es/projects/cloudsat/pvl8dagans/. Finally, when we use the proposed DA transformation and apply the CD model trained in Landsat-Upscaled images we observe an increase in cloud detection accuracy of two points.

With this work, we demonstrated how to construct DA transformations between two different sensors to transfer the *style* of the other sensor while maintaining the *content* of the observed image. It is important to stress that the proposed methodology does not require paired samples which makes it easier to apply it to many remote sensing use-cases. We use this transformation to apply a model trained in a source domain to test images of the target domain; however, this methodology is intrinsically general and could be used for other use-cases. For instance, it could be used to produce harmonized fusion products where images are transformed to the style of the sensor with better radiometric quality. Another extension, for sensors where atmospheric correction is challenging (i.e. for sensors with few bands and without dedicated bands for atmospheric retrievals), could be to produce atmospherically corrected images. For example, we could do this by using Level 2 (atmospherically corrected) Landsat-8 scenes instead of Level 1 TOA reflectance in the setting propose in this article. This would produce Proba-V adapted images statistically similar to Landsat-8 surface reflectance. Nonetheless, this application would require a very thorough validation before testing it operationally.

Finally, for this work, we also made a significant effort to enable reproducibility: we open-sourced the paired dataset, a visualization tool with all the test images with the proposed methodology (https://isp.uv.es/projects/cloudsat/pvl8dagans/), and the code to train and to apply the transformation to a new Proba-V scene (https://github.com/IPL-UV/pvl8dagans). Additionally, in that repository, we also open-sourced parts of the previous work such as the upscaling transformation and the models trained in the upscaled L8Biome dataset.

6.1.4 Towards global flood mapping onboard low cost satellites with machine learning

The fourth contribution of this Thesis, Mateo-Garcia et al. (2021) [Appendix IV], describes an onboard ML-system to segment flooding water. Onboard processing is one of the latest trends in remote sensing. Running software to build products onboard has some advantages: to optimize the communication bandwidth of the satellite data download to ground stations (e.g. to only downlink images that are *useful*, for instance discarding overly cloudy scenes Giuffrida et al. (2021)); to speed up the downlinking of certain critical observed information (e.g. a methane leak, a wildfire or a flood); or to trigger a retrieval with a different instrument (e.g. for satellites with different sensors, onboard processing could be used to trigger the retrieval of another instrument aboard).

Onboard processing is not a new idea, the pioneer EO-1 mission launched on November 2000 demonstrated some onboard capabilities on the Autonomous Sciencecraft Experiment (ASE); these experiments include onboard cloud detection (Griggin et al., 2003), ice monitoring (Doggett et al., 2006), and even flood mapping (Ip et al., 2006). The differentiating factor of nowadays proposals is that the current payloads contain dedicated hardware aimed to accelerate neural network applications. This is the case of the ESA ΦSat-1 mission, which contains a hyperspectral camera (HyperScout-2) and an Intel Movidius Myriad2 chip to accelerate computer vision applications. In this framework, our work describes a simple MLPayload to do flood extent segmentation that is tested on hardware similar to the ΦSat-1. The proposed model to segment floods and clouds is again a FCNN, which we showed to work well for the cloud detection case in previous contributions. In particular, in this work we propose a simpler architecture that produces masks fast in order to cope with the requirements of the ΦSat-1 hardware.

For training the proposed flood segmentation model we need a dataset with images and flood segmentations masks to be used as ground truth. At the time of developing this work, there was no operational hyperspectral sensor with sufficient data over flooded areas to build a global model. Hence, we choose the multi-spectral Sentinel-2 as a good proxy since their bands overlap the area of the spectrum sampled by HyperScout-2. Nevertheless, a curated dataset of flood extent maps and Sentinel-2 images did not exist either at that time; therefore, we decided to create that dataset ourselves. The collected dataset, that we called *WorldFloods*, is perhaps the most important contribution of this work.

WorldFloods used flood extent maps created by three different organizations and its closest Sentinel-2 image after the event. We compiled a curated dataset with more than 400 flood extent maps from more than 100 verified flood events that we used to train and validate flood segmentation models. In order to create a model for the HyperScout-2 sensor, we followed a transfer learning approach very similar to the one proposed in the second contribution of this Thesis. In particular, to simulate the HyperScout-2 data, we upscale Sentinel-2 images from 10 m to the 80 m nominal resolution of HyperScout-2 and select the nine bands of Sentinel-2 that overlap the spectrum sampled by the sensor. Additionally, since the CubeSat is expected to have worse radiometric quality, we introduce noise in the images at training time to mimic the expected degradation of the images.

Finally, in this work, we also made an important contribution towards reproducibility and open science: we published the *WorldFloods* dataset in a common format, the trained models and the code to run inference and reproduce all the experiments in this GitLab repository: https://gitlab.com/frontierdevelopmentlab/disaster-prevention/cubesatfloods).

6.2 Conclusion

In a nutshell, this Thesis proposes different data-driven methods, most of them based on deep learning, to address different remote sensing problems for cloud detection. Satellite images are at the core of all data-driven methods proposed in this Thesis and preparing this data has been the most critical and time consuming (and underrepresented) part of all these works. Data preparation is critical in data-driven systems because errors or miss-understandings in the inputs propagate to the outputs and, given the complexity of the models, they are very difficult to detect and track (see Sambasivan et al. (2021) for a good survey of data cascades caused by bad data engineering practices). Data preparation is also very time consuming because of the volumes of data that are needed to train and validate the models. Although this cost could be cut down by transfer learning or more data-efficient models, large, global and accurate datasets are still needed to validate and compare any proposed DL methodology. Dealing with large volumes of data requires software expertise and domain knowledge, yet a good understanding of the datasets is of uttermost importance to propose meaningful solutions and build accurate models (see e.g. Karpathy (2019) for good data practices, where the author suggests to *"become one with the data"* and thoroughly inspect all your data samples before modeling). At the end of the day, data is the basis of the empirical method in Science and, as a community, we should make an effort to encourage and foster good and open data practices.

The contributions of these works lie in between the ML and the RS fields. In general, one could say that most of our contributions consist of *'taking a successful computer vision model and adapting it to remote sensing data'*. In essence this is true, nevertheless all these works are specifically designed for the remote sensing problem we address and all the models have been thoroughly validated thinking in the use-cases these models could have. It should be pointed out that the adaptations of the computer vision methods that we propose are tailored to the specific remote sensing problem that we are aiming to solve. For instance, the third contribution of this Thesis, Mateo-García et al. (2020a), proposes a domain adaptation transformation that seeks to maintain the radiometric calibration of the sensors. This is needed if we wish the proposed transformation to be used by other downstream applications that might require calibrated reflectance. In a more broad manner, all models on this Thesis have been validated in good quality with representative data that we have carefully selected. This data is chosen to be representative of the global Earth conditions each sensor could observe.

To conclude, in this Thesis we propose different solutions to improve cloud masking and develop methodologies to transfer deep learning models across different sensors. In particular, we demonstrate that we could improve cloud detection using either the temporal or the spatial dimension of optical satellite images. We also show that we could transfer FCNN models trained on data of one optical satellite to another with compatible spatio-spectral characteristics and that learning-based domain adaptation transformations could boost the transferability of these methods. Finally, we show that these models could be even deployed onboard CubeSats, opening the door to many future developments.

6.3 Related outcomes and projects

During the course of this Thesis, I have been lucky to participate in different projects where I had the opportunity to learn and collaborate with many different people. Most of these projects, are somehow related to this Thesis and have influenced this work. In the following, I will briefly cover some of them and their most significant outcomes. It should be noticed that none of these projects have been carried out in solitude; hence, I take the opportunity here to thank all the collaborators that made this possible.

6.3.1 Google Earth Engine award: Cloud Detection in the Cloud

This Google Earth Engine Award, granted to Prof. Luis Gomez-Chova, was my first opportunity as a researcher. The goal of the project, as the title suggest, was to develop cloud detection models in the GEE platform. The first contribution of this Thesis (Mateo-García et al., 2018) and the following journal article (the contribution of this Thesis is a continuation of that work) are direct contributions of this project:

- L. Gómez-Chova, J. Amorós-López, **G. Mateo-García**, J. Muñoz-Marí, and G. Camps-Valls, Cloud masking and removal in remote sensing image time series, *Journal of Applied Remote Sensing JARSC4*, vol. 11, no. 1, p. 015005, Jan. 2017, doi: 10.1117/1.JRS.11.015005.

6.3.2 Operational cloud detection models for Proba-V

In 2016 we participated in the 'Proba-V Cloud Detection Round Robin exercise' (Iannone et al., 2017). Our proposed cloud masking solution based on Neural Networks ended up in second place with a difference of 0.1% with the first ranked model. The ESA Proba-V Quality Working Group (PVQWG) decided to implement our solution in the Proba-V ground segment to be the operational cloud masking model disseminated with all Proba-V products. Over the course of a year we developed three models for Proba-V one for each of the resolutions that Proba-V images are published (100 m, 333 m and 1 km). During 2021 the Proba-V team has been carrying out a reprocessing of the full archive; this will produce a new Proba-V collection (called Collection 2, C2) where, among different improvements, it will include our cloud mask as the official cloud masking product (Toté et al., 2021).

The Proba-V labeled dataset that we used in the second and third contributions of this Thesis are an outcome of this project. See section 1.4.1 for more details about the Proba-V cloud detection manually labeled dataset (PV72). Additionally, I participated in the following conference publications:

- R. Q. Iannone, F. Niro, P. Goryl, S. Dransfeld, B. Hoersch, K. Stelzer, G. Kirches, M. Paperin, C. Brockmann, L. Gómez-Chova, **G. Mateo-García**, R. Preusker, J. Fischer, U. Amato, C. Serio, U. Gangkofner, B. Berthelot, M. D. Iordache, L. Bertels, E. Wolters, W. Dierckx, I. Benhadj, E. Swinnen, Proba-V Cloud Detection Round Robin: Validation Results and Recommendations, *9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp)*, 1–8, 2017. doi: 10.1109/Multi-Temp.2017.8035219.
- L. Gómez-Chova, **G. Mateo-García**, J. Muñoz-Marí and G. Camps-Valls, Cloud detection machine learning algorithms for PROBA-V. *IGARSS 2017 - 2017 IEEE International Geoscience and Remote Sensing Symposium*, 2017
doi: 10.1109/IGARSS.2017.8127437

6.3.3 Cloud Masking Intercomparison eXercise (CMIX)

The Cloud Masking Inter-comparison eXercise (CMIX) organized jointly by ESA and NASA is a similar exercise to the Proba-V Round Robin but for Sentinel-2 and Landsat-8. The preceding Atmospheric Correction Intercomparison Exercise (AMIX) revealed several cloud masking issues for both Landsat-8 and Sentinel-2. Hence, the CMIX seeks to compare different cloud detection schemes in eight different manually labeled datasets for Landsat-8 and Sentinel-2. We participated in this exercise with a FCNN trained in the L8Biome and L8SPARCS datasets. For Sentinel-2, we propose transfer learning of this network. The final results of the study can be found in Skakun et al. (Submitted). The models that we trained for the contest for Landsat-8 and Sentinel-2 are open-sourced at https://github.com/IPL-UV/DL-L8S2-UV. Additionally, the following list of journal publications contains a more detailed description of our approach validated in different Landsat-8 and Sentinel-2 datasets and the joint work with all participants of CMIX which is currently under review:

- D. López-Puigdollers, **G. Mateo-García**, and L. Gómez-Chova, Benchmarking Deep Learning Models for Cloud Detection in Landsat-8 and Sentinel-2 Images, *Remote Sensing*, vol. 13, no. 5, Art. no. 5, Jan. 2021, doi: 10.3390/rs13050992.
- S. Skakun, J. Wevers, C. Brockmann, G. Doxani, M. Aleksandrov, M. Batic, D. Frantz, F. Gascon Roca, L. Gómez-Chova, O. Hagolle, D. López-Puigdollers, J. Louis; M. Lubej, **G. Mateo-García**, J. Osman, D. Peressutti, B. Pflug, J. Puc, R. Richter, J.C. Roger, P. Scaramuzza, E. Vermote, N. Vesel, A. Zupanc, L. Zust, Cloud Mask Intercomparison eXercise (CMIX): an evaluation of cloud masking algorithms for Landsat 8 and Sentinel-2, *Remote Sensing of Environment*, (in revision)

6.3.4 FDL 2019 Research Sprint: onboard flood detection

The Frontier Development Lab (FDL) is a research program organized in partnership with ESA in Europe and NASA in the United States. During the summer of 2019, I participated in ESA FDL, which is a fully funded eight week program that took place at ESA ESRIN (Rome, Italy) and at the University of Oxford (UK). As mentioned in chapter 5, this research sprint was the starting point of the fourth contribution of this Thesis. Apart of the journal publication of contribution four, we published an early version of this work in the Humanitarian Assistance and Disaster Response Workshop in NeurIPS conference:

- **G. Mateo-García**, Silviu Oprea, Lewis Smith, Josh Veitch-Michaelis, Guy Schumann, Yarin Gal, AtılımGüneş Baydin and Dietmar Backes, Flood Detection On Low Cost Orbital Hardware. *Artificial Intelligence for Humanitarian Assistance and Disaster Response Workshop, 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada*. arXiv: 1910.03019

6.3.5 FDL 2020 Research Sprint: waters of the United States

In 2020, I also participated in the FDL program, this time in the American version funded by NASA and the USGS. Due to the Covid-19 pandemic this time the sprint took place on-remote. In this case, the goal of the challenge was to map narrow water streams in order to produce early warnings for droughts. The motivation of this is that it has been shown in several studies of precipitation levels that there is an unaccounted-for volume of flowing surface water that has yet to be considered. This is because current satellite approaches are limited to scarce observations of Landsat-8 and Sentinel-2 that map only the widest streams (up to 90 m) (Pekel et al., 2016). Smaller tributaries that make up to almost 50% of the dendritic surface network (Allen & Pavelsky, 2018) remain unobserved. Mapping those streams over time could give us early warnings of droughts and could provide a better understanding of the impermanence of our waters, showing where to expect water, and where not to.

In order to produce such a map, we got access to different Very High Resolution (VHR) data sources over 4 AoIs. This data includes: few WorldView-3 images with a nominal resolution of 0.5 m, LiDAR derived products from the 3D Elevation Program (3DEP) with 1 m to 3 m resolution, daily PlanetScope time series over a two year time period at 3 m and few labeled polygons that indicate presence or absence of water in few places of the WorldView-3 imagery. With this, we propose a two-stage pipeline, that we called Pix2Streams, to produce per-reach estimations of presence or absence of water in PlanetScope imagery. In the first stage, we use a multi-sensor FCNN that fuses a multi-day window of 3m PlanetScope imagery with 1m LiDAR derivative products to produce higher resolution water probability maps at the resolution of the labels (0.5 m). The second step aggregates these maps over an elevation-derived synthetic valley network to produce a snapshot of water occurrence at the stream level.

We ran Pix2Streams on a 24 km^2 area over a 2-year daily PlanetScope time-series to produce a daily product of water probability per-stream that is used to derive flow frequency and could be used to produce early warnings of droughts in the future. A video with the results over some locations can be seen at http://bit.ly/pix2streams.

This work has been continued by a contract of the USGS with Trillium (the company managing the FDL program). I have participated in this follow-up project which consisted of consolidating the results and providing a professional implementation of the aforementioned pipeline. The results of this work are being prepared for publication; an early version of those were published at the AI for the Earth Sciences Workshop of the NeurIPS conference:

- Dolores Garcia, **Gonzalo Mateo-Garcia**, Hannes Bernhardt, Ron Hagensieker, Ignacio G. Lopez-Francos, Jonathan Stock, Guy Schumann, Kevin Dobbs and Alfredo Kalaitzis. Pix2Streams: Dynamic Hydrology Maps from Satellite-LiDAR Fusion. *AI for Earth Scienes Workshop, 34rd Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada*. axXiv: 2011.07584

6.3.6 FDL 2021 Research Mini-Sprint: Sentinel-2 super-resolution

Early in 2021, I participated in other short international collaboration funded by ESA to develop multi-image super-resolution models for Sentinel-2. Multi-image super-resolution (MISR) consists of producing a super-resolved image from a set of lower resolution images. Multi-image super-resolution is an ill-posed problem albeit much better conditioned than single image super-resolution since the multiple inputs provide more sub-pixel information to go beyond the Nyquist theoretical limit. In this work, we tackle super-resolution as a regression problem using the state-of-the-art network HighResNet (Deudon et al., 2020). To that end, we curate a dataset, using PlanetScope imagery from the SpaceNet-7 challenge as the high resolution reference (4.77 m) and multiple Sentinel-2 revisits over the same location as its low-resolution counterparts (10 m). This contrasts with other works that use images from a single satellite that are artificially upscaled; we argue that our approach, although it requires more data wrangling for training, it should work better in the test case since there is not data-shift in the inputs (i.e. if we use artificially upscaled PlanetScope data for training the model might not transfer well to Sentinel-2). We show that MISR is superior to single-image super-resolution and other baselines on a range of image fidelity metrics. Additionally, we introduce a radiometric consistency module into MISR model to preserve the high radiometric resolution of the Sentinel-2 sensor, which is significantly better than that of PlanetScope. Finally, we conduct the first assessment of the utility of multi-image super-resolution on building delineation, showing that utilizing multiple images results in better performance in these downstream tasks. This work is currently in preparation, an early version can be found in arXiv:

- M. Razzak, **G. Mateo-Garcia**, L. Gómez-Chova, Y. Gal, F. Kalaitzis, Multi-Spectral Multi-Image Super-Resolution of Sentinel-2 with Radiometric Consistency Losses and Its Effect on Building Delineation. arXiv:2111.03231

6.3.7 Onboard flood segmentation on the WildRide mission

The fourth contribution of this Thesis describes a system for onboard flood segmentation. This system was designed for ΦSat-1 and its hyperspectral camera; however it could not be deployed on this platform due to timing and contracting constraints. Nevertheless, D-Orbit's WildRide mission[1] has been a new opportunity to deploy the system onboard. D-Orbit's Nebula is a vision for a service for in-orbit cloud computing and data storage. Each Nebula processor has specific hardware to accelerate computer vision applications.

Targeting Nebula, we adapted the inference pipeline proposed for ΦSat-1 into a dockerised payload that we also called 'WorldFloods'. Additionally, we designed a set of experiments to be tested in space onboard this platform. The 'WorldFloods' payload consists of an inference pipeline to detect flood water based on the FCNN proposed in the fourth contribution of this Thesis and an extra vectorization step to produce polygons from the per-pixel predictions. Vectorizing the outputs produce an even more compressed product that we tested to be between 1,000 and 10,000 times smaller than Sentinel-2 images.

In June 30th 2021, D-Orbit launched the WildRide mission with a first prototype of Nebula into space on a SpaceX Falcon 9 rocket from Cape Canaveral. This prototype includes the aforementioned 'WorldFloods' payload. From September to December 2021 we tested 'WorldFloods' in orbit and demonstrated the three main goals of the mission[2]:

1. The payload has been run on a full Sentinel-2 image acquisition of 120M pixels and on six smaller Sentinel-2 tiles with flooding downlinking the resulting vector products together with timing statistics.
2. 'WorldFloods' has been repurposed to work on images from the onboard D-sense camera. The D-sense Camera is a general purpose RGB sensor, used for star-tracking, attitude control and verifying payload deployment. It can also be used to acquire images of the Earth although it is not its original goal. Since Earth images of Sentinel-2 and D-sense are completely different, the segmentation models were fine-tuned in order to produce good results in D-sense imagery. For this, we retrained the model on four downlinked D-sense images that we manually labeled.
3. This new model was re-uploaded to the satellite and successfully tested onboard on a D-sense acquisition.

This mission demonstrates some of the proposals of this Thesis. In particular, FCNN for semantic segmentation and inductive transfer learning to improve a FCNN model after its deployment. We are currently preparing a publication with the onboard results and the lessons learned.

[1] Booklet of the mission: https://75a8451e-2fb7-4c8f-830f-36057291f2fe.filesusr.com/ugd/64a0e4_1b982a9343a547e38ed10502b0e25fff.pdf

[2] https://www.dorbit.space/wildride-mission-updates

6.3.8 ML4Floods and pilot study at UNOSAT

ML4Floods is an open-source project led by Trillium where I participated in the first semester of 2021. The long-term goal of the project is to democratise the use of machine learning for flood extent monitoring using Copernicus data. For this project, we develop an *end-to-end* package for flood extent mapping targeting data scientists and domain experts. This python package extends and operationalizes the code of the fourth contribution of this Thesis Mateo-Garcia et al. (2021). In a nutshell, ML4Floods has pipelines for flood extent estimation: from data downloading, preprocessing, model training, model deployment to visualization. We published the code in a public GitHub repository https://github.com/spaceml-org/ml4floods and we made an special effort to create a very comprehensive set of tutorials with different use-cases to foster its adoption here: http://trillium.tech/ml4floods.

The United Nations Satellite Center (UNOSAT) was one of the early testers of this work[3]. UNOSAT deployed recently FloodAI, a flood mapping platform for Sentinel-1 based on the work of Nemni et al. (2020). They are interested in deploying a sister system for Sentinel-2 that together could monitor floods with higher frequency an accuracy. During the pilot study, they collected a dataset of 21 recent flood events (2019-2021) over Africa and they compared the segmentation of the models proposed in the fourth contribution of this Thesis, Mateo-Garcia et al. (2021), with several remote sensing indexes (e.g. MNDWI, NDWI, AWEI). Their results show that our model has similar performance to the best of these indices and that our segmentation was complementary to those. During the last three months Trillium partnered with other researchers of the ISP group of the Universidad de Valencia to improve these models: it must be taken into account that this model was specifically designed for onboard processing and therefore it has significant room for improvement. We have been improving the quality of the data, the methods for testing and benchmarking the models, and we developed a new network architecture that improves the predictions especially in partially cloud covered areas. The ML4Floods package, the results of the UNOSAT pilot, and the latest models were presented in the AGU21 conference. Additionally, we are finalizing the models that will be deployed in the UNOSAT system and preparing a journal publication:

- **G. Mateo-Garcia**, Enrique Portales, Fei Liu, Edoardo Nemni, J. Emmanuel Johnson, Lucas Kruitwagen, Guy Schumann and Luis Gómez-Chova, Clouds aware flood extent segmentation for emergency response services. *American Geophysical Union 2021 Fall Meeting*. https://agu.confex.com/agu/fm21/meetingapp.cgi/Paper/981285

6.3.9 FDL 2021 Research Sprint: unsupervised onboard change detection

In 2021, I also participated in the Frontier Development Lab (FDL) Europe summer research sprint. This time my role was as a mentor/leader of one of the research teams. The challenge that I worked, alongside other six researchers, was about devising general-purpose onboard strategies to optimize data communication between the satellite and ground stations. For this we developed RaVÆn, a lightweight, unsupervised approach for change detection in satellite imagery based on Variational Auto-Encoders designed for on-board deployment. Applications, such as disaster management require low-latency satellite observations to speed up response after a catastrophic event. RaVÆnpre-processes the sampled data directly on the satellite and flags changed areas that could be used

[3] https://aiforgood.itu.int/about/un-ai-actions/unitar/

for prioritize downlinking, which would significantly shorten the response time. The system is tested on a dataset of Sentinel-2 time series over changing areas demonstrating that RaVÆnoutperforms different pixel-wise baselines. We also tested our approach on constrained hardware for assessing computational and memory trade-offs. This work has been presented in the Humanitarian Assistance and Disaster Response workshop in the NeurIPS conference:

- V. Ruzicka, A. Vaughan, D. De Martini, J. Fulton, V. Salvatelli, C. Bridges, **G. Mateo-Garcia**, V. Zantedeschi, Unsupervised Change Detection of Extreme Events Using ML On-Board. *Artificial Intelligence for Humanitarian Assistance and Dissaster Response Workshop, 35rd Conference on Neural Information Processing Systems (NeurIPS 2021), Vancouver, Canada*. arXiv: 2111.02995

6.3.10 Other related publications

This section contains a list of other related publications where I have been involved. Some of these are early versions of the published contributions that have been presented at international conferences. Others are collaborations with visitors and fellows at the Image and Signal Processing (ISP) group of the Universidad de Valencia.

Related journal papers

1. J. Munoz-Mari, E. Izquierdo-Verdiguier, M. Campos-Taberner, A. Perez-Suay, L. Gomez-Chova, **G. Mateo-Garcia**, A.B. Ruescas, V. Laparra, J.A. Padron, J. Amoros-Lopez and G. Camps-Valls, HyperLabelMe: A Web Platform for Benchmarking Remote-Sensing Image Classifiers. *IEEE Geoscience and Remote Sensing Magazine 2017* doi: 10.1109/MGRS.2017.2762476.
2. A. B. Ruescas, M. Hieronymi, **G. Mateo-Garcia**, S. Koponen, K. Kallio, and G. Camps-Valls, Machine Learning Regression Approaches for Colored Dissolved Organic Matter (CDOM) Retrieval with S2-MSI and S3-OLCI Simulated Data, *Remote Sensing*, vol. 10, no. 5, p. 786, May 2018, doi: 10.3390/rs10050786.
3. A. Wolanin, G. Camps-Valls, L. Gómez-Chova, **G. Mateo-García**, C. van der Tol, Y. Zhang, and L. Guanter, Estimating crop primary productivity with Sentinel-2 and Landsat 8 using machine learning methods trained with radiative transfer simulations, *Remote Sensing of Environment*, vol. 225, pp. 441–457, May 2019, doi: 10.1016/j.rse.2019.03.002.
4. A. Wolanin, **G. Mateo-García**, G. Camps-Valls, L. Gómez-Chova, M. Meroni, G. Duveiller, Y. Liangzhi and L. Guanter, Estimating and understanding crop yields with explainable deep learning in the Indian Wheat Belt, *Environmental Research Letters*, vol. 15, no. 2, p. 024019, Feb. 2020, doi: 10.1088/1748-9326/ab68ac.
5. R. Fernandez-Moran, L. Gómez-Chova, L. Alonso, **G. Mateo-García**, and D. López-Puigdollers, Towards a novel approach for Sentinel-3 synergistic OLCI/SLSTR cloud and cloud shadow detection based on stereo cloud-top height estimation, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 181, pp. 238–253, Nov. 2021, doi: 10.1016/j.isprsjprs.2021.09.013.

Conference papers

1. Adrián Pérez-Suay, Valero Laparra, **Gonzalo Mateo-García**, Jordi Muñoz-Marí, Luis Gómez-Chova and Gustau Camps-Valls, Fair Kernel Learning, *European Con-*

ference of Machine Learning and Knowledge Discovery in Databases, 2017. doi: 10.1007/978-3-319-71249-9_21

2. **G. Mateo-García**, L. Gómez-Chova and G. Camps-Valls Convolutional neural networks for multispectral image cloud masking *IGARSS 2017 - 2017 IEEE International Geoscience and Remote Sensing Symposium*. doi: 10.1109/IGARSS.2017.8127438
3. A. B. Ruescas, G. **Mateo-Garcia, G**. Camps-Valls, and M. Hieronymi, Retrieval of Case 2 Water Quality Parameters with Machine Learning, *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium* doi: 10.1109/IGARSS.2018.8518810
4. **G. Mateo-García** and L. Gómez-Chova Convolutional Neural Networks for Cloud Screening: Transfer Learning from Landsat-8 to Proba-V. *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*. doi: 10.1109/IGARSS.2018.8517975
5. **G. Mateo-García**, V. Laparra and L. Gómez-Chova, Optimizing Kernel Ridge Regression for Remote Sensing Problems. *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium* doi: 10.1109/IGARSS.2018.8518016
6. **G. Mateo-García**, Jose E. Adsuara, Adrián Pérez-Suay and Luis Gómez-Chova, Convolutional Long Short-Term Memory Network for Multitemporal Cloud Detection Over Landmarks. *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium* doi: 10.1109/IGARSS.2019.8897832
7. **G. Mateo-García**, Valero Laparra and Luis Gómez-Chova, Domain Adaptation of Landsat-8 and Proba-V Data Using Generative Adversarial Networks for Cloud Detection. *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium* doi: 10.1109/IGARSS.2019.8899193
8. María Piles, Valero Laparra Pérez-Muelas, Adrian Peréz-Suay, **Gonzalo Mateo-García**, Vicent Girbés-Juan, Maria Moreno-Llácer, Jordi Muñoz-Marí, Estrategia de enseñanza y aprendizaje de programación basada en la idea de 'hackathon'. *IN-RED 2021: VII Congreso de Innovación Educativa y Docencia en Red* doi: 10.4995/INRED2021.2021.13785

Book chapters

1. Gustau Camps-Valls, Luis Gómez-Chova, Valero Laparra, Luca Martino, **Gonzalo Mateo-García**, Jordi Muñoz-Marí, Daniel H. Svendsen and Jochem Verrelst, Chapter 2.13 - Statistical biophysical parameter retrieval and emulation with Gaussian processes. Data *Handling in Science and Technology*. doi: 10.1016/B978-0-444-63977-6.00015-8
2. **Gonzalo Mateo-García**, Valero Laparra, Christian Requena-Mesa and Luis Gómez-Chova, Generative Adversarial Networks in the Geosciences. *Deep Learning for the Earth Sciences* doi: 10.1002/9781119646181.ch3

6.3.11 Acknowledgements

This Thesis is part of the R&D projects TEC2016-77741-R and PID2019-109026RB-I00, funded by MCIN/AEI/10.13039/501100011033/ and the European Social Fund "ERDF a way to make Europe".

Esta Tesis es parte de los proyectos de I+D+i TEC2016-77741-R y PID2019-109026RB-I00, financiada por MCIN/AEI/10.13039/501100011033/ y el Fondo Social Europeo "FEDER una manera de hacer Europa".

7. Resumen global de la Tesis

7.1 Motivación y objetivos

La observación de la Tierra con sensores ópticos a bordo de satélites es una tecnología imprescindible para monitorizar nuestro planeta. En los últimos años estamos asistiendo a un incremento exponencial en el número de instrumentos ópticos puestos en órbita. Estos nuevos instrumentos, considerados en conjunto, adquieren un flujo de datos sin precedentes que nos proporcionan información de alta resolución espacial y temporal en todo el mundo. Un tratamiento adecuado de estos datos está mejorando nuestra comprensión de la biosfera (Wolanin et al., 2019), los océanos (Sauzède et al., 2020), nuestra capacidad para responder rápidamente a desastres naturales (Rudner et al., 2019) y, en última instancia, nos está ayudando a adaptarnos al cambio climático. Sin embargo, esta abundancia de datos también tiene sus desafíos, ya que los datos brutos proporcionados por estos sensores no son suficientes para abordar estos problemas. Por tanto, para explotar realmente estos datos, es necesario desarrollar productos precisos de teledetección que se adapten a cada uno de estos nuevos sensores.

La teledetección con sensores ópticos tiene actualmente dos propiedades singulares. En primer lugar, es **heterogénea**: hay muchos sensores diferentes con diferentes características espacio-espectrales que proporcionan diferentes *vistas* de la Tierra. En segundo lugar, los datos de teledetección son **abundantes** y en muchos casos **están disponibles libremente**: en 2008 el USGS abrió el archivo de Landsat lo que hizo que el resto de actores importantes de la comunidad espacial abrieran también sus productos. Hoy en día, plataformas como GEE brindan acceso a la comunidad científica a cientos de diferentes productos de teledetección sin coste alguno.

Estas características están causando un **cambio de paradigma** sobre cómo se desarrollan los productos de teledetección. Los productos de teledetección tradicionales **knowledge-based** se basan en un conocimiento muy profundo de la espectroscopia y la física óptica y se basan en comprender muy bien las características de los instrumentos y las órbitas de los sensores. A lo largo de esta Tesis, hemos sido testigos de cómo una nueva tendencia basada en datos (**data-driven** en inglés); que busca explotar la abundancia de éstos para desarrollar mejores productos. Este nuevo enfoque, tal como lo entendemos, tiene el potencial de sobrepasar los casos en los que los modelos físicos son computacionalmente costosos o no están bien resueltos para producir productos de teledetección más precisos. Además, dado que hay archivos de datos abiertos de imágenes de infinidad de sensores ópticos diferentes, se pueden crear nuevos productos utilizando éstos para nuevos sensores que todavía no se han lanzado. En esta Tesis, llamamos a este proceso: *transferir un modelo a un nuevo sensor*. Algunas de las contribuciones de esta Tesis están dedicadas a explorar metodologías para transferir modelos basados en datos entre sensores ópticos similares.

Un producto que es necesario para prácticamente todos los sensores ópticos que observan la Tierra (desde el rango visible del espectro electromagnético a 390 nm hasta el SWIR a 2500 nm) son las máscaras de nubes. Las nubes son masas de partículas de agua suspendidas en la atmósfera terrestre que reflejan la luz solar captada por nuestros satélites ópticos y nos impiden observar la superficie de la Tierra. Sabemos, por estos sensores, que las nubes son omnipresentes en la atmósfera y que cubren en promedio casi el 70% de la superficie de nuestro planeta. Por lo tanto, diferenciar entre píxeles de nubes y píxeles de superficie es el primer paso en la mayoría de las aplicaciones, ya sea en aquellas que observan la superficie (por ejemplo, Álvaro Moreno-Martínez et al. (2018)) o la atmósfera (por ejemplo, Zantedeschi et al. (2019)). Esto es necesario porque antes de comenzar cualquier análisis, estas aplicaciones necesitan saber si están observando la superficie o no, para descartar (o usar) esos píxeles. Por lo tanto, obtener máscaras de nubes precisas es vital para muchas aplicaciones de teledetección. Además, dado que muchas de estas aplicaciones utilizan grandes cantidades de datos, requieren que estas máscaras de nubes se generen automáticamente sin intervención humana; por ejemplo, en el trabajo de Wolanin et al. (2020) o Mateo-Sanchis et al. (2019), los autores proponen explotar series temporales de imágenes de un año para estimar el rendimiento de los cultivos en diferentes regiones (en estados de la India o en el ”cinturón del maíz” de Estados Unidos respectivamente); Por tanto, filtrar manualmente las imágenes con nubes en series temporales de imágenes tan largas con una resolución temporal 5 días no es factible.

Los algoritmos actuales *knowledge-based* para el enmascaramiento de nubes (también conocidos como basados en umbrales) tienen errores en muchas situaciones. Durante esta Tesis, hemos participado en varios proyectos con el objetivo de mejorar la calidad de la detección de nubes de los modelos operativos de Proba-V en el ‘Proba-V Cloud Detection Round Robin’ (Iannone et al., 2017) y para Landsat-8 y Sentinel-2 en ‘Cloud Masking Inter-comparison eXercise (CMIX)’ (European Space Agency, 2019). Estos proyectos ponen de manifiesto las deficiencias actuales de los algoritmos basados en umbrales y la necesidad de productos de detección de nubes novedosos y más precisos. Algunas de las contribuciones de esta Tesis tienen como objetivo mejorar la detección de nubes para satélites ópticos de observación de la Tierra.

La mayoría de los modelos basados en datos que proponemos para la detección de nubes y para la transferencia de aprendizaje en esta Tesis se basan en el aprendizaje profundo (*deep learning* en inglés). Entre las metodologías basadas en datos, elegimos el aprendizaje profundo porque: (a) Tiene la capacidad de escalar a conjuntos de datos arbitrariamente grandes sin que se estanque su rendimiento (es decir, su acierto sigue aumentando con mayores volúmenes de datos). Esto los hace especialmente adecuados para la teledetección donde los archivos de imágenes de satélite están en el orden de los petabytes. (b) Han mostrado un nivel de acierto sin precedentes en muchas tareas con imágenes naturales gracias al uso de redes convolucionales. En esta Tesis, los modelos que proponemos son Fully Convolutional Neural Networks (FCNN), estos modelos están especialmente diseñados para obtener predicciones por píxel (una categoría asignada a cada píxel de la imagen). En particular, las contribuciones de esta Tesis incluyen algunos de los primeros trabajos utilizando FCNN para la detección de nubes (Mateo-García et al., 2017) y para la adaptación de dominio con imágenes de distintos sensores (Mateo-García et al., 2020a).

En los siguientes apartados comentaremos brevemente las aportaciones de cada una de las cuatro publicaciones de la Tesis y su impacto.

7.2 Multi-temporal Cloud Masking in the Google Earth Engine

La primera contribución de esta Tesis, Mateo-García et al. (2018), propone modelos de detección de nubes multitemporales para imágenes Landsat-8. Esta primera contribución me sirvió para adentrarme en el mundo de la teledetección en general y el problema de la detección de nubes en particular. En este trabajo nos centramos en Landsat-8, que posiblemente sea el satélite con un mayor número de modelos de detección de nubes propuestos. Ampliando el trabajo de Gómez-Chova et al. (2017a), proponemos un algoritmo de detección de nubes multitemporal que se basa en dos fases: *estimación de la reflectividad de superficie* y *detección de cambios*. En la primera fase usamos imágenes previas sin nubes para estimar la reflectancia de la superficie probando diferentes métodos (añadiendo métodos sencillos a los propuestos en Gómez-Chova et al. (2017a)). En la segunda fase, comparamos la reflectividad estimada con la imagen actual y ajustamos una serie de umbrales globales que se utilizan para enmascarar las nubes posteriormente. El enfoque propuesto es un método simplificado muy similar a otros métodos multitemporales propuestos en la literatura (por ejemplo, los propuestos en Zhu & Woodcock (2014) o Hagolle et al. (2010)). La novedad del trabajo es que el algoritmo propuesto se puede implementar de manera eficiente en una plataforma donde los modelos multitemporales podrían ejecutarse operativamente: el Google Earth Engine (GEE). En esto nuestro trabajo difiere de los anteriormente propuestos los cuales requieren descargar imágenes previas sin nubes para la fase de estimación de superficie. Esto limita mucho la aplicabilidad del modelo de detección de nubes. En nuestro trabajo, podemos ejecutar el modelo de detección de nubes en cualquier ubicación nueva directamente en usando el GEE sin ninguna descarga de datos. Otra contribución significativa de este trabajo es su validación en un gran corpus de imágenes etiquetadas manualmente del conjunto de datos L8Biome (Foga et al., 2017). Los problemas anteriormente mencionados de ejecutar modelos multitemporales de detección de nubes en nuevas ubicaciones hacen que esos métodos estén validados en pocas ubicaciones; es común ver que estos trabajos solo se validan en un pequeño conjunto de imágenes y en muchos casos esta validación solo incluye la inspección visual de las máscaras. Los resultados de nuestra validación muestran un rendimiento significativamente mejor que los métodos basados en umbrales monotemporales operativos, como FMask (Zhu & Woodcock, 2012) o ACCA (Scaramuzza et al., 2012). Por último, hemos hecho un esfuerzo importante en proporcionar en código abierto los modelos asegurando la reproducibilidad de nuestros resultados. El repositorio de código https://github.com/IPL-UV/ee_ipl_uv contiene la implementación de la metodología propuesta y varios tutoriales con ejemplos. Además, la página web https://isp.uv.es/projects/cdc/viewer_l8_GEE.html muestra la comparación de nuestras máscaras con las máscaras etiquetadas manualmente del L8Biome en todas las adquisiciones utilizadas para la validación.

7.3 Transferring deep learning models for cloud detection between Landsat-8 and Proba-V

La segunda contribución de esta Tesis, Mateo-García et al. (2020b), demuestra la transferencia de modelos FCNN entre dos instrumentos ópticos (Proba-V y Landsat -8). La transferencia de aprendizaje de los modelos de detección de nubes basados en ML es uno de los principales objetivos de la Tesis doctoral. Por un lado, no se había demostrado antes si era posible y tampoco se habían cuantificado el acierto de estos modelos en comparación con modelos diseñados con los datos del propio satélite. Por otro lado, la transferencia de modelos de ML podría permitir el desarrollo de modelos de detección de nubes basados en ML para los sensores que todavía no han sido lanzados, lo que podría reducir significativamente la cantidad de datos de entrenamiento necesarios para crear dichos modelos.

En este trabajo realizamos un estudio muy completo de transferencia de aprendizaje transductiva e inductiva utilizando varios conjuntos de datos etiquetados de Proba-V y Landsat-8. Primero proponemos una transformación de adaptación de dominio basada en las propiedades físicas de los sensores que aplicamos a las imágenes Landsat-8 para hacerlas similares a las adquisiciones Proba-V (para que las imágenes tengan bandas espectrales similares y la misma resolución espacial nominal). Llamamos a las imágenes transformadas *Landsat-Upscaled*. Usando esta transformación llevamos a cabo tres tipos diferentes de experimentos de transferencia de aprendizaje: (a) *de Landsat-8 a Proba-V*, donde mostramos que los modelos entrenados solo con imágenes Landsat-Upscaled producen máscaras de nubes 5 puntos más precisas que las obtenidas con el modelo de Proba-V operacional basado en umbrales (Toté et al., 2018); (b) *de Proba-V a Landsat-8*, donde los modelos que usan solo datos Proba-V para el entrenamiento tienen una precisión similar a la de FMask en Landsat-8 obtenidas sobre el dataset L8Biome (87.79–89.77% de nuestros métodos contra 88.48% de FMask); y (c) *conjuntamente de Proba-V y Landsat-8 a Proba-V*, donde demostramos la transferencia de aprendizaje inductiva usando imágenes Landsat-Upscaled y muy pocas imágenes etiquetadas de Proba-V. En este caso, la precisión aumenta de 1 a 5 puntos en comparación con el uso únicamente del conjunto de datos etiquetado de Landsat-8.

Con este trabajo cumplimos con la mayoría de los objetivos de la Tesis Doctoral ya que demostramos en la misma publicación que los modelos FCNN producen máscaras de nubes muy precisas y que estos modelos se pueden transferir a un sensor compatible con pérdidas asumibles en acierto. También nos gustaría señalar que llevamos a cabo las comparaciones de los modelos propuestos de una manera muy completa y rigurosa: entrenamos diferentes copias de la misma red con diferentes semillas aleatorias para tener en cuenta de la estocasticidad de las redes neuronales y comparamos con otras trabajos de la literatura que utilizan FCNN (fundamentalmente Jeppesen et al. (2019) y Li et al. (2019)) y con los métodos operativos de detección de nubes de los satélites.

7.4 Cross sensor adversarial domain adaptation of Landsat-8 and Proba-V images for cloud detection

El tercer trabajo de esta Tesis, Mateo-García et al. (2020a), explora la adaptación de dominios basada en aprendizaje también entre Landsat-8 y Proba-V. El propósito de las transformaciones de adaptación de dominio es hacer que las imágenes en un dominio *origen* sean similares a las imágenes en un dominio *destino*.

Este trabajo fue motivado por las discrepancias observadas entre las imágenes Landsat-Upscaled (introducidas en la sección anterior) y las adquisiciones reales de Proba-V. En particular, observamos que la distribución de los colores en las distintas bandas espectrales y su resolución espacial medida por su transformada de Fourier es significativamente diferente incluso en imágenes adquiridas con pocos minutos de diferencia. Nuestra hipótesis en este trabajo es que estas diferencias son la causa de la caída en acierto de detección de nubes en imágenes Proba-V de modelos entrenados con datos Landsat-Upscaled (en la literatura de ML las diferencias en las entradas de los modelos entre entrenamiento y validación se llama *data-shift*).

Por lo tanto, en este trabajo, adaptamos uno de los últimos métodos de adaptación de dominio basados en ML que ha demostrado funcionar de manera muy eficiente en imágenes naturales y sintéticas para cerrar la brecha entre las imágenes Proba-V y Landsat-Upscaled. En particular, proponemos una variante de Cycle-Consistent Adversarial Domain Adaptation (CyCADA) (Hoffman et al., 2018) con una penalización específica para el caso de la teledetección que tiene como objetivo mantener los valores calibrados de las imágenes de satélite (llamado *identity loss* en el artículo). Una de las principales ventajas de CyCADA es que no requiere datos pareados; es decir, no requiere de imágenes adquiridas sobre la misma ubicación y al mismo tiempo para el entrenamiento; esto se debe a que la supervisión viene a través de discriminadores entrenados simultáneamente a los modelos adaptación de dominio. Esto es crucial para el uso previsto de la aplicación (enmascaramiento de nubes) ya que las nubes se mueven rápido e incluso en imágenes tomadas con diferencias de minutos se puede observar un desplazamiento de éstas.

Para este trabajo, recopilamos un conjunto de datos diverso de imágenes Landsat-Upscaled y Proba-V cercanas en el tiempo; esas imágenes se dividen en conjuntos de entrenamiento y validación de acuerdo a su adquisición para mantener la independencia del conjunto de validación del de entrenamiento. Sobre este conjunto, observamos que las estadísticas globales de las imágenes Proba-V adaptadas con el modelo entrenado son similares a las de Landsat-Upscaled. Además, vemos que el contenido de las imágenes Proba-V se mantiene mientras que los colores son más similares a las imágenes Landsat-Upscaled; y que las imágenes adaptadas son ligeramente más nítidas que las imágenes reales de Proba-V y con significativamente menos saturaciones. El conjunto de prueba completo se puede visualizar en https://isp.uv.es/projects/cloudsat/pvl8dagans/. Finalmente, cuando usamos la transformación de adaptación de dominio propuesta y aplicamos el modelo de detección de nubes entrenado en imágenes Landsat-Upscaled, observamos un aumento en la precisión de en torno a dos puntos.

Con este trabajo demostramos cómo construir transformaciones de adaptación de dominio entre dos sensores diferentes para transferir el *estilo* del otro sensor mientras se mantiene el *contenido* de la imagen observada. Es importante destacar que la metodología propuesta no requiere muestras pareadas, lo que facilita su aplicación a muchos problemas en teledetección. Usamos esta transformación para aplicar un modelo entrenado en un dominio origen sobre imágenes de un dominio destino; sin embargo, esta metodología podría usarse para otros problemas. Por ejemplo, podría usarse para producir productos de fusión armonizados donde las imágenes se transforman al estilo del sensor con mejor calidad radiométrica. Otra extensión, para sensores donde la corrección atmosférica es un desafío (es decir, para sensores con pocas bandas y sin bandas dedicadas para recuperaciones atmosféricas), podría ser producir imágenes adaptadas con corrección atmosférica; para esto podríamos usar escenas Landsat-8 de Nivel 2 (corregidas atmosféricamente) en lugar de reflectancias de Nivel 1 que son las que usamos en este artículo. Esto produciría imágenes adaptadas a Proba-V estadísticamente similares a las reflectancias de superficie de Landsat-8. No obstante, esta aplicación requeriría una validación muy completa antes de poder aplicarla operativamente.

Finalmente, para este trabajo también hicimos un esfuerzo significativo para asegurar la reproducibilidad de nuestros resultados. Para ello, publicamos el conjunto de datos de entrenamiento, una visualización con todas las imágenes de prueba con la metodología propuesta (https://isp.uv.es/projects/cloudsat/pvl8dagans/) y el código para entrenar y aplicar la transformación a una nueva escena de Proba-V (https://github.com/IPL-UV/pvl8dagans). Además, en ese repositorio incluimos partes del trabajo anterior, como la transformación para producir imágenes Landsat-Upscaled y los modelos entrenados en el conjunto de datos L8Biome mejorado.

7.5 Towards global flood mapping onboard low cost satellites with machine learning

La cuarta contribución de esta Tesis, Mateo-Garcia et al. (2021), describe un sistema de ML para segmentar inundaciones directamente a bordo de un satélite. El procesamiento a bordo es una de las últimas tendencias en teledetección. Ejecutar software para crear productos a bordo tiene ventajas como: la optimización del ancho de banda de comunicación del satélite (por ejemplo, para solo transmitir imágenes *útiles*, por ejemplo descartando las escenas con demasiadas nubes (Giuffrida et al., 2021)); para acelerar la descarga de información crítica observada por los sensores (por ejemplo, una fuga de metano, un incendio forestal o una inundación); o para activar una adquisición con un instrumento diferente (por ejemplo, para satélites con diferentes sensores, el procesado a bordo podría utilizarse para decidir el área sobre el que realizar una adquisición con un instrumento de mayor precisión espacial o espectral).

El procesamiento a bordo no es una idea nueva, la misión pionera EO-1 lanzada en noviembre de 2000 demostró algunas capacidades a bordo a través del ASE; Esta serie de experimentos incluyeron detección de nubes a bordo (Griggin et al., 2003) monitorización de hielo (Doggett et al., 2006) e incluso mapeo de inundaciones (Ip et al., 2006). El factor diferencial de las propuestas actuales es que los procesadores actuales contienen además hardware específico destinado a acelerar las aplicaciones de redes neuronales. Este es el caso de la misión ESA ΦSat-1 que contiene una cámara hiperespectral (HyperScout-2) y un chip Intel Movidius Myriad2 para acelerar las aplicaciones de visión por computadora. En este marco, nuestro trabajo describe un sistema para realizar segmentación de extensión de inundaciones que se testeamos en un hardware similar al que contiene ΦSat-1. El modelo propuesto para segmentar inundaciones son nuevamente FCNN, las cuales mostramos que funcionan muy bien para la detección de nubes en los trabajos referidos anteriormente. En particular, en este trabajo proponemos una arquitectura más simple que produce máscaras con mucha rapidez y que tiene en cuenta los requisitos del hardware de ΦSat-1.

Para entrenar el modelo de segmentación de inundaciones propuesto, necesitamos un conjunto de datos con imágenes y máscaras de segmentación de inundaciones. En el momento en el que desarrollamos este trabajo no existía un sensor hiperespectral operativo con datos suficientes sobre áreas inundadas para construir un modelo global. Por lo tanto, elegimos el satélite multiespectral Sentinel-2 como un buen proxy, ya que contiene diferentes bandas sobre la zona del espectro muestreado por HyperScout-2. Para entrenar estos modelos tampoco existían en ese momento un conjunto de datos curado de mapas de extensión de inundaciones e imágenes de Sentinel-2; por lo tanto, decidimos crear ese conjunto de datos nosotros mismos. El conjunto de datos recopilados, que llamamos *WorldFloods*, es quizás la contribución más importante de este trabajo.

WorldFloods contiene mapas de extensión de inundaciones creados por tres organizaciones diferentes y su imagen Sentinel-2 más cercana en tiempo después del evento. En *WorldFloods* hemos compilado un conjunto de datos con más de 400 mapas de extensión de inundaciones de unos 100 eventos de inundaciones verificados. Estos datos los usamos para entrenar y validar los modelos propuestos. Para crear un modelo para el sensor HyperScout-2 seguimos un enfoque de transferencia de aprendizaje muy similar al propuesto en la segunda contribución de esta Tesis. En particular, para simular adquisiciones de HyperScout-2, reducimos la escala de las imágenes de Sentinel-2 (10 m) a la resolución nominal de HyperScout-2 (80 m) y seleccionamos las nueve bandas de Sentinel-2 que se superponen a la zona del espectro muestreada por el sensor. Además, dado que se espera que el CubeSat tenga peor calidad radiométrica, introducimos ruido en las imágenes en el entrenamiento para imitar las degradaciones esperadas de las imágenes (emborronamiento por desplazamiento, descolocación de las bandas o ruido por tener un peor aislamiento térmico).

Finalmente, en este trabajo también hicimos un esfuerzo en asegurar la reproducibilidad y la accesibilidad (open Science): publicamos el conjunto de datos de *WorldFloods*, los modelos entrenados y el código para realizar inferencias y reproducir todos los experimentos: https://gitlab.com/frontierdevelopmentlab/disaster-prevention/cubesatfloods.

7.6 Conclusión

Esta Tesis propone diferentes métodos basados en datos, en la mayoría de los artículos que componen la Tesis utilizamos modelos de aprendizaje profundo para abordar diferentes problemas de teledetección. Los datos están en el centro de todos los métodos propuestos en esta Tesis y la preparación de estos datos ha sido la parte más crítica y tediosa (y subrepresentada) de todas. La correcta preparación de datos es fundamental para que estos sistemas funcionen correctamente. Los errores o una incorrecta comprensión de los datos de entrada se propagan y acentúan en las salidas de los modelos y, dada la complejidad de éstos, estos problemas son muy difíciles de detectar y rastrear (en Sambasivan et al. (2021) se desgranan muchos problemas de ”cascadas de datos” (data cascades) causadas por malas prácticas que han tenido grandes impactos en modelos desplegados en entornos reales). La preparación de datos también consume mucho tiempo debido a los volúmenes de datos que se necesitan para entrenar y validar los modelos. Aunque este coste podría reducirse mediante la transferencia de aprendizaje o modelos más eficientes en el uso de datos, conjuntos de datos grandes, globales y precisos son imprescindibles para validar y comparar cualquier metodología propuesta. Tratar con grandes volúmenes de datos requiere experiencia en desarrollo de software y conocimiento del dominio de aplicación; no obstante, comprender y ser capaz de inspeccionar los datos con los que trabajamos es de suma importancia para proponer soluciones útiles y construir modelos precisos (ver, por ejemplo, Karpathy (2019) para una guía de buenas prácticas de tratamiento de datos; allí el autor sugiere: *"convertirse en uno con los datos"* esto requiere inspeccionar concienzudamente las muestras del conjunto de datos con el que trabajemos antes de realizar cualquier modelado). En resumen, es importante destacar que los datos son la base del método empírico científico y, por tanto, como comunidad, debemos hacer un esfuerzo para fomentar y recompensar las buenas prácticas de ingeniería de datos.

Las contribuciones de estos trabajos se encuentran entre los campos del aprendizaje máquina y la teledetección. En general, se podría decir que la mayoría de nuestras contribuciones consisten en *'tomar un modelo exitoso de visión por computador y adaptarlo a datos de teledetección'*. En esencia esto es cierto, sin embargo, todos los trabajos que componen esta Tesis están diseñados específicamente para el problema de teledetección que abordan y todos los modelos han sido validados a fondo pensando en los casos reales que estos modelos podrían tener. Cabe señalar que las adaptaciones de los métodos de visión por ordenador que proponemos se adaptan al problema específico de teledetección que pretendemos solucionar. Por ejemplo, la tercera contribución de esta Tesis, Mateo-García et al. (2020a), propone una transformación de adaptación de dominio que busca mantener la calibración radiométrica de los sensores; esto es necesario si deseamos que la transformación propuesta sea utilizada por otras aplicaciones posteriores que puedan requerir reflectancias calibradas. De una manera más amplia, todos los modelos de esta Tesis han sido validados con datos representativos de buena calidad que hemos seleccionado cuidadosamente. Estos datos se han elegido para que sean representativos de las condiciones globales de la Tierra que cada sensor podría observar.

Para concluir, en esta Tesis proponemos diferentes soluciones para mejorar el enmascaramiento de nubes y desarrollar metodologías para transferir modelos de aprendizaje profundo a través de diferentes sensores. En particular, demostramos que podríamos mejorar la detección de nubes utilizando la dimensión temporal o espacial de las imágenes de satélite ópticas. También mostramos que podríamos transferir modelos FCNN entrenados con datos de un satélite óptico a otro con características espacio-espectrales compatibles y que las transformaciones de adaptación de dominio basadas en aprendizaje máquina mejoran la capacidad de transferencia de estos métodos. Finalmente, mostramos que estos modelos podrían incluso ser implementados a bordo de micro-satélites (CubeSats) lo cual abre la puerta a desarrollos futuros.

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. URL: http://tensorflow.org/.

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning From Data*. AMLBook.

Allen, G. H., & Pavelsky, T. M. (2018). Global extent of rivers and streams. *Science*, . doi: 10.1126/science.aat0636.

Bottou, L. (1998). Online algorithms and stochastic approximations. In D. Saad (Ed.), *Online Learning and Neural Networks*. Cambridge, UK: Cambridge University Press. URL: http://leon.bottou.org/papers/bottou-98x revised, oct 2012.

Bulgin, C. E., Merchant, C. J., Ghent, D., Klüser, L., Popp, T., Poulsen, C., & Sogacheva, L. (2018). Quantifying Uncertainty in Satellite-Retrieved Land Surface Temperature from Cloud Detection Errors. *Remote Sensing*, *10*, 616. doi: 10.3390/rs10040616. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

Candra, D. S., Phinn, S., & Scarth, P. (2017). Cloud and cloud shadow removal of landsat 8 images using multitemporal cloud removal method. In *2017 6th International Conference on Agro-Geoinformatics* (pp. 1–5). doi: 10.1109/Agro-Geoinformatics.2017.8047007.

Cervera-Alamar, M. (2019). *Staphylococcus aureus pathogenicity islands mobilisation and their role in chromosomally-encoded virulence gene expression*. Ph.D. thesis Universidad de Valencia.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2015). Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. In *International Conference on Learning Representations (ICLR)* (pp. 1–14). ArXiv: 1412.7062.

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1800–1807). doi: 10.1109/CVPR.2017.195.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Deudon, M., Kalaitzis, A., Goytom, I., Arefin, M. R., Lin, Z., Sankaran, K., Michalski, V., Kahou, S. E., Cornebise, J., & Bengio, Y. (2020). HighRes-net: Recursive Fusion for Multi-Frame Super-Resolution of Satellite Imagery. *arXiv:2002.06460 [cs, eess, stat]*, . ArXiv: 2002.06460.

Doggett, T., Greeley, R., Chien, S., Castano, R., Cichy, B., Davies, A., Rabideau, G., Sherwood, R., Tran, D., Baker, V., Dohm, J., & Ip, F. (2006). Autonomous detection of cryospheric change with hyperion on-board earth observing-1. *Remote Sensing of Environment*, *101*, 447 – 462. URL: http://www.sciencedirect.com/science/article/pii/S0034425706000216. doi: 10.1016/j.rse.2005.11.014.

Drönner, J., Korfhage, N., Egli, S., Mühling, M., Thies, B., Bendix, J., Freisleben, B., & Seeger, B. (2018). Fast Cloud Segmentation Using Convolutional Neural Networks. *Remote Sensing*, *10*, 1782. doi: 10.3390/rs10111782.

European Space Agency (2019). CEOS-WGCV ACIX II CMIX Atmospheric Correction Inter-comparison Exercise Cloud Masking Inter-comparison Exercise 2nd workshop. URL: https://earth.esa.int/eogateway/events/ceos-wgcv-acix-ii-cmix-atmospheric-correction-inter-comparison-exercise-cloud-masking-inter-comparison-exercise-2nd-workshop online; accessed 14 October 2021.

Farabet, C., Couprie, C., Najman, L., & LeCun, Y. (2013). Learning Hierarchical Features for Scene Labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*, 1915–1929. doi: 10.1109/TPAMI.2012.231.

Foga, S., Scaramuzza, P. L., Guo, S., Zhu, Z., Dilley, R. D., Beckmann, T., Schmidt, G. L., Dwyer, J. L., Hughes, M. J., & Laue, B. (2017). Cloud detection algorithm comparison and validation for operational Landsat data products. *Remote Sensing of Environment*, *194*, 379–390. doi: 10.1016/j.rse.2017.03.026.

Frantz, D., Röder, A., Udelhoven, T., & Schmidt, M. (2015). Enhancing the Detectability of Clouds and Their Shadows in Multitemporal Dryland Landsat Imagery: Extending Fmask. *IEEE Geoscience and Remote Sensing Letters*, *12*, 1242–1246. doi: 10.1109/LGRS.2015.2390673.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., & Lempitsky, V. (2016). Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, *17*, 1–35. URL: http://jmlr.org/papers/v17/15-239.html.

Giuffrida, G., Diana, L., de Gioia, F., Benelli, G., Meoni, G., Donati, M., & Fanucci, L. (2020). CloudScout: A Deep Neural Network for On-Board Cloud Detection on Hyperspectral Images. *Remote Sensing*, *12*, 2205. doi: 10.3390/rs12142205.

Giuffrida, G., Fanucci, L., Meoni, G., Batic, M., Buckley, L., Dunne, A., Van Dijk, C., Esposito, M., Hefele, J., Vercruyssen, N., Furano, G., Pastena, M., & Aschbacher, J. (2021). The phi-Sat-1 mission: the first on-board deep neural network demonstrator for satellite earth observation. *IEEE Transactions on Geoscience and Remote Sensing*, (pp. 1–1). doi: 10.1109/TGRS.2021.3125567. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

Gómez-Chova, L., Camps-Valls, G., Bruzzone, L., & Calpe-Maravilla, J. (2010). Mean map kernel methods for semisupervised cloud classification. *IEEE Trans. on Geoscience and Remote Sensing*, *48*, 207–220. doi: 10.1109/TGRS.2009.2026425.

Gomez-Chova, L., Camps-Valls, G., Calpe-Maravilla, J., Guanter, L., & Moreno, J. (2007). Cloud-Screening Algorithm for ENVISAT/MERIS Multispectral Images. *IEEE Transactions on Geoscience and Remote Sensing*, *45*, 4105–4118. doi: 10.1109/TGRS.2007.905312.

Gómez-Chova, L., Muñoz-Marí, J., Amorós-López, J., Izquierdo-Verdiguier, E., & Camps-Valls, G. (2013). Advances in synergy of AATSR-MERIS sensors for cloud detection. In *Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International* (pp. 4391–4394). doi: 10.1109/IGARSS.2013.6723808.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. URL: http://www.deeplearningbook.org.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, *202*, 18–27. doi: 10.1016/j.rse.2017.06.031.

Griggin, M., Burke, H., Mandl, D., & Miller, J. (2003). Cloud cover detection algorithm for EO-1 Hyperion imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2003)* (pp. 86–89 vol.1). volume 1. doi: 10.1109/IGARSS.2003.1293687.

Gómez-Chova, L., Amorós-López, J., Mateo-García, G., Muñoz-Marí, J., & Camps-Valls, G. (2017a). Cloud masking and removal in remote sensing image time series. *Journal of Applied Remote Sensing*, *11*, 015005. doi: 10.1117/1.JRS.11.015005.

Gómez-Chova, L., Mateo-García, G., Muñoz-Marí, J., & Camps-Valls, G. (2017b). Cloud detection machine learning algorithms for PROBA-V. In *IGARSS 2017 - 2017 IEEE International Geoscience and Remote Sensing Symposium* (pp. 2251–2254). doi: 10.1109/IGARSS.2017.8127437.

Hagolle, O., Huc, M., Pascual, D. V., & Dedieu, G. (2010). A multi-temporal method for cloud detection, applied to FORMOSAT-2, VENμS, LANDSAT and SENTINEL-2 images. *Remote Sensing of Environment*, *114*, 1747–1755. doi: 10.1016/j.rse.2010.03.002.

Hardt, M., & Recht, B. (2021). *Patterns, predictions, and actions: A story about machine learning*. https://mlstory.org. `arXiv:2102.05242`.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., & Darrell, T. (2018). CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In *ICML 2018* (pp. 1989–1998).

Hollstein, A., Segl, K., Guanter, L., Brell, M., & Enesco, M. (2016). Ready-to-Use Methods for the Detection of Clouds, Cirrus, Snow, Shadow, Water and Clear Sky Pixels in Sentinel-2 MSI Images. *Remote Sensing*, *8*, 666.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, *160*, 106–154.2. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/.

Hughes, M. J., & Hayes, D. J. (2014). Automated Detection of Cloud and Cloud Shadow in Single-Date Landsat Imagery Using Neural Networks and Spatial Post-Processing. *Remote Sensing*, *6*, 4907–4926. doi: 10.3390/rs6064907.

Iannone, R. Q., Niro, F., Goryl, P., Dransfeld, S., Hoersch, B., Stelzer, K., Kirches, G., Paperin, M., Brockmann, C., Gómez-Chova, L., Mateo-García, G., Preusker, R., Fischer, J., Amato, U., Serio, C., Gangkofner, U., Berthelot, B., Iordache, M. D., Bertels, L., Wolters, E., Dierckx, W., Benhadj, I., & Swinnen, E. (2017). Proba-V cloud detection Round Robin: Validation results and recommendations. In *2017 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp)* (pp. 1–8). doi: 10.1109/Multi-Temp.2017.8035219.

Ip, F., Dohm, J., Baker, V., Doggett, T., Davies, A., Castaño, R., Chien, S., Cichy, B., Greeley, R., Sherwood, R., Tran, D., & Rabideau, G. (2006). Flood detection and monitoring with the autonomous sciencecraft experiment onboard eo-1. *Remote Sensing of Environment*, *101*, 463 – 481. doi: 10.1016/j.rse.2005.12.018.

Ishida, H., Oishi, Y., Morita, K., Moriwaki, K., & Nakajima, T. Y. (2018). Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions. *Remote Sensing of Environment*, *205*, 390–407. doi: 10.1016/j.rse.2017.11.003.

Jeppesen, J. H., Jacobsen, R. H., Inceoglu, F., & Toftegaard, T. S. (2019). A cloud detection algorithm for satellite imagery based on deep learning. *Remote Sensing of Environment*, *229*, 247–259. doi: 10.1016/j.rse.2019.03.039.

Karpathy, A. (2016). Yes, you should understand backprop. https://karpathy.medium.com/yes-you-should-understand-backprop-e2f06eab496b. Online; accessed 27 October 2021.

Karpathy, A. (2019). A recipe for training neural networks. http://karpathy.github.io/2019/04/25/recipe/. Online; accessed 29 December 2021.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25* (pp. 1097–1105).

Kruitwagen, L., Story, K. T., Friedrich, J., Byers, L., Skillman, S., & Hepburn, C. (2021). A global inventory of photovoltaic solar energy generating units. *Nature*, *598*, 604–610. doi: 10.1038/s41586-021-03957-7.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural*

Computation, *1*, 541–551. doi: 10.1162/neco.1989.1.4.541. Conference Name: Neural Computation.

Lee, J., Weger, R., Sengupta, S., & Welch, R. (1990). A neural network approach to cloud classification. *IEEE Transactions on Geoscience and Remote Sensing*, *28*, 846–855. doi: 10.1109/36.58972. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

Li, Z., Shen, H., Cheng, Q., Liu, Y., You, S., & He, Z. (2019). Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. *ISPRS Journal of Photogrammetry and Remote Sensing*, *150*, 197–212. doi: 10.1016/j.isprsjprs.2019.02.017.

Lisens, G., Kempencers, P., Fierens, F., & Van Rensbergen, J. (2000). Development of cloud, snow, and shadow masking algorithms for VEGETATION imagery. In *IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No.00CH37120)* (pp. 834–836 vol.2). volume 2. doi: 10.1109/IGARSS.2000.861719.

Lohmann, U., Lüönd, F., & Mahrt, F. (2016). *An Introduction to Clouds: From the Microscale to Climate*. Cambridge University Press. doi: 10.1017/CBO9781139087513.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3431–3440). doi: 10.1109/CVPR.2015.7298965.

López-Puigdollers, D., Mateo-García, G., & Gómez-Chova, L. (2021). Benchmarking Deep Learning Models for Cloud Detection in Landsat-8 and Sentinel-2 Images. *Remote Sensing*, *13*, 992. doi: 10.3390/rs13050992.

Mateo-Garcia, G., Veitch-Michaelis, J., Smith, L., Oprea, S. V., Schumann, G., Gal, Y., Baydin, A. G., & Backes, D. (2021). Towards global flood mapping onboard low cost satellites with machine learning. *Scientific Reports*, *11*, 7249. doi: 10.1038/s41598-021-86650-z.

Mateo-García, G., Gómez-Chova, L., Amorós-López, J., Muñoz-Marí, J., & Camps-Valls, G. (2018). Multitemporal Cloud Masking in the Google Earth Engine. *Remote Sensing*, *10*, 1079. doi: 10.3390/rs10071079.

Mateo-García, G., Gómez-Chova, L., & Camps-Valls, G. (2017). Convolutional neural networks for multispectral image cloud masking. In *IGARSS 2017 - 2017 IEEE International Geoscience and Remote Sensing Symposium* (pp. 2255–2258). doi: 10.1109/IGARSS.2017.8127438.

Mateo-García, G., Laparra, V., & Gómez-Chova, L. (2019). Domain Adaptation of Landsat-8 and Proba-V Data Using Generative Adversarial Networks for Cloud Detection. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium* (pp. 712–715). doi: 10.1109/IGARSS.2019.8899193 iSSN: 2153-6996.

Mateo-García, G., Laparra, V., López-Puigdollers, D., & Gómez-Chova, L. (2020a). Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images for Cloud Detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 747–761. doi: 10.1109/JSTARS.2020.3031741.

Mateo-García, G., Laparra, V., López-Puigdollers, D., & Gómez-Chova, L. (2020b). Transferring deep learning models for cloud detection between Landsat-8 and Proba-V. *ISPRS Journal of Photogrammetry and Remote Sensing*, *160*, 1–17. doi: 10.1016/j.isprsjprs.2019.11.024.

Mateo-Sanchis, A., Piles, M., Muñoz-Marí, J., Adsuara, J. E., Pérez-Suay, A., & Camps-Valls, G. (2019). Synergistic integration of optical and microwave satellite data for crop yield estimation. *Remote Sensing of Environment*, *234*, 111460. doi: https://doi.org/10.1016/j.rse.2019.111460.

Álvaro Moreno-Martínez, Camps-Valls, G., Kattge, J., Robinson, N., Reichstein, M., van Bodegom, P., Kramer, K., Cornelissen, J. H. C., Reich, P., Bahn, M., Ülo Niinemets, Peñuelas, J., Craine, J. M., Cerabolini, B. E., Minden, V., Laughlin, D. C., Sack, L., Allred, B., Baraloto, C., Byun, C., Soudzilovskaia, N. A., & Running, S. W. (2018). A methodology to derive global maps of leaf traits using remote sensing and climate data. *Remote Sensing of Environment*, *218*, 69–88. doi: https://doi.org/10.1016/j.rse.2018.09.006.

Murphy, K. P. (2013). *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press.

Nemni, E., Bullock, J., Belabbes, S., & Bromley, L. (2020). Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery. *Remote Sensing*, *12*, 2532. doi: 10.3390/rs12162532.

Neumann, M., Pinto, A. S., Zhai, X., & Houlsby, N. (2020). Training General Representations for Remote Sensing Using in-Domain Knowledge. In *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium* (pp. 6730–6733). doi: 10.1109/IGARSS39084.2020.9324501.

Ng, A. (2017). *Machine Learning Yearning*. Online Draft. URL: http://www.mlyearning.org/,/bib/ng/ng2017mlyearning/Ng_MLY01_13.pdf.

Pan, S. J., & Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*, 1345–1359. doi: 10.1109/TKDE.2009.191.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32* (pp. 8024–8035).

Pekel, J.-F., Cottam, A., Gorelick, N., & Belward, A. S. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, *540*, 418–422. doi: 10.1038/nature20584.

Pipia, L., Muñoz-Marí, J., Amin, E., Belda, S., Camps-Valls, G., & Verrelst, J. (2019). Fusing optical and SAR time series for LAI gap filling with multioutput Gaussian processes. *Remote Sensing of Environment*, *235*, 111452. doi: 10.1016/j.rse.2019.111452.

Ploton, P., Mortier, F., Réjou-Méchain, M., Barbier, N., Picard, N., Rossi, V., Dormann, C., Cornu, G., Viennois, G., Bayol, N., Lyapustin, A., Gourlet-Fleury, S., & Pélissier, R. (2020). Spatial validation reveals poor predictive performance of large-scale ecological mapping models. *Nature Communications*, *11*, 4540. doi: 10.1038/s41467-020-18321-y.

Robbins, H., & Monro, S. (1951). A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, *22*, 400 – 407. doi: 10.1214/aoms/1177729586.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* LNCS (pp. 234–241). Springer, Cham. doi: 10.1007/978-3-319-24574-4_28.

Rudner, T. G. J., Rußwurm, M., Fil, J., Pelich, R., Bischke, B., Kopackova, V., & Biliński, P. (2019). Multi3Net: Segmenting Flooded Buildings via Fusion of Multiresolution, Multisensor, and Multitemporal Satellite Imagery. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 702–709. doi: 10.1609/aaai.v33i01.3301702. Number: 01.

Ruescas, A. B., Hieronymi, M., Mateo-Garcia, G., Koponen, S., Kallio, K., & Camps-Valls, G. (2018). Machine Learning Regression Approaches for Colored Dissolved Organic Matter (CDOM) Retrieval with S2-MSI and S3-OLCI Simulated Data. *Remote Sensing*, *10*, 786. doi: 10.3390/rs10050786.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536. doi: 10.1038/323533a0.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, *115*, 211–252. doi: 10.1007/s11263-015-0816-y.

Sambasivan, N., Kapania, S., Highfill, H., Akrong, D., Paritosh, P., & Aroyo, L. M. (2021). Everyone wants to do the model work, not the data work: Data Cascades in High-Stakes AI. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* 39 (pp. 1–15). New York, NY, USA: Association for Computing Machinery.

Sauzède, R., Johnson, J. E., Claustre, H., Camps-Valls, G., & Ruescas, A. B. (2020). Estimation of Oceanic Particulate Organic Carbon with Machine Learning. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *5.2*, 949–956. doi: 10.5194/isprs-annals-V-2-2020-949-2020. ADS Bibcode: 2020ISPAn.5.2..949S.

Scaramuzza, P. L., Bouchard, M. A., & Dwyer, J. L. (2012). Development of the Landsat Data Continuity Mission Cloud-Cover Assessment Algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, *50*, 1140–1154. doi: 10.1109/TGRS.2011.2164087.

Schuegraf, P., & Bittner, K. (2019). Automatic Building Footprint Extraction from Multi-Resolution Remote Sensing Images Using a Hybrid FCN. *ISPRS International Journal of Geo-Information*, *8*, 191. doi: 10.3390/ijgi8040191.

Sentinel Hub team (2017). Sentinel Hub's cloud detector for Sentinel-2 imagery. https://github.com/sentinel-hub/sentinel2-cloud-detector and https://medium.com/sentinel-hub/improving-cloud-detection-with-machine-learning-c09dc5d7cf13. Accessed 28 January 2020.

Skakun, S., Vermote, E. F., Artigas, A. E. S., Rountree, W. H., & Roger, J.-C. (2021). An experimental sky-image-derived cloud validation dataset for Sentinel-2 and Landsat 8 satellites over NASA GSFC. *International Journal of Applied Earth Observation and Geoinformation*, *95*, 102253. doi: 10.1016/j.jag.2020.102253.

Skakun, S., Wevers, J., Brockmann, C., Doxani, G., Aleksandrov, M., Batic, M., Frantz, D., Gascon Roca, F., Gómez-Chova, L., Hagolle, O., López-Puigdollers, D., Louis, J., Lubej, M., Mateo-García, G., Osman, J., Peressutti, D., Pflug, B., Puc, J., Richter, R., Roger, J.-C., Scaramuzza, P., Vermote, E., Vesel, N., Zupanc, A., & Zust, L. (Submitted). Cloud Mask Intercomparison eXercise (CMIX): an evaluation of cloud masking algorithms for Landsat 8 and Sentinel-2. *Remote Sensing of Environment*, .

Slawinski, O., Kowalski, J., & Cornillon, P. (1991). A neural network approach to cloud detection in AVHRR images. In *IJCNN-91-Seattle International Joint Conference on Neural Networks* (pp. 283–288 vol.1). volume i. doi: 10.1109/IJCNN.1991.155190.

Stubenrauch, C., Rossow, W., Kinne, S., Ackerman, S., Cesana, G., Chepfer, H., Girolamo, L., Getzewich, B., Guignard, A., Heidinger, A., Maddux, B., Menzel, W., Minnis, P., Pearl, C., Platnick, S., Poulsen, C., Riedi, J., Sun-Mack, S., Walther, A., & Zhao, G. (2013). Assessment of global cloud datasets from satellites: Project and database initiated by the gewex radiation panel. *Bulletin of the American Meteorological Society*, *94*, 1031–1049. doi: 10.1175/BAMS-D-12-00117.1.

Sumbul, G., Charfuelan, M., Demir, B., & Markl, V. (2019). Bigearthnet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium* (pp. 5901–5904). doi: 10.1109/IGARSS.2019.8900532.

Sumbul, G., de Wall, A., Kreuziger, T., Marcelino, F., Costa, H., Benevides, P., Caetano, M., Demir, B., & Markl, V. (2021). BigEarthNet-MM: A Large-Scale, Multimodal, Multilabel Benchmark Archive for Remote Sensing Image Classification and Retrieval [Software and Data Sets]. *IEEE Geoscience and Remote Sensing Magazine*, *9*, 174–180. doi: 10.1109/MGRS.2021.3089174.

Tangseng, P., Wu, Z., & Yamaguchi, K. (2017). Looking at Outfit to Parse Clothing. *arXiv:1703.01386 [cs]*, . ArXiv: 1703.01386.

Tatem, A. J., Goetz, S. J., & Hay, S. I. (2008). Fifty Years of Earth Observation Satellites. *American scientist*, *96*, 390–398. doi: 10.1511/2008.74.390.

Torralba, A., & Efros, A. A. (2011). Unbiased look at dataset bias. In *CVPR 2011* (pp. 1521–1528). doi: 10.1109/CVPR.2011.5995347.

Toté, C., Swinnen, E., Sterckx, S., Adriaensen, S., Benhadj, I., Iordache, M.-D., Bertels, L., Kirches, G., Stelzer, K., Dierckx, W., Van den Heuvel, L., Clarijs, D., & Niro, F. (2018). Evaluation of PROBA-V Collection 1: Refined Radiometry, Geometry, and Cloud Screening. *Remote Sensing*, *10*, 1375. doi: 10.3390/rs10091375.

Toté, C., Swinnen, E., Sterckx, S., Benhadj, I., Dierckx, W., Gómez-Chova, L., Ramon, D., Stelzer, K., Van den Heuvel, L., Clarijs, D., & Niro, F. (2021). The Reprocessed Proba-V Collection 2: Product Validation. In *IGARSS 2021 - IEEE International Geoscience and Remote Sensing Symposium* (pp. 8084–8086). doi: 10.1109/IGARSS47720.2021.9553376.

Tuia, D., Persello, C., & Bruzzone, L. (2016). Domain adaptation for the classification of remote sensing data: An overview of recent advances. *IEEE Geoscience and Remote Sensing Magazine*, *4*, 41–57. doi: 10.1109/MGRS.2016.2548504.

UCS Satellite Database (2021). UCS Satellite Database. https://www.ucsusa.org/resources/satellite-database. Accessed: 2021-12-12.

Wieland, M., Li, Y., & Martinis, S. (2019). Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network. *Remote Sensing of Environment*, *230*, 111203. doi: 10.1016/j.rse.2019.05.022.

Wolanin, A., Camps-Valls, G., Gómez-Chova, L., Mateo-García, G., van der Tol, C., Zhang, Y., & Guanter, L. (2019). Estimating crop primary productivity with Sentinel-2 and Landsat 8 using machine learning methods trained with radiative transfer simulations. *Remote Sensing of Environment*, *225*, 441–457. doi: 10.1016/j.rse.2019.03.002.

Wolanin, A., Mateo-García, G., Camps-Valls, G., Gómez-Chova, L., Meroni, M., Duveiller, G., Liangzhi, Y., & Guanter, L. (2020). Estimating and understanding crop yields with explainable deep learning in the Indian Wheat Belt. *Environmental Research Letters*, *15*, 024019. doi: 10.1088/1748-9326/ab68ac.

Yhann, S., & Simpson, J. (1995). Application of neural networks to AVHRR cloud segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, *33*, 590–604. doi: 10.1109/36.387575. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

Zantedeschi, V., Falasca, F., Douglas, A., Strange, R., Kusner, M. J., & Watson-Parris, D. (2019). Cumulo: A dataset for learning cloud classes. In *Tackling Climate Change with Machine Learning, 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada*. ArXiv: 1911.04227.

Zhu, Z., & Woodcock, C. E. (2012). Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sensing of Environment*, *118*, 83–94. doi: 10.1016/j.rse.2011.10.028.

Zhu, Z., & Woodcock, C. E. (2014). Automated cloud, cloud shadow, and snow detection in multitemporal Landsat data: An algorithm designed specifically for monitoring land cover change. *Remote Sensing of Environment*, *152*, 217–234. doi: 10.1016/j.rse.2014.06.012.

Appendix: Scientific Publications

Publication I

G. Mateo-García, L. Gómez-Chova, J. Amorós-López, J. Muñoz-Marí, and G. Camps-Valls, 2018. Multitemporal Cloud Masking in the Google Earth Engine. *Remote Sensing*, vol. 10, no. 7, 1079, pp.68-81, doi: 10.3390/rs10071079.

Q1: Environmental Sciences, Q1: Geosciences, Multidisciplinary, Q1: Remote Sensing, Q1: Imaging Science & Photographic Technology, IF = 4.848

Publication II

G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, Transferring deep learning models for cloud detection between Landsat-8 and Proba-V, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 160, pp. 1–17, Feb. 2020, doi: 10.1016/j.isprsjprs.2019.11.024.

Q1: Geography, Physical, Q1: Geosciences, Multidisciplinary, Q1: Remote Sensing, Q1: Imaging Science & Photographic Technology, IF = 8.979

Publication III

G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images for Cloud Detection, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 747–761, Oct. 2020, doi: 10.1109/JSTARS.2020.3031741.

Q2: Geography Physical, Q2: Remote Sensing, Q2: Imaging Science & Photographic Technology, Q1: Engineering, Electrical and Electronic, IF = 3.784

Publication IV

G. Mateo-Garcia, J. Veitch-Michaelis, L. Smith, S. Oprea, G. Schumann, Y. Gal, A. Güneş Baydin, and D. Backes, Towards global flood mapping onboard low cost satellites with machine learning, *Scientific Reports*, vol. 11, no. 1, Mar. 2021, doi: 10.1038/s41598-021-86650-z.

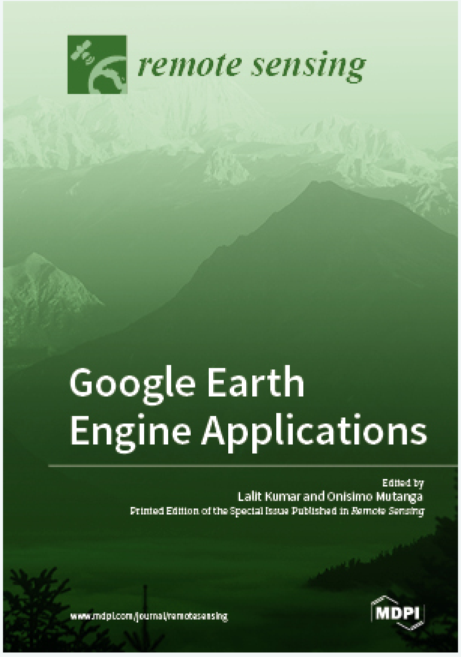
Q1: Multidisciplinary Sciences, IF = 4.379

Publication I: Multitemporal Cloud Masking in the Google Earth Engine

Publication I

G. Mateo-García, L. Gómez-Chova, J. Amorós-López, J. Muñoz-Marí, and G. Camps-Valls, 2018. Multitemporal Cloud Masking in the Google Earth Engine. *Remote Sensing*, vol. 10, no. 7, 1079, pp.68-81, doi: 10.3390/rs10071079.

Q1: Environmental Sciences, Q1: Geosciences, Multidisciplinary, Q1: Remote Sensing, Q1: Imaging Science & Photographic Technology, IF = 4.848


remote sensing
Google Earth
Engine Applications
Edited by
Lalit Kumar and Onisimo Mutanga
Printed Edition of the Special Issue Published in *Remote Sensing*
www.mdpi.com/journal/remotesensing
MDPI

remote sensing

MDPI

Article

Multitemporal Cloud Masking in the Google Earth Engine

Gonzalo Mateo-García *, Luis Gómez-Chova, Julia Amorós-López, Jordi Muñoz-Marí and Gustau Camps-Valls

Image Processing Laboratory, University of Valencia, 46980 Paterna, Spain; luis.gomez-chova@uv.es (L.G.-C.); julia.amoros@uv.es (J.A.-L.); jordi.munoz@uv.es (J.M.-M.); gustau.camps@uv.es (G.C.-V.)

* Correspondence: gonzalo.mateo-garcia@uv.es

Received: 28 May 2018; Accepted: 5 July 2018; Published: 6 July 2018

Abstract: The exploitation of Earth observation satellite images acquired by optical instruments requires an automatic and accurate cloud detection. Multitemporal approaches to cloud detection are usually more powerful than their single scene counterparts since the presence of clouds varies greatly from one acquisition to another whereas surface can be assumed stationary in a broad sense. However, two practical limitations usually hamper their operational use: the access to the complete satellite image archive and the required computational power. This work presents a cloud detection and removal methodology implemented in the Google Earth Engine (GEE) cloud computing platform in order to meet these requirements. The proposed methodology is tested for the Landsat-8 mission over a large collection of manually labeled cloud masks from the Biome dataset. The quantitative results show state-of-the-art performance compared with mono-temporal standard approaches, such as FMask and ACCA algorithms, yielding improvements between 4–5% in classification accuracy and 3–10% in commission errors. The algorithm implementation within the Google Earth Engine and the generated cloud masks for all test images are released for interested readers.

Keywords: image time series; multitemporal analysis; change detection; cloud masking; Landsat-8; Google Earth Engine (GEE)

1. Introduction

Reliable and accurate cloud detection is a mandatory first step towards developing remote sensing products based on optical satellite images. Undetected clouds in the acquired satellite images hampers their operational exploitation at a global scale since cloud contamination affects most Earth observation applications [1]. Cloud masking of time series is thus a priority to obtain a better monitoring of the land cover dynamics and to generate more elaborated products [2].

Cloud detection approaches are generally based on the assumption that clouds present some useful features for their identification and discrimination from the underlying surface. On the one hand, a simple approach to cloud detection consists then in applying thresholds over a set of selected features, such as reflectance or temperature of the processed image, based on the physical properties of the clouds [3–6]. Apart from its simplicity, such approaches produce accurate results for satellite instruments that acquire enough spectral information, but it is challenging to adjust a set of thresholds that work at a global level. On the other hand, there is empirical evidence that supervised machine learning approaches outperform threshold-based ones in single scene cloud detection [1,7–9]. For instance, Ref. [1,7,9] show that neural networks are good candidates for cloud detection. However, they present practical limitations since they need a statistically significant, large collection of labeled images to learn from. This is because, in order to design algorithms capable of working globally over different types of surfaces and over different seasons, a huge number of image pixels labeled as cloudy or cloud free must be available to train the models. This labelling process usually requires a large amount of tedious manual work, which is also not exempt from errors. Furthermore, additional independent data has to be gathered to validate the performance of the algorithms, which increases the data requirements and dedication. In any case, both threshold and machine learning based cloud detection algorithms relying only on the information of the analyzed image are still far from being perfect and produce systematic errors specially over high reflectance surfaces such as urban areas, bright sand over coastlines, snow and ice covers [10].

In this complex scenario, including temporal information helps to distinguish clouds from the surface, since the latter usually remains stable over time. Cloud detection methodologies can thus be divided into monotemporal single scene and multitemporal approaches. Single scene approaches only use the information from a given image to build the cloud mask, while multitemporal approaches also exploit the information of previously acquired images, collocated over the same area, to improve the cloud detection accuracy. Multitemporal cloud detection is therefore an intrinsically easier problem because location and features of clouds vary greatly between acquisitions, whereas the surface is to a certain extent stable. However, multitemporal methods are computationally demanding, and the lack of accessibility to previous data usually hampers their operational application to most satellite missions. Therefore, in order to exploit the wealth of the temporal information, long-term missions with a granted access to the satellite images archive, and suitable computing platforms, are required. A clear example fulfilling these requirements is the Landsat mission from NASA [11], which provides global image data over land since 1972. For this reason we will focus here on Landsat images, although the methodology and the subsequent discussion can also be applied to other similar satellites [12].

There exists a wide variety of multitemporal approaches for cloud detection that have been applied to Landsat imagery [13–19]. In the Multitemporal Cloud Detection (MTCD) algorithm [14], the authors use a composite cloud-free image as reference, then they detect clouds by setting a threshold on the difference between the target and the reference in the blue band. In order to reduce false positives, they use an extra correlation criteria with at least 10 previous images. In Ref. [15], a previous spatially collocated cloud-free image from the same region is manually selected as the reference image. Then a set of thresholds over the reflectance in some Landsat bands (B1, B4 and B6) and over the difference in reflectance between the target and the reference image are set. The Temporal mask (TMask) algorithm [16] builds a pixel-wise time series regression to model the cloud-free reflectance of each pixel. It uses the FMask algorithm [5] to decide which pixels to include in such a regression model. Then, it applies a set of thresholds over the difference in reflectance between the estimated and the target image in Landsat bands B2, B4 and B5. The work presented in Ref. [17] is also based on FMask. In this case, they first remove one of the FMask tests to reduce over-detection, and compute the FMask cloud probability for each image in the time series. Afterwards, they compute the pixel-wise median FMask cloud probability and the standard deviation over the time series. Then, analyzed pixels are masked as cloudy if (a) the modified FMask says it is a cloud; or (b) if the cloud probability exceeds 3.5 standard deviations the median value. Recently, Ref. [19] proposed to also use a composite reference image and a set of thresholds over the difference in reflectance between the target and the reference in Landsat bands B3, B4 and B5. Thus the method is similar to the one presented in Ref. [14] but without the correlation criteria over the time series. Finally, in Ref. [18], we modeled the background surface from the three previous collocated cloud-free images using a non-linear kernel ridge regression that minimizes both prediction and estimation errors simultaneously. Then, the difference image between this background surface reference and the target is clustered and a threshold over the mean difference reflectance is applied to each cluster to decide if it belongs to a cloudy or cloud-free area. In summary, one can see how most of the multitemporal cloud detection schemes proposed in the literature cast the cloud detection problem as a change detection problem [20]: a reference image is built using cloud-free pixels and clouds are detected as particular changes over this reference. To decide whether the change is relevant enough, several thresholds are usually proposed based on heuristics.

Three main issues not properly addressed can be identified in all multitemporal approaches proposed so far:

- **Data access and image retrieval**. Most of the proposed methods assume that a sufficiently long time series of collocated images is available. It is worth pointing out that retrieving the images to build the time series in an easy and operational manner is technically difficult. We need access to the full catalog and powerful enough GIS software to select and co-register the images. We overcome this limitation using the Google Earth Engine (GEE) platform.
- **Computational cost**. Most of the proposed methods require a sufficiently long time series of images to operate: at least 15 in the case of TMask [16] and at least 10 in the case of MTCD [14]. This is a critical problem if the algorithm cannot be implemented using parallel computing techniques. We again solve this issue ensuring that our algorithm can be implemented on the GEE cloud computing platform.
- **Validation of results**. A consistent drawback in most cloud detection studies is the lack of quantitative validation over a large collection of independent images. On the one hand, as we have mentioned in the two previous points, if the multitemporal algorithm is computationally demanding and required images are hard to retrieve, it will be difficult to test the method over a large dataset. On the other hand, simultaneous collocated observations informing about the presence of clouds or independent datasets of manually annotated clouds are often not available. Therefore, without a comprehensive *ground truth*, validation of cloud detection results is usually limited to a visual inspection of the generated cloud masks. In this work we take advantage of the recently released Landsat-8 Biome Cloud Masks dataset [10], which contains manually generated cloud masks from 96 images from different Biomes around the world.

Therefore, we propose a multitemporal cloud detection algorithm that is also based on the hypothesis that surface reflectance smoothly varies over time, whereas abrupt changes are caused by the presence of clouds. Our proposed methodology extends the work we presented in Ref. [18]. In particular, the proposed methodology presented in this paper consists of four main steps. First, the surface background is estimated using few previous cloud-free images that are automatically retrieved from the Landsat archive stored in the GEE catalog. Then, the difference between the analyzed cloudy image (target) and the cloud-free estimated background (reference) is computed in order to enhance the changes due to the presence of clouds. This difference image is then processed to find homogeneous clusters corresponding to clouds and surface. Finally, the obtained clusters are labelled as cloudy or cloud-free areas by applying a set of thresholds on the difference intensity and on the reflectance of the representative clusters.

In addition, the surface background estimated from the previous cloud-free images can be also used to perform a *cloud removal* (or *cloud filling*) in the analyzed cloudy image [21,22]. Pixels masked as clouds can be replaced by the estimated surface background at these locations obtaining a completely cloud-free image [23,24]. The improved frequency of the satellite images time series can then be used to better monitor land cover dynamics and to generate more elaborated products.

The proposed algorithm is fully implemented in the GEE platform, which grants access to the complete Landsat-8 catalog, reducing the technical complexity of the multitemporal cloud detection and transferring the computational load to the GEE parallel computing infrastructure. The potential of the proposed approach is tested over 2661 500×500 patches extracted from the Biome dataset [10], and the obtained results are available online for the interested readers (http://isp.uv.es/projects/cdc/viewer_l8_GEE.html).

The rest of the paper is organized as follows. In Section 2 the Landsat-8 data, the GEE platform, and the Biome dataset are presented. In Section 3, we explain the proposed methodology for cloud detection and removal. Section 4 presents the evaluation of the proposed methodology. It shows the predictive power of the proposed variables over the dataset, the accuracy, commission and omission errors, some illustrative scenes with the proposed cloud mask, and the cloud removal errors. The algorithm implementation in the Google Earth Engine is briefly described in Section 5. Finally, Section 6 discusses the results and summarizes the conclusions.

2. Satellite Data and Ground Truth

2.1. Landsat-8 Data

The Landsat Program [11] consists of a series of Earth observation satellite missions jointly managed by NASA and the United States Geological Survey (USGS). Landsat is a unique resource with the world's longest continuously acquired image collection of the Earth's land areas at moderate to high resolution to support resource assessment, land-cover mapping, and to track inter-annual changes. It started with the first Landsat satellite launched in 1972, and is continued with both Landsat 7 and 8, which are still operational. Landsat 9 is expected to be launched in late 2020 ensuring the Landsat data continuity.

The Landsat-8 payload consists of two science instruments: the Operational Land Imager (OLI) and the Thermal InfraRed Sensor (TIRS), acquiring multispectral images with 11 spectral bands that cover from deep blue to the thermal infrared: B1—*Coastal and Aerosol* (0.433–0.453 μm), B2—*Blue* (0.450–0.515 μm), B3—*Green* (0.525–0.600 μm), B4—*Red* (0.630–0.680 μm), B5—*Near Infrared* or *NIR* (0.845–0.885 μm), B6—*Short Wavelength Infrared* or *SWIR* (1.560–1.660 μm), B7—*SWIR* (2.100–2.300 μm), B8—*Panchromatic* (0.500–0.680 μm), B9—*Cirrus* (1.360–1.390 μm), B10—*Thermal Infrared* or TIR (10.30–11.30 μm) and B11—*TIR* (11.50–12.50 μm). Note that the visible channels are B1-B4 (and B8), which are useful to distinguish the white and bright clouds. Additionally, Landsat-8 presents a band (B9) specifically designed to detect cirrus and high clouds.

2.2. Google Earth Engine Platform

The Google Earth Engine platform [25] is a cloud computing platform for geographical data analysis. It gives access to a full complete catalog of remote sensing products together with the capability to process these products quickly online through massive parallelization. The GEE data catalog includes data from Landsat 4, 5, 7 and 8 processed by the United States Geological Survey (USGS), several MODIS products, including global composites, recently imagery from Sentinel 1, 2 and 3 satellites, and many more. All data are pre-processed and geo-referenced, facilitating its direct use. In addition, user data in raster or vector formats can be uploaded (*ingested* using GEE terminology) and processed in the GEE. We took advantage of this feature for uploading the manual cloud masks used as ground truth in our experiments.

In this work, all required Landsat images were retrieved from the `LANDSAT/LC8_L1T_TOA_FMASK` *Image Collection* available in the GEE. These images consist of top of atmosphere (TOA) reflectance (calibration coefficients are included in metadata [26]). These products also include two additional bands: the quality assessment band (BQA) and the FMask cloud mask [5]. We use the cloud flag included in the BQA nominal product [11] to assess if previous images over each test site location are cloud free or not, which allows us to easily and automatically retrieve cloud-free images from the entire archive. In addition to the Automated Cloud Cover Assessment (ACCA) cloud masking algorithm in the BQA band presented in Ref. [27], the FMask [5] is used to benchmark the proposed cloud detection algorithm. Both algorithms are single-image approaches mainly based on combination of rules and thresholds over a set of spectral indexes.

The GEE computation engine offers both JavaScript and Python application programming interfaces (API), which allow to easily develop algorithms that work in parallel on the Google data computer facilities. The programming model is object oriented and based on the MapReduce paradigm [28]. On the one hand, the GEE engine is accessible through a web-based integrated development environment (IDE) using the JavaScript API. The web-based IDE allows the user to visualize images, results, tables and charts that can be easily exported. On the other hand, the Python API offers the same set of methods, which allow to make requests to the Engine and access the catalog, but without the visualization capabilities of the web-based IDE. However, we chose the Python API to develop our cloud detection scheme because it is easier to integrate with long running tasks, which are essential to run the full validation study in an automatic manner.

2.3. Cloud Detection Ground Truth

Validation of cloud detection algorithms is an extremely difficult task due to the lack of accurate simultaneous collocated information per pixel about the presence of clouds. In this scenario one is forced to manually generate a labeled dataset of annotated clouds, which is time consuming and always includes some uncertainties. Recent validation studies carried out for single scene cloud detection, e.g. for Landsat-8 [10] and for Proba-V [9], are extremely important efforts for the development and validation of cloud screening algorithms. The public dissemination of this data gives the opportunity to fairly benchmark the results on independent datasets and allows to quantify and analyze the cloud screening quality. This is the case of the Landsat 7 Irish dataset [3,29], the Landsat-8 SPARCS dataset [7,30], the Landsat-8 Biome dataset [10,31] or the Sentinel 2 Hollstein dataset [8]. In this work, we take advantage of the Landsat-8 Biome dataset [31] created in Ref. [10]. The Biome dataset consists of 96 Landsat-8 acquisitions (∼7500 × 7500 pixels approximately) from eight different biomes around the world, in which all pixels have been manually labeled. Figure 1 shows the geographic location of the 96 images that form the dataset.

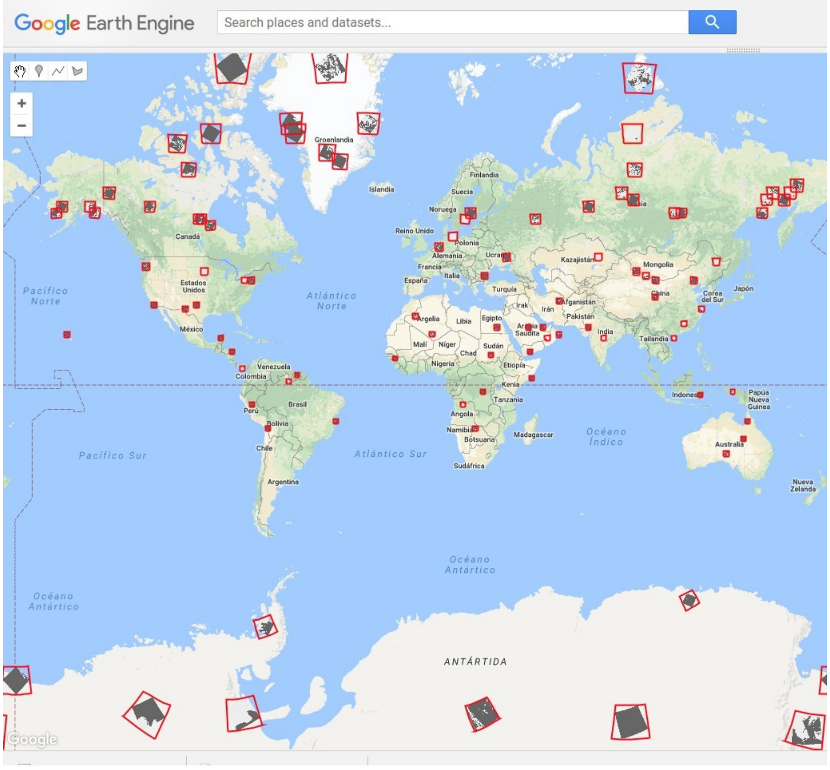


Figure 1. Geographic location of the 96 images from the Landsat-8 Biome dataset [31] ingested on the Google Earth Engine.

We add these cloud masks with the corresponding Landsat-8 products by ingesting this dataset in the GEE. Then, a few previous cloud-free images for each acquisition were automatically retrieved using the GEE API. From the original 96 Biome products, only 23 of them have enough (three) previous cloud-free images. This is mainly because unfortunately most of the labeled acquisitions selected for the Biome dataset are close to the launch of the Landsat-8 satellite. Therefore, we divided these 23 images in smaller patches of 500 × 500 pixels for our analysis, resulting in 2661 patches.

It is worth noting that validation studies of cloud detection algorithms over large datasets are scarce in the literature and, in the particular case of multitemporal cloud detection, the algorithms have been usually validated on a few images. The use of processing platforms such as the GEE make our study much more feasible.

3. Methodology

The proposed methodology for multitemporal cloud detection is based on our previous work [18]. It works under the assumption that surface reflectance is stable over time or at least follows smooth variations compared to the abrupt changes induced by the presence of clouds. Therefore, this work follows the widespread approach for cloud detection based on multitemporal background modeling with difference change detection extensively used in the remote sensing literature [13–19]. Figure 2 shows a diagram summarizing the proposed multitemporal cloud detection approach. The following sections describe the main methodological steps.

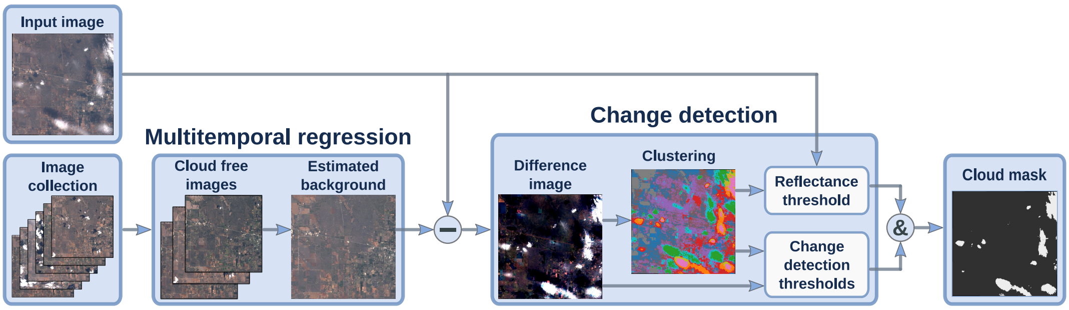


Figure 2. Multitemporal cloud detection scheme implemented on the Google Earth Engine platform.

3.1. Background Estimation

One of the main challenges of the background modeling step is to make it computationally scalable: previous attempts in Ref. [14,16] are computationally demanding, which make them difficult to apply in operational settings. In order to alleviate these problems, in this study we limit the proposed algorithm to work with only three previous collocated images for the surface background estimation. The key for this process to be fully operational is that the selection and retrieval of the three previous cloud-free collocated images has to be carried out automatically. We use the BQA band included in the Landsat products to discriminate if an image is cloud free; and, as we have mentioned, one of the main advantages of using the GEE Python API together with the Landsat image collection is that this step can be fully automated requiring no human intervention.

We call *pre-filtering* to the first image retrieval step, which consists of assessing if previous images are cloud free or not. Pre-filtering can be solved applying some rough cloud detection method, e.g., setting a threshold over the brightness or over the blue channel as proposed in Ref. [14], or taking advantage of automatic single scene cloud detection schemes if they exist for the given satellite. For this study we use the cloud flag from the Level 1 BQA band of Landsat-8 [27]. We consider an image cloud free if less than 10% of its pixels are masked as cloudy. This raises an important consideration on the design of the cloud detection scheme: it should be robust to errors on the pre-filtering method. An extremely inaccurate pre-filtering algorithm can undermine the performance of the method since cloudy pixels will be used to model the background surface. We will see that these methods are robust enough to work on situations where previous images have some clouds. It is worth pointing out that, since we limit the cloud cover to be less than 10% in each selected image and we assume that clouds are randomly located from one image to another, the probability that the same pixel is cloudy in all three images is expected to be really low.

The estimation of the background from the cloud-free image time series is one of the critical steps of the method. We compare four different background estimation methodologies presented in the literature, from simpler to more complex:

- **Nearest date**: It consists of taking the nearest cloud-free image in time as the background. This is the approach used in Ref. [15], however they rely on human intervention to assess that the image does not present any cloud.
- **Median filter**: It takes the pixel-wise median over time using the three previous cloud-free images. This is the approach suggested in TMask [16] for pixels where the time series is not long enough.
- **Linear regression**: It fits a standard regularized linear regression using the time series of the previous cloud-free images [18]. Similarly, TMask [16] used an iterative re-weighted least squares regression at pixel level, which mitigates the effect of eventual cloudy pixels in the time series.
- **Kernel regression**: The nonlinear version of the former method. It is based on a specific kernel ridge regression (KRR) formulation for change detection presented in Ref. [18].

3.2. Change Detection and Clouds Identification

Once the background is estimated, we use it as a cloud-free reference image to tackle the cloud detection as a change detection problem. Therefore, we compute the difference image between the cloudy target image and the estimated cloud-free reference, which is the base for most change detection methods [20].

However, we do not find changes by applying thresholds directly to the difference image, i.e., target minus estimated. Instead, we previously apply a *k*-means clustering algorithm over the difference image using all Landsat-8 bands. Afterwards, specific thresholds are applied at a cluster level, i.e., to some features computed over the pixels belonging to each cluster. In particular, we compute three different features for each cluster i: (a) the norm (intensity) of the difference reflectance image over the visible bands (B2, B3 and B4 for Landsat 8), we denote this quantity with α_i; (b) the mean of the difference reflectance image over visible bands, β_i; and (c) the norm of reflectance image over the visible bands, γ_i. A cluster is classified as cloudy if the three following tests over these features are satisfied: $\alpha_i \geq 0.04$, $\beta_i \geq 0$ and $\gamma_i \geq 0.175$.

The threshold 0.04 on the difference of reflectance image is ubiquitous in the existing literature. For instance, TMask [16] also suggested 0.04 for the B4 channel, MTCD [14] suggests 0.03 on the blue band weighted by the difference between the acquisition time of the image and the reference. The method proposed in Ref. [19] also used 0.04 in the B3 and B4 bands. In contrast, in our previous work [18] the threshold was higher (0.09) since we used the norm over all the reflectance bands. Here we select the norm as a more robust indicator but restricted to the visible bands (B2, B3 and B4). This threshold is intended to detect significant differences, i.e., with a sufficient intensity to be considered changes, while the other two conditions to be satisfied are specifically included to distinguish clouds from the rest of possible changes in the surface. On the one hand, clouds are usually brighter than the surface so clouds imply an increase in reflectance with respect to the reference background image. By imposing the temporal difference over the visible bands to be positive we exclude intense changes decreasing the reflectance, such as shadows, flooded areas, agricultural changes, etc. On the other hand, we also want to discard changes that increase the brightness but do not look like a cloud in the target image, e.g., agricultural crops. Therefore, we also impose that the norm of the top of atmosphere (TOA) reflectance over the visible bands is higher than 0.175 in order to consider that the cluster corresponds to a cloud. The norm of the visible reflectance bands is also used in Ref. [17] to distinguish potentially cloudy pixels, although in this work they set a lower threshold of 0.15 because they wanted to over-detect cloudy areas.

Modifying these thresholds will make the algorithm more or less cloud conservative. We believe that the subsequent user of the cloud mask should have some flexibility to choose to be more or less cloud conservative. For instance, applications like land use or land cover classification are less affected by the presence of semitransparent cirrus whereas for instance estimating the water content of canopy should be much more cloud conservative. Providing the receiver operator curve (ROC) [32] for the entire dataset allows the users to better select these thresholds in order to obtain a trade-off between commission (false positives) and omission (false negatives) errors for their particular application.

3.3. Remarks

One of the main differences of our proposal for cloud detection is the clustering step. We apply a *k*-means clustering over the difference image over all bands of the satellite. We fixed the number of clusters to 10; this number is related to the size of the image (500 × 500 pixels in the experiments) so if larger images are used this number should be increased. We tried however different numbers of clusters (5, 15 and 20) but we did not observe major differences in performance. The clustering step seeks to capture patterns over all the bands that cannot be captured with a single static threshold. For example, it is well known that the Thermal Infrared Bands (TIR, B10 and B11) have good predictive power for the cloud detection problem. However, setting a global threshold independently of location and season is very difficult since surface temperature greatly varies over places and surfaces. In addition, working with time series exacerbates this problem since the surface temperature might vary quite a lot with the date of the acquired image. Therefore, *k*-means clustering is intended to group similar patterns, e.g., in temperature, and pixels assigned to the same cluster will be classified afterwards to the same class (cloudy or clear). The clustering step simplifies the problem since instead of classifying pixels we have to classify clusters. However, it might introduce errors in mixed clusters where not all the pixels are purely from one of the two classes (cloudy or clear). In our case, if we classify each cluster according to its majority class using the ground truth, we obtain a classification error lower than 3% for all the proposed background estimation methods in the used dataset. This error can be considered a lower bound of the classification error for the presented results. Finally, it is worth mentioning that if we apply the thresholds directly over the difference image, i.e., without the clustering step, numerical accuracy is not significantly affected, but visual inspection showed less consistency on the masks and higher salt-and-pepper effects.

4. Experimental Results

This section contains the experimental results. First we describe an illustrative example where we show some intermediate results of the method, then the analysis over the full dataset is presented. For these results, we will first explore the parameters and the discriminative power of the multitemporal difference, then we will show the results over the complete dataset for cloud detection and cloud removal.

4.1. Cloud Detection Example

Figure 3 shows the cloud detection results for a cloudy image over Texas (USA). The top right corner shows the RGB composite of the acquired image included in the Biome dataset. We see that it contains several thin clouds scattered across the image. In the bottom left image, the manually labeled ground truth cloud mask (in yellow) from the Biome dataset overlay the RGB composite. We can see here that some very thin clouds are not included in the provided ground truth. The three top left images are the previous cloud-free images retrieved automatically with the GEE API from the GEE Landsat image collection. We see that the top left one is not completely free of clouds: this is because it has less than 10% of clouds according to the ACCA algorithm of the BQA band. The image of the bottom right corner corresponds to the cloud-free estimated background using the median method. We see that this estimation method is robust enough in this case since it has not been affected by the unscreened clouds present in the previous "cloud-free" images, and it correctly preserves other bright surfaces such as urban areas. Finally, we compare both the proposed and the Fmask cloud masks with the available Biome ground truth. The second image of the bottom starting from the left shows the differences between our proposed method and the ground truth. In white it shows the true positives (clouds) of the method with respect the ground truth, in orange the false negatives (omissions) and in blue the false positives (commissions). We see that the overall agreement is very high and most of the discrepancies are on the borders of the clouds. The image to the right corresponds to the differences between FMask and the ground truth, in this case we see that FMask missed some thin clouds in the bottom and in the left part of the image.

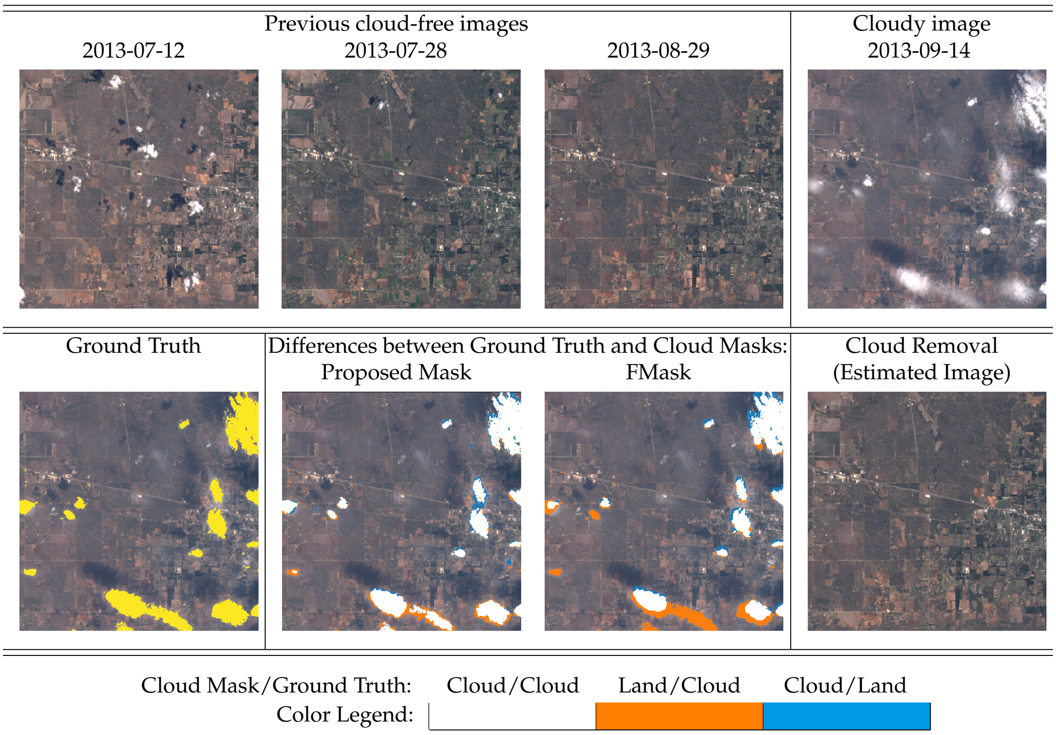


Figure 3. Illustration of the Cloud detection scheme. Comparison between the ground truth and the proposed cloud mask algorithm and the FMask. Discrepancies are shown in blue when the proposed method detects 'cloudy', and in orange when pixels are classified as 'cloud-free'.

4.2. Parameters and Errors Analysis

We evaluate the results in terms of commission errors, omission errors and overall accuracy. Table 1 contains the definition of these metrics. Generally, we can obtain a trade-off between commission and omission errors depending on the requirements. For instance, to reduce the omission error we can reduce the threshold over the reflectance which will make the algorithm more cloud conservative and, as a result, the commission error will increase. On the other hand, if we increase the threshold the commission error will decrease and the algorithm will be more clearly conservative and will probably raise the omission error.

First we want to demonstrate the discriminating capability of the norm of the difference image (α_i) for cloud detection. The receiver operator curve (ROC) shows the true positive rate vs. false positive rate trade-off as we vary the threshold over the predictive variable α_i. Figure 4 shows four curves corresponding to the four different background estimation methods (nearest date, median filter, Linear regression, and Kernel ridge regression). A cross is displayed for the case of the proposed threshold (0.04). We also show a cross indicating the TPR and FPR values for FMask [5] and for ACCA (BQA) [27] over this dataset. As we see, using the nearest date or the median filter to estimate the background and a single threshold over α we outperform FMask and ACCA (BQA) since those points are below the obtained ROC curves. This means that we can reduce commission error while having the same omission error as FMask, or reduce omission error while maintaining the same commission error as FMask.

Table 1. Validation metrics: True Positive Rate (TPR), False Positive Rate (FPR), Commission Error, Omission Error, Overall Accuracy.

True Positive Rate	TPR	$\frac{\text{cloudy pixels predicted as clouds}}{\text{cloudy pixels}}$
False Positive Rate (Commission Error)	FPR	$\frac{\text{clear pixels predicted as clouds}}{\text{clear pixels}}$
Omission Error	1- TPR	$\frac{\text{cloudy pixels predicted as clear}}{\text{cloudy pixels}}$
Overall Accuracy		$\frac{\text{cloudy pixels pred. as clouds+clear pixels pred. as clear}}{\text{total pixels}}$

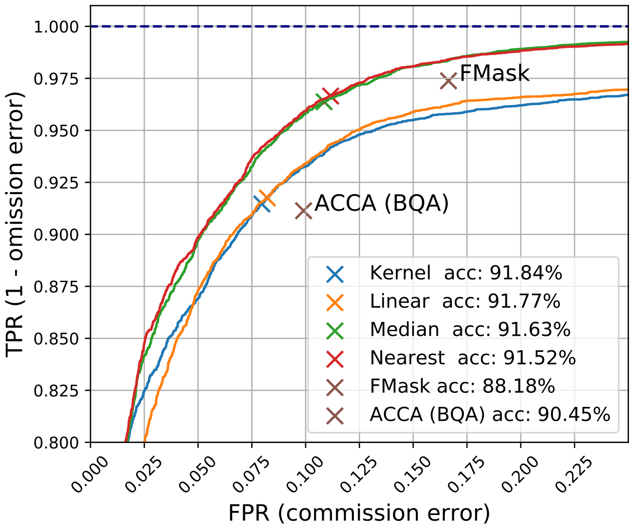


Figure 4. This figure shows ROC curves for the four proposed background estimation methods. Crosses show the TPR and FPR values for the proposed threshold (0.04) and for FMask [5] and ACCA (BQA) [27] on the same dataset.

It is worth mentioning that the simpler background estimation methods (Median and Nearest) have better performance in terms of cloud detection accuracy. In the case of the median, it is more robust to outliers (e.g., clouds contaminating the images used for the background estimation) than the linear or kernel regression approaches. For the nearest date, it might be because the closer in time the image is, the more similar it is to the target image in terms of surface changes. Nevertheless, we will see in the next sections that the kernel and linear regression methods obtain better results in terms of mean squared error in reflectance and, therefore, will be the more appropriate for the cloud removal task.

Figure 4 shows that using only a threshold over the norm difference has a very good performance on cloud detection. However, as we have mentioned in Section 3.2, by doing this we detect all high differences (changes) in reflectance. Whereas most of these differences are because of the presence of clouds, some of them are due to changes in the surface. Figure 5 shows an example of agricultural crops in Bulgaria. The image on the center shows that some of those fields are detected as clouds if we use only a threshold over the differences.

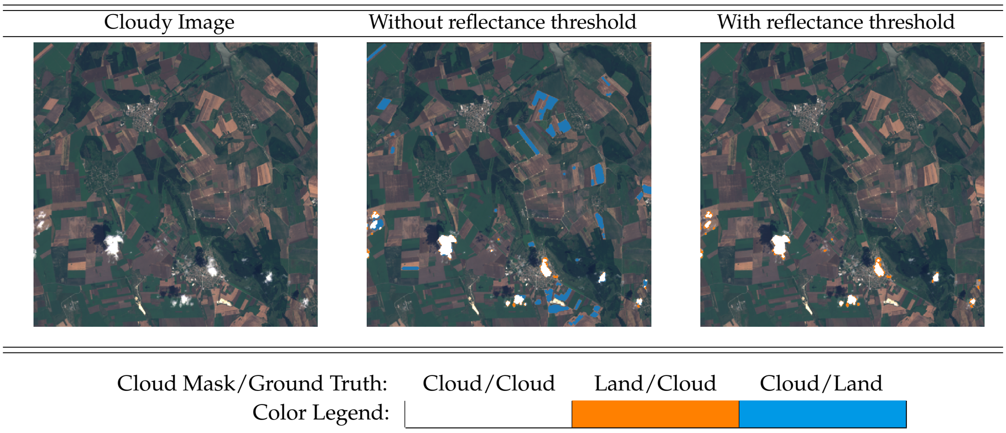


Figure 5. Landsat 8 image (LC81820302014180LGN00 006_013) acquired on 10 June 2013. Rural area in Bulgaria presenting crops misclassified as clouds when the threshold in reflectance is not applied.

In order to reduce these false positive cases we added an additional threshold applied directly over the reflectance of the cloudy image instead of over the difference image. In particular, we applied the threshold over the norm of the visible bands, γ. This is physically grounded since clouds have high reflectance on the visible spectral range. In addition, it has been exploited before in Ref. [17] as a measure of potentially cloudy pixels. Figure 6 confirms this approach. The left plot shows cluster centers colored in orange if the majority of their pixels are cloudy and in blue if most of them are clear. The X-axis shows the norm of the difference in reflectance, α, and the Y-axis shows the norm of the reflectance, γ. We can see that the threshold of 0.04 in α was correctly fixed and that 0.175 is a natural threshold in γ for this dataset. The right plot in Figure 6 shows the ROC curves with and without the extra threshold on reflectance γ. We show the ROC curves corresponding to the thresholds 0.15 and 0.175. We can see that the inclusion of this additional threshold (0.175) increases the overall accuracy from 91 to 94%. The threshold at 0.15 could be used instead of 0.175 for cloud conservative applications. Overall we see that by including this additional restriction (either in 0.175 or 0.15) the resulting algorithm is more accurate and less cloud conservative.

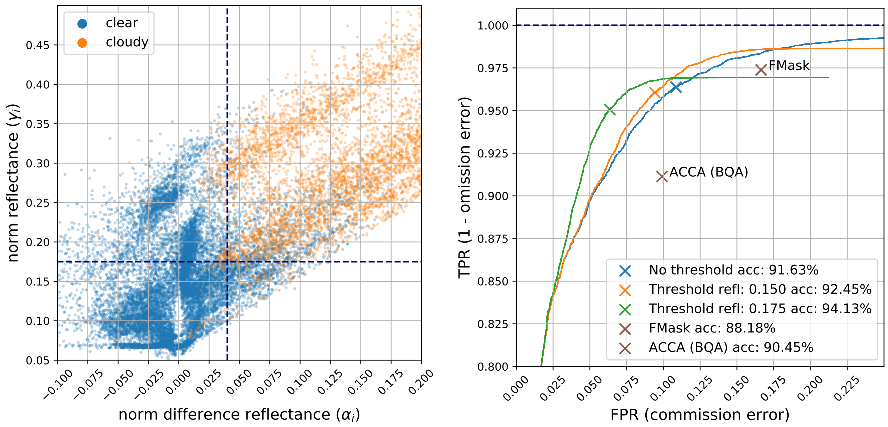


Figure 6. (**Left**) Scatter plot of the clusters. Norm of TOA reflectance of the visible bands on the Y-axis (γ) and norm of difference in reflectance on the X-axis (α). Each point corresponds to one of the 10 clusters from each of the 2661 image patches. Vertical and horizontal lines show the proposed thresholds. (**Right**) ROC curves with and without the extra threshold on reflectance γ. The median is used for background estimation in both cases.

4.3. Cloud Detection Results

Once the parameters and methodology have been fixed we analyze the cloud detection results over the whole dataset. Table 2 shows the cloud detection statistics using both the proposed thresholds for the four background estimation models and for the independent FMask and ACCA (BQA) cloud detection algorithms. We see that multitemporal methods yield higher overall accuracy than single scene methods. In addition, we see that the simpler background estimations, such as the median filter and the nearest date, yield a good trade-off in commission and omission errors. Figure 7 shows the mean accuracy and standard deviation over the patches for each of the 23 Landsat-8 acquisitions. We see here again that the multitemporal approach using the median as background estimator consistently outperforms FMask.

Table 2. Cloud detection statistics over all pixels of the used Landsat-8 Biome Dataset.

Method	Overall Accuracy	Kappa Statistic	Commission Error	Omission Error
FMask [5]	88.18%	0.7550	16.64%	2.62%
ACCA (BQA) [27]	90.45%	0.7933	9.90%	8.86%
Nearest date	94.18%	0.8733	6.31%	4.87%
Median filter	94.13%	0.8720	6.36%	4.94%
Linear regression	93.66%	0.8593	4.78%	9.32%
Kernel regression	93.56%	0.8572	4.82%	9.53%

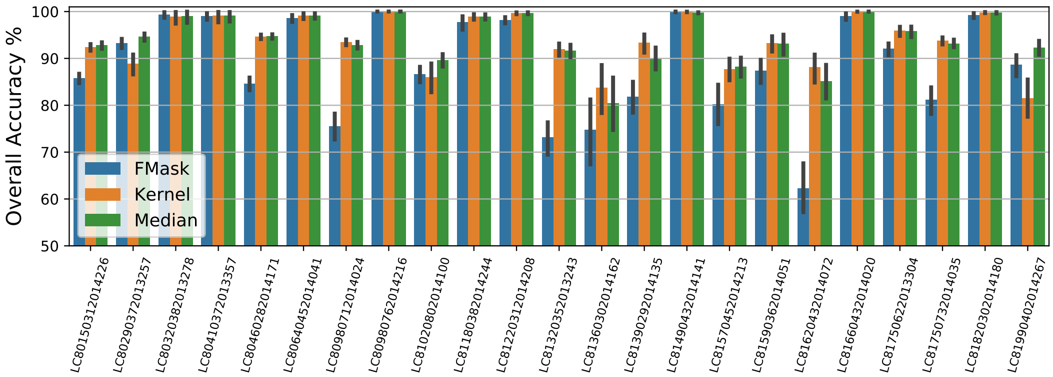


Figure 7. Average accuracy over the image patches for each of the 23 different Landsat-8 acquisitions selected from the Biome dataset.

Finally, Figure 8 shows some cherry-picked results of the proposed method using the median to estimate the background. Rows 1, 3 and 5 show some systematic errors of FMask over cities, coastal areas and riversides. Row 2 presents small errors in semitransparent clouds in the middle of the image that are not correctly labeled in the manual ground truth cloud mask but that our method correctly identifies. On the other hand, thin clouds on Rows 1 and 7 are misclassified by the proposed method whereas FMask identifies them correctly. Row 6 again shows errors in the ground truth labels. In this case, a path is falsely identified as a cloud. We found these errors specially over bright surfaces, which remind us that single scene cloud detection is challenging even for human experts. Actually, in some cases, we detected them only because we have previous images from the same location with which to compare. Interested readers can visually inspect cloud detection results and the comparison of both the proposed method and FMask [5] with the Biome ground truth, which are available online at http://isp.uv.es/projects/cdc/viewer_l8_GEE.html for all 2661 patches from the Biome dataset.

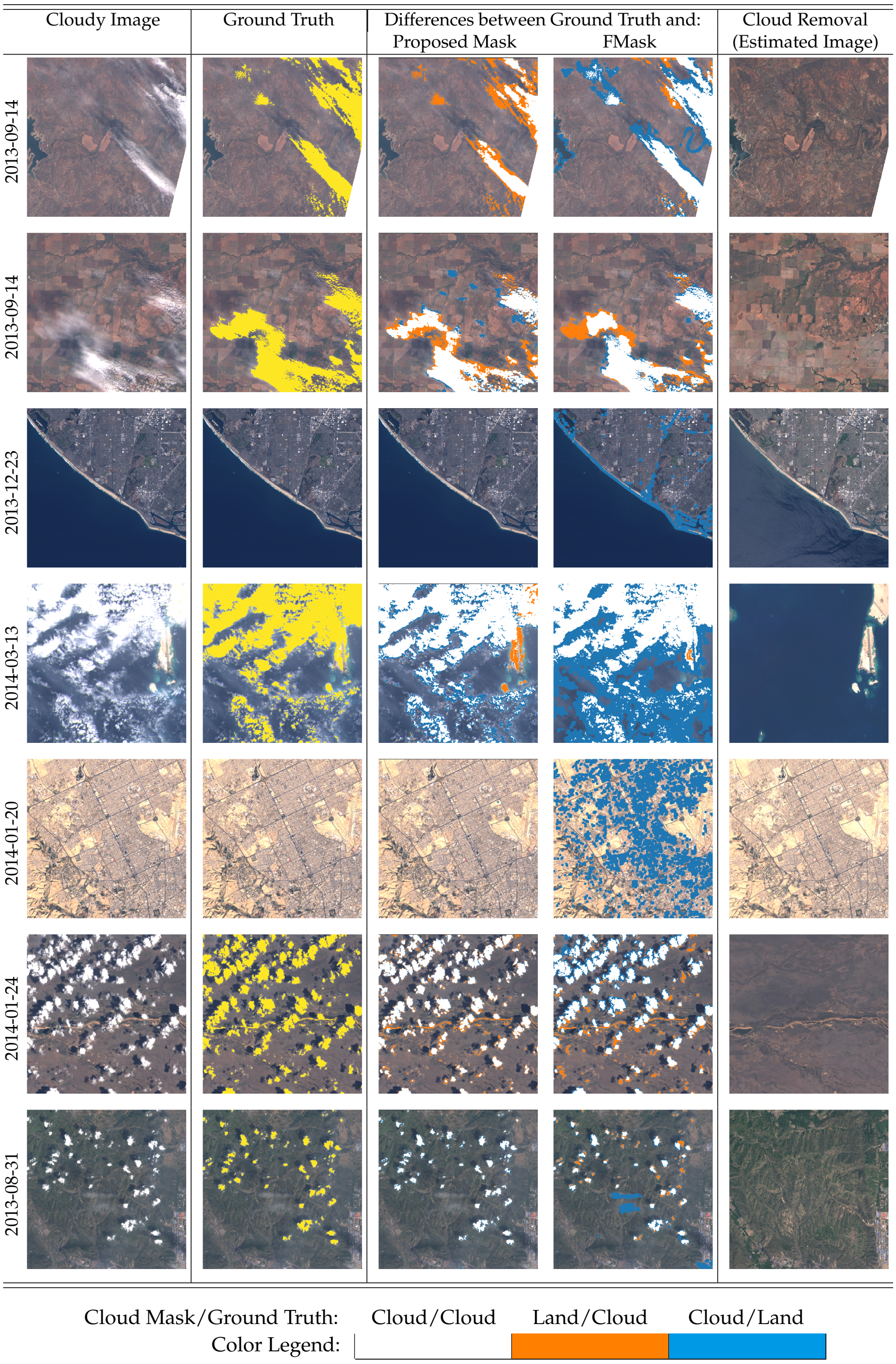


Figure 8. Patches of 500 × 500 pixels from Biome dataset. From left to right: RGB scene, RGB scene with ground truth cloud mask in yellow, differences between ground truth and the proposed cloud mask (using the median as background estimation), difference between ground truth and FMask, and estimated cloud-free image.

4.4. Cloud Removal

In addition to the cloud detection problem, we consider also the task of cloud removal (or cloud filling). Most land applications generally discard cloud contaminated pixels for the estimation of biophysical parameters. In cloudy areas, this causes a big amount of missing values in the processed time series that undermine the statistical significance of the subsequent analysis. In this subsection, we benchmark the different background estimation methodologies proposed in this paper and evaluate their suitability for cloud removal in a large dataset. In Ref. [18], we proposed to use previous images together with the current one to estimate the TOA reflectance of the cloud contaminated areas. This idea is also presented in Ref. [33] using more sophisticated methods, however, our proposal in Ref. [18] can be directly implemented in the GEE platform. We compare these linear and kernel based regression approaches with the two simplest baselines, i.e., using the latest available cloud-free pixel or the pixel-wise median filter. The performance of the cloud removal is quantified and evaluated in terms of the error between the estimated and actual background pixels in the cloud-free areas (since cloud contaminated pixels cannot be compared with the background). The accurate cloud mask and the posterior cloud removal provide cloud-free time series that allow a better monitoring of land cover dynamics and the generation of further remote sensing products. The last column in Figure 8 shows the estimated cloud-free image for some scenes. The plots show the estimated image where clouds have been removed. In fact, we show estimated values for the whole scene and not clouded areas only. We can see how the spatial and radiometric features are well preserved and no cloud residuals can be observed.

Quantitative results for the cloud removal are shown in Figure 9. It shows the distribution of the root mean square error on the 2661 patches separately for each spectral band. In the plots, the mid lines represent the mean RMSE for all the (cloud-free) image pixels, the boxes define the 25 and 75 percentiles of the RMSE distribution, and the vertical lines define the maximum and minimum RMSE values. We see here that the more sophisticated methods for background estimation (Linear and Kernel regression) perform better than the simpler ones (median and nearest). This confirms the results presented in Ref. [18] and shows that estimated reflectance is an accurate option to fill the gaps caused by cloud contamination.

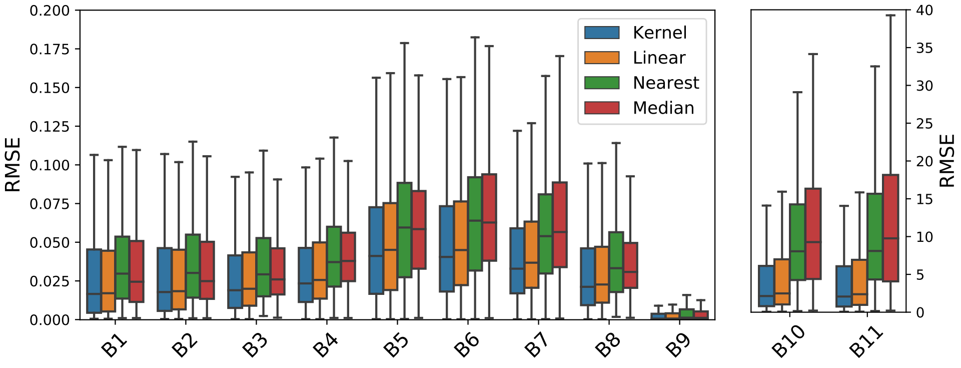


Figure 9. Root mean square error between the estimated and actual background pixels in the cloud-free areas. Distribution of the errors is shown separately for all Landsat bands for the four background estimation approaches.

5. Algorithm Implementation in the Google Earth Engine

The data in the GEE platform is organized in collections, usually composed of images or features. Images contain bands (spectra, masks, products, etc.), properties, and metadata. Features can contain any kind of information needed to process data, such as labels for supervised algorithms, polygons to define geographical areas, etc. Users can apply their own defined functions, or use the ones provided by the API, using an operation called *mapping*, which essentially applies a function over any given collection independently. This allows a straightforward processing of large amounts of images and data in parallel. Using this computational paradigm we implemented the full proposed cloud detection scheme using the Python API. In particular, given an input target image, we *map* and *filter* the Landsat-8 `LANDSAT/LC8_L1T_TOA_FMASK` *Image Collection*. Then the filtered collection is *reduced* to produce an *Image* which is the background estimation. With this reference image we compute the difference image and apply the *k*-means *clustering*. Finally, we apply the thresholds as defined in Section 3.

The manual cloud masks from the Biome dataset were ingested in the GEE. Therefore, the proposed methodology together with the comparison with the ground truth is implemented using only the Python API of the GEE platform. The developed code has been published in GitHub at https://github.com/IPL-UV/ee_ipl_uv. In that package we provide a function that computes the cloud mask following the proposed methodology for a given GEE image. In addition, some Python notebooks with examples that go step by step on the proposed methodology have been included in the software package.

Finally, as we have mentioned, in order to show the potential of the GEE platform the proposed algorithms have been tested over 2661 patches extracted from the Biome dataset. The obtained cloud masks can be inspected online for the whole dataset at http://isp.uv.es/projects/cdc/viewer_l8_GEE.html.

6. Discussion

In previous sections we presented a simple yet efficient multitemporal algorithm for cloud detection. The results show an overall increase in detection accuracy and commission error compared to state-of-the-art mono-temporal approaches such as FMask and ACCA. In addition, omission error could be reduced slightly more for the same commission error than FMask using a lower threshold in reflectance ($\gamma = 0.15$), as can be seen in Figure 6 (left). For cloud detection, it is normally taken for granted that commission errors are better than omissions, thus operational algorithms tend to overmask in order to avoid false negatives. However, we think that the proliferation of open access satellite image archives implies that in the future more advanced users will be interested in controlling by themselves the trade-off between commission and omission errors depending on their underlying application. To this end, we provide Table 3 as a guide to help in tuning the thresholds of the current algorithm, where we can see the selected combination providing the best trade-off highlighted in bold. From results shown in Tables 2 and 3, we can see that the proposed method presents improvements between 4–5% in classification accuracy and 3–10% in commission errors, compared with FMask and ACCA algorithms.

Table 3. Cloud detection statistics for different thresholds combinations over all selected pixels of the Landsat-8 Biome Dataset, using the *median* as background estimation method.

Thresholds		Overall Accuracy	Kappa Statistic	Commission Error	Omission Error
Difference (α)	Reflectance (γ)				
0.02	0.000	85.54%	0.7076	21.53%	0.95%
0.02	0.150	89.52%	0.7820	15.16%	1.54%
0.02	0.175	92.59%	0.8415	9.61%	3.21%
0.03	0.000	89.56%	0.7823	14.90%	1.93%
0.03	0.150	91.43%	0.8187	11.83%	2.35%
0.03	0.175	93.63%	0.8624	7.76%	3.74%
0.04	0.000	91.63%	0.8217	10.86%	3.63%
0.04	0.150	92.45%	0.8382	9.43%	3.95%
0.04	**0.175**	**94.13%**	**0.8720**	**6.36%**	**4.94%**
0.05	0.000	92.50%	0.8378	8.54%	5.53%
0.05	0.150	92.97%	0.8474	7.70%	5.76%
0.05	0.175	94.26%	0.8738	5.32%	6.54%
FMask		88.18%	0.7550	16.64%	2.62%
ACCA (BQA)		90.45%	0.7933	9.90%	8.86%

The proposed multitemporal methodology resembles popular multitemporal algorithms such as TMask [16] and MTCD [14] since all of them are based on background estimation and thresholds over the difference image. However, our methodology is simpler and requires less images in the time series to operate. For this reason we consider the current work as a baseline to evaluate trade-offs in processing performance for these more complex multitemporal schemes. It would be of great interest for the community to compare all these approaches in a common benchmark; unfortunately, to this end, we would need labeled images and common open-sourced versions of the algorithms to evaluate the models.

Obviously there are limitations to the proposed multitemporal methodology: for instance, it might fail in situations with sudden changes in the underlying surface, such as permanent snow in upper latitudes. The current dataset lacks these situations, hence we do not recommend its use in such cases.

In addition, another limitation of current and future works on cloud detection is the quality of the ground truth masks: for the Irish dataset [3], the work [34] estimated a mean overall disagreement of 7% over the manual cloud masks labelled by three different experts. The labelling procedure to create the Irish dataset [10] is similar to the Biome dataset that we use in the present work. Therefore, current overall errors are in line with the intrinsic error of human experts following the current labelling procedure. This indicates that in the future, in order to increase the performance, we should develop better labelling methods and provide results by cloud type and underlying surface.

7. Conclusions

In this work, we proposed a multitemporal cloud detection methodology that can be applied at a global scale using the GEE cloud computing platform. We applied the proposed approach to Landsat-8 imagery and we validated it using a large independent dataset of manually labelled images.

The approach is based on a simple multitemporal background modelling algorithm together with a set of tests applied over the segmented difference image, which has shown a high cloud detection power. Our principal findings and contributions can be summarized as follows. This approach outperforms single-scene threshold-based cloud detection approaches for Landsat such as FMask [5] and ACCA (BQA) [27]. We provided an evaluation of different background estimation methods and different variables and thresholds in terms of commission and omission errors. In particular, we showed that simple background detection models such as the median or the nearest cloud-free image are both accurate and robust for the cloud detection task. In addition, for the first time to the authors knowledge, a multitemporal cloud detection scheme is validated over a large collection of independent manually labelled images. The whole process has been implemented within the GEE cloud computing platform and a ready to use implementation has been provided. Compared to previous multitemporal open source implementations, our approach also includes the image retrieval and coregistration steps, which are essential for the operational use of the algorithm. The generated cloud masks can be inspected at http://isp.uv.es/projects/cdc/viewer_l8_GEE.html.

Future lines of research include the application to other optical multispectral satellites requiring accurate and automatic cloud detection. For example, the satellite constellations of Sentinel missions from the European Copernicus programme aim to optimize global coverage and data delivery. In particular, Sentinel-2 mission [12] acquires image time series with a high temporal frequency and unprecedented spatial resolution for satellite missions providing open access data at a global scale. Additionally, another line of research consists of using the multitemporal cloud masks as a proxy of a ground truth that can be used to train single scene supervised machine learning cloud detection algorithms. This approach has been recently successfully applied to image classification tasks [35] and would alleviate data requirements of machine learning methods.

Supplementary Materials: An interactive tool for the visualization of the validation results is available online at http://isp.uv.es/projects/cdc/viewer_l8_GEE.html. The code is published at https://github.com/IPL-UV/ee_ipl_uv.

Author Contributions: G.M.G. and L.G.C. conceived and designed the experiments; G.M.G. performed the experiments; G.M.G. and L.G.C. analyzed the data; J.M.M. contributed to develop the analysis tools; G.C.V. and J.A.L. contributed to write and review the manuscript.

Funding: This research has been funded by the Spanish Ministry of Economy and Competitiveness (MINECO-ERDF, TEC2016-77741-R and TIN2015-64210-R projects) and by Google Inc. under the Google Earth Engine (GEE) Award titled 'Cloud detection in the Cloud' granted to Luis Gómez-Chova. Camps-Valls was supported by the European Research Council (ERC) through the ERC Consolidator Grant SEDAL (project id 647423).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gómez-Chova, L.; Camps-Valls, G.; Calpe, J.; Guanter, L.; Moreno, J. Cloud-Screening Algorithm for ENVISAT/MERIS Multispectral Images. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 4105–4118. [CrossRef]
2. Malenovsky, Z.; Rott, H.; Cihlar, J.; Schaepman, M.E.; García-Santos, G.; Fernandes, R.; Berger, M. Sentinels for science: Potential of Sentinel-1, -2, and -3 missions for scientific observations of ocean, cryosphere and land. *Remote Sens. Environ.* **2012**, *120*, 91–101. [CrossRef]
3. Irish, R.R.; Barker, J.L.; Goward, S.N.; Arvidson, T. Characterization of the Landsat-7 ETM+ Automated Cloud-Cover Assessment (ACCA) Algorithm. *Photogramm. Eng. Remote Sens.* **2006**, *72*, 1179–1188. [CrossRef]
4. Zhu, Z.; Woodcock, C.E. Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sens. Environ.* **2012**, *118*, 83–94. [CrossRef]
5. Zhu, Z.; Wang, S.; Woodcock, C.E. Improvement and expansion of the Fmask algorithm: Cloud, cloud shadow, and snow detection for Landsats 4–7, 8, and Sentinel 2 images. *Remote Sens. Environ.* **2015**, *159*, 269–277. [CrossRef]
6. Mei, L.; Vountas, M.; Gómez-Chova, L.; Rozanov, V.; Jäger, M.; Lotz, W.; Burrows, J.P.; Hollmann, R. A Cloud masking algorithm for the XBAER aerosol retrieval using MERIS data. *Remote Sens. Environ.* **2017**, *197*, 141–160. [CrossRef]
7. Hughes, M.J.; Hayes, D.J. Automated Detection of Cloud and Cloud Shadow in Single-Date Landsat Imagery Using Neural Networks and Spatial Post-Processing. *Remote Sens.* **2014**, *6*, 4907–4926. [CrossRef]
8. Hollstein, A.; Segl, K.; Guanter, L.; Brell, M.; Enesco, M. Ready-to-Use Methods for the Detection of Clouds, Cirrus, Snow, Shadow, Water and Clear Sky Pixels in Sentinel-2 MSI Images. *Remote Sens.* **2016**, *8*, 666. [CrossRef]
9. Iannone, R.Q.; Niro, F.; Goryl, P.; Dransfeld, S.; Hoersch, B.; Stelzer, K.; Kirches, G.; Paperin, M.; Brockmann, C.; Gómez-Chova, L.; et al. Proba-V cloud detection Round Robin: Validation results and recommendations. In Proceedings of the 2017 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp), Brugge, Belgium, 27–29 June 2017; pp. 1–8.
10. Foga, S.; Scaramuzza, P.L.; Guo, S.; Zhu, Z.; Dilley, R.D.; Beckmann, T.; Schmidt, G.L.; Dwyer, J.L.; Joseph Hughes, M.; Laue, B. Cloud detection algorithm comparison and validation for operational Landsat data products. *Remote Sens. Environ.* **2017**, *194*, 379–390. [CrossRef]
11. Loveland, T.R.; Dwyer, J.L. Landsat: Building a strong future. *Remote Sens. Environ.* **2012**, *122*, 22–29. [CrossRef]
12. Drusch, M.; Bello, U.D.; Carlier, S.; Colin, O.; Fernandez, V.; Gascon, F.; Hoersch, B.; Isola, C.; Laberinti, P.; Martimort, P.; et al. Sentinel-2: ESA's Optical High-Resolution Mission for GMES Operational Services. *Remote Sens. Environ.* **2012**, *120*, 25–36. [CrossRef]
13. Wang, B.; Ono, A.; Muramatsu, K.; Fujiwara, N. Automated Detection and Removal of Clouds and Their Shadows from Landsat TM Images. *IEICE Trans. Inf. Syst.* **1999**, *82*, 453–460.
14. Hagolle, O.; Huc, M.; Pascual, D.V.; Dedieu, G. A multi-temporal method for cloud detection, applied to FORMOSAT-2, VENμS, LANDSAT and SENTINEL-2 images. *Remote Sens. Environ.* **2010**, *114*, 1747–1755. [CrossRef]
15. Jin, S.; Homer, C.; Yang, L.; Xian, G.; Fry, J.; Danielson, P.; Townsend, P.A. Automated cloud and shadow detection and filling using two-date Landsat imagery in the USA. *Int. J. Remote Sens.* **2013**, *34*, 1540–1560. [CrossRef]
16. Zhu, Z.; Woodcock, C.E. Automated cloud, cloud shadow, and snow detection in multitemporal Landsat data: An algorithm designed specifically for monitoring land cover change. *Remote Sens. Environ.* **2014**, *152*, 217–234. [CrossRef]

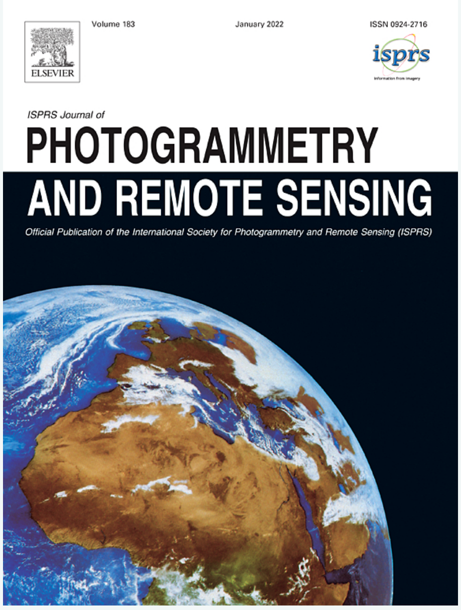
17. Frantz, D.; Röder, A.; Udelhoven, T.; Schmidt, M. Enhancing the Detectability of Clouds and Their Shadows in Multitemporal Dryland Landsat Imagery: Extending Fmask. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1242–1246. [CrossRef]
18. Gómez-Chova, L.; Amorós-López, J.; Mateo-García, G.; Muñoz-Marí, J.; Camps-Valls, G. Cloud masking and removal in remote sensing image time series. *J. Appl. Remote Sens.* **2017**, *11*, 015005. [CrossRef]
19. Candra, D.S.; Phinn, S.; Scarth, P. Cloud and cloud shadow removal of landsat 8 images using Multitemporal Cloud Removal method. In Proceedings of the 2017 6th International Conference on Agro-Geoinformatics, Fairfax, VA, USA, 7–10 August 2017; pp. 1–5.
20. Bovolo, F.; Bruzzone, L. The Time Variable in Data Fusion: A Change Detection Perspective. *IEEE Geosci. Remote Sens. Mag.* **2015**, *3*, 8–26. [CrossRef]
21. Melgani, F. Contextual reconstruction of cloud-contaminated multitemporal multispectral images. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 442–455. [CrossRef]
22. Lin, C.H.; Tsai, P.H.; Lai, K.H.; Chen, J.Y. Cloud Removal From Multitemporal Satellite Images Using Information Cloning. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 232–241. [CrossRef]
23. Hu, G.; Li, X.; Liang, D. Thin cloud removal from remote sensing images using multidirectional dual tree complex wavelet transform and transfer least square support vector regression. *J. Appl. Remote Sens.* **2015**, *9*, 095053. [CrossRef]
24. Chen, B.; Huang, B.; Chen, L.; Xu, B. Spatially and Temporally Weighted Regression: A Novel Method to Produce Continuous Cloud-Free Landsat Imagery. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 27–37. [CrossRef]
25. Gorelick, N.; Hancher, M.; Dixon, M.; Ilyushchenko, S.; Thau, D.; Moore, R. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sens. Environ.* **2017**, *202*, 18–27. [CrossRef]
26. Chander, G.; Markham, B.L.; Helder, D.L. Summary of current radiometric calibration coefficients for Landsat MSS, TM, ETM+, and EO-1 ALI sensors. *Remote Sens. Environ.* **2009**, *113*, 893–903. [CrossRef]
27. Scaramuzza, P.L.; Bouchard, M.A.; Dwyer, J.L. Development of the Landsat Data Continuity Mission Cloud-Cover Assessment Algorithms. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 1140–1154. [CrossRef]
28. Dean, J.; Ghemawat, S. MapReduce: Simplified Data Processing on Large Clusters. In Proceedings of the Sixth Symposium on Operating System Design and Implementation (OSDI'04), San Francisco, CA, USA, 6–8 December 2004; pp. 137–150.
29. U.S. Geological Survey. *L7 Irish Cloud Validation Masks*; Data Release; U.S. Geological Survey: Reston, VA, USA, 2016.
30. U.S. Geological Survey. *L8 SPARCS Cloud Validation Masks*; Data Release; U.S. Geological Survey: Reston, VA, USA, 2016.
31. U.S. Geological Survey. *L8 Biome Cloud Validation Masks*; Data Release; U.S. Geological Survey: Reston, VA, USA, 2016.
32. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [CrossRef]
33. Meng, F.; Yang, X.; Zhou, C.; Li, Z. A Sparse Dictionary Learning-Based Adaptive Patch Inpainting Method for Thick Clouds Removal from High-Spatial Resolution Remote Sensing Imagery. *Sensors* **2017**, *17*, 2130. [CrossRef] [PubMed]
34. Oreopoulos, L.; Wilson, M.J.; Várnai, T. Implementation on Landsat Data of a Simple Cloud-Mask Algorithm Developed for MODIS Land Bands. *IEEE Geosci. Remote Sens. Lett.* **2011**, *8*, 597–601. [CrossRef]
35. Mahajan, D.; Girshick, R.; Ramanathan, V.; He, K.; Paluri, M.; Li, Y.; Bharambe, A.; van der Maaten, L. Exploring the Limits of Weakly Supervised Pretraining. *arXiv* **2018**, arXiv:1805.00932.

Publication II: Transferring Deep Learning Models for Cloud Detection between Landsat-8 and Proba-V

Publication II

G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, Transferring deep learning models for cloud detection between Landsat-8 and Proba-V, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 160, pp. 1–17, Feb. 2020, doi: 10.1016/j.isprsjprs.2019.11.024.

Q1: Geography, Physical, Q1: Geosciences, Multidisciplinary, Q1: Remote Sensing, Q1: Imaging Science & Photographic Technology, IF = 8.979



ISPRS Journal of Photogrammetry and Remote Sensing 160 (2020) 1–17

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Transferring deep learning models for cloud detection between Landsat-8 and Proba-V

Gonzalo Mateo-García*, Valero Laparra, Dan López-Puigdollers, Luis Gómez-Chova*

Image Processing Laboratory, University of Valencia, Valencia, Spain

ARTICLE INFO

Keywords:
Transfer learning
Cloud masking
Convolutional neural networks
Deep learning
Multispectral sensors
Domain adaptation

ABSTRACT

Accurate cloud detection algorithms are mandatory to analyze the large streams of data coming from the different optical Earth observation satellites. Deep learning (DL) based cloud detection schemes provide very accurate cloud detection models. However, training these models for a given sensor requires large datasets of manually labeled samples, which are very costly or even impossible to create when the satellite has not been launched yet. In this work, we present an approach that exploits manually labeled datasets from one satellite to train deep learning models for cloud detection that can be applied (or transferred) to other satellites. We take into account the physical properties of the acquired signals and propose a simple transfer learning approach using Landsat-8 and Proba-V sensors, whose images have different but similar spatial and spectral characteristics.

Three types of experiments are conducted to demonstrate that transfer learning can work in both directions: (*a*) *from Landsat-8 to Proba-V*, where we show that models trained only with Landsat-8 data produce cloud masks 5 points more accurate than the current operational Proba-V cloud masking method, (*b*) *from Proba-V to Landsat-8*, where models that use only Proba-V data for training have an accuracy similar to the operational FMask in the publicly available Biome dataset (87.79–89.77% vs 88.48%), and (*c*) *jointly from Proba-V and Landsat-8 to Proba-V*, where we demonstrate that using jointly both data sources the accuracy increases 1–10 points when few Proba-V labeled images are available. These results highlight that, taking advantage of existing publicly available cloud masking labeled datasets, we can create accurate deep learning based cloud detection models for new satellites, but without the burden of collecting and labeling a large dataset of images.

1. Introduction

The number of new satellites and sensors with the objective of monitoring the Earth system and understanding its dynamics is growing exponentially. Among these sensors, optical instruments measure radiance coming from the Earth in the visible and infra-red part of the electromagnetic spectrum. Data from optical sensors is used in a wide range of applications such as estimating biophysical parameters, monitoring land use over time, assessing damages after natural disasters, or monitoring urban areas among others. In most of those applications, the presence of clouds and their shadows affects the signal and can be considered as a source of uncertainty (Gómez-Chova et al., 2007). Whereas, on a single scene, cloud masking might be handled manually, on operational applications exploiting image time series or multiple locations, this is not feasible. Thus, in order to automatically process imagery from optical sensors, accurate and automatic cloud masking algorithms are mandatory.

Cloud masking algorithms assign a *clear* or *cloudy* binary label to each of the pixels within a satellite image. Most basic approaches to cloud masking are the so called threshold based approaches, which consist of a set of thresholds applied on one or more of the spectral bands of the images, or on extracted features trying to enhance physical properties of the clouds. In general, thresholding is simple and easy to implement and it works well when the spectral information provided by the satellite is sufficiently rich in terms of cloud discrimination. Examples of current operational threshold based approaches to cloud masking include FMask (Zhu and Woodcock, 2012; Zhu et al., 2015) for Landsat-7 and Landsat-8, Sen2Cor (Richter et al., 2012) for Sentinel-2, and several recent works that propose improvements to them (e.g. Zhai et al., 2018; Qiu et al., 2019; Frantz et al., 2018). On the other hand, machine learning (ML) based approaches handle cloud detection as a statistical classification problem. These methods *learn* a cloud detection

* Corresponding authors.
E-mail addresses: gonzalo.mateo-garcia@uv.es (G. Mateo-García), valero.laparra@uv.es (V. Laparra), dan.lopez@uv.es (D. López-Puigdollers), luis.gomez-chova@uv.es (L. Gómez-Chova).

https://doi.org/10.1016/j.isprsjprs.2019.11.024
Received 14 July 2019; Received in revised form 23 November 2019; Accepted 25 November 2019
Available online 10 December 2019

model based on a set of examples: data pairs of observations and labels. When the quality of the training data is sufficiently good, machine learning based approaches outperform threshold based ones (Gómez-Chova et al., 2007; Li et al., 2019; Jeppesen et al., 2019). Machine learning approaches to cloud detection can be further divided into *classical* and *deep learning* approaches. In classical ones, a set of manually selected spatial and spectral features are extracted for each pixel in the training set, afterwards a classifier is optimized to distinguish the label of those pixels based on these features. In the simplest case, only two classes are considered: cloud and clear pixel; however, several works consider a wider range including cirrus, cloud shadows, ice/snow, water, etc. (Hollstein et al., 2016; Hughes and Hayes, 2014; Wieland et al., 2019). Classical machine learning approaches are normally pixelwise, in the sense that the trained classifier can be applied independently to each pixel in the test image after the feature extraction. The classifiers used by these approaches widely vary including: kernel methods and support vector machines (Azimi and Zekavat, 2000; Bai et al., 2016; Ishida et al., 2018; Gómez-Chova et al., 2010), neural networks (Torres Arriaza et al., 2003; Hughes and Hayes, 2014) or trees and ensemble methods (Ghosh et al., 2006; Hollstein et al., 2016; Ghasemian and Akhoondzadeh, 2018; Wieland et al., 2019). On the other hand, deep learning approaches for cloud masking are end-to-end models where the input is the raw image and the output the cloud mask. If the model is defined as a set of stacked convolutional operations, then it constitutes a fully convolutional neural network (FCNN) (Long et al., 2015). In these models, the convolutional filters weights are parameters to optimize, thus the model can learn to exploit the spatial information of surrounding pixels directly from the data. FCNNs applied to cloud detection have shown *state-of-the-art* performance for several satellites, such as Landsat-7 (Li et al., 2019), Landsat-8 (Jeppesen et al., 2019; Li et al., 2019), GaoFen-1 (Li et al., 2019), or MSG SEVIRI (Drönner et al., 2018).

Independently of the selected cloud masking approach, the method has to be validated. This is a bottleneck in the development of cloud masking algorithms for most satellite sensors, since usually there is no independent and simultaneous information about the presence of clouds in the images. Therefore, in order to perform a quantitative validation, the standard approach is to manually label a set of pixels or images by human experts, which will constitute the *ground truth*. This approach has been extensively applied in the literature, e.g. to Landsat-7 (Irish et al., 2006), Envisat/MERIS (Gómez-Chova et al., 2007), Landsat-8 (Foga et al., 2017), Proba-V (Iannone et al., 2017), or Sentinel-2 (Coluzzi et al., 2018; Baetens et al., 2019). In some cases, only some pixels within an image are labeled as cloudy or cloud free, whereas in other cases, all the pixels of the image are labeled, capturing also the spatial distribution of clouds. In either case, this process is not exempt of errors: e.g., in Scaramuzza et al. (2012), authors reported a mean overall error of 7% for 11 Landsat-7 scenes fully labeled by three different experts. Labeling pixels individually is more accurate, however it requires a higher dedication and hence the total amount of labeled pixels is usually considerably lower. This makes results statistically less significant which could be a problem when the goal is to validate cloud detection algorithms that work globally under different seasons and climatic conditions.

Moreover, if the proposed method for cloud detection is based on machine learning, in addition to the validation data, an independent comprehensive set of labeled samples is also required to train the models. If the goal is to provide an accurate global cloud detection method, this training set should be representative enough of natural statistics, with data from different land covers, climate zones, and seasons. Therefore, for machine learning approaches, the effort to generate a ground truth and develop a cloud detection algorithm is huge. Another disadvantage of machine learning approaches is that they cannot be applied until the satellite is launched and data is available, since a comprehensive archive of images with the corresponding ground truth is required to develop the models. For these reasons, it is still very common that most satellite missions use empirically designed threshold based methods for cloud detection at their launching time. Afterwards, if the operational cloud detection performance is an issue, the original algorithm is replaced by an improved one based on the acquired data during the mission lifetime. This is the case of Proba-V mission (Sterckx et al., 2014), in which case the European Space Agency (ESA) recently organized a Cloud Detection Round Robin experiment (Iannone et al., 2017) aimed at the inter-comparison of different cloud detection algorithms in order to improve the current operational algorithm (Wolters et al., 2015).

Taking into account the aforementioned issues, we can conclude that the lack of an accurate and representative ground truth for the particular satellite sensor will hamper the development of accurate machine learning models. However, the amount of available Earth observation data is huge nowadays and it is increasingly common to publish not only the algorithms but also the manually labeled cloud masks datasets as a good practice to foster research in the field of remote sensing. In particular, for cloud detection, in Foga et al. (2017) the authors published more than 250 scenes of Landsat-7 and Landsat-8; in Mohajerani and Saeedi (2019) they published an additional 38 scenes for Landsat-8; the works (Li et al., 2017, 2019) published[1] 108 GaoFen-1 images and 150 high resolution scenes from Google Earth, respectively; and the works (Hollstein et al., 2016; Liu et al., 2019; Baetens et al., 2019) also published their manually labeled cloud masks for Sentinel-2. In this context, we propose to exploit the wealth of information contained in available labeled datasets to transfer previous knowledge about the problem *between* similar satellites. This approach allows us to address some of the drawbacks of machine learning approaches. Firstly, from a methodological point of view, the size of the manually labeled training set required to build an accurate cloud detection model for the new satellite is drastically reduced. Secondly, from an operational point of view, since the training data from an existing satellite is already available, the machine learning based cloud detection algorithm could be developed before the launch of the satellite, and thus it can be applied from the first day.

In this paper, we focus on the Proba-V and the Landsat-8 satellites, which have different spatial resolution, different spectral bands and from which there are manually labeled cloud detection datasets available to train and evaluate the models. Proba-V is a small satellite with medium spatial resolution and with only four spectral bands (Sterckx et al., 2014); we will take advantage of manually labeled datasets for cloud detection from the recent ESA Round Robin experiment (Iannone et al., 2017). Landsat-8 (Irons et al., 2012) has higher spatial and spectral resolutions compared to Proba-V, and, as mentioned previously, there exists a large collection of manually labeled images for cloud detection (U.S. Geological Survey, 2016a,b).

Our proposed approach to transfer knowledge between Landsat-8 and Proba-V is based on two components. The first one is a domain adaptation transformation of Lansdsat-8 data to resemble Proba-V images in terms of both spectral and spatial characteristics. The objective is to carry out a simple physically-based conversion between the two sources in order to facilitate the transfer learning from the available manually labeled dataset (i.e., the *source domain*) to the satellite images where we want to detect clouds (i.e., the *target domain*). The second component is a fully convolutional neural network model capable of learning as much as possible spectral and spatial information from the training data. FCNNs excel in image segmentation tasks (Xie et al., 2017; Chen et al., 2018a,b; Lin et al., 2018; Drozdzal et al., 2016; Breininger et al., 2018; Schuegraf and Bittner, 2019), they integrate spectral but also spatial information in a hierarchical manner: in our view, spatial information is crucial specially in the context of Proba-V, which has a limited number of spectral bands.

Using the domain adaptation transformation and the FCNN models,

[1] Upon request and for academic purpose.

we performed three types of transfer learning experiments: (*a*) *transductive transfer learning from Landsat-8 to Proba-V*, where we used only Landsat-8 annotated data to develop a model that works in the Proba-V domain; (*b*) *transductive transfer learning from Proba-V to Landsat-8*, where labeled Proba-V data is used to train a cloud detection model for Landsat-8; and (*c*) *inductive transfer learning from Landsat-8 to Proba-V*, where we use few Proba-V labeled images together with the annotated Landsat-8 dataset to generate a cloud detection model for Proba-V.

In these experiments, we show that the proposed models trained only on Landsat-8 data (previous item *a*) outperforms by at least 5 points in accuracy the current Proba-V operational cloud detection algorithm (Wolters et al., 2015). This model does not use any Proba-V image for training. In the more challenging Proba-V to Landsat-8 transfer direction, the model trained only with Proba-V data (previous item *b*), which works on a 10 times lower spatial resolution scale, is only 2 point less accurate than the *state-of-the-art* deep learning models for Landsat-8 (Jeppesen et al., 2019; Li et al., 2019) and it is as accurate as the operational FMask (Zhu and Woodcock, 2012) on the analyzed dataset. Finally, the performance of models that exploit labeled data from both sensors (previous item *c*) shows that models trained only with few Proba-V images are significantly less accurate than models trained jointly with these few Proba-V images together with the available Landsat-8 data. In particular, results show a boost between 1 to 10 points in detection accuracy depending on the amount of Proba-V data used when networks are trained jointly using both data sources.

The paper is organized as follows. In Section 2, we frame our proposal in the current literature context and we detailed our contributions. In Section 3, we present the physically-based image conversion scheme, which facilitates transfer learning between sensors, the transfer learning schemes, and the proposed network architecture. In Section 4, we present the Landsat-8 and Proba-V datasets. Section 5 contains the experimental design with the detailed description of the transfer learning experiments. In Section 6, the results are shown and discussed. Finally, Section 7 presents the conclusions.

2. Background and related work

There has been a large amount of remote sensing papers that use transfer learning in a plethora of different manners. The objective is to improve machine learning models performance by reusing data or by training the models in different but related tasks (Jean et al., 2016; Li et al., 2017; Helber et al., 2018; Lu and Li, 2018; Kemker et al., 2018; Wurm et al., 2019). Transfer learning has thus become a buzzword with different meanings depending on the particular context. In this work, we follow the definition of transfer learning given in the literature survey in Pan and Yang (2010). In this view, the goal of transfer learning is to find a trade-off between the two most fundamental assumptions of machine learning: (1) the training set is representative enough of the underlying data distribution, and (2) future test data is drawn from the same exact distribution. In particular, transfer learning seeks to relax this second constraint at the expense of the first one by using data from a different domain and/or from a different task when training the models. Following this definition, transfer learning is further categorized depending on the data we have from the related domain (called *source domain*) and from the domain of interest (called *target domain*). In this work, we restrict ourselves to two of these categories: *transductive transfer learning* and *inductive transfer learning*.

Transductive transfer learning (Pan and Yang, 2010), assumes that, at training time, we only have data from the source domain. This corresponds, in our setting, to use data only from Landsat-8 in order to learn a cloud detection model for Proba-V (or vice versa). This transfer learning approach is common in remote sensing when machine learning is used to invert radiative transfer models (RTM) (Wolanin et al., 2019). In that case, the machine learning model is trained using simulated radiance data as input, and the variables used as inputs to RTMs as outputs. At the test phase, the machine learning model is applied to data from the target domain, which in this case corresponds to real observed satellite radiances. In the context of cloud detection, models trained on RTM simulated data has been proposed for MERIS (Preusker et al., 2006) to estimate cloud optical thickness, for MERIS and AATSR (Gómez-Chova et al., 2013), for Proba-V (Iannone et al., 2017), and also recently for MODIS (Chen et al., 2018c). The main difference with our approach is that here we transfer from a real sensor to another one. Transferring the model from a real dataset, instead that from RTM simulated radiance, has the advantage that we can exploit the natural statistics and spatial information of clouds over different surfaces, which is conveniently used by convolutional neural networks.

The second transfer learning category that we have explored is *inductive transfer learning*. In this setting, it is assumed to have, in addition to the source domain data, some labeled data from the target domain. Inductive transfer learning has also been explored in hybrid RTM inversions by using jointly real and RTM simulated data (Gómez-Chova et al., 2013; Svendsen et al., 2018). In our setting, we will explore inductive transfer learning using all available Landsat-8 datasets and a limited number of Proba-V real labeled images for training.

In the context of neural networks, *inductive transfer learning* is performed in at least two different ways: the first one, called *joint training*, consists of simply joining the training sets of the two domains. The second one, *fine-tuning*, consists of pre-training the network using the source domain and then use the adjusted weights as initialization for a second training using the target domain data. In the case of CNN, using the weights from ImageNet (Deng et al., (CVPR09), 2009,) as the source domain is by far the most common approach also in remote sensing applications (Li et al., 2017; Helber et al., 2018; Lu and Li, 2018). This approach has been explored for cloud detection of Landsat-8 images in Li et al. (2019) and Chai et al. (2019), but both works showed better performance by training a tailored fully convolutional neural network from scratch. In Section 6, *joint training* and *fine-tuning* are compared experimentally.

There is a vast recent literature of deep learning applied to cloud detection on satellite imagery (Zhan et al., 2017; Mateo-García et al., 2017; Jeppesen et al., 2019; Li et al., 2019; Shao et al., 2019; Chai et al., 2019; Liu et al., 2019; Drönner et al., 2018; Mohajerani and Saeedi, 2019, 2018). Fully convolutional neural network is the model of choice in all those cases. In the work (Li et al., 2019), the authors proposed a FCNN, named multi-scale convolutional feature fusion (MSCFF), for remote sensing images of different sensors, this network shows better detection accuracy than other FCNN architectures such as (Zhan et al., 2017; Chen et al., 2018d). In the work (Jeppesen et al., 2019), authors target Landsat-8 cloud detection. They use the U-Net architecture modifying the input and output layers to accommodate for multispectral images. They designed a experimental setup where they train on the L8Biome (U.S. Geological Survey, 2016b) dataset and test on the L8SPARCS (U.S. Geological Survey, 2016a) dataset and the other way around; thus showing generalization across datasets labeled using different experts and different labeling methodologies. In this work we use the same methodology when our networks are trained and evaluated in the Landsat-8 domain.

Nevertheless, the goal of this work is not only to find an accurate FCNN architecture but to show that these FCNN models can be transferred between different sensors with very good cloud detection accuracy. Our networks are, however compared with some of these state-of-the-art methods in the Landsat-8 domain.

Finally, it is worth to mention that detection of cloud shadows is an important issue intimately related to cloud detection. Usually, it involves two steps: first, cloud detection to locate the clouds in the image and, then, a geometry-based cloud shadow detection method (Sun et al., 2018). This geometry-based approach is used in both Landsat-8 (Fmask) and Proba-V operational detection methods, but it could be solved also in one step in a CNN framework as shown in Chai et al. (2019). However, datasets including a shadow ground truth are scarcely available for Landsat (Foga et al., 2017) and are not available for

Proba-V. Hence, the domain adaptation proposed in this work focuses only on cloud detection, and cloud shadows are not distinguished from other cloud-free pixels.

3. Methodology

In this section, we first introduce the Proba-V and Landsat-8 characteristics. Then, we propose two transfer learning (TL) schemes: the first of them will be used to transfer learning from Landsat-8 to Proba-V and the second one from Proba-V to Landsat-8. These TL schemes specify how training and testing can be done in the source and the target domain, respectively. Each scheme can be applied to different situations depending if the domain adaptation is done from the source to the target domain or on the opposite direction. Afterwards, we detailed the domain adaptation transformation that will be used in our experiments for both TL schemes. The transformation is based on the instrumental characteristics of the sensors in order to adapt Landsat-8 images to the Proba-V domain. Finally, subsection 3.4 explains the fully convolutional neural network architecture used in the experiments. In this paper, we focus on the Landsat-8 and Proba-V case, however, this procedure for transfer learning could be reproduced in other sensors with similar characteristics, since we only require that the two sensors have some spectral bands with overlapping response.

3.1. The Landsat-8 and the Proba-V sensors

Proba-V is a small satellite designed for global vegetation monitoring (Sterckx et al., 2014). It was launched in 2013 to bridge the gap between Envisat/MERIS and SPOT Vegetation and the recently launched Sentinel-3. Proba-V is an experimental satellite with a constrained budget designed to be much smaller than the former MERIS and SPOT. It acquires top of atmosphere (TOA) radiance in four bands of the visible (BLUE and RED), the near infrared (NIR) and short-wave infrared (SWIR). Proba-V has three cameras: one central camera, with nadir pointing, and two more on its sides. These three cameras provide a wide swath to Proba-V which enables a short revisiting period of 1–2 days. The central (nadir) camera acquires data at 100 m (from 90 to 110 m) whereas the spatial resolution of the two side cameras ranges from 110 m to 350 m. The operational Level 2A processing projects this varying resolution data into a uniform 333 m Plate Carrée projection using Lanczos interpolation (Dierckx et al., 2014). Cloud detection in Proba-V is specially challenging due to the limited amount of spectral information. The current operational Proba-V cloud detection algorithm based on thresholds (Wolters et al., 2015) has been modified several times and it still presents several drawbacks such as a dependency on illumination and viewing geometry, the detection at edges, and the high amount of commission errors (Stelzer et al., 2016).

Landsat-8 (Irons et al., 2012) measures TOA radiance in 11 bands of the electromagnetic spectrum with a revisiting period of 15 days. Landsat-8 has two sensors: the Operational Land Imager (OLI), which collects data from nine spectral bands at 30 m resolution; and the Thermal Infrared Sensor aimed for thermal imaging, which measures data from two more wavelengths in a 100 m spatial resolution scale. There are two factors that make cloud detection an easier problem for Landsat-8 compared with Proba-V. First, the band 9 from the OLI sensor is specially designed for detection of cirrus clouds. Secondly, the thermal bands are particularly discriminative for clouds since some clouds are significantly cooler than the underlying surface. Algorithms such as FMask (Zhu et al., 2015) take advantage of these facts to design simple, yet robust, cloud detection algorithms based on thresholds that can be applied globally.

3.2. Transfer learning schemes

As we discussed previously in Section 2, transfer learning consists in exploiting data from one (source) domain to solve a problem in a similar but different (target) domain. However, there are different possibilities to perform TL depending on the relationships between the source and the target domains. In this work, we consider two different TL schemes. These schemes assume that we have labeled data in the source domain (S), that we want to perform predictions in the target domain (T), and that a domain adaptation transformation (DA) can be applied between both domains. The applicability of each TL scheme depends on the direction of the domain adaptation transformation:

- **Scheme 1 – training models in the target domain**. In this case we have a domain adaptation transformation from the source to the target domain. We first adapt the labeled dataset from the source domain to the target domain using the domain adaptation transformation and then we train a model using the adapted data. Since both images and labels are transformed to the target domain to train the model, applying the learned model to data from the target domain is straightforward (because the model has been constructed already in this domain). On the other hand, if we want to test the model in the source domain, we would need to transform the source domain inputs to the target domain, apply the predictive model and transform back the predictions to the source domain.
- **Scheme 2 – training models in the source domain**. This scheme is based on training the model directly in the source domain. In this scheme we have a domain adaptation transformation form the target to the source domain. In order to apply the model to new data from the target domain one has to first adapt the input sample to the source domain, then apply the predictive model, and finally transform the predictions back to the target domain. Note that, in this case, testing the model on data from the source domain is direct.

In this work, we will use Scheme 1 for transfer learning from Landsat-8 to Proba-V and Scheme 2 for transfer learning in the opposite direction. Schemes are summarized in Fig. 1. In particular, in this work, X are satellite images (either Landsat or Proba-V, Section 3.1); Y are cloud mask labels; DA_X represents adaptation from Landsat-8 images to Proba-V (Section 3.3); DA_Y is an adaptation of the labels by upscaling or downscaling the masks (Section 3.3); and f is the prediction model, $Y = f(X)$, implemented with Fully Convolutional Neural Networks (Section 3.4). In addition, Fig. 1 illustrates the transformations of the data to train the models and the procedure to test them in datasets from either the target or the source domains.

It is worth noting that the intended use of the proposed transfer learning models is to apply them in data from the target domain. However, it is also interesting to analyze their performance on data from the source domain since we have labeled data for validating the models and, moreover, because there are independent results from the literature to compare with. Therefore, we will evaluate the proposed models on both the source and the target domains.

3.3. Landsat-8 to Proba-V domain adaptation

In order to apply the TL schemes (Section 3.2) we need a procedure to adapt images from one domain to the other (DA_X in Fig. 1). While there is vast amount of methods to learn the domain adaptation transformation from data (see e.g. Tuia et al., 2014; Hoffman et al., 2018; Csurka, 2017), in this paper we employ a methodology based only on the physical properties of the acquired signals. Learning the transformation would imply having data from both sensors to train a model which in some cases might not be feasible (for instance if our target satellite has not been launched yet). Hence, several parts of this study assumes there is no data (or very few) from the target domain. In cases where learning the domain adaptation transformation can be done, this simpler approach can serve as a baseline for such methods to compare with. Our proposed transformation can be applied in general when there is spectral overlap between the acquired signals in both domains. In our particular case, these domains are the Proba-V and

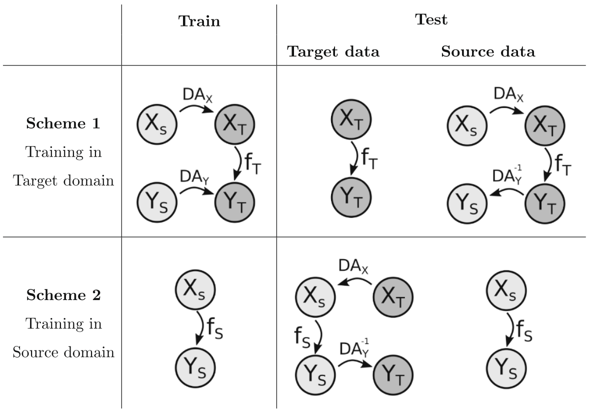


Fig. 1. Transfer learning schemes. In both schemes, we assume that there is labeled data in the source (S) domain but we want to perform predictions in the target (T) domain. The adaptation of images X and labels Y is performed using the transformations DA_X and DA_Y, respectively. The trained ML model is denoted by f_T or f_S, depending if it is trained in the source or the target domain, respectively.

Landsat-8 satellite images. Among the two possible domain adaptation transformations (from Landsat-8 to Proba-V or from Proba-V to Landsat-8), and given the characteristics of Landsat-8 and Proba-V images, the transformation from Landsat-8 to Proba-V seems the more natural one since it goes from the higher to the lower spatial and spectral resolutions. The opposite transformation could also be possible; however, the interpolation to a 30 m spatial resolution from Proba-V images is an ill-posed problem and it is unlikely that the interpolated image has the spatio-spectral quality of a Landsat-8 image. For this reason, in all the paper, we will only consider domain adaptation from Landsat-8 to Proba-V for the transformation of TOA reflectance images.

Our proposed image conversion from Landsat-8 to Proba-V (Fig. 3) is based on the instrumental characteristics of both sensors. This conversion consists of two adaptation steps: firstly, the more suited spectral bands are selected and, secondly, we scale the Landsat-8 image to match the spatial properties of Proba-V. The spectral transformation takes into account the spectral response function (SRF) of both satellites. It consists basically in selecting the overlapping spectral bands between both satellites and eventually weight their contribution as a function of their spectral overlap. Fig. 2(left) shows the spectral response function of common bands in Proba-V (solid) and Landsat-8 (dashed). One can see a good agreement in the case of the SWIR band and also in the RED one. In the case of NIR band, the spectral response of Proba-V is wider and its peak is not aligned with Landsat-8 B5 band, which might led to differences in the the retrieved radiance. Finally, for the Proba-V BLUE band, there are two bands on Landsat-8 in the same spectral range. In this case, the contribution of B1 and B2 bands of Landsat-8 is weighted according to the overlapping area of the spectral responses as is shown in Fig. 2(right), which corresponds to 25% for B1 and 75% for B2.

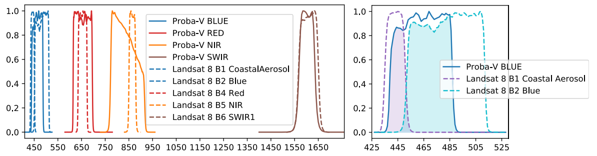


Fig. 2. Left: spectral responses of Landsat-8 and Proba-V. Right: zoom of the blue region of the spectrum; we weight the contributions of bands B1 and B2 of Landsat-8 according to its overlap with the spectral response function of Proba-V.

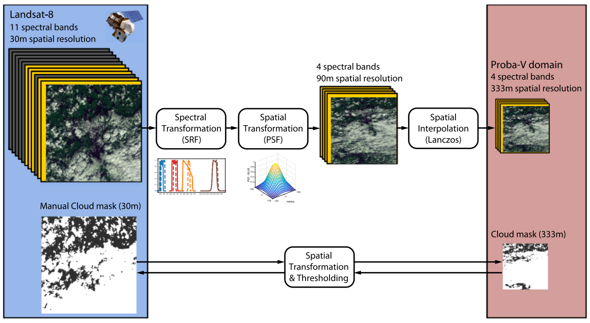


Fig. 3. Transformation of Landsat-8 products and masks to resemble Proba-V characteristics.

The second adaptation step changes the spatial resolution of Landsat-8 images. In order to resemble as much as possible the spatial properties of Proba-V, we upscale the Landsat-8 image to the coarser Proba-V resolution. First, we used the point spread function (PSF) of each Proba-V spectral band to convert the Landsat-8 observations to the nominal Proba-V spatial resolution at nadir. The ground sampling distance (GSD) for the Proba-V center camera is about 96.9 m for the BLUE, RED and NIR channels, while the SWIR center camera resolution is 184.7 m (Wouter Dierckx personal communication (Dierckx et al., 2014), June 26, 2018). The SWIR PSF is about twice as wide as the PSF of the other bands, which stresses the fact that a distinct spatial adaptation might be applied to each band. The PSFs of the bands are modeled as 2 dimensional Gaussian filters, which are applied to the 30 m resolution Landsat-8 bands. The filtered image is upscaled to the nominal 90 m resolution at nadir by taking 1 out of every 3 pixels. Finally, Lanczos interpolation is applied to upscale the image to the final 333 m Proba-V resolution. Notice that Lanczos is the interpolation method used at the Proba-V ground segment processing to upscale the acquired raw Proba-V data to the 333 m Plate Carée grid (Dierckx et al., 2014).

We transformed the associated ground truth (DA_Y in Fig. 1) of the Landsat-8 datasets using basically the same procedure. For the binary cloud mask, we apply the Gaussian filter, the 3 × 3 upscaling, and the lanczos interpolation to produce a 333 m resolution image; afterwards, the image is binarized applying a threshold, which is set to 0.5 for cloudy pixels. For the transformation of the cloud masks from the Proba-V 333 m resolution to the 30 m resolution, we use a simple bicubic interpolation. Spectral and spatial transformations of both Landsat-8 images and associated cloud masks are depicted in Fig. 3.

3.4. Fully convolutional neural networks

Fully convolutional neural networks (FCNN) are the model of choice to learn the mapping function (f_S and f_T in Fig. 1). FCNN are state-of-the-art models for image segmentation because of their capacity to exploit the spatio-spectral information of the input data. FCNNs, when provided with a large amount of training data, have shown very high accuracy levels on several image segmentation tasks (Chen et al., 2018a; Lin et al., 2018; Chen et al., 2018b). Although the reasons of their success are still poorly understood (Zhang et al., 2017; Szegedy et al., 2014), it is acknowledged that the hierarchy of stacks of spatio-spectral convolutions are good priors for vision systems (Yosinski et al., 2014). In addition, it has been shown in many works that they usually attain higher performance than classification methods with manually designed spatio-spectral features (Wieland et al., 2019; Mateo-García et al., 2017).

In this work, fully convolutional neural networks solve a standard multi-output binary classification problem where the input is a 4-band image and the output is a two-dimensional map. This output has values

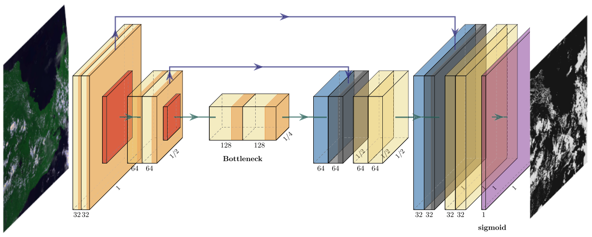


Fig. 4. Proposed FCNN architecture, based on Ronneberger et al. (2015), for cloud detection: inputs are 4-band TOA ref.lectance images.

between 0 to 1 that can be interpreted as the probability of cloud of the underlying pixel. The stacked set of convolutional filters seek to exploit the spatial information of nearby pixels to provide the cloud mask of each pixel, which is crucial in the context of reduced spectral information, with only 4 spectral bands, as in Proba-V.

Fully convolutional neural networks design has been constantly evolving since the burst of deep learning applications for image segmentation (Farabet et al., 2013; Long et al., 2015; Chen et al., 2015). In most of these applications, the FCNN architectures consist of an encoder module formed of convolutional filters that pool the image several times plus a decoder module that unpool the reduced feature vectors to the original image size to conform the prediction. Since all operations are convolutions and point-wise non-linearities, the networks can be applied to images of arbitrary size with fast inference times. The U-Net architecture proposed in Ronneberger et al. (2015) is a well-known fully convolutional architecture that has been applied in several fields from computer vision to medical imagery (Ronneberger et al., 2015; Drozdzal et al., 2016; Breininger et al., 2018). It has been extensively employed also in remote sensing (Schuegraf and Bittner, 2019; Wieland et al., 2019; Jeppesen et al., 2019; Drönner et al., 2018) and, in particular, for cloud detection with the RS-Net network (Jeppesen et al., 2019) and in Wieland et al. (2019) for Landsat-8. It has 5 pooling/unpooling stages and it adds skip connections between feature maps of the same resolution. Overall, the U-Net is conceptually simple yet accurate and provides fast predictions, which is mandatory in remote sensing and for cloud detection in particular. In this work, we adapted the U-Net architecture by reducing the number of pooling steps from five to two, by using separable convolutions layers (Chollet, 2017), and by replacing the output of the network to work with binary classification instead of the multiclass classification. These modifications follow the hypothesis that cloud detection at 333 m resolution can be solved with less parameters and with less downscaling steps. The RS-Net (Jeppesen et al., 2019) for Landsat-8 used 5 downscaling steps whereas we use 2, which makes sense since they were working with 30 m resolution data.

Fig. 4 shows an scheme of the proposed architecture. The encoder part consists of 2 blocks of two times 3×3 separable convolution, batch normalization (Ioffe and Szegedy, 2015) and ReLU activation followed by a 2×2 max pooling. The bottleneck is also a block of two 3×3 separable convolution, batch normalization and ReLU activation. The decoder consists of two blocks of transpose convolution that is concatenated with the previous activations of the encoder, and two times 3×3 separable convolution, batch normalization and ReLU activation. Finally, a 1×1 convolution is applied to obtain the outputs (log-odds) that are passed through a sigmoid activation to obtain the final cloud probabilities.[2] In total, our FCNN architecture has 95,769 trainable parameters and it does 2.18 M floating point operations to compute the cloud mask of a 256×256 image. Compared to the U-Net architecture proposed in Jeppesen et al. (2019),Wieland et al. (2019), our proposed architecture has 99% less parameters and 92% less floating point operations: the U-Net has around 7.8 million parameters and needs 27.97 M floating point operations to compute a cloud mask of a 256×256 image.

In this work, two different training strategies are used: networks are either trained from *scratch* and using *fine-tuning*. Training from *scratch* refers to initialize the weights of the network randomly, while *fine-tuning* corresponds to use the weights from a previously trained network for initialization. Since the optimization of the neural network is in general a non convex problem, a different initialization of the weights may lead to different local minimum of the loss function, which could have a different test performance.

Once weights are initialized, we used mini-batch stochastic gradient descent to minimize the standard binary cross entropy loss with respect to those weights. This loss is defined as:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{i,j,k}^{B,S_1,S_2} y_{i,j,k}\log(\hat{y}_{i,j,k}) + (1 - y_{i,j,k})\log(1 - \hat{y}_{i,j,k}),$$

where $\hat{y}_{i,j,k}$ is the predicted network output in the (j, k) pixel of the ith image in the batch; $y_{i,j,k}$ is its corresponding label in the ground truth; B is the batch size; and $S_1 \times S_2$ is the size of the image.

4. Labeled datasets

This section describes the labeled datasets used for Landsat-8 and for Proba-V. Manually annotated cloud masks are essential to train and validate cloud detection algorithms designed to work globally, over different land covers, and with different atmospheric conditions. In this work, we use the publicly available L8Biome (U.S. Geological Survey, 2016b) and L8SPARCS (U.S. Geological Survey, 2016a) datasets for Landsat-8, and an improved version of the dataset developed in the context of the ESA Round Robin exercise (Iannone et al., 2017) for Proba-V (Fig. 5).

4.1. Landsat-8 datasets and ground truth

As mentioned before, one of the motivations to explore TL across Landsat-8 and Proba-V is the availability of public Landsat-8 image datasets with the corresponding cloud mask, which are used as ground truth by supervised machine learning algorithms. We use the open access L8Biome (U.S. Geological Survey, 2016b) and L8SPARCS (U.S. Geological Survey, 2016a) datasets as provided by Foga et al. (2017).

The L8Biome dataset was developed by the authors of Foga et al. (2017). It contains 96 Landsat-8 Level 1T products fully labeled using three classes: clear, thin cloud, and cloud. We fused the last two (thin cloud and cloud) to obtain a binary cloud mask. The products are scattered around the world covering the 8 major biomes. The average size of each product is 8000×8000 pixels. For some of the experiments, we used the same train-test split as in Li et al. (2019), containing 73 training and 19 testing images, respectively.

The L8SPARCS dataset was collected for the validation of the method proposed in Hughes and Hayes (2014). It contains 80 Landsat-8 Level 1T subscenes. They were manually labeled using five different classes: cloud, cloud-shadow, snow/ice, water, flooded, and clear-sky. We merged all the non-cloud classes (cloud-shadow, snow/ice, water, flooded, and clear-sky) in the *clear* class for this work. Each subscene is 1000×1000 pixels, hence the amount of data compared with the L8Biome dataset is much lower.

4.2. Proba-V dataset and ground truth

The Proba-V dataset is formed by 72 Proba-V level 2A products (processing version v101) that were manually labeled by the authors. This dataset is a corrected and extended version of the dataset created in the framework of the ESA Round Robin exercise (Iannone et al.,

[2] The detailed implementation of the model is available at https://gist.github.com/gonzmg88/8a27dab653982817034938b0af1a2bf7.

2017), which was also employed in Mateo-García et al. (2017), Mateo-García and Gómez-Chova (2018). For this work, the manual labels have been extensively improved following a manual procedure by two different experts. All pixels within the 72 scenes are annotated as *cloudy*, *clear*, or *uncertain*. For *uncertain* pixels the human expert could not clearly decide whether they were cloud contaminated or cloud free. *Uncertain* pixels are thus not considered for neither training nor testing purposes.

In order to assess the quality of the ground truth, 950 pixels coming from 12 different images were also labeled pixel-by-pixel by independent experts (Stelzer et al., 2017). The disagreement between these pixel-wise labels and the fully labeled scenes is 6.62%. This error is similar to the 7% error reported in Scaramuzza et al. (2012). In addition, a further analysis of the discrepancies shows that they arise mainly in semi-transparent thin clouds over the ocean, where it was difficult even for an experienced user to distinguish clouds. Nevertheless, this error constitutes a lower bound on the error a model can achieve using these labels; i.e. we cannot really distinguish between models with errors below 6% with this dataset.

We split the Proba-V dataset into train and test. The train dataset is formed by 48 of those images: we will refer to this dataset as PV48. The test dataset is formed by the remaining 24 products, which we will call PV24. Fig. 5 shows the location of the training and testing products. These labeled products are also available for inspection.[3]

5. Experimental setup

The experimental design seeks to answer several questions which can be summarized in three: (1) Can models trained with Landsat-8 data be adapted to work in the Proba-V domain? (2) Can models trained with Proba-V images (333 meter resolution) be applied to Landsat-8 images (30 meter resolution)? and (3) Does combining data from the source and target domains increase accuracy of trained models? Questions 1 and 2 are thus related to *transductive TL* problems, while Question 3 involves *inductive TL*. To answer these questions, we performed two blocks of experiments summarized in Table 1: one for transductive TL and one for inductive TL.

In the transductive TL, we explored two different scenarios: (1) only having Landsat-8 labeled data and (2) only having Proba-V labeled data. For each scenario, once the models are trained, we performed two tasks: first we validate the models in the source domain, and then we evaluate them on the target domain where they were designed to work. As explained before, in the first scenario we will use the TL Scheme 1 and in the second scenario the TL Scheme 2 (Section 3.2).

The inductive TL experiment answers the third question formulated above. It consists of training a model in the Proba-V domain using simultaneously Proba-V data and adapted Landsat-8 data. Landsat-8 data is transformed to the Proba-V domain for training using the spatio-spectral transformation explained in Section 3.3.

The employed TL models of each experiment are summarized in Table 2. The models are denoted by $\text{TL}_{\text{Sat,SR}}$, where *Sat* makes reference to the satellite from which the training data came from (L8, Landsat-8, and PV, Proba-V); *SR* refers to the spatial resolution used when training the model, which can be 30 meters (the Landsat-8 resolution) or 333 meters (the Proba-V resolution). Note that when the satellite is L8 and the resolution is 333 m it means that, for training the model, the L8 images and the ground truth have been transformed to the Proba-V domain using the spatio-spectral domain adaptation in Section 3.3.

5.1. Transductive transfer learning: from Landsat-8 to Proba-V

In this experiment, we assume that we only have labeled data from Landsat-8 for training. In this setting, we trained two models that follow the TL Scheme 1. The first model, $\text{TL}_{\text{L8,30}}$, uses as domain adaptation step only the spectral transformation and does not apply the spatial transformation (Section 3.3). The second model, $\text{TL}_{\text{L8,333}}$, uses both steps, the spectral and the spatial one, to adapt Landsat-8 labeled images to the Proba-V domain. Both models can be directly applied to Proba-V data. In the case of the first model, this is technically possible even though it is trained on images of different spatial resolution since the model is based on a Fully Convolutional architecture (Section 3.4) thus it can be applied to images of any size.

In order to ensure that the models are working properly, we perform a preliminary test on Landsat-8 data. Notice that this is a realistic situation since we assume we only have labeled data from this domain. While testing the first model in the Landsat-8 domain is straightforward (i.e. the spatial resolution of the predicted cloud mask is the same as the original one), to test the second model in the Landsat-8 domain we have to undo the spatial adaptation. In order to do so, we downscale the resulting cloud mask back to the 30 m resolution by simple bicubic interpolation (Section 3.3). Specifically, we preform two tests on the Landsat-8 source domain in order to compare with the works (Li et al., 2019 and Jeppesen et al., 2019): in the first one, we follow the experimental setup of Li et al. (2019), which consists in using 73 images from the L8Biome dataset for training and the remaining 19 for testing[4]. The second one, following the setup of Jeppesen et al. (2019), uses all the L8Biome images for training and the L8SPARCS for testing.

Once we checked that the trained models work in the source domain, we evaluate their performance in the target domain (i.e. in Proba-V images from the PV24 test dataset). With this experiment we want to demonstrate that (1) transductive transfer learning works from Landsat-8 to Proba-V, and (2) that both of the domain adaptation steps (spectral and spatial) are required to enable transfer learning.

5.2. Transductive transfer learning: from Proba-V to Landsat-8

When assuming that we only have Proba-V labeled data for training (Proba-V is the source domain and Landsat-8 the target domain) we will apply the TL Scheme 2 (Section 3.2). This model ($\text{TL}_{\text{PV,333}}$ in Table 2) is first trained and evaluated in the Proba-V domain using the PV48 and the PV24 datasets, respectively. Afterwards we evaluate its performance in Landsat-8 images. To apply this model to Landsat-8 images, the TL Scheme 2 consists of (1) applying the spatio-spectral domain adaptation transformation to the Landsat-8 image (Section 3.3), (2) applying the Proba-V trained model, and (3) downscaling the resulting cloud mask prediction back to the 30 m resolution by simple bicubic interpolation.

5.3. Inductive transfer learning: from Landsat-8 to Proba-V

Finally, in the inductive block, we evaluate several models trained with an increasing amount of data coming from Proba-V and with all Landsat-8 data. Notice that, as in the first scenario of the transductive TL experiments, we use the TL Scheme 1 which trains the model in the target domain; therefore, it is straightforward to include extra labeled images from Proba-V for the joint training experiments. We analyze two different training strategies: (1) train models from *scratch* including simultaneously both the Landsat-8 and the Proba-V images, and (2) models are initialized using the parameters of the model $\text{TL}_{\text{L8,333}}$ and *fine-tuned* with the Proba-V images. Moreover, we compare with models trained only with the same Proba-V images from scratch. The training details of all models can be found in Appendix A.

[3] http://isp.uv.es/projects/cdc/probav_dataset.html

[4] They discarded 4 images from the 96 of the L8Biome because of errors in the labels.

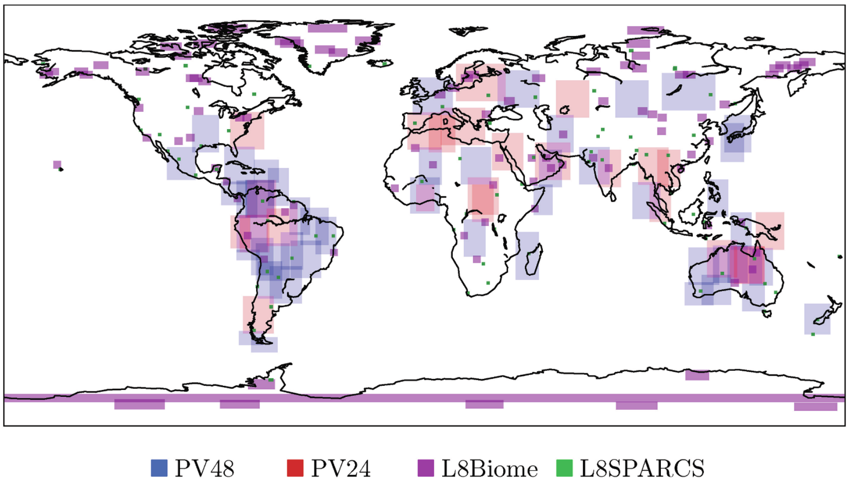


Fig. 5. Location of the used Landsat-8 and Proba-V datasets. Each image has a manually generated cloud mask.

Table 1
Experimental setup summary.

Experiment	Source domain	Training and Testing Tasks
Transductive TL	Landsat-8	(1) Training and validation on Landsat-8 (2) Evaluation on Proba-V
	Proba-V	(1) Training and validation on Proba-V (2) Evaluation on Landsat-8
Inductive TL	Landsat-8	(1) Training on Landsat-8 and Proba-V (2) Evaluation on Proba-V

6. Experimental results and discussion

In this section, we discuss the results for the different transfer learning experiments described in Section 5, and summarized in Table 1. We first present the transductive transfer learning results: we start with TL from Landsat-8 to Proba-V (Section 6.1), then TL from Proba-V to Landsat-8 (Section 6.2), afterwards results related to the robustness of the transductive models (Section 6.3), and finally a summary of all transductive transfer learning models in both domains and a comparison with independent state-of-the-art models (Section 6.4). Finally, we present the inductive transfer learning results (Section 6.5).

In order to test the models, we use the PV24 dataset in the Proba-V domain. In the Landsat-8 domain, we use the L8SPARCS and L8Biome datasets when they were not used for training. Testing is always performed in the native resolution of the given domain; hence, in the case of Proba-V, predicted masks are obtained at the 333 m resolution domain and, in the case of Landsat-8, the predicted cloud masks are obtained at the 30 m resolution of Landsat-8 images.

6.1. Transductive transfer learning results: Landsat-8 to Proba-V

In this subsection, we show results of the experimental setup explained in Section 5.1. First, we show results of our models, evaluated using the same train-test split used in Li et al. (2019) for the L8Biome dataset, and compare our results with theirs. Then, we show results of our models trained using all images from the L8Biome dataset, which are evaluated first in the Landsat-8 domain using the L8SPARCS dataset and later in the target Proba-V domain using the PV24 test dataset. The goal of this section is to demonstrate that the transfer learning between Landsat-8 and Proba-V using the proposed spatio-spectral domain adaptation is useful. Moreover, a complementary result is that using only the spectral domain adaptation is not sufficient to obtain an accurate model.

We evaluate the models on the L8Biome dataset using the train-test split proposed in Li et al. (2019). In particular, we use the same 73 images for training and 19 for testing, so results can be directly compared with (Li et al., 2019). We trained two models following the TL Scheme 1: the first model, $\mathrm{TL}_{\mathrm{L8,30}}$, using as domain adaptation transformation only the spectral step and the second one, $\mathrm{TL}_{\mathrm{L8,333}}$, using the whole spectral and spatial adaptation (cf. Section 3.3). It is worth to emphasize that, in order to apply the models to the Landsat-8 images, the images have to be previously transformed using the corresponding domain adaptation transform. After the model is applied, the corresponding cloud mask has to be transformed back to the source domain. In the case of the $\mathrm{TL}_{\mathrm{L8,333}}$, the mask is downscaled to 30 m using bicubic

Table 2
Experimental setup of the different trained models depending on the transfer learning direction across sensors (i.e. data used for training and testing).

Model name	Source Domain (train data)	Target Domain (test data)	TL Scheme (Section 3.2)	Domain Adaptation	TL direction
$\mathrm{TL}_{\mathrm{L8,30}}$	Landsat-8 (30m)	Proba-V (333m)	Sch.1: train in target domain	Spectral	L8 to PV
$\mathrm{TL}_{\mathrm{L8,333}}$	Landsat-8 (333m)	Proba-V (333m)	Sch.1: train in target domain	Spectral & Spatial	
$\mathrm{TL}_{\mathrm{PV,333}}$	Proba-V (333m)	Landsat-8 (30m)	Sch.2: train in source domain	Spectral & Spatial	PV to L8
$\mathrm{TL}_{\mathrm{L8+PV,333}}$	Landsat-8 & Proba-V (333m)	Proba-V (333m)	Sch.1: train in target domain	Spectral & Spatial	L8 to PV

Table 3
Results over the 19 test images of the L8Biome dataset used in Li et al. (2019). Proposed models ($TL_{L8,333}$ and $TL_{L8,30}$) and the model from Li et al. (2019) (MSCFF) were all trained using the same 73 images of the L8Biome dataset.

Model	Commission Error%	Omission Error%	Overall Accuracy%	F_1 score%
$TL_{L8,333}$	6.48	7.67	92.90	93.11
$TL_{L8,30}$	6.63	5.58	93.92	94.17
FMask (Zhu et al., 2015)	–	6.99	89.59	89.3
MSCFF (Li et al., 2019) (all bands)	–	6.07	94.96	94.5
MSCFF (Li et al., 2019) (NRGB)	–	5.48	93.94	92.6

interpolation. Table 3 shows the results for the Landsat-8 test images. As one can see, both proposed models have a similar performance. Although the model $TL_{L8,333}$ works in a different spatial resolution, it is only one point less accurate than the model that uses directly the 30 m resolution data ($TL_{L8,30}$).

Results from the MSCFF network (Li et al., 2019) and from FMask (Zhu et al., 2015) are included in Table 3 for comparison purposes. For the MSCFF network, we consider results using all the bands and using only the NIR, Red, Green, and Blue bands (NRGB). We can see that our network trained at 30 m resolution has a similar performance than MSCFF using NRGB bands, which indicates that our FCNN architecture squeeze a similar amount of information than MSCFF even though it has much less trainable parameters and pooling steps. In addition, the network trained with the 333 m resolution data ($TL_{L8,333}$) is 2 points less accurate than MSCFF using all bands (Li et al., 2019). However, it provides a more accurate cloud mask than the operational Landsat-8 cloud detection algorithm, FMask (Zhu et al., 2015), for these 19 images. This highlights that the 333 m resolution image retains sufficient information to provide an accurate cloud mask even for the 30 m product; i.e. the implicit smoothing effect of the employed upscaling-downscaling approach does not affect the overall cloud detection accuracy —although some effects at cloud borders might be expected.

Since these preliminary results were satisfactory, we retrained both networks from scratch using all the images of the L8Biome dataset as described in Section 5.1. In order to analyze the robustness of the networks to different weight initialization, we trained 10 copies of the network that uses the spectro-spatial domain adaptation ($TL_{L8,333}$) using different random seeds. Robustness results will be further analyzed in Section 6.3.

In Table 4, the results of these models tested on the Landsat-8 L8SPARCS dataset are shown. First of all, we see that, as expected, results of the 10 copies of $TL_{L8,333}$ exhibit a low variability for the ten different runs. This agrees with our hypothesis that a different initialization of the weights leads to consistent train and test accuracy values. Regarding the networks performance, networks trained using the spatio-spectral domain adaptation ($TL_{L8,333}$) are around 2 points less accurate compared with the network that work in 30 m resolution ($TL_{L8,30}$) and the RS-Net network of the work (Jeppesen et al., 2019). For the RS-Net, we consider again results using the RGB bands plus NIR (NRGB) and results using all bands except the thermal (all-NT), which were the best performing model for the L8SPARCS dataset in Jeppesen et al. (2019). In this case, it is also worth to mention that the network that only uses the spectral domain adaptation transformation ($TL_{L8,30}$) has almost the same accuracy than RS-Net (Jeppesen et al., 2019) even tough (*a*) the network has 99% less trainable parameters and (*b*) it uses less Landsat-8 spectral bands.

Table 4
Results of models trained with the L8Biome dataset and tested in the source Landsat-8 domain using L8SPARCS dataset. Both RS-Net models and our models using the L8Biome dataset for training.

Model	Acc%	F_1%
$TL_{L8,333}$	91.25–91.81	73.48–74.73
$TL_{L8,30}$	93.20	79.98
FMask (Foga et al., 2017)	92.47	81.61
RS-Net (Jeppesen et al., 2019) (all-NT)	93.26	80.62
RS-Net (Jeppesen et al., 2019) (NRGB)	92.53	76.99

Table 5
Results of the models trained with the L8Biome dataset and tested on the Proba-V target domain using the PV24 dataset.

Model	Acc%	F_1%
$TL_{L8,333}$	88.84–91.87	87.95–89.71
$TL_{L8,30}$	80.37	73.48
Operational Proba-V v101 (Wolters et al., 2015)	83.01	83.00

Once we showed that the proposed models trained in a transfer learning framework have a competitive performance, even with models trained specifically for the source domain, we evaluate the performance of the models in the target domain. Table 5 shows the trasfer learning results of Landsat-8 models into Proba-V data. In particular, this table shows the test results of the models trained using the Landsat-8 L8SPARCS dataset and tested in the Proba-V domain using the PV24 test set. Firstly, we see that the model trained using the spatio-spectral domain adaptation ($TL_{L8,333}$) is much more accurate than the Operational Proba-V cloud mask. This suggests that the proposed strategy could be used to design accurate ML models even before the satellite is launched. On the other hand, $TL_{L8,333}$ provides results between 8 to 10 points more accurate than the model trained using only the spectral transformation ($TL_{L8,30}$). This demonstrates that FCNN learn spatial patterns that are dependent on the spatial resolution and, therefore, in order to transfer learning between sensors of different spatial resolutions, a domain adaptation transformation that takes into account the spatial scale is required. Results of the 10 runs of the $TL_{L8,333}$ network show an unusual behaviour: the cloud detection accuracy and F_1 score vary within 3 points for the different random initializations. This dependency on the initialization contrasts with the results of these 10 runs in the Landsat-8 domain showed in Table 4. Our hypothesis is that a *data-shift* (Torralba and Efros, 2011) between the distribution of the Landsat-8 adapted data and the real Proba-V distribution still exists after the proposed adaptation. In our view, for some images in the real Proba-V domain, the networks extrapolate. Hence, predictions on these regions are correct for some networks and incorrect for others depending on its initialization. However, this implicit extrapolation does not significantly affect the quality of the predictions; we can see that, even in the worse case scenario, the proposed network trained with the spatio-spectraly adapted Landsat-8 data outperforms the Proba-V operational cloud detection algorithm (Wolters et al., 2015) by a large margin.

Finally, Fig. 6 shows some illustrative results of the cloud masks of three different models, all applied to Proba-V images not used for training. Those images have been selected to highlight critical cloud detection cases such as cloud ice discrimination, bright impervious surfaces, and sand and coastal areas. We show in white the agreement in cloudy pixels between the model prediction and the ground truth, in orange omission errors (the model predicts clear and the ground truth cloudy), and in blue commission errors (predictions indicate cloud and the ground truth clear). First example presents commission errors in the operational PV cloud mask over sandy beaches and water. The convolutional models do not exhibit those problems, although the model trained on the L8Biome dataset still has several omission errors mainly in cloud borders. Second example shows a winter acquisition over the Andes, in South America. In this case, the operational algorithm produce commission errors in the snowy mountains that convolutional models correctly detect; specially the model trained with Proba-V images, $TL_{PV,333}$. The last example also highlights several commission

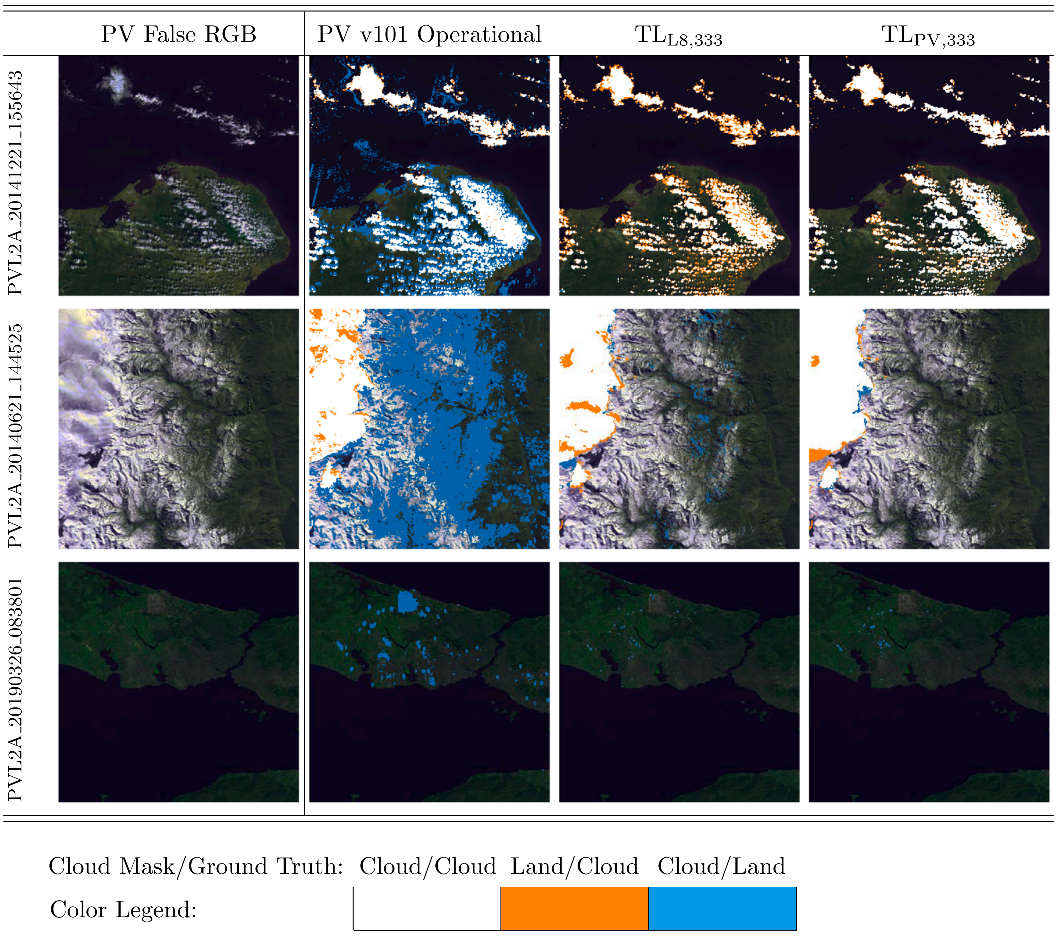


Fig. 6. Discrepancies between the ground truth and three models applied to three test sites of the PV24 test dataset.

errors of the operational algorithm over the city of Istanbul, in Turkey. Again the convolutional models exhibit a much lower amount of commissions. In Appendix B, we present more examples on Proba-V images for the interested reader.

6.2. Transductive transfer learning results: Proba-V to Landsat-8

In this section, we present and analyze the results of the networks trained only with Proba-V data. These networks are first tested in the Proba-V source domain using the PV24 test set and afterwards in the target Landsat-8 domain using the L8Biome dataset. The objective is to prove that models trained in the 10 times lower resolution Proba-V data can be also transferred to the 30 m Landsat-8 resolution with a negligible loss in accuracy.

We trained the model TL$_{PV,333}$ using the transfer learning Scheme 2 (Fig. 1) following the setup explained in Section 5.2. In particular, we use the PV48 dataset for training and we also train 10 copies of the network to evaluate the robustness to initialization. Table 6 shows results of this model in the source Proba-V domain using the PV24 test dataset. We can see that the model achieves a very high accuracy and that is not very sensitive to the initialization of the network.

In order to test the models in the Landsat-8 domain, we follow the procedure described in Section 5.2. Notice that, as we discussed before, using 333 m resolution data to resolve 30 m resolution images is an under-determined problem since there is information loss in the resampling process. Table 7 shows the performance metrics of the proposed model on the Landsat-8 domain using the L8Biome dataset as test set. We can see that the accuracy of the models trained with the Proba-V data, TL$_{PV,333}$, is similar to the accuracy of FMask (Zhu and Woodcock, 2012) and it is not far from deep learning approaches of recent works (Jeppesen et al., 2019). This highlight that cloud detection for a given resolution can be solved reasonably well using data with lower resolution, which indicates that much of the information loss due to the upscaling does not affect the final cloud mask predictions. Appendix B presents the cloud mask of this model for some additional cherry-picked Landsat-8 images.

6.3. FCNN robustness

As previously mentioned, we trained ten copies of the same network changing the random seed for both TL directions experiments in order to test the robustness of the transductive transfer learning models to the initialization. Fig. 7 shows the test accuracy of these models (TL$_{PV,333}$ and TL$_{L8,333}$) in both Proba-V and Landsat-8 domains using the PV24 and the SPARCS datasets, respectively. The most clear pattern we can see is that when the networks are tested in the source domain (i.e. in the same domain that they were trained), the accuracy is higher and with lower variability than when they are tested in the target domain. As we explained before, we ascribe this behaviour to the implicit extrapolation of the networks in the target domain: the different trained networks give different predictions in some parts of the target domain that is unknown to them. These results should be taken into account when

Table 6
Results of models trained in the Proba-V PV48 dataset over the Proba-V source domain using the PV24 dataset.

Model	Commission Error%	Omission Error%	Overall Accuracy%	F_1 score%
$TL_{PV,333}$	4.32–5.61	4.66–6.01	94.81–95.10	94.14–94.43
Operational Proba-V v101 (Wolters et al., 2015)	25.86	5.70	83.01	83.00

Table 7
Results of the model $TL_{PV,333}$ over the L8Biome dataset compared with other published results. The RS-Net (Jeppesen et al., 2019) model uses the L8SPARCS dataset for training.

Model	Commission Error%	Omission Error%	Overall Accuracy%	F_1 score%
$TL_{PV,333}$	10.99–17.13	6.01–10.55	87.79–89.77	87.95–89.71
RS-Net (Jeppesen et al., 2019)	–	5.51	91.59	91.52
FMask (Foga et al., 2017)	–	9.69	88.48	85.03

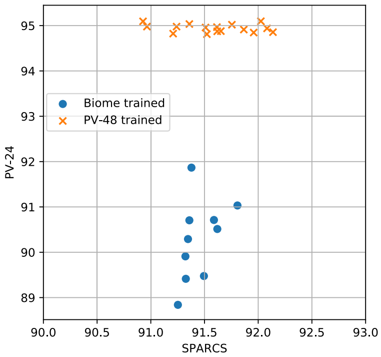


Fig. 7. Test accuracy of the models trained with Landsat-8 data from L8Biome dataset $TL_{L8,333}$ (blue) and with Proba-V data from PV48 dataset $TL_{PV,333}$ (orange). X-axis: accuracy in the L8SPARCS dataset; Y-axis: accuracy in the PV24 dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the hyper-parameters of the networks are tuned, since differences between hyper-parameters configuration might be due to noise caused by this extrapolation effect. It is also worth to mention that networks trained in the Proba-V domain ($TL_{PV,333}$) have similar accuracy, although with higher variability, than the networks trained on the Landsat-8 data adapted with the spatio-spectral domain adaptation: $TL_{L8,333}$.

6.4. Summary of transductive results

In this subsection we explore the connections between all previous experiments and compare their results. Since the proposed models can be evaluated in both domains (Landsat-8 and Proba-V), they can be inter-compared. Table 8 shows a summary of the results from the previous sections. The first column, *model*, refers to a particular architecture and TL scheme employed. The TL scheme of a particular model specifies how this model is tested in the source and target domain (see Section 5 for the details). The second column shows the dataset used to train the model; and the third column shows the dataset where the model is tested. The remainder columns are different measures of the performance of tested models. Notice that, if a given model is trained using a different dataset, it will end up with different parameters (i.e. the weight values of the network will be different).

Firstly, in the case of testing in the Proba-V domain, we can see that the model trained only with Landsat-8 is still much better than the threshold-based Proba-V Operational Cloud Detection model (Wolters et al., 2015); however, it is still far from the network trained with real Proba-V images. Therefore, it proofs to be a valid strategy with perspectives of improvement. Secondly, there is a dependency on the manual labeling procedure employed by the experts developing the ground truth: we see that models trained on data which was labeled using the same methodology for training and for testing have significantly higher accuracy. For example, networks trained on L8Biome data using the train-test split of Li et al. (2019) have a significantly higher accuracy (92.90%) than networks trained with all L8Biome dataset and tested in the L8SPARCS dataset (91.25–91.81%). In the case of the Proba-V domain, we see that networks trained with the PV48 dataset have also a very high accuracy in the PV24 dataset (94.81–95.10%), which may be also due to the fact that the PV48 and PV24 datasets were developed by the same experts using the same manual labeling approach. This dependence is also documented in other contexts involving classification like in Recht et al. (2018),Torralba and Efros (2011). In the case of cloud detection, this could be exacerbated by different criteria in the inclusion of thin clouds in the datasets, since in one dataset very thin semitransparent clouds might have been considered as clear pixels whereas for other this pixels might have been annotated as cloudy. Finally, in these results, it is important to consider the errors in the labeling procedure. These errors were estimated to be around 7% for Landsat (Scaramuzza et al., 2012) and 6.62% for Proba-V (see Section 4.2). Hence, models over 93% accuracy cannot be really compared or ranked, from a statistical point of view, using these datasets.

6.5. Inductive transfer learning results

In this section, we present and discuss results of models that use both datasets for training, simultaneously. This setting seeks to explore a scenario where there are few labeled images from a given (target) satellite sensor, which is often the case due to the high cost of manual labeling of clouds, and a larger corpus of labeled images from a different but similar sensor. Proba-V will be in these experiments the target domain, where few labeled images with cloud mask are available, whereas the Landsat-8 satellite will be the source domain with the L8Biome dataset as the large corpus of labeled images. The goal of the experiments is thus to test if networks trained using fine-tuning or joint training with the L8Biome dataset have a significantly better performance than networks trained from scratch using the few Proba-V images.

In order to train the models with Landsat-8 data we apply the TL Scheme 1 (Section 3.2) with the proposed spectro-spatial domain adaptation as explained in Section 5.3. In this setting, models are trained in the Proba-V domain hence, joint training consists of merging the dataset of the few Proba-V images with the dataset of Landsat-8 adapted images.

We trained several networks with an increasing number of real Proba-V images d from the PV48 dataset. For each number of Proba-V images, d, we selected 8 disjoint subsets of the PV48 dataset containing d images. For each of such subsets, three models were trained: (a) from *scratch* using the d Proba-V images in the subset; (b) *fine-tuning*, which uses those d Proba-V images to fine-tune a network trained previously

Table 8
Table with results over the different test sets of the proposed models and selected models of the literature. Ranges show minimum and maximum values obtained in 10 runs changing the random seed value for the initialization of the network weights.

Model	Train Set	Test Set	Commission Error%	Omission Error%	Overall Accuracy%	F_1 score%
TL$_{L8,333}$	L8Biome	L8SPARCS	1.16–1.86	36.34–37.82	91.25–91.81	73.48–74.73
TL$_{L8,30}$	L8Biome	L8SPARCS	1.24	29.91	93.20	79.98
TL$_{PV,333}$	PV48	L8SPARCS	1.05–3.26	33.08–40.84	90.93–92.14	71.68–76.27
FMask (Foga et al., 2017)	–	L8SPARCS	–	13.79	92.47	81.61
RS-Net (Jeppesen et al., 2019)	L8Biome	L8SPARCS	–	27.66	93.26	80.62
TL$_{L8,333}$	L8Biome (73)	L8Biome (19)	6.78	7.67	92.90	93.11
TL$_{L8,30}$	L8Biome (73)	L8Biome (19)	6.63	5.58	93.92	94.17
TL$_{PV,333}$	PV48	L8Biome (19)	7.32–10.5	6.83–9.79	90.85–91.89	91.11–92.22
FMask (Foga et al., 2017)	–	L8Biome (19)	–	6.99	89.59	89.3
MSCFF (Li et al., 2019) (all bands)	L8Biome (73)	L8Biome (19)	–	6.07	94.96	94.5
MSCFF (Li et al., 2019) (NRGB)	L8Biome (73)	L8Biome (19)	–	5.48	93.94	92.6
TL$_{PV,333}$	PV48	L8Biome	10.99–17.13	6.01–10.55	87.79–89.77	87.95–89.71
FMask (Foga et al., 2017)	–	L8Biome	–	9.69	88.48	85.03
RS-Net (Jeppesen et al., 2019)	L8SPARCS	L8Biome	–	5.51	91.59	91.52
TL$_{L8,333}$	L8Biome	PV24	5.10–12.18	8.42–14.63	88.84–91.87	87.20–90.69
TL$_{L8,30}$	L8Biome	PV24	5.00	38.23	80.37	73.48
TL$_{PV,333}$	PV48	PV24	4.32–5.61	4.66–6.01	94.81–95.10	94.14–94.43
Oper. PV v101 (Wolters et al., 2015)	–	PV24	25.86	5.70	83.01	83.00

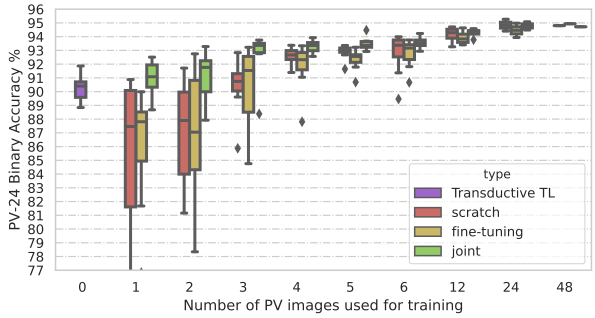


Fig. 8. Test accuracy over the PV24 test set of FCNN Joint models trained with different numbers of Proba-V images in red from scracth, in yellow using fine-tuning and in green using joint training. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

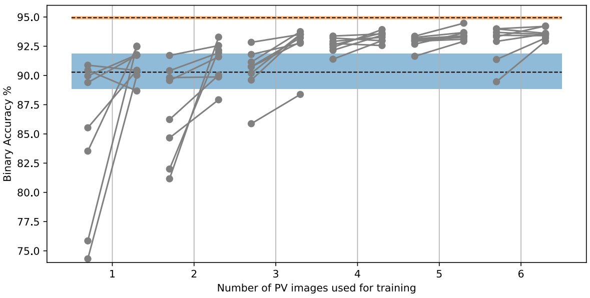


Fig. 9. Test accuracy of models trained using different number of Proba-V images. For each value, on left, only Proba-V data is used; on right, models trained jointly on Landsat-8 and Proba-V data. Blue shaded area depicts the accuracy of the models trained only in the L8Biome dataset. Orange area depicts the accuracy of models trained on all Proba-V images (PV48). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

in the L8Biome dataset[5]; and (*c*) *joint training*, which trains from scratch using the *d* Proba-V images together with all the images in the L8Biome dataset.

Fig. 8 shows the results of this experiment tested over the PV24 test set. Overall, we see that joint training has a better performance than training from scratch or using fine-tuning, which shows similar accuracy. In particular, we can see that using joint training increases the mean accuracy between 2 and 4 points in the scarce data scenarios with 1 to 3 images. In scenarios with 4 to 6 images for training, joint training still gives an small boost in accuracy and also reduces the variance of the resulting accuracy values. This indicates more robustness of the joint training solution. In scenarios with a larger amount of data, we see that the three methods have a similar performance. It is also worth to mention that joint training provides systematically an increase in the mean accuracy over the models trained only with Landsat-8 data without any Proba-V image (Section 6.1).

Fig. 9 compares joint training with training from scratch. In this figure, each point represent a subset of *d* images with *d* varying in the x-axis. Points on the left show the accuracy on the PV24 test set of the model trained from scratch using only those *d* images, whereas points on the right use these *d* images together with the images in the L8Biome dataset for training (joint learning). We see that, for the vast majority of those subsets, using joint training has a positive impact on the final performance of the model (points on the right). We see again that the variance of the joint models is reduced and that joint training consistently take advantage of the new Proba-V data to also perform better than the model that does not use it, which is trained only with the L8Biome dataset and depicted by the blue shaded area. Note that the orange area depicts the accuracy of models trained on all Proba-V images (PV48) that provides an upper bound for the cloud detection accuracy.

[5] For fine-tuning we used the network TL$_{L8,333}$ from subsection 6.1.

7. Conclusions

In this paper, we explored different transfer learning (TL) approaches to train machine learning (ML) methods for cloud detection in remote sensing images. In particular, we analyzed transductive and inductive TL frameworks using Landsat-8 and Proba-V as case studies. Both frameworks depend on a domain adaptation transformation that converts images from one satellite to resemble images acquired with the other satellite.

We proposed an image conversion method to adapt Landsat-8 images to the Proba-V spectral and spatial characteristics that enables TL across satellites. Our results suggest that it is important to use both the spatial and the spectral adaptation in order to fully exploit TL advantages.

The transductive transfer learning framework assumes that we only have data from one satellite. In this context, two different TL schemes were proposed and successfully tested. Each scheme allows for a different TL depending on the particular direction of the domain adaptation transformation: from the source domain to the target domain or from the target domain to the source domain. We show that ML models trained only with data from Landsat-8 can have a very good performance on Proba-V surpassing current operational algorithm (Wolters et al., 2015). This means that ML methods can be trained even before the satellite is launched, and obtain better performance than threshold-based approaches. We evaluated the proposed methods results in the context of state-of-the-art cloud detection methodologies based on deep learning (Jeppesen et al., 2019; Li et al., 2019).

In order to use the Proba-V data for predicting on Landsat-8 images, we proposed a TL scheme that takes advantage of the proposed Landsat-8 to Proba-V domain adaptation transformation. We showed that ML models trained only with Proba-V data have similar accuracy than operational Landsat-8 approaches such as FMask (Zhu and Woodcock, 2012) and are only two points less accurate than (Li et al., 2019), even though our method is trained with data on a 11 times lower spatial resolution.

The inductive transfer learning framework relies on merging data from two different domains. We showed that joining data from both satellites increases accuracy specially in the regimes where there is few data from the target Proba-V domain, although we do not see a significant improvement using fine-tuning.

We show that training only with the adapted Landsat-8 data suffers the *data-shift* (Torralba and Efros, 2011) problem. In particular, we trained 10 copies of the same network with different initialization weights and show that the error in the adapted Landsat-8 domain is lower and with less variance than in the Proba-V domain. We see that, in the former, the error ranges from 91.4% to 91.9% whereas for Proba-V the error is 88.8–90.7%. This contrasts with the belief that CNN initialization does not affect much the obtained solution. In this respect, there is still margin to improve the transfer learning results by improving the domain adaptation transformation and thus reducing the data-shift problem. Our next steps are fostered to improve the cloud detection accuracy by using the generative adversarial networks (GANs) framework (Mateo-García et al., 2019) to learn a transformation between Landsat-8 and Proba-V data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science, Innovation and Universities (MINECO, TEC2016-77741-R, ERDF), the European Social Fund and the European Space Agency (ESA IDEAS+ research grant, CCN008). We also thank the Proba-V Quality Working Group members and the Proba-V Cloud Detection Round Robin exercise participants, with a special mention to Wouter Dierckx (VITO) and Fabrizio Niro (ESA).

Appendix A. Models training technical details

For convenience, all networks were trained using as input patches of size 32×32, although the model is independent of this size. The test size corresponds to the size of the images: for example, in the case of the L8SPARCS dataset, the size of each test image is 1000×1000 pixels. The input of the networks is always top of atmosphere reflectance for both Proba-V (Wolters et al., 2018) and Landsat-8 (U.S. Geological Survey, 2019). The training patches were taken from the training images with a 16 pixel overlap as a form of data augmentation; we also employed horizontal and vertical flips and 90 degree rotations as data augmentation techniques. In all experiments, we used 64 as the batch size and the Adam optimizer (Kingma and Ba, 2015). All networks were trained using TensorFlow library (v1.12) (Abadi et al., 2015) and the weights of the network were initialized with the default initialization of each of the corresponding layers.

The networks $TL_{L8,333}$ are trained in Landsat-8 data from the L8Biome dataset transformed using the spatio-spectral domain adaptation transformation. After the spatial domain adaptation, the L8Biome dataset has 243,430 patches. We first train the network using the train-test split of Li et al. (2019) to compare with their approach. Then, the network is trained using all the L8Biome dataset. Networks were trained until no improvement was observed for 15 epochs. We used a learning rate of 10^{-4} and weight decay of $5 \cdot 10^{-3}$ for regularization purposes. We trained 10 copies of the same network architecture with different random seed initialization to ensure that results do not depend on the weight initialization or optimization process.

Networks $TL_{PV,333}$ are trained on the Proba-V data using the PV48 Proba-V dataset which corresponds to 1,891,095 patches. For these experiments we reduced the learning rate and the weight decay. In particular, we used a learning rate of 10^{-5} and weight decay of $5 \cdot 10^{-4}$. We also trained 10 copies of the same network to ensure consistency across initializations.

Networks $TL_{L8,30}$ are trained in the L8Biome dataset transformed using only the spectral domain adaptation. Since there is no spatial upscaling, the total amount of patches is much bigger than in the $TL_{L8,333}$ case (14,531,228 patches). We also trained the network first using the train-test split of Li et al. (2019) to compare with their approach. Networks were trained for 50 epochs using a learning rate of 10^{-5} and weight decay of $5 \cdot 10^{-4}$.

When the networks are trained jointly, we used the L8Biome dataset transformed using the spatio-spectral domain adaptation (i.e. same as above in $TL_{L8,3333}$), and an increasing number of images from the PV48 dataset. For the fine-tuning we used as initial weights the aforementioned network ($TL_{L8,333}$) trained with the L8Biome dataset.

Appendix B. Visual inspection of cloud detection results

In this appendix, we show additional results of the produced cloud masks for Proba-V and for Landsat-8, and compare them against the ground truth. All shown images have not been used for training by none of the models. As in Section 6, we show in white agreement in cloudy pixels between the model prediction and the ground truth, in orange omission errors (the model predicts clear and the ground truth cloudy), and in blue commission errors (predictions indicate cloud and the ground truth clear).

Fig. B.10 shows four additional results for Proba-V. Models shown are the operational Proba-V cloud detection (Wolters et al., 2015), the model trained on Landsat-8 data with the proposed domain adaptation $TL_{L8,333}$ and the model trained on Proba-V $TL_{PV,333}$. The first example presents omission errors of the operational Proba-V algorithm over cloudy areas with saturated pixels in the blue band. We see that both convolutional models solve this issue, nevertheless, the model trained on the L8Biome dataset still has several omission errors in cloud borders. Second example shows an acquisition over Corsica island, where all models capture most of the cloudy pixels. However, in this case, the operational model has commission errors in the snowy mountains in Corsica that convolutional models correct; specially the FCNN trained with Proba-V images $\mathrm{TL}_{\mathrm{PV},333}$. The third example also highlights several commission errors of the operational algorithm, in this case over coastal waters. Again the convolutional models do not exhibit this problem although there are very thin clouds over land that are undetected. Finally, last acquisition shows a salty lake in Central Anatolia (Tuz Lake), where we can see that the operational algorithm incurs in several commission errors. The models based on FCNNs exhibit less commissions in the case of $TL_{L8,333}$ and none in the model trained with Proba-V data $TL_{PV,333}$.

Fig. B.11 shows results for Landsat-8 of four images in the L8SPARCS dataset. In the case of Landsat-8, the models selected are the operational FMask (Zhu et al., 2015), the model trained with Proba-V data $TL_{PV,333}$ and the model trained with Landsat-8 data at its original resolution $TL_{L8,30}$.

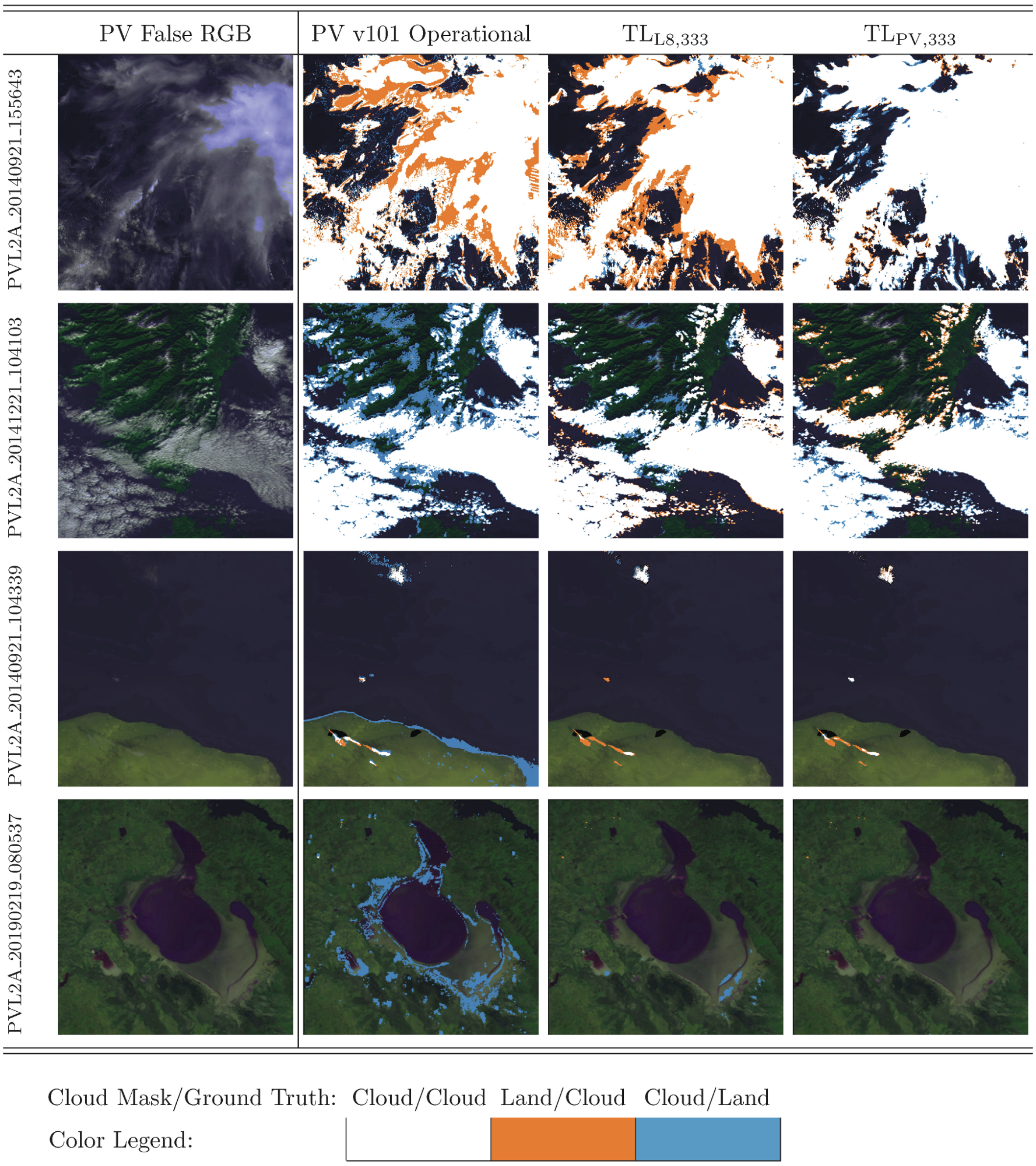


Fig. B.10. Discrepancies between the ground truth and three models applied to four test sites of Proba-V.

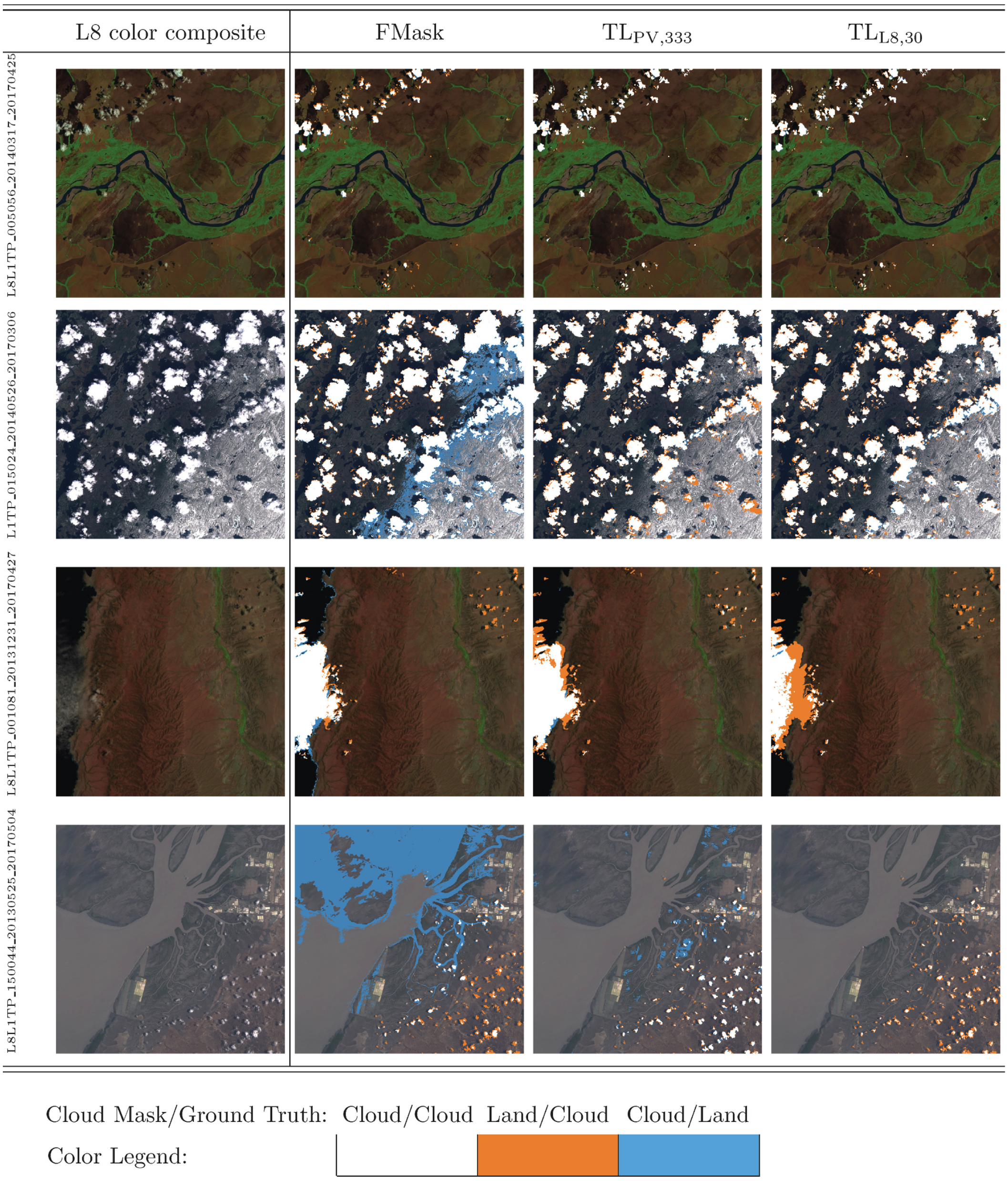

L8 color composite
FMask
$TL_{PV,333}$
$TL_{L8,30}$
L8L1TP_005056_20140317_20170425
L1TP_015024_20140526_20170306
L8L1TP_001081_20131231_20170427
L8L1TP_150044_20130525_20170504
Cloud Mask/Ground Truth: Cloud/Cloud Land/Cloud Cloud/Land
Color Legend:

Fig. B.11. Discrepancies between the ground truth and three models applied to four test sites of the L8SPARCS dataset.

First row shows the Vichada river in the border of Colombia and Venezuela. We can see that overall all three models provide sensible cloud masks. FMask exhibit a slightly higher amount of omission errors for very thin clouds in the bottom part of the images which the models based on FCNN do not exhibit. Second row shows the tundra in the North of Quebec in late spring. We can see several commission errors of FMask in regions where the ice is melting; these false positives are not present in the FCNN models predictions nevertheless the model trained in Proba-V data omits some clouds in the icy surface. Third row, from the chilean coast in South America, shows very thin clouds in the upper right part of the images that the three models mainly omit. In addition, FMask shows systematic commission errors in the coast pixels that the FCNN models do not have. Last row is an acquisition from a salt marsh in the Little Rann of Kutch in India. It contains muddy water with a big amount of suspended sediments and salt evaporation ponds. We can see that FMask has large commission errors in these muddy waters and it also failed to identify thin clouds in the bottom right of the image. In contrast, the model trained on Proba-V data shows few commission errors mostly in the salt pans and it is able to capture most of thin clouds in the image. The model trained with 30 m Landsat-8 data does not show commission errors, however, it also failed to identify several thin clouds.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems, software available from tensorflow.org. http://tensorflow.org/.

Azimi, M., Zekavat, S.A., 2000. Cloud classification using support vector machines. In: IEEE Int. Geoscience And Remote Sensing Symposium. IGARSS'2000, vol. 2, Hawaii, USA, pp. 669–671.

Baetens, L., Desjardins, C., Hagolle, O., 2019. Validation of Copernicus Sentinel-2 Cloud Masks Obtained from MAJA, Sen2cor, and FMask Processors Using Reference Cloud Masks Generated with a Supervised Active Learning Procedure. Remote Sens. 11 (4), 433. https://doi.org/10.3390/rs11040433.

Bai, T., Li, D., Sun, K., Chen, Y., Li, W., 2016. Cloud detection for high-resolution satellite imagery using machine learning and multi-feature fusion. Remote Sens. 8 (9), 715. https://doi.org/10.3390/rs8090715.

Breininger, K., Albarqouni, S., Kurzendorfer, T., Pfister, M., Kowarschik, M., Maier, A., 2018. Intraoperative stent segmentation in X-ray fluoroscopy for endovascular aortic repair. Int. J. Comput. Assist. Radiol. Surg. 13 (8), 1221–1231. https://doi.org/10.1007/s11548-018-1779-6.

Chai, D., Newsam, S., Zhang, H.K., Qiu, Y., Huang, J., 2019. Cloud and cloud shadow detection in Landsat imagery based on deep convolutional neural networks. Remote Sens. Environ. 225, 307–316. https://doi.org/10.1016/j.rse.2019.03.007.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2015. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: International Conference on Learning Representations (ICLR), pp. 1–14 arXiv: 1412.7062.

Chen, L.-C., Collins, M., Zhu, Y., Papandreou, G., Zoph, B., Schroff, F., Adam, H., Shlens, J., 2018a. Searching for efficient multi-scale architectures for dense image prediction. In: Advances in Neural Information Processing Systems 31, Curran Associates Inc, pp. 8699–8710.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Computer Vision – ECCV 2018, Lecture Notes in Computer Science, Springer International Publishing, pp. 833–851.

Chen, N., Li, W., Gatebe, C., Tanikawa, T., Hori, M., Shimada, R., Aoki, T., Stamnes, K., 2018c. New neural network cloud mask algorithm based on radiative transfer simulations. Remote Sens. Environ. 219, 62–71. https://doi.org/10.1016/j.rse.2018.09.029.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018d. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. 40 (4), 834–848. https://doi.org/10.1109/TPAMI.2017.2699184.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1800–1807. https://doi.org/10.1109/CVPR.2017.195.

Coluzzi, R., Imbrenda, V., Lanfredi, M., Simoniello, T., 2018. A first assessment of the Sentinel-2 Level 1-C cloud mask product to support informed surface analyses. Remote Sens. Environ. 217, 426–443. https://doi.org/10.1016/j.rse.2018.08.009.

Csurka, G. (Ed.), 2017. Domain Adaptation in Computer Vision Applications, Advances in Computer Vision and Pattern Recognition. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-58347-1.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009,. ImageNet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR09), 2009, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848.

Dierckx, W., Sterckx, S., Benhadj, I., Livens, S., Duhoux, G., Van Achteren, T., Francois, M., Mellab, K., Saint, G., 2014. PROBA-V mission for global vegetation monitoring: standard products and image quality. Int. J. Remote Sens. 35 (7), 2589–2614. https://doi.org/10.1080/01431161.2014.883097.

Drönner, J., Korfhage, N., Egli, S., Mühling, M., Thies, B., Bendix, J., Freisleben, B., Seeger, B., 2018. Fast cloud segmentation using convolutional neural networks. Remote Sens. 10 (11), 1782. https://doi.org/10.3390/rs10111782.

Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., Pal, C., 2016. The Importance of Skip Connections in Biomedical Image Segmentation, in: Deep Learning and Data Labeling for. In: Deep Learning and Data Labeling for Medical Applications, Lecture Notes in Computer Science. Springer International Publishing, pp. 179–187.

Farabet, C., Couprie, C., Najman, L., LeCun, Y., 2013. Learning hierarchical features for scene labeling. IEEE Trans. Pattern Anal. Mach. Intell. 35 (8), 1915–1929. https://doi.org/10.1109/TPAMI.2012.231.

Foga, S., Scaramuzza, P.L., Guo, S., Zhu, Z., Dilley, R.D., Beckmann, T., Schmidt, G.L., Dwyer, J.L., Joseph Hughes, M., Laue, B., 2017. Cloud detection algorithm comparison and validation for operational Landsat data products. Remote Sens. Environ. 194 (Supplement C), 379–390. https://doi.org/10.1016/j.rse.2017.03.026.

Frantz, D., Haß, E., Uhl, A., Stoffels, J., Hill, J., 2018. Improvement of the Fmask algorithm for Sentinel-2 images: separating clouds from bright surfaces based on parallax effects. Remote Sens. Environ. 215, 471–481. https://doi.org/10.1016/j.rse.2018.04.046.

Ghasemian, N., Akhoondzadeh, M., 2018. Introducing two Random Forest based methods for cloud detection in remote sensing images. Adv. Space Res. 62 (2), 288–303. https://doi.org/10.1016/j.asr.2018.04.030.

Ghosh, A., Pal, N., Das, J., 2006. A fuzzy rule based approach to cloud cover estimation. Remote Sens. Environ. 100, 531–549. https://doi.org/10.1016/j.rse.2005.11.005.

Gómez-Chova, L., Camps-Valls, G., Calpe, J., Guanter, L., Moreno, J., 2007. Cloud-screening algorithm for ENVISAT/MERIS multispectral images. IEEE Trans. Geosci. Remote Sens. 45 (12, Part 2), 4105–4118. https://doi.org/10.1109/TGRS.2007.905312.

Gómez-Chova, L., Camps-Valls, G., Bruzzone, L., Calpe-Maravilla, J., 2010. Mean map kernel methods for semisupervised cloud classification. IEEE Trans. Geosci. Remote Sens. 48 (1), 207–220. https://doi.org/10.1109/TGRS.2009.2026425.

Gómez-Chova, L., Muñoz-Marí, J., Amorós-López, J., Izquierdo-Verdiguier, E., Camps-Valls, G., 2013. Advances in synergy of AATSR-MERIS sensors for cloud detection. In: Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International, pp. 4391–4394. https://doi.org/10.1109/IGARSS.2013.6723808.

Helber, P., Bischke, B., Dengel, A., Borth, D., 2018. Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. In: IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium, pp. 204–207. https://doi.org/10.1109/IGARSS.2018.8519248.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., Darrell, T., 2018. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In: International Conference on Machine Learning, pp. 1989–1998.

Hollstein, A., Segl, K., Guanter, L., Brell, M., Enesco, M., 2016. Ready-to-use methods for the detection of clouds, cirrus, snow, shadow, water and clear sky pixels in Sentinel-2 MSI Images. Remote Sens. 8 (8), 666. https://doi.org/10.3390/rs8080666.

Hughes, M.J., Hayes, D.J., 2014. Automated detection of cloud and cloud shadow in single-date landsat imagery using neural networks and spatial post-processing. Remote Sens. 6 (6), 4907–4926. https://doi.org/10.3390/rs6064907.

Iannone, R.Q., Niro, F., Goryl, P., Dransfeld, S., Hoersch, B., Stelzer, K., Kirches, G., Paperin, M., Brockmann, C., Gómez-Chova, L., Mateo-García, G., Preusker, R., Fischer, J., Amato, U., Serio, C., Gangkofner, U., Berthelot, B., Iordache, M.D., Bertels, L., Wolters, E., Dierckx, W., Benhadj, I., Swinnen, E., 2017. Proba-V cloud detection Round Robin: Validation results and recommendations. In: 2017 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp), pp. 1–8. https://doi.org/10.1109/Multi-Temp.2017.8035219.

Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456.

Irish, R.R., Barker, J.L., Goward, S.N., Arvidson, T., 2006. Characterization of the Landsat-7 ETM+ Automated Cloud-Cover Assessment (ACCA) Algorithm. Photogramm. Eng. Remote Sens. 72 (10), 1179–1188.

Irons, J.R., Dwyer, J.L., Barsi, J.A., 2012. The next Landsat satellite: The Landsat Data Continuity Mission. Remote Sens. Environ. 122, 11–21. https://doi.org/10.1016/j.rse.2011.08.026. (landsat Legacy Special Issue).

Ishida, H., Oishi, Y., Morita, K., Moriwaki, K., Nakajima, T.Y., 2018. Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions. Remote Sens. Environ. 205, 390–407. https://doi.org/10.1016/j.rse.2017.11.003.

Jean, N., Burke, M., Xie, M., Davis, W.M., Lobell, D.B., Ermon, S., 2016. Combining satellite imagery and machine learning to predict poverty. Science 353 (6301), 790–794. https://doi.org/10.1126/science.aaf7894.

Jeppesen, J.H., Jacobsen, R.H., Inceoglu, F., Toftegaard, T.S., 2019. A cloud detection algorithm for satellite imagery based on deep learning. Remote Sens. Environ. 229, 247–259. https://doi.org/10.1016/j.rse.2019.03.039.

Kemker, R., Salvaggio, C., Kanan, C., 2018. Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning. ISPRS J. Photogramm. Remote Sens. 145, 60–77. https://doi.org/10.1016/j.isprsjprs.2018.04.014.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings, pp. 1–13.

Li, Z., Shen, H., Li, H., Xia, G., Gamba, P., Zhang, L., 2017. Multi-feature combined cloud and cloud shadow detection in GaoFen-1 wide field of view imagery. Remote Sens. Environ. 191, 342–358. https://doi.org/10.1016/j.rse.2017.01.026.

Li, X., Zhang, L., Du, B., Zhang, L., Shi, Q., 2017. Iterative reweighting heterogeneous transfer learning framework for supervised remote sensing image classification. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 10 (5), 2022–2035. https://doi.org/10.1109/JSTARS.2016.2646138.

Li, Z., Shen, H., Cheng, Q., Liu, Y., You, S., He, Z., 2019. Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. ISPRS J. Photogramm. Remote Sens. 150, 197–212. https://doi.org/10.1016/j.isprsjprs.2019.02.017.

Lin, D., Ji, Y., Lischinski, D., Cohen-Or, D., Huang, H., 2018. Multi-scale context intertwining for semantic segmentation. In: The European Conference on Computer Vision (ECCV), pp. 603–619.

Liu, C.-C., Zhang, Y.-C., Chen, P.-Y., Lai, C.-C., Chen, Y.-H., Cheng, J.-H., Ko, M.-H., 2019. Clouds Classification from Sentinel-2 Imagery with Deep Residual Learning and Semantic Image Segmentation. Remote Sens. 11 (2), 119. https://doi.org/10.3390/rs11020119.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431–3440. https://doi.org/10.1109/CVPR.2015.7298965.

Lu, C., Li, W., 2018. Ship classification in high-resolution sar images via transfer learning with small training dataset. Sensors 19 (1). https://doi.org/10.3390/s19010063.

Mateo-García, G., Gómez-Chova, L., 2018. Convolutional neural networks for cloud screening: transfer learning from Landsat-8 to Proba-V. In: IGARSS 2018, pp. 2103–2106. https://doi.org/10.1109/IGARSS.2018.8517975.

Mateo-García, G., Gómez-Chova, L., Camps-Valls, G., 2017. Convolutional neural networks for multispectral image cloud masking. IEEE International Geoscience and Remote Sensing Symposium (IGARSS) 2017, 2255–2258. https://doi.org/10.1109/IGARSS.2017.8127438.

Mateo-García, G., Laparra, V., Gómez-Chova, L., 2019. Domain adaptation of Landsat-8 and Proba-V data using generative adversarial networks for cloud detection. In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS) 2019, pp. 712–715. https://doi.org/10.1109/IGARSS.2019.8899193.

Mohajerani, S., Saeedi, P., 2019. Cloud-Net: An End-to-end Cloud Detection Algorithm for Landsat 8 Imagery. In: IGARSS 2019, 2019, to appear at 2019 IEEE International Geoscience and Remote Sensing Symposium (IGARSS).

Mohajerani, S., Krammer, T.A., Saeedi, P., 2018. A cloud detection algorithm for remote sensing images using fully convolutional neural networks. In: 2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP), pp. 1–5. https://doi.org/10.1109/MMSP.2018.8547095.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Trans. Knowl. Data Eng. 22 (10), 1345–1359. https://doi.org/10.1109/TKDE.2009.191.

Preusker, R., Huenerbein, A., Fischer, J., 2006. Cloud detection with MERIS using oxygen absorption measurements. Geophys. Res. Abstracts 8, 09956.

Qiu, S., Zhu, Z., He, B., 2019. Fmask 4.0: Improved cloud and cloud shadow detection in Landsats 4–8 and Sentinel-2 imagery. Remote Sens. Environ. 231, 111205. https://doi.org/10.1016/j.rse.2019.05.024.

Recht, B., Roelofs, R., Schmidt, L., Shankar, V., 2018. Do CIFAR-10 Classifiers Generalize to CIFAR-10?, arXiv:1806.00451 [cs, stat].

Richter, R., Louis, B.J., Muller-Wilm, U., 2012. Sentinel-2 MSI–level 2A products algorithm theoretical basis document, Tech. rep., ESA. https://earth.esa.int/c/document_library/get_file?folderId=349490&name=DLFE-4518.pdf.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, LNCS. Springer, Cham, pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28.

Scaramuzza, P.L., Bouchard, M.A., Dwyer, J.L., 2012. Development of the Landsat data continuity mission cloud-cover assessment algorithms. IEEE Trans. Geosci. Remote Sens. 50 (4), 1140–1154. https://doi.org/10.1109/TGRS.2011.2164087.

Schuegraf, P., Bittner, K., 2019. Automatic building footprint extraction from multi-resolution remote sensing images using a hybrid FCN. ISPRS Int. J. Geo-Information 8 (4), 191. https://doi.org/10.3390/ijgi8040191.

Shao, Z., Pan, Y., Diao, C., Cai, J., 2019. Cloud detection in remote sensing images based on multiscale features-convolutional neural network. IEEE Trans. Geosci. Remote Sens. 1–15. https://doi.org/10.1109/TGRS.2018.2889677.

Stelzer, K., Paperin, M., Kirches, G., B.C., 2016. Proba-V Cloud Mask Validation, Tech. rep., QWG (April 2016). http://proba-v.vgt.vito.be/sites/proba-v.vgt.vito.be/files/documents/probav_cloudmask_validation_v1.0.pdf.

Stelzer, K., Paperin, M., Benhadj, I., Kirches, G., 2017. PROBA-V Cloud Round Robin Validation Report, Tech. rep., QWG. https://earth.esa.int/documents/700255/2362868/ProbaV_CloudContest_ValidationReport_1_3.pdf.

Sterckx, S., Benhadj, I., Duhoux, G., Livens, S., Dierckx, W., Goor, E., Adriaensen, S., Heyns, W., Van Hoof, K., Strackx, G., Nackaerts, K., Reusen, I., Van Achteren, T., Dries, J., Van Roey, T., Mellab, K., Duca, R., Zender, J., 2014. The PROBA-V mission: image processing and calibration. Int. J. Remote Sens. 35 (7), 2565–2588. https://doi.org/10.1080/01431161.2014.883094.

Sun, L., Liu, X., Yang, Y., Chen, T., Wang, Q., Zhou, X., 2018. A cloud shadow detection method combined with cloud height iteration and spectral analysis for Landsat 8 OLI data. ISPRS J. Photogramm. Remote Sens. 138, 193–207. https://doi.org/10.1016/j.isprsjprs.2018.02.016.

Svendsen, D.H., Martino, L., Campos-Taberner, M., García-Haro, F.J., Camps-Valls, G., 2018. Joint gaussian processes for biophysical parameter retrieval. IEEE Trans. Geosci. Remote Sens. 56 (3), 1718–1727. https://doi.org/10.1109/TGRS.2017.2767205.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R., 2014. Intriguing properties of neural networks. In: International Conference on Learning Representations (ICLR), pp. 1–10.

Torralba, A., Efros, A.A., 2011. Unbiased look at dataset bias. In: CVPR 2011, pp. 1521–1528. https://doi.org/10.1109/CVPR.2011.5995347.

Torres Arriaza, J.A., Guindos Rojas, F., Peralta López, M., Cantón, M., 2003. An automatic cloud-masking system using Backpro. Neural nets for AVHRR scenes. IEEE Trans. Geosci. Remote Sens. 41 (4), 826–831. https://doi.org/10.1109/TGRS.2003.809930.

Tuia, D., Volpi, M., Trolliet, M., Camps-Valls, G., 2014. Semisupervised manifold alignment of multimodal remote sensing images. IEEE Trans. Geosci. Remote Sens. 52 (12), 7708–7720. https://doi.org/10.1109/TGRS.2014.2317499.

U.S. Geological Survey, 2016a. L8 SPARCS Cloud Validation Masks, data release. doi: 10.5066/F7FB5146.

U.S. Geological Survey, 2016b. L8 Biome Cloud Validation Masks, data release. doi: 10.5066/F7251GDH.

U.S. Geological Survey, 2019. Landsat 8 Data Users Handbook, Tech. Rep. LSDS-1574, USGS, https://www.usgs.gov/media/files/landsat-8-data-users-handbook.

Wieland, M., Li, Y., Martinis, S., 2019. Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network. Remote Sens. Environ. 230, 111203. https://doi.org/10.1016/j.rse.2019.05.022.

Wolanin, A., Camps-Valls, G., Gómez-Chova, L., Mateo-García, G., van der Tol, C., Zhang, Y., Guanter, L., 2019. Estimating crop primary productivity with Sentinel-2 and Landsat 8 using machine learning methods trained with radiative transfer simulations. Remote Sens. Environ. 225, 441–457. https://doi.org/10.1016/j.rse.2019.03.002.

Wolters, E., Swinnen, E., Benhadj, I., Dierckx, W., 2015. PROBA-V cloud detection evaluation and proposed modification, Tech. Rep. Technical Note, 17/7/2015, QWG.

Wolters, E., Dierckx, W., Iordache, M.-D., Swinnen, E., 2018. PROBA-V products user manual, Tech. Rep. Technical Note, 16/03/2018, QWG. http://www.vito-eodata.be/PDF/image/PROBAV-Products_User_Manual.pdf.

Wurm, M., Stark, T., Zhu, X.X., Weigand, M., Taubenböck, H., 2019. Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 150, 59–69. https://doi.org/10.1016/j.isprsjprs.2019.02.006.

Xie, F., Shi, M., Shi, Z., Yin, J., Zhao, D., 2017. Multilevel cloud detection in remote sensing images based on deep learning. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 10 (8), 3631–3640. https://doi.org/10.1109/JSTARS.2017.2686488.

Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks?. In: Advances in Neural Information Processing Systems 27, Curran Associates Inc, pp. 3320–3328.

Zhai, H., Zhang, H., Zhang, L., Li, P., 2018. Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery. ISPRS J. Photogramm. Remote Sens. 144, 235–253. https://doi.org/10.1016/j.isprsjprs.2018.07.006.

Zhan, Y., Wang, J., Shi, J., Cheng, G., Yao, L., Sun, W., 2017. Distinguishing cloud and snow in satellite images via deep convolutional network. IEEE Geosci. Remote Sens. Lett. 14 (10), 1785–1789. https://doi.org/10.1109/LGRS.2017.2735801.

Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2017. Understanding deep learning requires rethinking generalization. In: International Conference on Learning Representations (ICLR), pp. 1–15.

Zhu, Z., Woodcock, C.E., 2012. Object-based cloud and cloud shadow detection in Landsat imagery. Remote Sens. Environ. 118 (Supplement C), 83–94. https://doi.org/10.1016/j.rse.2011.10.028.

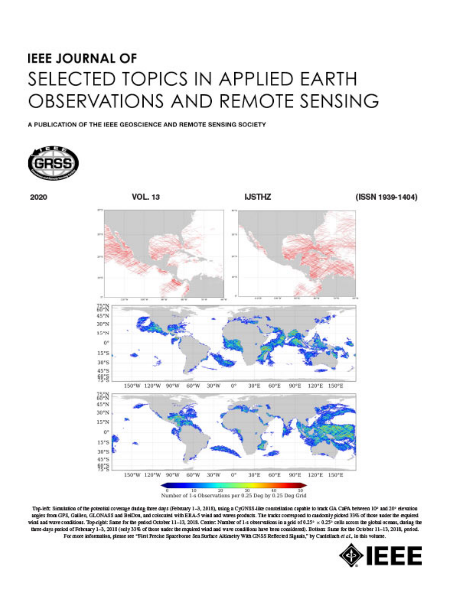
Zhu, Z., Wang, S., Woodcock, C.E., 2015. Improvement and expansion of the Fmask algorithm: cloud, cloud shadow, and snow detection for Landsats 4–7, 8, and Sentinel 2 images. Remote Sens. Environ. 159 (Supplement C), 269–277. https://doi.org/10.1016/j.rse.2014.12.014.

Publication III: Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V images for Cloud Detection

Publication III

G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images for Cloud Detection, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 747–761, Oct. 2020, doi: 10.1109/JSTARS.2020.3031741.

Q2: Geography Physical, Q2: Remote Sensing, Q2: Imaging Science & Photographic Technology, Q1: Engineering, Electrical and Electronic, IF = 3.784



IEEE JOURNAL OF
SELECTED TOPICS IN APPLIED EARTH
OBSERVATIONS AND REMOTE SENSING

A PUBLICATION OF THE IEEE GEOSCIENCE AND REMOTE SENSING SOCIETY

2020 VOL. 13 IJSTHZ (ISSN 1939-1404)

Top-left: Simulation of the potential coverage during three days (February 1–3, 2018), using a CyGNSS-like constellation capable to track GA CaPA between 10° and 20° elevation angles from GPS, Galileo, GLONASS and BeiDou, and colocated with ERA-5 wind and waves products. The tracks correspond to randomly picked 33% of those under the required wind and wave conditions. Top-right: Same for the period October 11–13, 2018. Center: Number of 1-s observations in a grid of 0.25° × 0.25° cells across the global oceans, during the three-days period of February 1–3, 2018 (only 33% of those under the required wind and wave conditions have been considered). Bottom: Same for the October 11–13, 2018, period. For more information, please see "First Precise Spaceborne Sea Surface Altimetry With GNSS Reflected Signals," by Cardellach *et al.*, in this volume.

IEEE

Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images for Cloud Detection

Gonzalo Mateo-García, Valero Laparra, Dan López-Puigdollers, and Luis Gómez-Chova, *Senior Member, IEEE*

***Abstract*—The number of Earth observation satellites carrying optical sensors with similar characteristics is constantly growing. Despite their similarities and the potential synergies among them, derived satellite products are often developed for each sensor independently. Differences in retrieved radiances lead to significant drops in accuracy, which hampers knowledge and information sharing across sensors. This is particularly harmful for machine learning algorithms, since gathering new ground-truth data to train models for each sensor is costly and requires experienced manpower. In this work, we propose a domain adaptation transformation to reduce the statistical differences between images of two satellite sensors in order to boost the performance of transfer learning models. The proposed methodology is based on the cycle consistent generative adversarial domain adaptation framework that trains the transformation model in an unpaired manner. In particular, Landsat-8 and Proba-V satellites, which present different but compatible spatio-spectral characteristics, are used to illustrate the method. The obtained transformation significantly reduces differences between the image datasets while preserving the spatial and spectral information of adapted images, which is, hence, useful for any general purpose cross-sensor application. In addition, the training of the proposed adversarial domain adaptation model can be modified to improve the performance in a specific remote sensing application, such as cloud detection, by including a dedicated term in the cost function. Results show that, when the proposed transformation is applied, cloud detection models trained in Landsat-8 data increase cloud detection accuracy in Proba-V.**

***Index Terms*—Generative adversarial networks, convolutional neural networks, domain adaptation, Landsat-8, Proba-V, cloud detection.**

I. Introduction

OVER the last decade, the number of both private and public satellite missions for Earth observation has explode. According to the UCS satellite database [1], there are currently around 500 orbiting satellites carrying passive optical sensors (multispectral or hyperspectral). Whereas each sensor is to some extent unique, in many cases, there are only small differences between them such as slightly different spectral responses, different ground sampling distances, or the different inherent noise of the instruments. Nevertheless, derived products from those images are currently tailored to each particular sensor, since models designed to one sensor often transfer poorly to a different one due to those differences [2]. In order to transfer products across sensors, we need to ensure that the underlying data distribution does not change from one sensor to the other. In machine learning, this is a long-standing problem that is called *data shift* [3]: differences between the training and testing dataset distributions yield significant drops in performance. In order to address this problem, the field of domain adaptation (DA) proposes to build a transformation between the different distribution domains such that, when images are transformed from one *source* domain to other *target* domain, the distribution shift is reduced.

In this work, we focus on the problem where the training (*source*) distribution corresponds to images and ground truth from one satellite, whereas the testing (*target*) distribution corresponds to images from another sensor. Notice that this is a very broad scenario that is found frequently in remote sensing (RS). Examples of products built in this manner include cloud masks [4], [5], but also land use and land cover classification [6], vegetation indexes retrieval [7], or crop yield estimation [8]. The goal of domain adaptation is thus to find a transformation that allows models working on a given satellite (source domain) to work accurately on another one (target domain).

As a representative case study, in this work, we focus on the Landsat-8 [9] and Proba-V [10] satellites. Transfer learning across these two sensors is particularly interesting since Landsat is a pioneering RS satellite program with a strong and well-established community, and hence, a good number of manually annotated datasets with cloud masks are publicly available, which could be very valuable to develop cloud detection models for Proba-V. Nevertheless, in order to build a model for Proba-V using Landsat-8 training data, differences in the imaging instruments on-board Landsat-8 and Proba-V must be taken in to account. On the one hand, the operational Land Imager instrument (OLI), on board of Landsat-8, measures radiance in nine bands in the visible and infrared part of the electromagnetic

Manuscript received June 10, 2020; revised August 12, 2020 and September 25, 2020; accepted October 8, 2020. Date of publication October 21, 2020; date of current version January 6, 2021. This work was supported in part by the Spanish Ministry of Science and Innovation under Project TEC2016-77741-R and Project PID2019-109026RB-I00 (ERDF) and in part by the European Space Agency (ESA IDEAS) under Research Grant CCN008. *(Corresponding author: Gonzalo Mateo-Garcia.)*

The authors are with the Image Processing Laboratory (IPL), University of Valencia, 46010 Valencia, Spain (e-mail: Gonzalo.Mateo-Garcia@uv.es; Valero.Laparra@uv.es; dan.lopez@uv.es; luis.gomez-chova@uv.es).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/JSTARS.2020.3031741

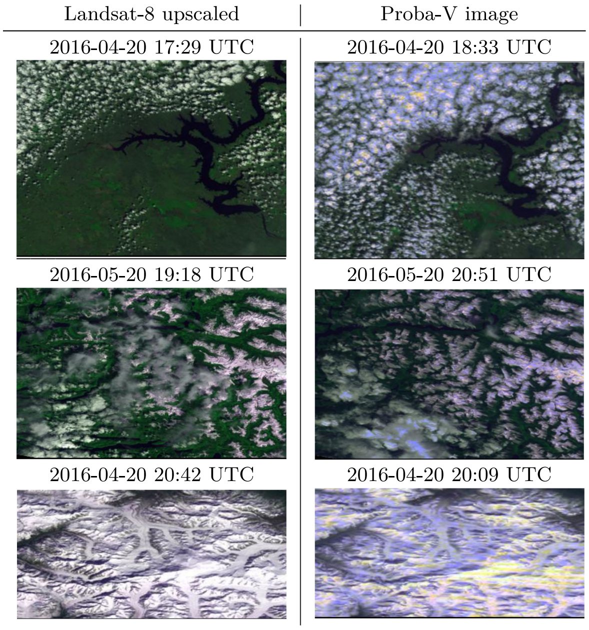


Fig. 1. Close-in-time acquisitions of Landsat-8 and Proba-V satellites. Landsat-8 image is transformed and upscaled to resemble the optical characteristics of the Proba-V sensor; however, differences in radiometry and texture between images still remain. First row: Missouri river in North America (April 20, 2016). Second and third rows: North West Pacific coast, North America (May 20, 2016 and April 20, 2016).

spectrum at 30-m resolution. On the other hand, the Proba-V instrument retrieves four bands in the blue, red, near-infrared, and short-wave infrared at 333-m resolution (full swath). Compared to Landsat-8, Proba-V has a much wider swath, which yields a more frequent revisiting time of around two days, whereas Landsat-8 revisit time ranges between 7 and 14 days. Fig. 1 shows three calibrated top of atmosphere (TOA) reflectance Landsat-8 and Proba-V images from the same location acquired with a difference of 30–90 min. We can see that, despite showing the same bands for Landsat-8 and Proba-V, and upscaling the Landsat-8 image to the Proba-V resolution (cf., Section III-A), differences between the images still remain [11]. In particular, Proba-V images are more blueish, due to saturation effects in the blue channel, and are more noisy [12]. This higher noise and lower spatial resolution can be appreciated, for example, in the bottom row of Fig. 1, which shows a mountainous area in British Columbia where details in the Landsat-8 image are sharper than in the Proba-V one. Hence, products built using data in the Landsat-8 domain, such as the cloud detection model proposed in [5], show a drop in detection accuracy when directly applied to Proba-V images.

In this work, we propose a DA methodology based on the state-of-the-art work in DA for computer vision of Hoffman *et al.*, CyCADA [13], to the remote sensing field. In particular, we propose to use cycle consistent generative adversarial networks (cycleGAN [14]) to find a DA transformation from the Proba-V domain to the Landsat-8 upscaled domain that removes noise and saturation of Proba-V images while preventing artifacts on the resulting images. One of the main advantages of the proposed methodology is that it is unpaired, i.e., it does not require a paired dataset of simultaneous and collocated Landsat-8 and Proba-V images to be trained. This is crucial for applications such as cloud detection since cloud's presence and location highly varies between acquisitions. This can also be viewed in Fig. 1, even though acquisitions are the closest possible in time for Landsat-8 and Proba-V, cloud location changes significantly between the acquisitions. This would make a paired approach unfeasible. Following the proposed methodology, Proba-V images are enhanced and are shown to be statistically more similar to the Landsat-8 upscaled ones. In addition, from a transfer learning perspective, it is important to remark that the cloud detection models applied to Proba-V are trained using only Landsat-8 images and their ground truth. In this context, a boost in the cloud detection accuracy is shown for Proba-V when the proposed adversarial domain adaptation transformation is applied.

The rest of this article is organized as follows: in Section II, we discuss related work in cloud detection and domain adaptation in RS; in Section III, we detailed the proposed methodology to upscale Landsat-8 images and to train the domain adaptation network; Section IV describes the Proba-V and Landsat-8 datasets where experiments are carried out; Section V contains the experimental results, and finally, Section VI concludes this article.

II. Related Work

There is a huge amount of work in both cloud detection and domain adaptation in the remote sensing literature. We discuss related work in both fields with a particular focus on approaches that deal with data coming from different sensors.

A. Transfer Learning for Cloud Detection

Cloud detection has been lately dominated by deep learning approaches where, given a sufficiently large corpus of manually annotated images with the corresponding ground-truth cloud masks, a network based on spatial convolutions is trained by back-propagation. Fully convolutional neural networks (FCNNs) [15], most of them based on the U-Net architecture [16], produce very accurate results and have the advantage that they can be applied to images of the arbitrary size with a fast inference time. Jeppesen *et al.* [17], Mohajerani and Sahedi [18], [19], Li *et al.* [20], and Yang *et al.* [21] tackle cloud detection in Landsat-8 using FCNNs trained in publicly available manually annotated datasets. They all show very high cloud detection accuracy outperforming the operational Landsat-8 cloud detection algorithm, FMask [22]. Hence, our work seeks to transfer those accurate cloud detection models to other satellite data with a minimal drop in performance. There are some very recent works that propose to transfer an FCNN cloud detection model between different sensors. For instance, in [23], an FCNN is trained with contrast and brightness data augmentation in the Landsat-8 *SPARCS* dataset [24] and it is tested on Sentinel-2, Landsat-7, and Landsat-8 images. Results show similar performance of the model on the three sensors; which suggest that the Sentinel-2 and the Landsat sensors are very similar, and thus, the data shift problem is not so relevant. On the other hand, in [25], an FCNN is trained on a manually annotated collection of World-View-2 images over the Fiji islands and tested in both World-View-2 and Sentinel-2 imagery. In this case, a significant drop in performance is observed in the Sentinel-2 domain; in order to correct this gap, the authors propose a simple domain adversarial method that obtains good results but it is still far from the accuracy obtained in Word-View-2 data. Shendryk *et al.* [26] also propose transfer learning, in this case, using PlanetScope and Sentinel-2 imagery. The proposed network classifies patches as cloudy or clear instead of providing a full segmentation mask at the pixel level. Nevertheless, a small gap in performance is observed between results in the source domain (PlanetScope) and the target domain (Sentinel-2). Finally, in our previous work [5], we showed that transfer learning from Proba-V to Landsat-8 and from Landsat-8 to Proba-V produce accurate results on a par with the FMask [22] model on Landsat-8 and surpassing the operational cloud detection model [27] for Proba-V, respectively. However, a significant gap between transfer learning approaches and *state-of-the-art* models trained with data from the same domain still exists, which is the focus of this work.

B. Domain Adaptation

The remote sensing community has traditionally addressed DA taking advantage of a deep understanding of the imaging sensors and exploiting their physical and optical characteristics to provide well-calibrated products [28]. Despite the efforts to provide a good calibration, small differences are found in retrieved radiance values due to different spectral response functions (SRFs), saturation effects, mixed pixels, etc. [11], [29], [30]. To this end, several works propose sensor intercalibrations or harmonizations to correct biases between the bands of different sensors [31]–[35]. These approaches train models in *paired* data using either real spectra of both satellites (retrieved in same location and close in time) or simulated radiances. As mentioned earlier, *paired* approaches cannot be applied to cloud detection due to the extreme variability of cloud location between acquisitions. Among *unpaired* approaches, histogram matching [36] aligns the observed radiance distribution of both satellites using the cumulative density function of the data. Histogram matching is fast and reliable, and thus, we use it as a baseline to compare our method. More complex *unpaired* methods include multivariate matching [37], graph matching [38], or manifold alignment [39].

All the methods discussed so far only focus on the spectral information of images disregarding the spatial dimension. In order to account for spatial changes, recent works propose DA using convolutional neural networks (CNNs). Most of these works use generative adversarial networks (GANs) [40] to align source and target distributions. Those works could be divided in *feature-level* DA and *pixel-level* DA. In *feature-level* or *discriminative* DA [41], [42], the model to be transferred is jointly trained with its normal loss and to make its internal representations (i.e., activations at some layers) invariant to the input distribution. Hence, *feature level* DA requires retraining the model with that extra penalty. An example of *feature level DA* is the work of Segal *et al.* [25] previously discussed in Section II-A. In *pixel-level* DA [43], [44] (also called *image-to-image* DA), extra networks are trained to transform images between domains. Hence, *pixel-level* DA is independent of the transferred model, and thus, it could be applied to other problems with same inputs. Our work falls into the *pixel-level* DA framework: we assume that the cloud detection model trained in the source domain is fixed, and thus, we focus on finding a DA transformation from the target to the source domain (see Section III).

Regarding the definition and the types of domains in remote sensing, works such as [45]–[48] consider data from a single sensor, where the source and target domains are represented by images from different locations or different time acquisitions. The DA works where domains are represented by different sensors are scarce; for instance, the work of Benjdira *et al.* [49] tackles urban segmentation in aerial imagery of two cities acquired with two different cameras. They obtain good results despite differences in spectral bands and spatial resolution of the instruments are not taken into account. Even though there are some works using GANs to transfer learning between different sensors, most of them involve training the classifiers using some labeled samples from the target domain. This is the case of works that tackle the DA between SAR and optical images [50]–[52]. It is important to remark that we are dealing with *unsupervised domain adaptation* [2], [41] (also known as *transductive transfer learning* [53]), which assumes there is no labeled data available in the target domain.

III. Methodology

We assume two independent datasets from two different sensors are given, but we only have labels for one dataset. The main idea is to be able to use the data from the labeled dataset in order to design algorithms to solve problems in the unlabeled dataset.

In our particular case, we have images for Landsat-8 (L8) with the corresponding ground-truth cloud masks (binary labels identifying *cloudy* or *clear* pixels), $\{X_{\mathrm{L8}}, y_{\mathrm{L8}}\}$; and we only have Proba-V (PV) images without ground truth, X_{PV}. Therefore, we want to perform cloud detection in X_{PV} using algorithms trained with $\{X_{\mathrm{L8}}, y_{\mathrm{L8}}\}$. Since we know the technical specifications of Landsat-8 and Proba-V, we can design an upscaling algorithm to convert images captured from Landsat-8 to resemble Proba-V spectral and spatial characteristics. This upscaling could work quite well, and actually classical remote sensing approaches follow this methodology to combine or perform transfer learning across different existing satellites. However, this upscaling transformation is not perfect since it is based on the prelaunch characterization of the instruments and is always susceptible to be affected by diverse uncertainty sources. Therefore, an extra adaptation step could be used in order to transform the Proba-V images before applying the transfer learning algorithms. In this work, we are going to explore how to design this extra step by using GANs.

In Fig. 2, we show the proposed adaptation scheme. The upscaling transformation, U, converts the Landsat-8 labeled data to a domain where it has similar spatio-spectral properties than Proba-V (i.e., the same number of bands and same spatial resolution). We can use the Landsat-8 upscaled (LU) data in

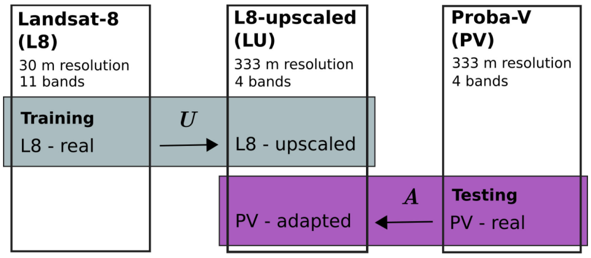


Fig. 2. Transfer learning and adaptation scheme: Landsat-8 and Proba-V datasets and how they are transformed between the three different domains. The transformations look for adaptation between the domains: U is the upscaling transformation applied to Landsat-8 to resemble the Proba-V instrument characteristics (see Section III-A); and A adapts from the Proba-V domain to the Landsat-8 upscaled domain (see Section III-C).

order to train an algorithm, in our case, we are going to design a cloud detection algorithm using FCNNs. While this model could be applied directly on Proba-V images, we will show that an extra adaptation step, A, applied to the Proba-V images to resemble even more the upscaled domain, could improve the similarity between the training and the testing data, and therefore, improve the performance of the cloud detection algorithm. Note that, if no adaptation is used, A would be the identity function. In the following subsections, we detail both the transformation U, based on the instruments characteristics, and the transformation A, which is based on CycleGANs.

A. Upscaling Transformation From Landsat-8 to Proba-V

In this section, we describe the upscaling transformation (U transformation in Fig. 2). In a standard transfer learning approach from one sensor to another, the first step is to transform Landsat-8 images to resemble Proba-V images in terms of spectral and spatial properties. Fig. 3 shows the proposed upscaling from Landsat-8 to Proba-V using the instrumental characteristics of both sensors, which are based on the prelaunch characterization and on-ground calibration of the instruments. First, we select the spectral bands from Landsat-8 that overlap with Proba-V in terms of the SRFs of both instruments. Fig. 4 shows the SRF of overlapping bands in Landsat-8 (dashed) and Proba-V (solid). SWIR and RED bands present the best agreement. However, the NIR SRF of Proba-V is wider and its peak is not aligned with Landsat-8 B5 band, which might led to differences in the retrieved radiance. Finally, the BLUE band of Proba-V overlaps with two different Landsat-8 bands. Therefore, the contribution of Landsat-8 B1 and B2 bands is weighted according to the overlapping area of the SRFs: 25% and 75% for B1 and B2, respectively.

Then, the selected bands of the Landsat-8 image are scaled to match the spatial properties of Proba-V. The 30-m resolution Landsat-8 bands are upscaled to the 333-m resolution of Proba-V. This upscaling takes into account the optical characteristics of the Proba-V sensor and the resampling of the 333-m product described in [10]. First, the point spread function (PSF) of each Proba-V spectral band is used to convert the Landsat-8 observations to the nominal Proba-V spatial resolution at nadir. The ground sampling distance (GSD) for the Proba-V center camera is about 96.9 m for the BLUE, RED, and NIR channels, while the SWIR center camera resolution is 184.7 m [10]. The SWIR PSF is about twice as wide as the PSF of the other bands, which stresses the fact that a distinct spatial adaptation might be applied to each band. The PSFs of the bands are modeled as 2-D Gaussian filters, which are applied to the 30-m resolution Landsat-8 bands. The filtered image is upscaled to the nominal 90-m resolution at nadir by taking 1 out of every 3 pixels. Finally, Lanczos interpolation is applied to upscale the image to the final 333-m Proba-V resolution. Notice that Lanczos is the interpolation method used at the Proba-V ground segment processing to upscale the acquired raw Proba-V data to the 333-m Plate Care grid [10]. Ground-truth labels, y_L at 30 m, must were also scaled to get a Landsat-8 upscaled dataset at 333 m: $\{X_{\rm LU}, y_{\rm LU}\}$.

B. Transfer Cloud Detection Model

The cloud detection model trained on the Landsat-8 upscaled dataset is an FCNN classifier based on the simplified U-Net architecture described in [5]. This model is trained in the Landsat-8 Upscaled dataset ($\{X_{\rm LU}, y_{\rm LU}\}$) to minimize the binary cross entropy between the model output and the labels $y_{\rm LU}$. Hence, the model input is a four-band 333-m resolution image and its output a cloud probability mask (additional details about this network can be found in Appendix). Therefore, it could be applied directly to Proba-V images. Nevertheless, as explained before, statistical differences between Landsat-8 upscaled and Proba-V images make that the performance of this model is not as good as expected. This effect is related to the different sensor spectral response functions, saturation effects, radiometric calibration, modulation transfer functions, or mixed pixels. For instance, as shown in Fig. 1, Proba-V contains many saturated pixels, especially in the blue channel, which is a known issue. This suggests that an extra domain adaptation step could be added to improve the transfer learning results reported in [5].

C. Generative Adversarial Domain Adaptation

In this section, we describe the training process for the extra adaptation transformation (A in Fig. 2) that we propose to improve the performance of the transferred models. This training process is based on the GAN [40] framework.

The main idea of GANs is to train two networks, a generative one and a discriminative one, with opposite objectives, simultaneously. This adversarial training fits a data generator that minimizes the Jensen–Shannon divergence between the real and the generated data distribution. An extension of the original GANs formulation, the conditional GANs [54], was proposed to train a model that generates samples from a conditional distribution. One application of conditional GANs is the generative adversarial domain adaptation proposed in [13], [42], and[44]. In those works, the conditional GANs formulation was modified to solve domain adaptation problems.

Probably, the most complete approximation for domain adaptation based on GANs is the one proposed in CyCADA [13]. Unlike the classical GANs, where adaptation is performed in one

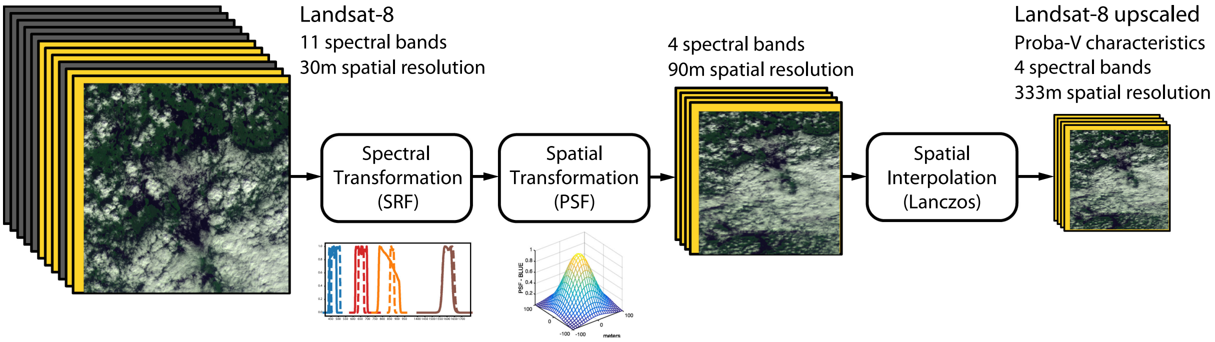


Fig. 3. Upscaling transformation (U in Fig. 2) applied to Landsat-8 in order to resemble the Proba-V instrument characteristics.

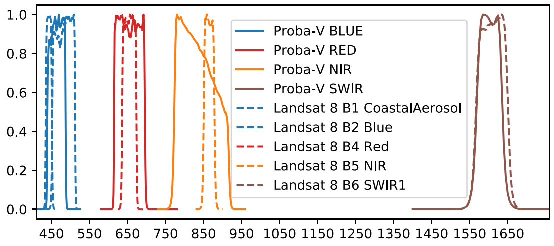


Fig. 4. Spectral response of Landsat-8 and Proba-V channels.

direction only, this approach proposes a double simultaneous adaptation between the two domains. This allows to include several consistency terms in order to impose restrictions on the two adaptation directions. Our approach has a similar structure with CyCADA (see Fig. 5). It has two generators and two discriminators: $G_{\mathrm{LU}\to\mathrm{PV}}$, $G_{\mathrm{PV}\to\mathrm{LU}}$, D_{PV}, D_{LU}. We are interested in using $G_{\mathrm{PV}\to\mathrm{LU}}$ to adapt the Proba-V images to better match the upscaled ones that have been used to train the cloud detection algorithm, i.e., $A \equiv G_{\mathrm{PV}\to\mathrm{LU}}$.

On the one hand, the discriminators are trained to minimize the binary cross entropy loss between the real and the generated images

$$\begin{aligned}\mathcal{L}_D(D_{\mathrm{LU}}) &= \sum_i -\log(D(X^i_{\mathrm{LU}}))+\\&\quad -\log(1-D(G_{\mathrm{PV}\to\mathrm{LU}}(X^i_{\mathrm{PV}})))\\\mathcal{L}_D(D_{\mathrm{PV}}) &= \sum_i -\log(D(X^i_{\mathrm{PV}}))+\\&\quad -\log(1-D(G_{\mathrm{LU}\to\mathrm{PV}}(X^i_{\mathrm{LU}}))).\end{aligned}$$

On the other hand, the generators are trained to fool the discriminators by minimizing the adversarial loss

$$\begin{aligned}\mathcal{L}_{\mathrm{GAN}}(G_{\mathrm{PV}\to\mathrm{LU}}) &= \sum_i -\log(D(G_{\mathrm{PV}\to\mathrm{LU}}(X^i_{\mathrm{PV}})))\\\mathcal{L}_{\mathrm{GAN}}(G_{\mathrm{LU}\to\mathrm{PV}}) &= \sum_i -\log(D(G_{\mathrm{LU}\to\mathrm{PV}}(X^i_{\mathrm{LU}}))).\end{aligned}$$

In this work, in order to ensure consistency between the real and the generated images, three extra penalties are added to the standard GAN generator loss: the *identity consistency loss*, the *cycle loss*, and the *segmentation consistency loss*. First, we take into account that data from both sensors are radiometrically calibrated and we do not want to significantly modify the original TOA values of the adapted images. Therefore, the *identity consistency loss*, introduced in our previous work [4], is added to make the input TOA reflectance values similar to those in the output

$$\begin{aligned}\mathcal{L}_{id}(G_{\mathrm{PV}\to\mathrm{LU}}) &= \sum_i \|X^i_{\mathrm{PV}} - G_{\mathrm{PV}\to\mathrm{LU}}(X^i_{\mathrm{PV}})\|_1\\\mathcal{L}_{id}(G_{\mathrm{LU}\to\mathrm{PV}}) &= \sum_i \|X^i_{\mathrm{LU}} - G_{\mathrm{LU}\to\mathrm{PV}}(X^i_{\mathrm{LU}})\|_1.\end{aligned}$$

Second, *the cycle consistency loss*, proposed in [54], is added to both generators to force them to act approximately as inverse functions one of each other

$$\begin{aligned}&\mathcal{L}_{\mathrm{cyc}}(G_{\mathrm{PV}\to\mathrm{LU}}, G_{\mathrm{LU}\to\mathrm{PV}}) =\\&\qquad \sum_i \|X^i_{\mathrm{PV}} - G_{\mathrm{LU}\to\mathrm{PV}}(G_{\mathrm{PV}\to\mathrm{LU}}(X^i_{\mathrm{PV}}))\|_1\\&\qquad + \sum_i \|X^i_{\mathrm{LU}} - G_{\mathrm{PV}\to\mathrm{LU}}(G_{\mathrm{LU}\to\mathrm{PV}}(X^i_{\mathrm{LU}}))\|_1.\end{aligned}$$

Finally, we additionally include a *segmentation consistency loss* that takes advantage of the cloud detection model trained in the LU domain. We apply this model to both LU and PV images even though the cloud detection classifier (f_{LU}) is trained using only LU images, the idea is that the LU classifier can act as a rough supervisor in the Proba-V domain.[1] This approach is also taken in CyCADA [13]. The selected semantic segmentation loss is the Kullback–Leibler divergence between the cloud probabilities of

[1] If ground-truth labels are available for some images of the source domain, the term $\mathcal{L}_{\mathrm{seg}}(G_{\mathrm{LU}\to\mathrm{PV}})$ could be changed to be the cross-entropy loss with the ground truth. In our experiments, we have not considered this option.

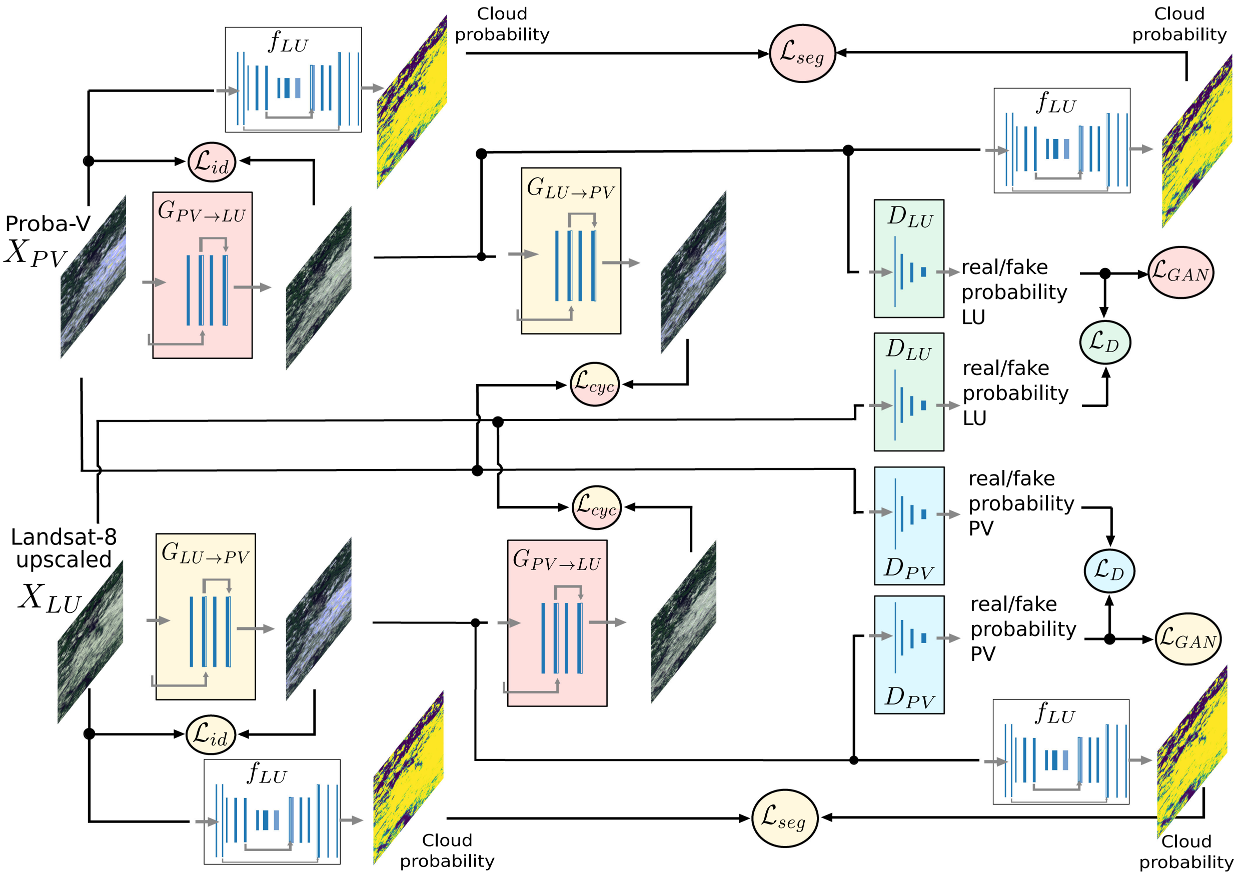


Fig. 5. Scheme of the forward passes for the training procedure of the proposed cycle consistent adversarial domain adaptation method. The four networks ($G_{\mathrm{PV}\to\mathrm{LU}}$, $G_{\mathrm{LU}\to\mathrm{PV}}$, D_{PV}, D_{LU}) have a different color. Losses are depicted with circles and their fill color corresponds to the color of the network that they penalize.

the model f_{LU} applied to real and generated images

$$\mathcal{L}_{\mathrm{seg}}(G_{\mathrm{PV}\to\mathrm{LU}}) = \sum_i \mathrm{KL}(f_{\mathrm{LU}}(X^i_{\mathrm{PV}})|f_{\mathrm{LU}}(G_{\mathrm{PV}\to\mathrm{LU}}(X^i_{\mathrm{PV}})))$$

$$\mathcal{L}_{\mathrm{seg}}(G_{\mathrm{LU}\to\mathrm{PV}}) = \sum_i \mathrm{KL}(f_{\mathrm{LU}}(X^i_{\mathrm{LU}})|f_{\mathrm{LU}}(G_{\mathrm{LU}\to\mathrm{PV}}(X^i_{\mathrm{LU}}))).$$

Therefore, the final loss of the generators is the weighted sum of the five described losses

$$\begin{aligned}\mathcal{L}(G_{\mathrm{PV}\to\mathrm{LU}}) &= \lambda_{GAN}\mathcal{L}_{GAN}(G_{\mathrm{PV}\to\mathrm{LU}}) + \lambda_{id}\mathcal{L}_{id}(G_{\mathrm{PV}\to\mathrm{LU}})\\ &\quad + \lambda_{\mathrm{cyc}}\mathcal{L}_{\mathrm{cyc}}(G_{\mathrm{PV}\to\mathrm{LU}}, G_{\mathrm{LU}\to\mathrm{PV}})\\ &\quad + \lambda_{\mathrm{seg}}\mathcal{L}_{\mathrm{seg}}(G_{\mathrm{PV}\to\mathrm{LU}})\\ \mathcal{L}(G_{\mathrm{LU}\to\mathrm{PV}}) &= \lambda_{\mathrm{GAN}}\mathcal{L}_{\mathrm{GAN}}(G_{\mathrm{LU}\to\mathrm{PV}}) + \lambda_{\mathrm{id}}\mathcal{L}_{\mathrm{id}}(G_{\mathrm{LU}\to\mathrm{PV}})\\ &\quad + \lambda_{\mathrm{cyc}}\mathcal{L}_{\mathrm{cyc}}(G_{\mathrm{PV}\to\mathrm{LU}}, G_{\mathrm{LU}\to\mathrm{PV}})\\ &\quad + \lambda_{\mathrm{seg}}\mathcal{L}_{\mathrm{seg}}(G_{\mathrm{LU}\to\mathrm{PV}})\end{aligned}$$

The weight parameters are set to $\lambda_{\mathrm{cyc}} = \lambda_{\mathrm{id}} = 5$ and $\lambda_{\mathrm{seg}} = \lambda_{\mathrm{GAN}} = 1$, so that losses are of the same magnitude. In addition, the two discriminators are regularized using a 0-centered gradient penalty with a weight of 10 [55]. In Section V, we conduct several experiments by setting some of these weights to zero in order to quantify the importance of each of these terms. In addition, notice that by setting some of these hyperparameters to zero, we obtain different adversarial domain adaptations proposals in the literature. In particular, if we set $\lambda_{\mathrm{id}} = 0$, we get the original CyCADA of Hoffman *et al.* [13]. When we set $\lambda_{\mathrm{cyc}} = 0$, we obtain one-direction GANs. By setting $\lambda_{\mathrm{cyc}} = \lambda_{\mathrm{seg}} = 0$, we get the approach of our previous work [4].

Details about the training procedure and particular network architectures of the generators G (FCNNs) and discriminators D (convolutional neural networks) of the proposed adaptation model can be found in Appendix.

Additionally, the implemented code is available.[2] From both a methodological and an operational perspective, the proposed approach has an important benefit: it does not require simultaneous and collocated pairs of Landsat-8, X^i_{LU}, and Proba-V, X^i_{PV}, images. Having coincident pairs from sensors on different platforms would be impossible in our case. Note that clouds' presence and location within an image highly vary even for small

[2] [Online]. Available: https://github.com/IPL-UV/pvl8dagans

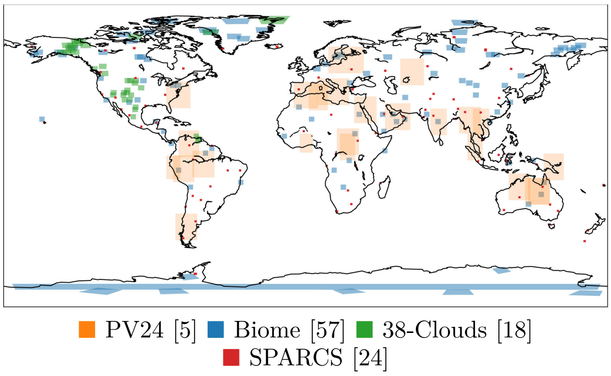


Fig. 6. Location of Landsat-8 and Proba-V products with manually annotated ground-truth cloud mask. For the *Biome* [57] and *38-Clouds* dataset [18], we additionally downloaded Proba-V images from the same location and acquisition time if available.

time differences. This problem prevents the use of other approaches such as canonical correlation analysis [56] or directly learning a generic transformation from X^i_{PV} to X^i_{LU} [31].

IV. Manually Annotated Datasets

Transfer learning from Landsat-8 to Proba-V is supported by the fact that there are several open access datasets with manually labeled clouds for the Landsat-8 mission. In this work, we use three of them that have a large coverage of acquisitions across different dates, latitudes, and landscapes. The *Biome* dataset [57], released for the Landsat-8 validation study of Foga *et al.* [58], is the largest among them. It consists of 96 full acquisitions covering the different biomes on Earth. The *SPARCS* dataset, collected in the study of Hughes and Hayes [59], contains 80 1000 × 1000 patches from different Landsat-8 acquisitions. Finally, the *38-Clouds* dataset of Mohajerani and Saeedi [18] has 38 full scenes mostly located in North America. Images and ground-truth cloud masks from these datasets have been upscaled, to match Proba-V spectral and spatial properties, following the procedure described in Section III-A. Hence, when we refer to those datasets, we assume four-band images and ground truth at 333 m. For testing the proposed cloud detection approach in Proba-V, since there are not publicly available datasets, we use the *PV24* dataset created by the authors in [60] and extensively curated in [5]. The *PV24* dataset contains 24 full Proba-V images at 333-m resolution and their corresponding manually annotated cloud masks. Fig. 6 shows the locations of the products of the aforementioned datasets. It is important to remark that we only use data from the *Biome* dataset for training the cloud detection models based on the FCNN (see Section III-B); the other three datasets (*SPARCS*, *38-Cloud*, and *PV24*) are only used for testing the models.

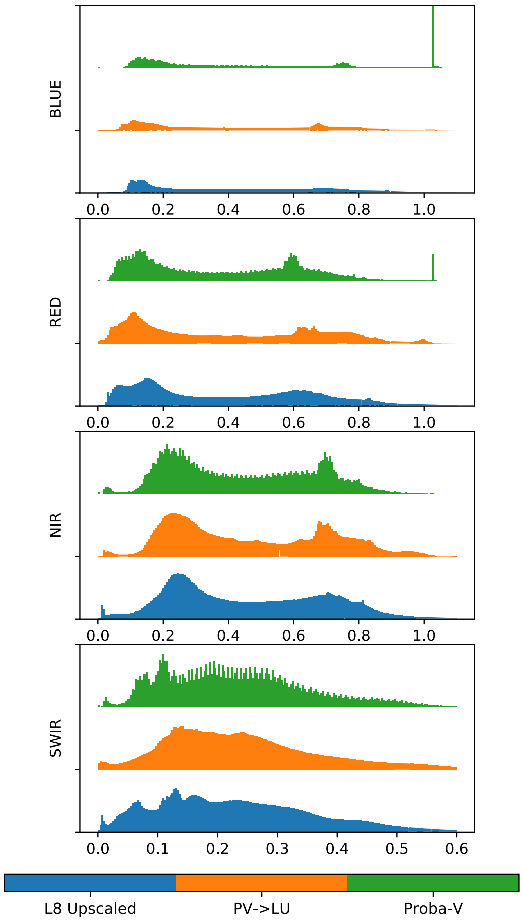


Fig. 7. TOA reflectance distribution on each of the spectral bands for Proba-V (green) images, Proba-V images transformed using the proposed DA method (orange), and pseudosimultaneous Landsat-8 Upscaled images (blue). Values measured across all image pairs in the *38-Clouds pseudosimultaneous dataset*.

On the other hand, to train the proposed domain adaptation model based on CycleGANs (see Section III-C), a set of 181 Proba-V products from the same locations and season as the *Biome* dataset has been selected. Using those Proba-V images and the Landsat-8 upscaled images from the *Biome* dataset, we created the *Biome Proba-V pseudosimultaneous dataset*, which contains 37 310 pairs of patches of 64 × 64 pixels, used to train the proposed DA method. Notice that, in this dataset, the pairs of images, one coming from Proba-V and the other from Landsat-8, are images from the same location and close-in-time acquisitions when available.[3] The same approach is followed to create the *38-Clouds Proba-V pseudosimultaneous dataset*, which is only used for testing the domain adaptation results. Images of this dataset, together with the results of the proposed domain adaptation and cloud detection models, are available.[4]

Finally, it is important to point out that images in this work are operational level-1TP products for Landsat-8 and level-2 A

[3] Some images from the *Biome* dataset are previous to the beginning of the Proba-V mission catalog; in this cases, we use images from same day of year in the next year.

[4] [Online]. Available: https://isp.uv.es/projects/cloudsat/pvl8dagans

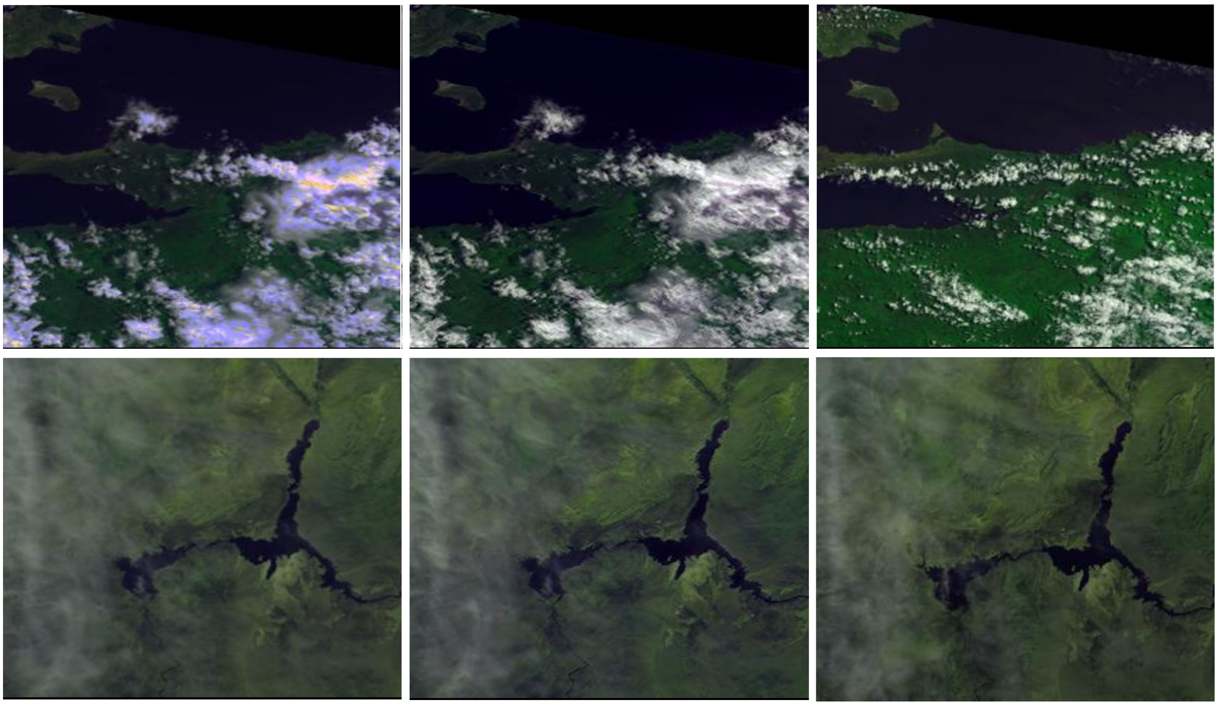


Fig. 8. Left: Proba-V images. Center: Proba-V adapted with the $G_{\mathrm{PV}\to\mathrm{LU}}$ transformation (A in Fig. 2). Right: Landsat-8 upscaled image (LU in Fig. 2). Top: 1 day difference acquisitions from Nueva Esparta island in the Caribbean sea. Bottom: Four minutes of difference acquisitions from lake made in North America.

products for Proba-V. These products have been preprocessed to TOA reflectance for both Landsat-8 [9] and Proba-V [61].

V. Experimental Results

This section is divided in two parts. In the first one, we analyze the radiometric and spatial properties of resulting images. The purpose is to assess the quality of the proposed transformation. We show that the proposed DA transformation produces reliable images (i.e., images without artifacts) that are statistically similar to Landsat-8 upscaled images. Additionally, we show that pixels flagged as good radiometric quality in Proba-V are much less changed by the proposed DA transformation. In the second part, we analyze the impact of DA on the cloud detection performance. Cloud detection results in the source domain (Landsat-8 upscaled) are compared with results in the target domain (Proba-V), with and without the DA transformation. In addition, we conduct an ablation study to show the relative importance of each of the proposed loss terms included to fit the generator network.

A. Domain Adaptation of Input Images

As explained in Section IV, we trained the DA method described in Section III-C using Proba-V and Landsat-8 upscaled patches from the *Biome Proba-V pseudosimultaneous dataset*. The generator and discriminator networks are trained simultaneously with minibatch stochastic gradient descent (see details in Section A). Afterwards, the trained Proba-V to Landsat-8 upscaled generator $G_{\mathrm{PV}\to\mathrm{LU}}$ (A in Fig. 2) is evaluated in the *38-Clouds Proba-V pseudosimultaneous dataset*, (i.e., the $G_{\mathrm{PV}\to\mathrm{LU}}$ network is applied to all Proba-V images in the dataset). Fig. 7 shows the distribution of TOA reflectance values for each of the bands without the domain adaptation step (green) and after applying $G_{\mathrm{PV}\to\mathrm{LU}}$ (orange) for all the Proba-V images in the dataset. One of the interesting results from these distributions is that the characteristic saturation in the Blue and Red bands of Proba-V disappears in the adapted images. In addition, the shape of the distribution of the adapted data is more similar to the shape of the pseudosimultaneous Landsat-8 upscaled images (blue).

Visual examples of the trained DA network are shown in Fig. 8. We show in the first column the Proba-V image, in the second one, the adapted Proba-V image using our DA method, and in the third column, the pseudosimultaneous Landsat-8 upscaled image (LU). In the first row, we can see that the location of clouds in the Proba-V image are preserved after the transformation while saturated blue values are removed. This provides a cloud appearance (and radiance) more similar to the pseudosimultaneous LU images (third column). In the second row, we can see slightly sharper edges in the DA transformed image compared to the original Proba-V image. This is because the Landsat-8 upscaled images have components of higher spatial frequency than Proba-V. This was also point out in the pair of images at the bottom in Fig. 1. In order to test this hypothesis, 64 × 64 pixels patches were extracted from the *38-Clouds Proba-V pseudosimultaneous dataset*. For each

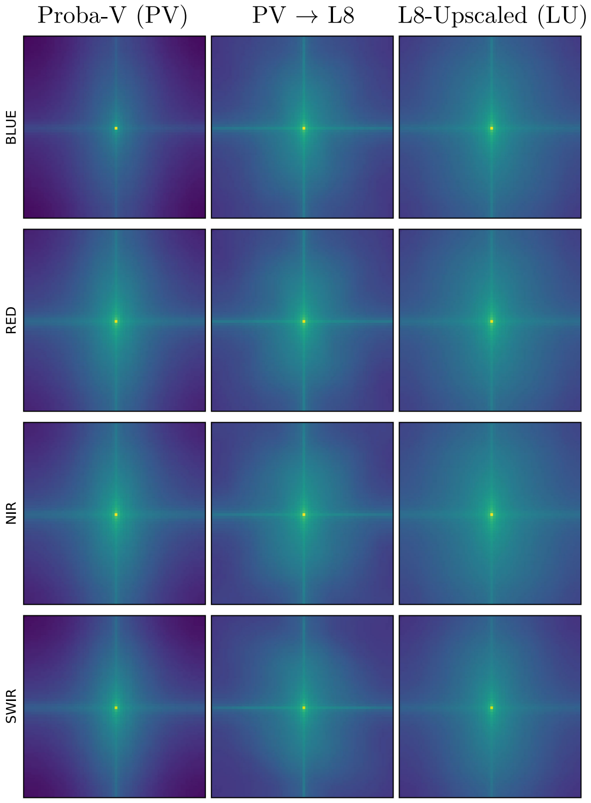


Fig. 9. 2-D Fourier transform in decibel for each of the four spectral channels averaged across all 64 × 64 images patches in the *38-Clouds Proba-V pseudosimultaneous dataset*. Left: Proba-V images. Center: Proba-V images adapted using the proposed domain adaptation method. Right: pseudosimultaneous Landsat-8 upscaled images.

patch, we computed the 2-D fast Fourier transform for each of the four spectral bands. Finally, the amplitude of the signal at each frequency is converted to decibels (dB) and averaged across all patches (see Fig. 9). As pointed out before, Proba-V images have less high frequency components, whereas the average frequency amplitudes for the adapted images are more similar to the Landsat-8 upscaled ones. This highlights the spatio-spectral nature of the proposed method: it does not only learn spectral changes between bands (colors) but also spatial relations.

Finally, Fig. 10 shows the difference in TOA reflectance between the original Proba-V images and the adapted ones ($X_{\rm PV} - G_{\rm PV\to LU}(X_{\rm PV})$) for all the pixels in the *38-Clouds Proba-V pseudosimultaneous dataset* and for each of the four Proba-V bands. In this case, we have stratified the pixels using the per pixel radiometric quality flag available in the status map (SM) of Proba-V products (see Proba-V User Manual [61, p. 6]). This quality indicator is a binary mask for each of the four Proba-V channels; pixels are flagged as *bad quality* for different reasons, including detector saturation [12]. In the Proba-V images of the *38-Clouds Proba-V pseudosimultaneous dataset*, approximately 30% of pixels in the blue band have a reported bad quality, in contrast to the 5% for the red band and 0.5% for the NIR and SWIR. One can see that differences in TOA reflectance for bad quality pixels is higher, whereas good quality pixels change much less.

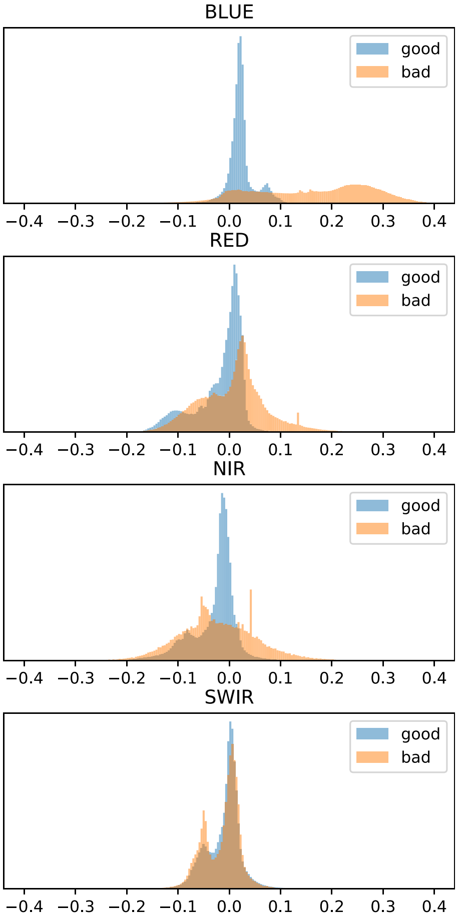


Fig. 10. Differences in TOA for Proba-V images before and after applying the proposed DA transformation ($X_{\rm PV} - G_{\rm PV\to LU}(X_{\rm PV})$).

TABLE I
ACCURACY FOR TEST IMAGES IN THE SOURCE LANDSAT-8 UPSCALED DOMAIN

Dataset	min	max	mean	std
SPARCS	91.67	92.46	92.12	0.20
38-Clouds	90.32	91.92	91.31	0.47

Results averaged over ten U-Net networks trained with different random initializations.

B. Domain Adaptation for Cloud Detection

In order to evaluate the DA methodology for the cloud detection application, we trained an FCNN in the *Biome* dataset. In order to account for the uncertainty at the weights initialization and ordering of the batches, we trained ten copies of the network using different random seed initializations. This procedure is also followed in [5]. Table I shows the cloud detection accuracy on the source domain by using the *SPARCS* and *38-Clouds*

TABLE II
ACCURACY OF DIFFERENT DA APPROACHES FOR CLOUD DETECTION OVER THE *PV24* DATASET

	min	max	mean	std
PV-trained [5]	94.83	95.15	94.98	0.09
full DA	**91.87**	**93.10**	**92.42**	0.37
$\lambda_{id} = 0$	**91.87**	93.02	92.34	0.36
$\lambda_{seg} = 0$	91.15	91.86	91.47	**0.24**
$\lambda_{cyc} = 0$, $\lambda_{seg} = 0$	90.35	91.26	90.79	0.32
$\lambda_{seg} = 0$, $\lambda_{id} = 0$	33.99	36.10	35.20	0.63
Histogram Matching [36]	89.09	91.31	90.18	0.72
no DA	89.05	91.88	90.41	0.82
Proba-V operational v101 [27]	-	-	82.97	-

Results averaged over ten FCNN networks trained with different random initialization.

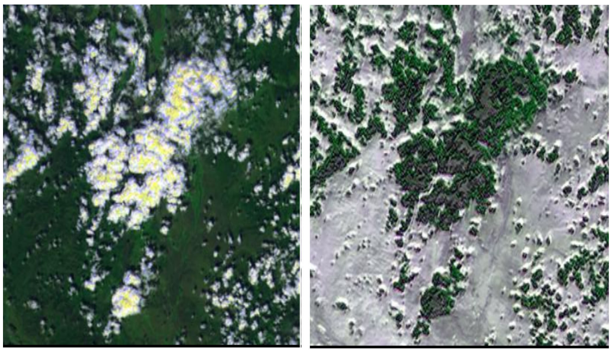


Fig. 11. Example of color inversion when neither the segmentation loss nor the identity loss are included. Left: Proba-V image. Right: Adapted image with the generator $G_{\text{PV}\to\text{LU}}$ trained with $\lambda_{\text{seg}} = 0$, $\lambda_{\text{id}} = 0$.

datasets. We see that overall the accuracy is relatively high and networks are not much affected by the weight's initialization.

Table II shows the results in the target domain (Proba-V) using the *PV24* dataset with the trained DA transformation $G_{\text{PV}\to\text{LU}}$ (called *full DA* in the table) and without it (called *no DA*). We also included the results of the ablation study, where we have set some of the weights of the generator losses to zero and the results using histogram matching [36] for domain adaptation as in [47]. In addition, results are compared with the FCNN trained in original Proba-V images and ground truths (PV-trained), which serves as an upper bound reference, and with the operational Proba-V cloud detection algorithm (v101) [27]. First of all, we see that the proposed DA method increases the mean overall accuracy and reduce the standard deviation of the metrics compared with direct transfer learning (no DA) or with adjusting the reflectance of each band with histogram matching. Second, there is a significant reduction in accuracy when the cycle loss is not included ($\lambda_{\text{cyc}} = \lambda_{\text{seg}} = 0$); notice that this setting is equivalent to a one-direction GAN. Third, we see that the segmentation loss and/or the identity loss must be included to obtain meaningful results: when none of those penalties are included ($\lambda_{\text{seg}} = 0$, $\lambda_{\text{id}} = 0$), cloud detection decreases abruptly. This is because, without those losses, generators are not constrained to maintain original radiance values and spectral signatures. Generated images displayed in Fig. 11 show that the generators trained without these two penalty terms replace clear surfaces with cloud spectra, and vice versa. Hence, this result shows that the cycle consistency loss is not sufficient to prevent big changes in the original input spectra.

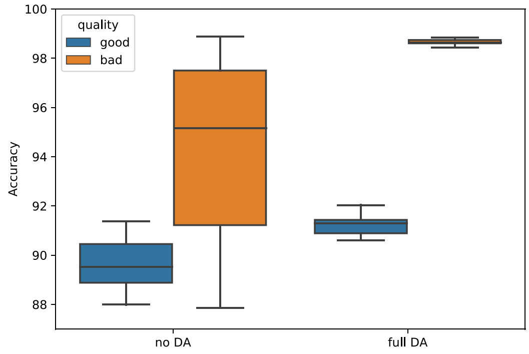


Fig. 12. Cloud detection accuracy in the *PV-24* dataset of the 10 cloud detection U-Net models with different weight initialization with and without the proposed DA transformation. Pixels stratified according to the quality indicator available in the SM flag of Proba-V images.

Fig. 12 shows the cloud detection accuracy with the proposed DA transformation (full DA) and without DA (no DA). In this case, results are stratified using the quality flag available in the SM band of Proba-V. In the *PV-24* dataset, 16.13% of pixels have at least one value in a band flagged as having bad quality. Within bad quality pixels, 96.8% are cloudy pixels. We see that, on the one hand, if the DA transformation is not used, the accuracy of the networks wildly varies especially for bad quality pixels. These differences in accuracy of models trained on the same data indicate that the networks are extrapolating in those regions. On the other hand, when the DA transformation is used, all the networks identify correctly most of the bad quality pixels.

Finally, Fig. 13 shows some cherry-picked examples of cloud detection with the proposed methodology. On each row, we show: the pseudosimultaneous Landsat-8 Upscaled image, the original Proba-V image, the Proba-V image after the domain adaptation $G_{\text{PV}\to\text{LU}}$, the cloud mask using as input the DA image, and the cloud mask obtained without the domain adaptation. First row shows a completely cloudy image with several blue saturated pixels in the original Proba-V image. We see that the DA image removes those saturated values and helps the cloud detection model to correctly predict all pixels. We see that, if no DA transformation is employed, the saturated values in Proba-V hinder the performance of the model with some cloud missclassifications. The second row shows an acquisition over the Canyon de Chelly, in North America. We see again that saturated values in the blue band disappear after the DA transformation; in this case, this help to reduce the false positives in the bottom and upper left part of the image. In the third row, we see an easier case where cloud masks, with and without DA, are both accurate. Finally, in the fourth row, a very challenging example of thin clouds over snowy mountains is shown. In this case, the DA method captures better the thin cloud in the

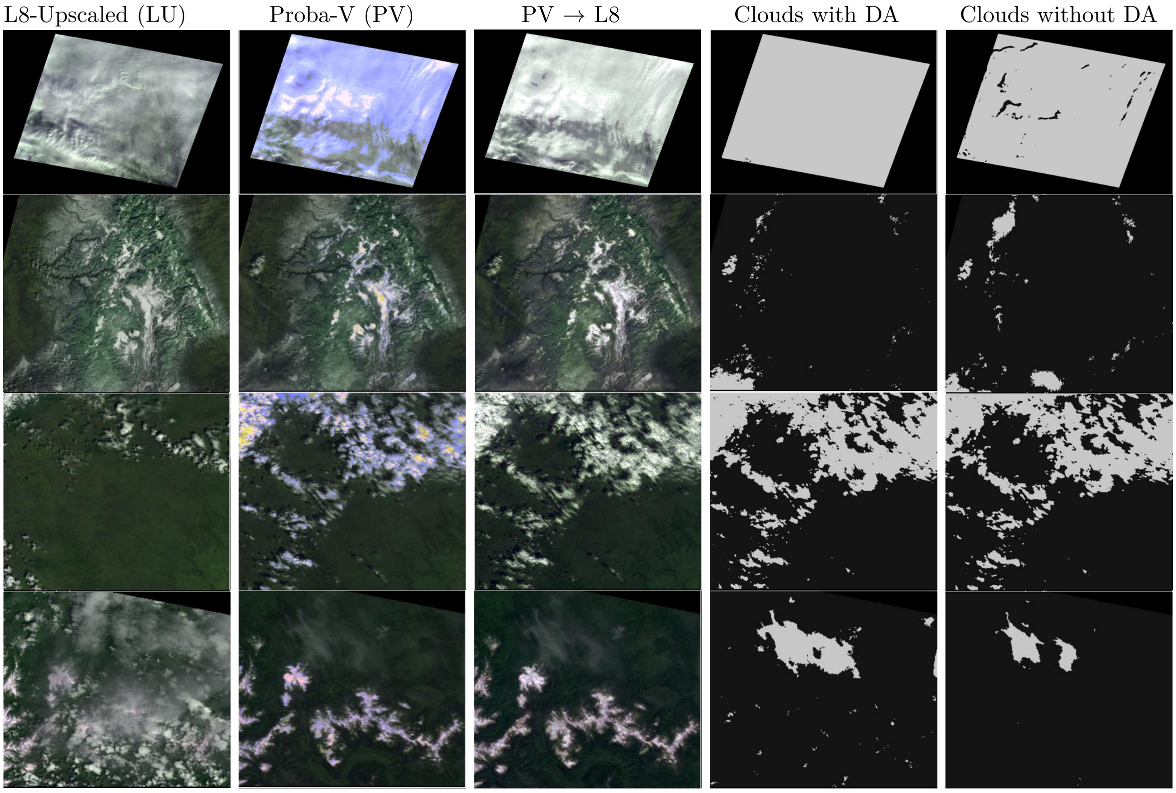


Fig. 13. From left to right: Landsat-8 Upscaled image, Proba-V image, Proba-V as Landsat-8 upscaled, Clouds from Proba-V as Landsat-8 upscaled, and Clouds without domain adaptation.

top of the image; however, it produces some false positives in mixed pixels where snow is melting. For results of all methods shown in Table II over all the images in the *38-Cloud Proba-V pseudosimultaneous dataset*, we refer the reader to the web application.[5]

VI. Discussion and Conclusion

The main motivation for this study has been to propose a domain adaptation algorithm to improve transfer learning from Landsat-8 to Proba-V for cloud detection. However, the obtained transformation is application-independent since its aim is to reduce spatial and spectral differences between the two sensors image datasets. It is worth noting that the main objective of the Proba-V mission was to ensure continuity and fill the gap between SPOT Vegetation (VGT) and the Sentinel-3 missions [62]. These multimission approaches provide long time series of surface vegetation products with complete Earth coverage, and require a high radiometric consistency across sensor datasets. For instance, differences between Proba-V and VGT2 spectral responses were of the same order as between VGT1 and VGT2 [63]. Also, the ESA Sentinel-3 synergy vegetation products replicate the characteristics of the 1-km SPOT VGT products using spectral remapping and colocation techniques. In this context, the proposed domain adaptation methodology is a new tool based on sound statistical methods that could be used to improve consistency across sensors.

Obtained results have shown that the proposed domain adaptation transformation, in addition to reduce the difference between the TOA reflectance distributions, also unintentionally fixes some radiometry artifacts. Looking at the resulting distributions, one can see that the characteristic saturation in the Blue and Red channels of Proba-V disappears in the adapted images (see Fig. 7). Moreover, although the developed model does not distinguish explicitly between *good* and *bad* pixels of the Proba-V products quality flag, results show that good quality pixels are much less changed than bad quality pixels (see Fig. 10). On the one hand, this result agrees with [11], where good quality pixels are similar between Landsat-8 and Proba-V, which implies that their TOA radiance calibration is quite good. On the other hand, the proposed adaptation method only changes those *good* pixels within the range of the radiometric error reported in [11] or in [12] (see Fig. 10). However, Proba-V products present a significant number of bad quality pixels: between 20% and 30% in the blue channel and around 5% in the red one. This can eventually have an important impact on derived products, since usually we are expected to provide results in the whole image. For instance, removing or ignoring those *bad* pixels

[5] [Online]. Available: https://isp.uv.es/projects/cloudsat/pvl8dagans

is not feasible for methods using the spatial context, such as CNNs, since the output for a pixel depends on the surrounding pixels. Therefore, the DA method improves the TOA reflectance image resemblance across sensors, and in this particular case, significantly increases the number of pixels that can be further processed: i.e., corrected bad quality pixels (see Fig. 12).

In addition, the proposed adversarial domain adaptation model has been modified to specifically improve the performance of a transfer learning cloud detection problem. In particular, the cost function used to train the DA network has been modified by including a dedicated term forcing similar cloud detection results across domains. Results show that, when the proposed transformation is applied, cloud detection models trained using only Landsat-8 data increase cloud detection accuracy in Proba-V. It is worth noting that results without the application dependent term ($\lambda_{\text{seg}} = 0$) are good enough. However, it is important to include either $\lambda_{\text{seg}} > 0$ or $\lambda_{\text{id}} > 0$ in order to constrain the method and to avoid artifacts in the adapted images.

The proposed adaptation framework can be extended in two ambitious directions. On the one hand, it would be possible to learn a domain adaptation transformation directly from Landsat-8 to Proba-V, without previously applying the upscaling transformation, which converted the Landsat-8 images in order to have similar spatio-spectral properties than Proba-V (number of bands and spatial resolution). However, this approach would imply to solve the more challenging super-resolution problem when transforming Proba-V to Landsat-8 in our cyclic GAN adaptation framework. On the other hand, an interesting option to explore would be to apply the transformation from top of atmosphere to surface reflectance data. In this case, the obtained transformation would be equivalent to learn an atmospheric correction transformation relying on the image data only. In addition, as mentioned before, the proposed framework can be applied to any other pair of similar sensors such as Proba-V and Sentinel 2 and 3.

Summarizing, in this article, a Cycle-GAN architecture has been proposed to train a domain adaptation method between Proba-V and upscaled Landsat images. The proposal includes two generators, two discriminators, and four different penalties. The GAN generator is used to modify the Proba-V images to better resemble the upscaled Landsat images that have also been used to train a cloud detection algorithm. Results on original Proba-V images demonstrate that when using the proposed model for the adaptation a higher cloud detection accuracy is achieved.

Appendix

This appendix presents the details about the network architectures and the training procedure of the generators and discriminators of the proposed generative adversarial adaptation model. It also has the details of the networks and training configuration of the cloud detection models. The implementation is available.[6]

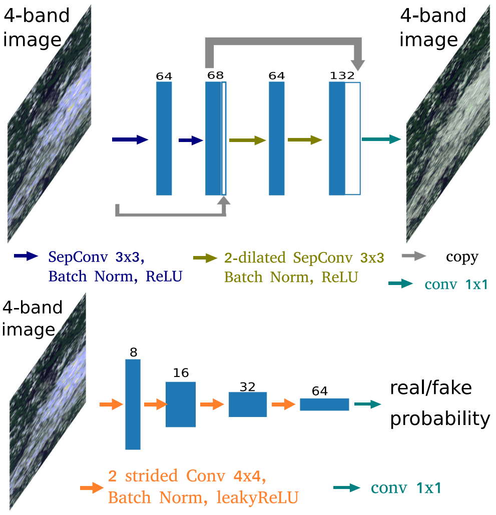

4-band image
64
68
64
132
4-band image
SepConv 3x3, Batch Norm, ReLU
2-dilated SepConv 3x3 Batch Norm, ReLU
copy
conv 1x1
4-band image
8
16
32
64
real/fake probability
2 strided Conv 4x4, Batch Norm, leakyReLU
conv 1x1

Fig. 14. (Top) Generator and (bottom) discriminator architectures. Implementation details available at https://github.com/IPL-UV/pvl8dagans.

We use the same network architecture for all generators G [see Fig. 14(top)]. In particular, G is a five-layer FCNN. It consists of the following.

1) Two layers: Convolution with 64 separable filters of size 3 × 3, reLU activation, and batch normalization.
2) Two layers: Convolution with 64 separable filters of size 3 × 3 with a dilation rate equal to 2, reLU activation, and batch normalization.
3) One layer: 1 × 1 convolution with four channels output.

We used residual connections between blocks and before the final layer.

As in the case of the generators, both discriminators, D, have the same architecture: A five-layer convolutional neural network adapted from [54] [see Fig. 14 (bottom)]. It consists of the following.

1) Four layers: 4 × 4 convolution, leakyReLU activation, and batch normalization. The number of filters starts in eight for the first convolution and grows by a factor two in every layer. The convolutions are applied with a stride of 2, thus reducing by this factor the spatial size of the input.
2) One layer: 1 × 1 convolution with one output channel and a sigmoid activation.

The output of the discriminators can be interpreted as the probability of an image to be *real*. Hence, the discriminator is trained to provide close-to-zero values for images generated by G and close-to-one values for real satellite images.

The proposed networks ($G_{\text{PV}\to\text{LU}}$, $G_{\text{LU}\to\text{PV}}$, D_{PV}, and D_{LU}) were trained simultaneously using 64 × 64 patches with stochastic gradient descent on their respective losses with a batch size of 48. Networks were trained for 25 epochs in the *Proba-V pseudosimultaneous dataset*, which corresponds to 14 574 steps where the weights are updated. In order to ensure convergence in the GAN training procedure, we regularized the discriminator using 0 centered gradient penalty on the real images [55] with a weight of 10. We used the Adam [64] optimizer with a learning

[6] [Online]. Available: https://github.com/IPL-UV/pvl8dagans

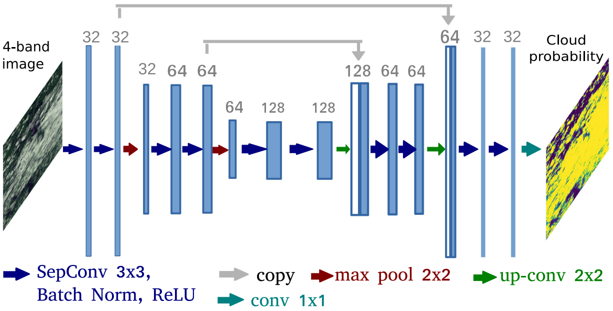


Fig. 15. Simplified U-Net architecture used for the cloud detection model. Implementation details are available at https://github.com/IPL-UV/pvl8dagans.

rate of 10^{-4} to update the weights of the networks at each step. Additionally, we apply data augmentations in form of 90 rotations and horizontal and vertical flips.

For the cloud detection model f_{LU}, we used the same simplified U-Net architecture as in [5]. Fig. 15 shows the configuration of layers; we used only two subsampling steps and separable convolutions [65] to reduce the number of trainable parameters and floating points operations (96 k parameters and 2.18 M FLOPS). The cloud detection networks are trained for 250 k steps using batches of 64 overlapping patches of 32 × 32 pixels from the Biome dataset (upscaled to 333 m as described in Section IV). The network is trained to minimize the binary cross-entropy loss between the model output and the ground-truth labels. We used a learning rate of 10^{-4} and the Adam [64] optimizer.

References

[1] *UCS Satellite Database*. Accessed: Apr. 25, 2020, [Online]. Available: https://www.ucsusa.org/resources/satellite-database

[2] D. Tuia, C. Persello, and L. Bruzzone, “Domain adaptation for the classification of remote sensing data: An overview of recent advances,” *IEEE Geosci. Remote Sens. Mag.*, vol. 4, pp. 41–57, Jun. 2016.

[3] A. Torralba and A. A. Efros, “Unbiased look at dataset bias,” in *Proc. Conf. Comput. Vision Pattern Recognit.*, Jun. 2011, pp. 1521–1528.

[4] G. Mateo-García, V. Laparra, and L. Gómez-Chova, “Domain adaptation of Landsat-8 and Proba-V data using generative adversarial networks for cloud detection,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 712–715.

[5] G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, “Transferring deep learning models for cloud detection between Landsat-8 and Proba-V,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 160, pp. 1–17, Feb. 2020.

[6] B. Banerjee and S. Chaudhuri, “Hierarchical subspace learning based unsupervised domain adaptation for cross-domain classification of remote sensing images,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 11, pp. 5099–5109, Nov. 2017.

[7] D. H. Svendsen, L. Martino, M. Campos-Taberner, F. J. García-Haro, and G. Camps-Valls, “Joint Gaussian processes for biophysical parameter retrieval,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1718–1727, Mar. 2018.

[8] A. Wolanin *et al.*, “Estimating and understanding crop yields with explainable deep learning in the Indian Wheat Belt,” *Environ. Res. Lett.*, vol. 15, no. 2, Feb. 2020, Art. no. 024019.

[9] U.S. Geological Survey, “Landsat 8 data users handbook,” USGS, Tech. Rep. LSDS-1574, Apr. 2019. [Online]. Available: https://www.usgs.gov/media/files/landsat-8-data-users-handbook

[10] W. Dierckx *et al.*, “PROBA-V mission for global vegetation monitoring: Standard products and image quality,” *Int. J. Remote Sens.*, vol. 35, no. 7, pp. 2589–2614, 2014.

[11] S. Sterckx and E. Wolters, “Radiometric top-of-atmosphere reflectance consistency assessment for Landsat 8/OLI, Sentinel-2/MSI, PROBA-V, and DEIMOS-1 over Libya-4 and RadCalNet calibration sites,” *Remote Sens.*, vol. 11, no. 19, Art. no. 2253, Jan. 2019.

[12] S. Sterckx, W. Dierckx, S. Adriaensen, and S. Livens, “PROBA-V commissioning report Annex 1-radiometric calibration results,” VITO, Belgium, Germany, Nov. 2013. [Online]. Available: https://earth.esa.int/documents/700255/1929094/US-20+Annex1-RadiometricCalibartion-v1_1.pdf/389c059f-4808-4f92-9642-2cab5a4450cb

[13] J. Hoffman *et al.*, “CyCADA: Cycle-consistent adversarial domain adaptation,” in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 1989–1998.

[14] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 2223–2232.

[15] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[16] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, Oct. 2015, pp. 234–241.

[17] J. H. Jeppesen, R. H. Jacobsen, F. Inceoglu, and T. S. Toftegaard, “A cloud detection algorithm for satellite imagery based on deep learning,” *Remote Sens. Environ.*, vol. 229, pp. 247–259, Aug. 2019.

[18] S. Mohajerani and P. Saeedi, “Cloud-Net: An end-to-end cloud detection algorithm for Landsat 8 imagery,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 1029–1032.

[19] S. Mohajerani, T. A. Krammer, and P. Saeedi, “A cloud detection algorithm for remote sensing images using fully convolutional neural networks,” in *Proc. IEEE 20th Int. Workshop Multimedia Signal Process.*, Aug. 2018, pp. 1–5.

[20] Z. Li, H. Shen, Q. Cheng, Y. Liu, S. You, and Z. He, “Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 150, pp. 197–212, Apr. 2019.

[21] J. Yang, J. Guo, H. Yue, Z. Liu, H. Hu, and K. Li, “CDnet: CNN-based cloud detection for remote sensing imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 6195–6211, Aug. 2019.

[22] Z. Zhu, S. Wang, and C. E. Woodcock, “Improvement and expansion of the Fmask algorithm: Cloud, cloud shadow, and snow detection for Landsats 4-7, 8, and Sentinel 2 images,” *Remote Sens. Environ.*, vol. 159, pp. 269–277, Mar. 2015.

[23] M. Wieland, Y. Li, and S. Martinis, “Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network,” *Remote Sens. Environ.*, vol. 230, Art. no. 111203, Sep. 2019.

[24] U. S. Geological Survey, “L8 SPARCS cloud validation masks,” 2016. [Online]. Available: https://landsat.usgs.gov/sparcs

[25] M. Segal-Rozenhaimer, A. Li, K. Das, and V. Chirayath, “Cloud detection algorithm for multi-modal satellite imagery using convolutional neural-networks (CNN),” *Remote Sens. Environ.*, vol. 237, Art. no. 111446, Feb. 2020.

[26] Y. Shendryk, Y. Rist, C. Ticehurst, and P. Thorburn, “Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 157, pp. 124–136, Nov. 2019.

[27] C. Toté, *et al.*, “Evaluation of PROBA-V collection 1: Refined radiometry, geometry, and cloud screening,” *Remote Sens.*, vol. 10, no. 9, p. 1375, Aug. 2018. [Online]. Available: https://www.mdpi.com/2072-4292/10/9/1375

[28] S. Liang, *Quantitative Remote Sensing of Land Surfaces*. New York, NY, USA: Wiley-Interscience, 2003.

[29] E. Mandanici and G. Bitelli, “Preliminary comparison of Sentinel-2 and Landsat 8 imagery for a combined use,” *Remote Sens.*, vol. 8, no. 12, Art. no. 1014, Dec. 2016.

[30] C. Revel *et al.*, “Sentinel-2A and 2B absolute calibration monitoring,” *Eur. J. Remote Sens.*, vol. 52, no. 1, pp. 122–137, Jan. 2019.

[31] Y. Zhao, L. Ma, C. Li, C. Gao, N. Wang, and L. Tang, “Radiometric cross-calibration of Landsat-8/OLI and GF-1/PMS sensors using an instrumented sand site,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 10, pp. 3822–3829, Oct. 2018.

[32] R. Houborg and M. F. McCabe, “A Cubesat enabled spatio-temporal enhancement method (CESTEM) utilizing Planet, Landsat and MODIS data,” *Remote Sens. Environ.*, vol. 209, pp. 211–226, May 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0034425718300786

[33] M. Claverie *et al.*, "The harmonized Landsat and Sentinel-2 surface reflectance data set," *Remote Sens. Environ.*, vol. 219, pp. 145–161, Dec. 2018.

[34] H. K. Zhang *et al.*, "Characterization of Sentinel-2A and Landsat-8 top of atmosphere, surface, and nadir BRDF adjusted reflectance and NDVI differences," *Remote Sens. Environ.*, vol. 215, pp. 482–494, Sep. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0034425718301883

[35] D. Scheffler, D. Frantz, and K. Segl, "Spectral harmonization and red edge prediction of Landsat-8 to Sentinel-2 using land cover optimized multivariate regressors," *Remote Sens. Environ.*, vol. 241, Art. no. 111723, May 2020.

[36] R. C. Gonzalez and R. E. Woods, *Digital Image Process*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2006.

[37] S. Inamdar, F. Bovolo, L. Bruzzone, and S. Chaudhuri, "Multidimensional probability density function matching for preprocessing of multitemporal remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1243–1252, Apr. 2008.

[38] D. Tuia, J. Muñoz-Marí, L. Gómez-Chova, and J. Malo, "Graph matching for adaptation in remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 329–341, Jan. 2013.

[39] D. Tuia, M. Volpi, M. Trolliet, and G. Camps-Valls, "Semisupervised manifold alignment of multimodal remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7708–7720, Dec. 2014.

[40] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Advances Neural Inf. Process. Syst.*, 2014, vol. 27, pp. 2672–2680.

[41] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, 2016. [Online]. Available: http://jmlr.org/papers/v17/15-239.html

[42] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jul. 2017, pp. 2962–2971.

[43] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proc. Advances Neural Inform. Process. Syst.*, 2017, vol. 30, pp. 700–708. [Online]. Available: http://papers.nips.cc/paper/6672-unsupervised-image-to-image-translation-networks.pdf

[44] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jul. 2017, pp. 95–104.

[45] A. Elshamli, G. W. Taylor, A. Berg, and S. Areibi, "Domain adaptation using representation learning for the classification of remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 9, pp. 4198–4209, Sep. 2017.

[46] S. Song, H. Yu, Z. Miao, Q. Zhang, Y. Lin, and S. Wang, "Domain adaptation for convolutional neural networks-based remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1324–1328, Aug. 2019.

[47] O. Tasar, S. L. Happy, Y. Tarabalka, and P. Alliez, "ColorMapGAN: Unsupervised domain adaptation for semantic segmentation using color mapping generative adversarial networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7178–7193, Oct. 2020.

[48] Y. Koga, H. Miyazaki, and R. Shibasaki, "A method for vehicle detection in high-resolution satellite images that uses a region-based object detector and unsupervised domain adaptation," *Remote Sens.*, vol. 12, no. 3, Art. no. 575, Jan. 2020.

[49] B. Benjdira, Y. Bazi, A. Koubaa, and K. Ouni, "Unsupervised domain adaptation using generative adversarial networks for semantic segmentation of aerial images," *Remote Sens.*, vol. 11, no. 11, Art. no. 1369, Jan. 2019.

[50] F. Ye, W. Luo, M. Dong, H. He, and W. Min, "SAR image retrieval based on unsupervised domain adaptation and clustering," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 9, pp. 1482–1486, Sep. 2019.

[51] L. Wang *et al.*, "SAR-to-optical image translation using supervised cycle-consistent adversarial networks," *IEEE Access*, vol. 7, pp. 129 136–129 149, 2019.

[52] L. Liu, Z. Pan, X. Qiu, and L. Peng, "SAR target classification with cycleGAN transferred simulated samples," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2018, pp. 4411–4414.

[53] S. J. Pan and Q. Yang, "A Survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[54] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 5967–5976.

[55] L. Mescheder, A. Geiger, and S. Nowozin, "Which training methods for GANs do actually converge?," in *Proc. Int. Conf. Mach. Learning*, J. Dy, and A. Krause, Eds. vol. 80, Jul. 2018, pp. 3481–3490.

[56] L. Denaro and C. Lin, "Hybrid canonical correlation analysis and regression for radiometric normalization of cross-sensor satellite imagery," *IEEE J. Sel. Topics Appl. Earth Observ Remote Sens.*, vol. 13, pp. 976–986, 2020.

[57] U. Geological Survey, "L8 bome cloud validation masks," 2016. [Online]. Available: http://doi.org/10.5066/F7251GDH

[58] S. Foga *et al.*, "Cloud detection algorithm comparison and validation for operational Landsat data products," *Remote Sens. Environ.*, vol. 194, pp. 379–390, Jun. 2017.

[59] M. J. Hughes and D. J. Hayes, "Automated detection of cloud and cloud shadow in single-date Landsat imagery using neural networks and spatial post-processing," *Remote Sens.*, vol. 6, no. 6, pp. 4907–4926, May 2014.

[60] L. Gómez-Chova, G. Mateo-García, J. Muñoz-Marí, and G. Camps-Valls, "Cloud detection machine learning algorithms for PROBA-V," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2017, pp. 2251–2254.

[61] E. Wolters, W. Dierckx, M.-D. Iordache, and E. Swinnen, "PROBA-V products user manual," VITO, Belgium, Germany, Mar. 16, 2018. [Online]. Available: http://www.vito-eodata.be/PDF/image/PROBAV-Products_User_Manual.pdf

[62] C. Toté and E. Swinnen, "Extending the spot/vegetation–Proba-V archive with Sentinel-3: A preliminary evaluation," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 8707–8710.

[63] S. Sterckx, S. Adriaensen, W. Dierckx, and M. Bouvet, "In-orbit radiometric calibration and stability monitoring of the PROBA-V instrument," *Remote Sens.*, vol. 8, no. 7, p. 546, Jul. 2016. [Online]. Available: https://www.mdpi.com/2072-4292/8/7/546

[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations, San Diego, CA, USA, May 7–9, 2015*, 2015, pp. 1–13.

[65] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jul. 2017, pp. 1800–1807.

Gonzalo Mateo-García received the double B.Sc. degree in mathematics and computer science from Universidad Autnoma de Madrid, Madrid, Spain, in 2011, and the M.Sc. degree in statistics from Universidad Complutense de Madrid, Madrid, in 2012. He is currently working toward the Ph.D. degree with the Image and Signal Processing group, Universidad de Valencia, Valencia, Spain, where he conducts research developing machine learning models for Earth observation problems.

He has been involved as a Researcher with Frontier Development Lab research sprints, in 2019 and 2020. He had previously worked in the private sector on renewable energy forecasting. He has authored and coauthored research on topics ranging from cloud screening to biophysical parameter retrieval and flood detection.

Valero Laparra received the B.Sc. degree in telecommunications engineering and the B.Sc. degree in electronics engineering from the Universitat de València, Valencia, Spain, in 2005 and 2007, respectively, the B.Sc. degree in mathematics from the Universidad Nacional de Educación a Distancia, Madrid, Spain, in 2010, and the Ph.D. degree in computer science and mathematics from the Universitat de València, in 2011.

He is currently an Assistant Professor with the Escuela Tcnica Superior de Ingenera, Universitat de Valècia, and Researcher with the Image Processing Laboratory, Universitat de València.

Dan López-Puigdollers received the B.Sc. degree in telecommunications engineering from Universidad Pública de Navarra, Pamplona, Spain, in 2016, and the M.Sc. degree in intelligent systems from Universitat Jaume I, Castelló, Spain, in 2017. He is currently working toward the Ph.D. degree with Image Processing Laboratory, Universitat de València, Valencia, Spain, working on the research topic of statistical learning for cloud detection in remote sensing satellite images, under the supervision of Luis Gómez-Chova.

During his M.Sc. studies, he participated in a research scholarship related to adaptive learning techniques and characterization in digital images for the automatic blood cell recognition.

Luis Gómez-Chova (Senior Member, IEEE) received the Ph.D. degree in electronics engineering from the University of Valencia, Valencia, Spain, in 2008.

He has completed different doctoral and postdoctoral stays at prestigious research centers. He is currently a Full Professor with the Department of Electronic Engineering, University of Valencia. He is also a Researcher with the Image and Signal Processing (ISP, isp.uv.es) group where his work is mainly related to pattern recognition and machine learning applied to remote sensing multispectral images and cloud screening. He conducts and supervises research on these topics within the frameworks of several national and international projects. He is the author or coauthor of more than 55 international journal papers (JCR), more than 140 international conference papers, and several international book chapters.

Dr. Gómez-Chova was the recipient of the National Award for Electronic Engineering by the Spanish Ministry of Education. He is a Referee for many international journals and conferences.

Publication IV: Towards global flood mapping onboard low cost satellites with machine learning

Publication IV

G. Mateo-Garcia, J. Veitch-Michaelis, L. Smith, S. Oprea, G. Schumann, Y. Gal, A. Güneş Baydin, and D. Backes, Towards global flood mapping onboard low cost satellites with machine learning, *Scientific Reports*, vol. 11, no. 1, Mar. 2021, doi: 10.1038/s41598-021-86650-z.

Q1: Multidisciplinary Sciences, IF = 4.379

nature

SCIENTIFIC REPORTS

Scientific Reports

scientific reports

OPEN

Towards global flood mapping onboard low cost satellites with machine learning

Gonzalo Mateo-Garcia[1,9]✉, Joshua Veitch-Michaelis[2,9], Lewis Smith[3,9], Silviu Vlad Oprea[4], Guy Schumann[5,6], Yarin Gal[3], Atılım Güneş Baydin[3] & Dietmar Backes[7,8]

Spaceborne Earth observation is a key technology for flood response, offering valuable information to decision makers on the ground. Very large constellations of small, nano satellites— 'CubeSats' are a promising solution to reduce revisit time in disaster areas from days to hours. However, data transmission to ground receivers is limited by constraints on power and bandwidth of CubeSats. Onboard processing offers a solution to decrease the amount of data to transmit by reducing large sensor images to smaller data products. The ESA's recent PhiSat-1 mission aims to facilitate the demonstration of this concept, providing the hardware capability to perform onboard processing by including a power-constrained machine learning accelerator and the software to run custom applications. This work demonstrates a flood segmentation algorithm that produces flood masks to be transmitted instead of the raw images, while running efficiently on the accelerator aboard the PhiSat-1. Our models are trained on *WorldFloods*: a newly compiled dataset of 119 globally verified flooding events from disaster response organizations, which we make available in a common format. We test the system on independent locations, demonstrating that it produces fast and accurate segmentation masks on the hardware accelerator, acting as a proof of concept for this approach.

Floods are among the most destructive extreme weather events—between 1995 and 2015, over 2.2 billion people were affected by floods comprising 53% of the total of people affected by all weather-related disasters[1,2]. Situational awareness on the ground is crucial for effective disaster response, and, today, satellite imagery is one of the most important sources of this information[3]. Both passive optical (multi-spectral) and synthetic-aperture radar (SAR) imagery are routinely used to determine flood extent and further derived products[4] (Fig. 1).

Some regions, like the USA, Europe and Japan have access to high-quality imaging resources from defence organisations and commercial satellite operators through domestic space agencies (i.e., NASA, ESA, JAXA). However, several of the worst flood-affected regions are in developing countries: of the top 20 countries by disaster mortality in proportion to their population for the years 1990–2017, the top five are low or lower-middle-income countries, and only five are upper-middle income[5].

Many of these countries have almost no means of getting access to higher quality imaging resources via domestic channels. To address this, organisations such as the International Charter "Space and Major Disasters"[7], initiated by the European Space Agency (ESA), liaise with space agencies and associated commercial organisations to produce free high resolution maps for end-users in the field. Despite best efforts it can take many days to provide actionable information, mainly due to image down-linking and subsequent image analysis[8]. Commercial organisations are able to provide the highest-frequency (daily) and highest-resolution (sub-metre) images, but their satellites must also be tasked and their images may only be freely available for a limited period of time during disasters via the International Charter Space and Major Disasters. ESA's Copernicus program[9] provides open data globally at 10 m resolution, but the optical component, Sentinel-2 (S2)[10], has a revisit time of five days at the equator and two to three days at mid-latitudes. This leads to wait periods much longer than two days in areas such as central Africa where alternatives for rapid data capture can be limited.

In this work we investigate how a constellation of small, inexpensive, nano satellites assembled from commercial off-the-shelf (COTS) hardware, also known as CubeSats[11], could be used for disaster response, using flooding as a case study. The main advantage of using CubeSats is an improved revisit time through larger constellations

[1]Universidad de Valencia, Valencia, Spain. [2]Liverpool John Moores University, Liverpool, UK. [3]University of Oxford, Oxford, UK. [4]University of Edinburgh, Edinburgh, UK. [5]University of Bristol, Bristol, UK. [6]RSS-Hydro, RED, Dudelange, Luxembourg. [7]University of Luxembourg, Luxembourg, Luxembourg. [8]University College London, London, UK. [9]These authors contributed equally: Gonzalo Mateo-Garcia, Joshua Veitch-Michaelis and Lewis Smith. ✉email: Gonzalo.Mateo-Garcia@uv.es

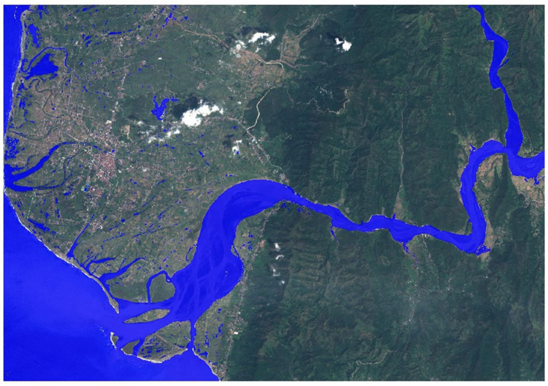


Figure 1. An example of a data product from the Copernicus EMS catalogue (activation EMSR312), in this case a map showing flood extent over the city of Vigan in the North West of Luzon island in the Philippines in September 2018. A blue water mask (here generated using an automatic method from a RADARSAT-2 image) is overlaid on top of a Sentinel-2 image, showing the extent of flooding. Sentinel 2 imagery and Copernicus EMS mapping products are provided as public domain. Base image and reference labels mask are included in the *WorldFloods* database and code for plotting this images may be found in our repository[6].

of satellites. Commercial organisations like Planet Labs, Inc. (California, USA) have demonstrated the potential for large fleets of low-cost satellites for Earth observation (EO), though their data are only freely available in small quantities. Tens of CubeSats similar to ESA's FSSCat mission[12] could be launched for the cost of a single conventional Earth observation satellite, with 30 CubeSats reducing the nominal revisit time from five days to around eight hours for a similar cost. However, CubeSats can have very limited downlink bandwidth, on the order of 1–10 Mbps[13], compared to around 0.5 Gbps for S2[10] (Downlink is communication from the satellite back to a ground station on Earth. It is very constrained for CubeSats because the satellite itself must act as a transmitter). In addition to this, there is a cost associated with downlinking data which is proportional to the transfer size, desired frequency and availability of ground stations.

Constrained downlink budgets are a common problem in space science and can be addressed using on-board processing for both targeted data acquisition and filtering. Examples include autonomously identifying science targets on Mars[14,15] and discarding cloud-obscured imagery on NASA's EO-1 satellite with the Autonomous Sciencecraft Experiment (ASE)[16,17]. On-board flood detection and mapping (an image segmentation task) has also been proven with ASE[18] using Hyperion, a 220-band hyperpsectral camera with a 30 m ground sample distance. The output was limited by the computational capability of the satellite and only a small 7.7 × 30 km region in the centre of the field of view could be processed using 12 of 220 bands. Flood detection was based on simple band thresholds, and an event was triggered based on the number of water pixels in a region compared to a baseline; the combination of three on-board classifiers achieved accuracies of 70–85.6%.

We propose to take this approach further leveraging modern deep learning[19] algorithms, to perform multiclass segmentation with high accuracy, on-board of very cheap satellite hardware. In order to demonstrate feasibility, we optimise our application for ESA's ΦSat-1, part of FSSCat[20]—a technology demonstrator mission— launched at 2nd of September 2020. Among other sensors, FSSCat carries a Cosine HyperScout 2 49-band hyperspectral camera (70 m ground sample distance at 500 km) which integrates an Intel Movidius Myriad2 vision processing unit (VPU) as a co-processor for performing on-board computer vision and neural network inference[12,21]. FSSCat is a 3 × 2U CubeSat, with HyperScout taking up 1U (10 × 10 × 11 cm) of space. The first machine learning application deployed on the satellite is a cloud detection model[22] similar to the system used on EO-1.

Using the on-board VPU to perform segmentation, an output two-bit flood map (up to four classes) would reduce the amount of data being down-linked by a factor of 100 (assuming 49 12-bit channels). Since segmented regions tend to be quite large and continuous, there could likely be further savings via simple compression methods like run-length encoding[23]. Our models are trained on a new extensive dataset of human-annotated flood maps covering more than 100 flood events and tested on five independent events from different locations around the globe. We made this dataset available at https://tinyurl.com/worldfloods. While we address flooding in this paper, satellites with on-board capability are attractive as they can potentially be re-targeted for multiple diverse missions, and on-board models can be improved over time if their weights are small enough.

The contributions of this paper are as follows:

1. We introduce a new dataset—*WorldFloods*—that combines, in "machine-learning ready form", several existing databases of satellite imagery of historical flood events. The dataset contains pairs of Sentinel-2 images and flood extent maps covering 119 global flood events.
2. Using this dataset, we train several convolutional neural network (CNN) architectures for flood segmentation and compare their performance against standard baselines: linear models and a per-image optimal threshold on the normalised difference water index (NDWI)[24].

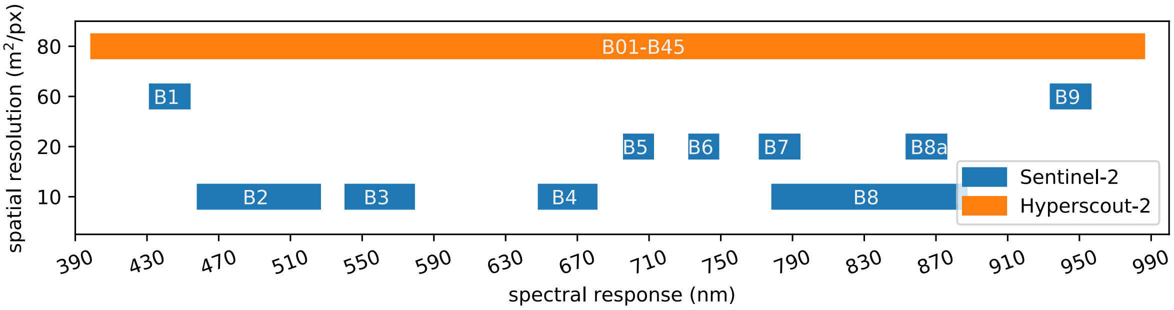


Figure 2. Spatial resolution and spectral response of Sentinel-2 and HyperScout-2 sensors.

3. We show that our models can process large volumes of hyperspectral data, yet fit the constraints of hardware deployed on the satellite. Specifically we report results on the on-board co-processor Intel Movidius Myriad2, which we found was able to process a 12 MP image in less than a minute.

Background

Flood mapping.

Water mapping, of which flood mapping is a special case, is a semantic segmentation task (also called land cover classification in remote sensing) that has been studied for decades. A simple approach to water mapping is to compute indices like the normalised difference water index (NDWI)[24] which exploits the difference in absorption of light by water bodies between the green and the near infrared part of the electromagnetic spectrum. However, this method can perform poorly because the spectral profile of flood water varies widely due to the presence of debris, pollutants and suspended sediments[25]. As a result, the main challenge with using indices at a global scale is that the threshold for water retrieval must be tuned per environment. SAR images (e.g., Sentinel-1) are commonly used for water retrieval as they are not affected by cloud cover[26,27], severe weather and lighting conditions. Since calm water strongly reflects radar wavelengths away from the receiving antenna (specular reflection), image thresholding is a straightforward way to identify water regions by their very low backscatter intensity. However, the presence of waves or wind causes significant backscatter, which can make inland water harder to identify. In addition, flooding in urban areas[28] is difficult to map due to multiple reflections by buildings and taller vegetation which produces an increase in backscatter. Additionally, as SAR is an active sensing technique with a high power requirement (e.g. Capella constellation, 600 Watts for transmission[29]), deployment on a small satellite is challenging; we therefore limit the scope of this paper to passive optical sensors, but we do use some training data derived from Sentinel 1 imagery.

More sophisticated segmentation techniques include rule-based classifiers[18,25] which use a fixed or tuned threshold on indices or individual bands; classical supervised machine learning[3]; and recently deep learning[30–33]. Among deep learning methods, fully convolutional neural networks (FCNNs)[34] produce state-of-the-art results in image segmentation tasks with fast inference time; they are thus the model proposed for this application.

Hyperspectral image processing.

One of the inherent difficulties of targeting a satellite that has yet to be launched is that no real-world orbital data are available. This problem is usually addressed by using data from a similar satellite and accounting for known differences in spectral sensitivity[35]. However, in the case of ΦSat-1, the problem is exacerbated as there are very few satellites with hyperspectral sensors and archival data are similarly limited[36,37]. Notably HyperScout-1 has been flown in space, on the GOMX-4B mission, but data from this mission are not publicly available[38]. Other aerial missions like AVIRIS (a NASA-modified U2 aircraft)[36,39] have a larger public archive, but these images are mostly limited geographically to the USA. Since we need labelled data, we have the additional constraint that we rely on serendipitous image acquisition coinciding with flood events.

The images that HyperScout-2 produces are relatively large—45 visible channels and four thermal infrared channels with a dynamic range of 12-bits per pixel. The output image has a spectral resolution of 15 nm over a range of 400–1000 nm. HyperScout-2 is a push-broom sensor; a nominal 2D frame represents approximately a 200 km by 300 km swath at a nominal orbital height of 500 km[38]. The ground sample distance (GSD) at this altitude is 70 m.

We propose to use Sentinel-2 data for model training, which is sensitive to a similar wavelength range, but with fewer bands. S2 spatial resolution varies for each spectral band from 10 to 60 m. In order to produce a model for HyperScout-2 images we follow an approach similar to two recent studies[40,41] which demonstrate models that show some generalisation to multiple sensors. In particular, we select the bands of Sentinel-2 that are common to HyperScout-2 (shown in Fig. 2) and reduce the spatial resolution of Sentinel-2 images to 80 m using bilinear interpolation. In addition, HyperScout-2 and ΦSat-1 are expected to have a worse signal-to-noise ratio compared to Sentinel-2 due to its reduced size and poorer direct georeference. In order to account for this, our models are trained with degradatations in form of Gaussian noise, channel jitter (translational offsets) and motion blur. These degradations are implemented as data augmentation functions[42,43].

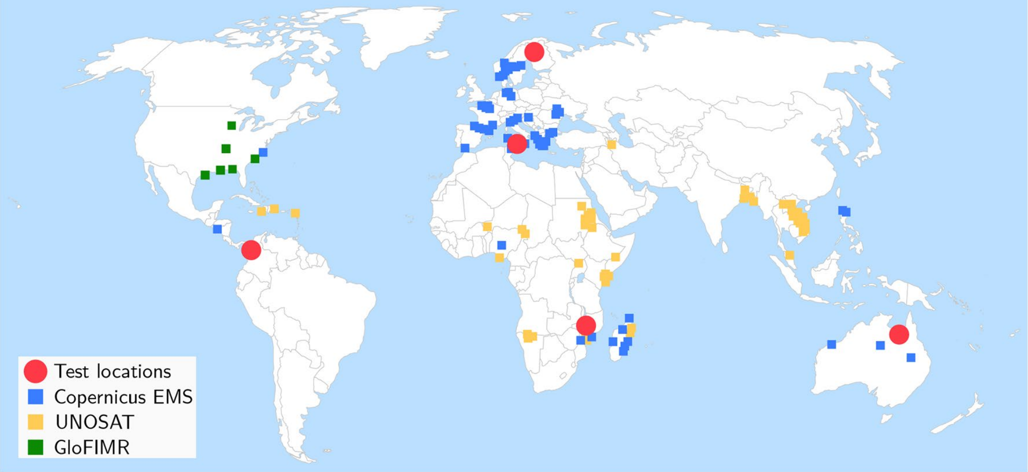


Figure 3. Locations of flood events contained in *WorldFloods*. Blue, orange and green areas denote Copernicus EMS, UNOSAT and GloFIMR data, respectively. Red circles denote test regions. Basemap credit: http://www.simplemaps.com.

Methods

Flood segmentation. Given a satellite image (with or without a flood), we wish to label each pixel as water/flood or land. As always with data coming from an optical sensor, we also have to deal with the problem of obstruction by clouds. Since we are targeting on-board processing, we choose to tackle this by adding a cloud class to the output of the model, so that we can maintain the workflow of a single pass over the image. Our models therefore have three output classes (land, water/flood and cloud), requiring two bits of data per pixel to store. Note our model does not distinguish water and flooded pixels; however we report segmentation results on flood and permanent water pixels using the JRC yearly permanent water layer[44].

***WorldFloods* dataset.** The development and evaluation of flooding response systems has been constrained so far by use of trusted, authoritative or validated datasets that are also often of limited geographical scope, with most studies only considering a single or very few flood events[33,45]. It is unclear whether such models would accurately generalise to the rest of the world due to variations in topography and land cover. To address this we collated a new global dataset called *WorldFloods*, which we believe is the largest collection of its kind.

WorldFloods contains 422 flood extent maps created by photo-interpretation either manually or semi-automatically, where a human validated machine-generated maps. A flood extent map is a vector layer (shapefile) derived from a satellite image with polygons indicating which part of that image has water (in some cases it distinguishes between flood water and permanent water and in other cases it does not); we assigned a date to each flood extent map which corresponds with the date of acquisition of the original satellite image that was used to derive it. Each flood extent map belongs to a flood event hence a flood event could have several flood maps which may cover different areas of interest or different days of the same area in the same flood event; in total the dataset covers 119 floods events that occurred between November 2015 and March 2019. We sourced all maps from three organisations: the Copernicus Emergency Management Service (Copernicus EMS)[46], the flood portal of UNOSAT[47], and the Global Flood Inundation Map Repository (GLOFIMR)[48]. The geographical distribution of flood maps is shown in Fig. 3.

For each flood event we provide the raw 13-band S2 image closest in time after the event, and rasterised *reference labels* (cloud, water and land) at 10 m resolution. (We explicitly avoid the term *ground truth* as labels are derived manually or semi-automatically by photo-interpretation and have not been validated by ground measurements). S2 images were downloaded from the Google Earth Engine[50]; S2 bands with spatial resolution larger than 10 m were resampled to 10 m using nearest neighbours interpolation. We generated cloud masks using `s2cloudless`[49]. The dataset contains in total more than 12 Gigapixels of labeled data which occupies around 266 GB of disk space. Figure 4 shows an example of S2 image and derived reference labels for a flood that occurred in Central-West Sicily in November 2018.

We manually validated the data to account for gross errors such as missing water bodies or invalid intensities. In some cases, missing water bodies were filled using the permanent water bodies dataset[44] available from the Google Earth Engine[50] (we also use this data to differentiate flood and permanent water in the results). Nevertheless, there are still mislabeled pixels specially in narrow streams, partially inundated crop fields and in the borders of clouds and water bodies. Some of these errors are caused by temporal misalignment, e.g., the closest S2 image may have been acquired some days after the map was produced. This happens, as is frequently the case, if the flood extent map was generated based on a satellite image other than S2. Figure 5 shows, on the left, the satellites used to derive each flood map and on the right, the difference in days between the flood extent map

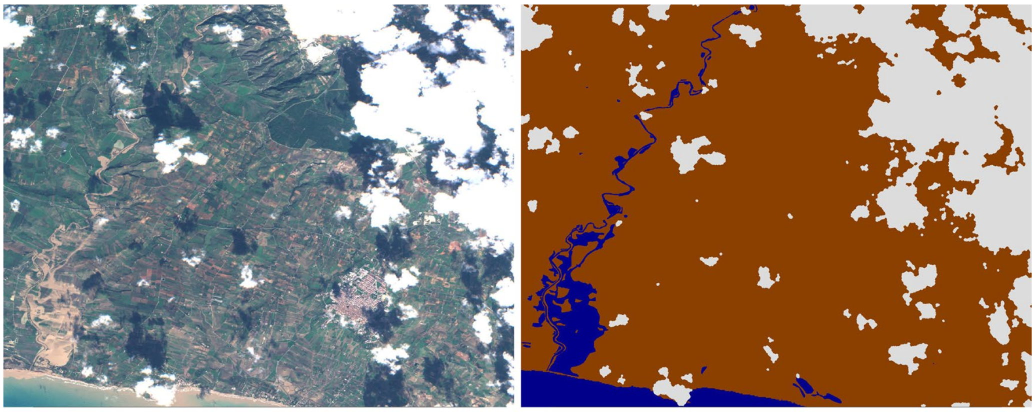


Figure 4. (**a**) Sentinel 2 RGB bands and (**b**) associated labelled map (land/brown, water/blue, cloud/white) over Porto Palo (Sicily) derived from Copernicus EMS 333 activation. Cloud mask obtained automatically with `s2cloudless`[49]. Base image and reference labels are included in the *WorldFloods* database and code for plotting this images may be found in our repository[6].

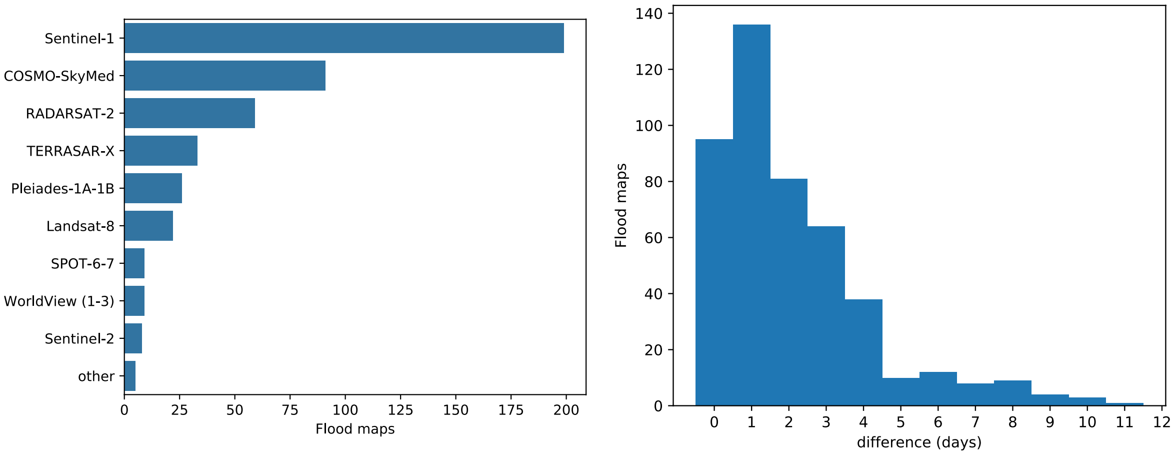


Figure 5. (**a**) Satellite used to derive each flood map in the *WorldFloods* data set. 'Other' satellites (all optical): GeoEye-1, PlanetScope, Earth Observing (EO)-1. (**b**) Difference in days between the flood map and the Sentinel-2 image (Sentinel-2 image is always posterior in time to the flood map).

and the next S2 overpass. As we can see, most of the flood extent maps where generated from radar imagery and most images are acquired within five days which suggests that the earliest available re-visit is used if available.

While including flood extent maps and S2 images from different days introduces label noise, this allows us to use a much larger training set than if we were restricted to images where the flood map was generated from S2. We were motivated by results from the authors of SEN12MS[51] who trained global high resolution (10 m) segmentation models using cloud-free S2 imagery and low resolution labels derived from MODIS (500 m), achieving 63–65% overall accuracy despite the coarseness of the available ground truth labels. In our results section we experimentally validate that this trade-off is justified for our dataset; that is, we achieve better segmentation results on a clean test set when we include these noisy labels in our training than if we restrict the training set to clean images.

Models trained on noisy labels in the training may appear to perform well, but it is important to ensure that the test set provides a clean measurement of the true performance of our system. In this direction, we manually selected test images from flood extent maps that were derived from S2 images which had no temporal misalignment. In addition, we visually inspected those images and fixed minor errors to improve the quality of their segmentation masks. To avoid data leakage, there was no spatial overlap between flood maps in the test set and the training and validation sets. Additionally, other flood extent maps from same flood events in the test set have also been removed from the training and validation sets. Table 1 shows the training, validation and test set

Dataset	Flood events	Flood maps	256x256 patches	Water pixels (%)		Land pixels	Cloud pixels	Invalid pixels
				Flood	Permanent[†]	(%)	(%)	(%)
Training	108	407	182,413	1.45	1.25	43.24	50.25	3.81
Validation	6	6	1132	3.14	5.19	76.72	13.27	1.68
Test	5	11	2029	20.23	1.16	59.05	16.21	3.34

Table 1. General statistics of the training, validation and test splits of the *WorldFloods* dataset. Since raw images from S2 can be many megapixels in size, we tile each image into 256-pixel square patches. The training set distribution has a higher percentage of cloudy pixels compared with the validation and test datasets; this is because we were interested in distinguishing water/flood pixels whereas detecting clouds is a byproduct of the model. [†] Permanent water obtained from the yearly water classification product of Pekel et al.[44] available at the Google Earth Engine[52].

statistics; there is a strong class imbalance in the training dataset with less than 3% of pixels belonging to the water class. From those, less than 50% are classified as permanent water in the JRC permanent water product[44]. The low occurrence of water pixels in the train dataset is because there is a high presence of clouds in the training data. Cloud occurrence in the validation and test sets is lower to provide more meaningful results of flood segmentation.

Results

In order to demonstrate that a FCNN-based flood detection model can segment floods accurately and could be deployed on ΦSat-1, we first train FCNN models on *WorldFloods* at its original resolution (10 m). We then train models on *degraded* imagery, mimicking the resolution of HyperScout-2 (80 m) by resampling the S2 images using bilinear interpolation and also by using only the overlapping bands between the sensors. Afterwards, models trained over the entire *WorldFloods* dataset are compared with models trained using only flood maps derived from Sentinel-2. Finally, we verify our trained (degraded) models can be run on a Intel Movidius Myriad2 chip and measure the processing speed; we use an Intel Neural Compute Stick v1 connected to a Raspberry Pi 3B+. Models tested on the Intel Movidius Myriad2 chip use all available S2 bands, in comparison to the cloud detection model[22] which uses three bands selected using Principle Component Analysis (PCA).

We focus on the segmentation accuracy of the water/flood class by measuring precision, recall and the intersection over union (IoU). Since missing flooded areas (false negatives) is more problematic than over-predicting floods (false positives), high recall is preferred to high precision. In practice the IoU is a good compromise if recall is sufficiently high (over 94%); with a lower recall we found that, even with a high IoU, the model misses entire water bodies in several scenes.

As baselines, we use NDWI (S2 band 3 and 8[24]) and a per-pixel linear model (all S2 bands) trained on *WorldFloods*. A range of NDWI thresholds have been suggested in the literature for flood water extraction[24,25,53]we chose 0 for our experiments since it is the most common one. In order to set a stronger baseline, we also report results for the threshold that maximizes the IoU in the test data providing a recall above 94% (threshold − 0.22). This represents the best case performance for the NDWI model. In addition, in order to strengthen the baseline results, the NDWI model assumes perfect cloud masking by using directly the s2cloudless cloud masking model. We compare our baselines to two FCNNs: a simple CNN (SCNN) comprising four convolutional layers (0.26M parameters) and a U-Net (7.8 M parameters)[54]. Although single-pixel classification methods like NDWI are common, we expect that models which can use larger contextual information, such as the extended shape of water bodies, will perform better. Therefore we calculated the receptive field of our models to ensure that larger features are considered during classification. Our smallest model has a receptive field of 9 × 9 pixels (700 × 700 m) which we judged to be sufficient. Details of our SCNN and UNet architectures can be found in the supplementary material for this paper; additionally our implementation and training code is provided in our GitLab repository[6].

Models were trained from scratch for 40 epochs using all 13 S2 bands with input patches of size 256 × 256 for 10 m data or 64 × 64 for 80 m data (2.5 km × 2.5 km). For data at 80 m resolution we also trained our models using only the 10 overlapping bands between HyperScout-2 and S2 (see Fig. 2). In order to achieve models with high recall we used a cross-entropy loss function that weights each class by the inverse of the observed frequency in Table 1, combined with a Dice loss[55]. Augmentation was applied during training including flips and rotations, per-channel jitter, Poisson (shot) noise and brightness/contrast adjustments. A flowchart showing the training and dataloading process is shown in Fig. 6. Models were tested on full S2 images as described in[56].

Table 2 shows the metrics for the different models and baselines. Specifically, we show IoU and recall for the water class (total water) as well as the recall stratified for flood and permanent water. Permanent water classification comes from the JRC permanent water layer[52]. Our three models (Linear, SCNN and UNet) all have a recall above 94%; NDWI with the threshold at zero generalises poorly, we suspect due to water with suspended matter. FCNN models performed best although there was only a small increase in performance between SCNN and U-Net, despite U-Net having 30 × more parameters. The drop in performance from 10 to 80 m is around two points for FCNN models which is acceptable taking into account that the spatial resolution is eight times worse. There is also a significant drop in performance when only the 10 overlapping bands of HyperScout-2 and S2 are used (bands B1 to B9) suggesting that the short-wave infrared (SWIR) bands of S2 (B10–B12) have high predictive power for water. This is expected since water reflectance is very low in the SWIR whereas soil and

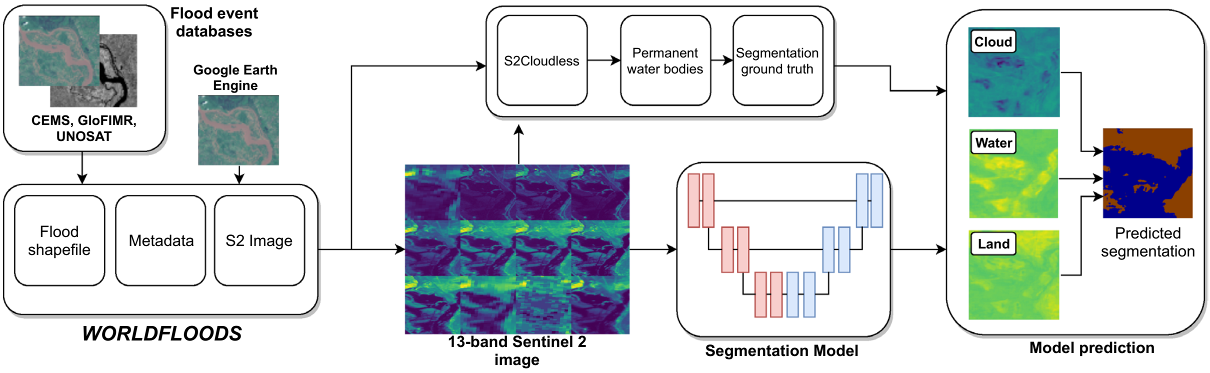


Figure 6. Overview of the model training pipeline used in this work. Note that *WorldFloods* provides images from S2, but reference flood extent maps may have been labelled from other sources, such as radar satellites.

	Model	IoU total water	Recall total water	Recall flood water	Recall permanent water
10 m	NDWI (thres -0.22)	65.12	**95.75**	95.53	**99.70**
	NDWI (thres 0)	39.99	44.84	42.43	86.65
	Linear	64.87	95.55	**95.82**	90.75
	SCNN	71.12	94.09	93.98	95.93
	U-Net	**72.42**	95.42	95.40	95.83
80 m	NDWI (thres -0.22)	64.10	94.76	94.57	98.15
	NDWI (thres 0)	39.07	44.01	41.69	84.55
	Linear	60.90	95.00	94.79	**98.58**
	SCNN	68.87	**96.03**	**96.11**	94.76
	U-Net	**70.22**	94.78	94.85	93.50
80 m HyperScout-2 overlapping bands	NDWI (thres -0.22)	64.10	**94.76**	94.57	**98.15**
	NDWI (thres 0)	39.07	44.01	41.69	84.55
	Linear	50.27	80.47	79.69	94.03
	SCNN	**65.82**	94.62	**95.17**	84.99
	U-Net	65.43	94.59	**95.17**	84.44

Table 2. IoU and recall results for models trained on *WorldFloods*. Bold values indicate highest metric value for each resolution and band combination.

vegetation reflectance is significantly higher[57]. Figure 7 shows the precision and recall for different thresholds on the total water class; again, our trained models beat NDWI and larger models tend to perform better.

Figure 8 shows the results of the models trained on the *WorldFloods* training dataset against models trained on clean S2–labelled data alone (Fig. 8). Results for the clean S2 labeled data have been computed by cross validation leaving one flood event out from the *WorldFloods* test dataset (details on this procedure and results for each flood event can be found in the supplementary material). We found that training using all data was better than training on S2-labelled data alone. Our hypothesis is that although reference labels from non-S2 satellites may be noisier, when considering the dataset in aggregate, this noise becomes less significant as most pixels are labelled correctly. This result also lends support to our argument that temporal misalignment between labels and imagery in our dataset was not significant. Similarly, this robustness should also extend to noisy ground truth which is semi-automatically labelled by humans.

The SCNN model was selected for testing on the Myriad 2 chip due to its similar accuracy, but lower computational footprint, compared to UNet (1 FLOPS vs 2.68 FLOPS for a 64 × 64 × 13 input). Figure 9 shows some example images segmented using the Myriad2. This model segments a 12 MP image—approximately the size acquired by HyperScout-2—in less than one minute, accounting for data transfer between the computer and the accelerator development board via a USB connection. We assume that the power required to downlink data is comparable to that of data processing (2.5 W for the Myriad2). Using a radio with a bandwidth of 10 Mbps, a 1GB image would take 13 minutes to transfer. Therefore we can reduce image transmission power consumption by an order of magnitude at least. On a fully integrated platform like a satellite, we would expect lower latency for data transfer and hence a slightly faster overall processing time.

In general, our models tend to over-predict water content; a common failure mode is to identify dark regions as water. False positives are mostly clustered in the surroundings of water bodies and in cloud shadows (see Fig. 9). For further work we are exploring other methods to improve this, for example by adding another input channel with elevation.

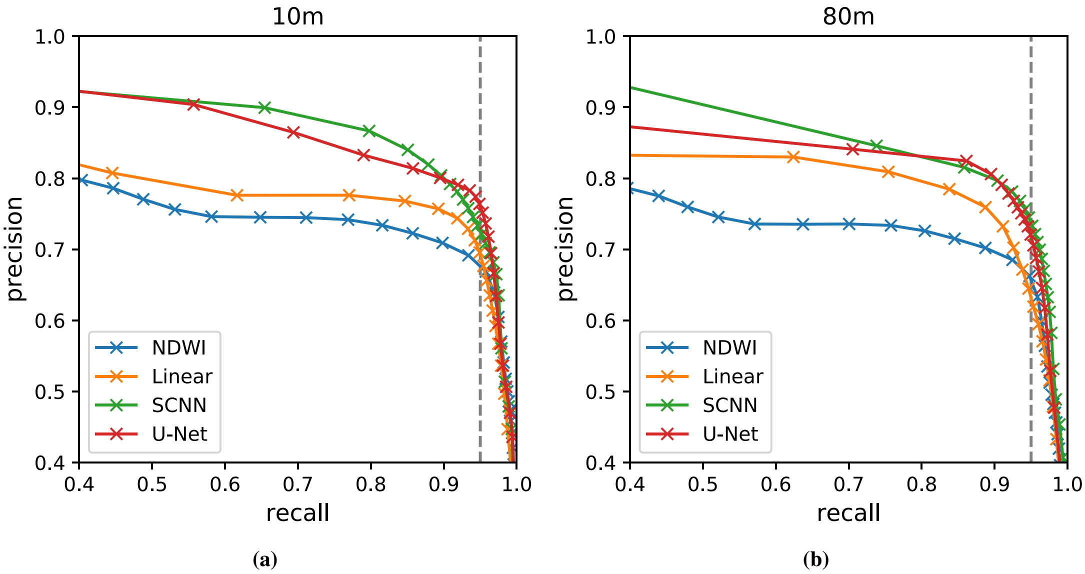


Figure 7. Precision–recall curves of different models trained on (**a**), the Sentinel-2 original resolution (10 m) and (**b**), in the degraded resolution of HyperScout-2 (80 m). In gray 95% recall threshold.

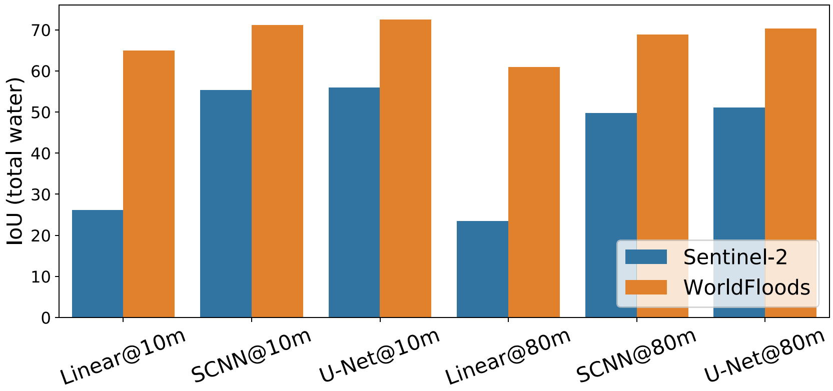


Figure 8. Performance of models trained with all *WorldFloods* flood maps compared with models trained only with flood maps derived from Sentinel-2.

Discussion and conclusions

The current proliferation of open-access satellite data complemented by imagery from commercial satellite operators has still only limited impact on assisting disaster response, primarily because of relatively low revisit times and long delays between image acquisition and product delivery. Here we propose a technical concept study for in-orbit flood mapping using low-cost hardware with machine learning capability to reduce the amount of data required to be downlinked. This concept will enable the use of large cubesat constellation to reliable monitor environmental phenomena such as flooding with high temporal resolution.

We have demonstrated that accurate flood segmentation in orbit is feasible to perform using low resolution images and available hardware. Our models outperform standard baselines and are favourably comparable to human annotation, while being efficiently computable with machine learning hardware on-board the current Φ Sat-1 technology demonstrator as well as future missions.

Recent works[58,59] have shown good performance of spectral indices such as NDWI for water detection on specific survey areas. In our experiments we see that our "best case" tuned NDWI results are also a strong baseline. However there are still examples where a fixed threshold in an image will incorrectly retrieve buildings and cloud shadows as water. Therefore we expect NDWI to perform well in some cases (in our dataset, Finland, for example) and poorly in others, which is perhaps reflected in our aggregated results (see table 3 in supplementary materials for the results for each flood event). Compared to previous works on flood detection[33,45], we have reported results on a wide range of geographical areas paying special attention to data leakage[60]. For our

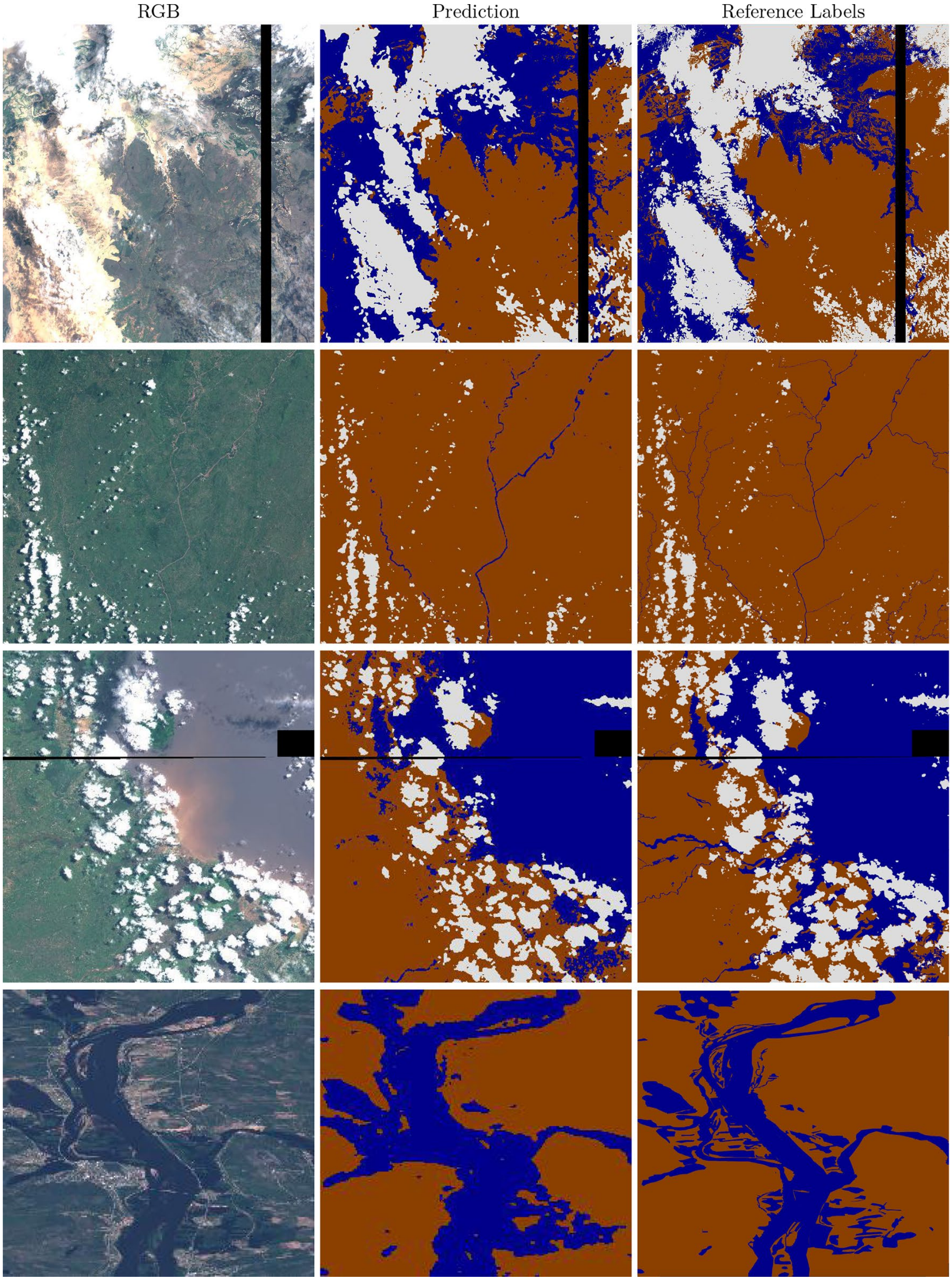


Figure 9. Segmentation results of degraded models (SCNN 80 m) run on Myriad 2 device. Sentinel 2 imagery and Copernicus EMS mapping products are provided as public domain. Base images and reference labels are included in the *WorldFloods* database and code for plotting these images may be found in our repository[6]. Colours are as follows: brown/land, blue/water, white/cloud.

application global generalisation is critical since its intended use is to automatically provide segmentation masks instead of heavier hyper-spectral images.

Downlinking only segmentation masks instead of complete images is not exempt from drawbacks. Firstly, the quality of the downloaded data only depends on the accuracy of the model. In other words, an erroneous segmentation can not be fixed on the ground since the original hyperspectral information is lost. This could be alleviated by periodically downlinking full images to assess and improve the segmentation algorithm's quality. The newly gained data could be added to the training dataset or even apply domain adaptation[61] to boost the segmentation networks. Secondly, by discarding the image, we lose information that could be used for advanced analysis. Hyperspectral information could be used to assess the presence of pollutants in the flood water. In this case, the segmentation masks could be used to guide the retrieval of relevant pixels. Guiding the retrieval of cloud free images is the current operational application onboard the ΦSat-1 satellite[22].

One of the contributions of this work is the release of the *WorldFloods* database alongside this paper, which we hope will serve as a useful tool to foster further research in disaster response. We are pleased to write that this approach is being increasingly explored - while this work was being prepared for publication, several other 'machine learning' ready datasets for segmentation from satellite imagery have been published; Rambour et. al.[62] demonstrated flood detection on time series of SAR and optical data, making their dataset publicly available, Bonafilia et. al.[63], who focus on Sentinel 1 data, but provide more detailed labels that we had available to us here and Nemni et al.[64] who has also made their dataset publicly accesible. The approach we explore here, of producing a 'machine learning ready' dataset as well as a concrete algorithm, has also been recently explored for other areas of disaster response[65], and we hope to see this continue.

Data availability

We are releasing the *WorldFloods* database alongside this paper at https://tinyurl.com/worldfloods. Users of this dataset should be aware of the varying quality of the reference labels that is pointed out in the paper; specifically some labels in the training and validation datasets have significant errors. In general the quality of the test dataset labels are higher and test images were curated to facilitate more accurate model evaluation. We hope to address any remaining label quality issues in future work. We provide a GitLab repository with our model architectures, model checkpoints and training/benchmarking code at: https://gitlab.com/frontierdevelopmentlab/disaster-prevention/cubesatfloods.

Received: 2 August 2020; Accepted: 4 March 2021
Published online: 31 March 2021

References

1. United Nations. *Global Assessment Report on Disaster Risk Reduction 2015* (United Nations International Strategy for Disaster Reduction, 2015).
2. Centre for Research on the Epidemiology of Disasters. *The human cost of weather-related disasters 1995-2015* (United Nations Office for Disaster Risk Reduction, 2015).
3. Serpico, S. B. *et al.* Information extraction from remote sensing images for flood monitoring and damage evaluation. *Proc. IEEE* **100**, 2946–2970. https://doi.org/10.1109/JPROC.2012.2198030 (2012).
4. Schumann, G.J.-P., Brakenridge, G. R., Kettner, A. J., Kashif, R. & Niebuhr, E. Assisting flood disaster response with earth observation data and products: a critical assessment. *Remote Sens.* **10**, 1230. https://doi.org/10.3390/rs10081230 (2018).
5. United Nations. *Global Assessment Report on Disaster Risk Reduction 2019* (United Nations International Strategy for Disaster Reduction, 2019).
6. WorldFloods GitLab repository. https://gitlab.com/frontierdevelopmentlab/disaster-prevention/cubesatfloods. Accessed: 2020-12-08.
7. International Charter "Space and Major Disasters". https://disasterscharter.org/. Accessed: 2020-06-15.
8. Havas, C. *et al.* E2mc: improving emergency management service practice through social media and crowdsourcing analysis in near real time. *Sensors* **17**, 2766 (2017).
9. Berger, M., Moreno, J., Johannessen, J. A., Levelt, P. F. & Hanssen, R. F. ESA's Sentinel missions in support of Earth system science. *Remote Sens. Environ.* **120**, 84–90 (2012).
10. Drusch, M. *et al.* Sentinel-2: ESA's optical high-resolution mission for GMES operational services. *Remote Sens. Environ.* **120**, 25–36 (2012).
11. Heidt, H., Puig-Suari, J., Moore, A., Nakasuka, S. & Twiggs, R. CubeSat: A new generation of picosatellite for education and industry low-cost space experimentation. In *14th Annual/USU Conference on Small Satellites* (2000).
12. Esposito, M., Conticello, S., Pastena, M. & Domínguez, B. C. In-orbit demonstration of artificial intelligence applied to hyperspectral and thermal sensing from space. In *CubeSats and SmallSats for Remote Sensing III*, vol. 11131, 111310C (International Society for Optics and Photonics, 2019).
13. Manzillo, P. F. *et al.* Hyperspectral imaging for real time land and vegetation inspection. In *The 4S Symposium* (2017).
14. Estlin, T. A. *et al.* AEGIS Automated Science Targeting for the MER Opportunity Rover. *ACM Trans. Intell. Syst. Technol.*https://doi.org/10.1145/2168752.2168764 *(2012).*
15. Francis, R. *et al.* AEGIS autonomous targeting for ChemCam on Mars Science Laboratory: deployment and results of initial science team use. *Sci. Robot.*https://doi.org/10.1126/scirobotics.aan4582 *(2017).*
16. Griggin, M., Burke, H., Mandl, D. & Miller, J. Cloud cover detection algorithm for EO-1 Hyperion imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2003)*, vol. 1, 86–89 vol.1, https://doi.org/10.1109/IGARSS.2003.1293687 (2003).
17. Doggett, T. *et al.* Autonomous detection of cryospheric change with hyperion on-board earth observing-1. *Remote Sens. Environ.* **101**, 447–462. https://doi.org/10.1016/j.rse.2005.11.014 (2006).
18. Ip, F. *et al.* Flood detection and monitoring with the autonomous sciencecraft experiment onboard eo-1. *Remote Sens. Environ.* **101**, 463–481. https://doi.org/10.1016/j.rse.2005.12.018 (2006).
19. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
20. Camps, A. *et al.* FSSCAT, the 2017 Copernicus Masters' "ESA Sentinel Small Satellite Challenge" Winner: A Federated Polar and Soil Moisture Tandem Mission Based on 6U Cubesats. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2018)*, 8285–8287 (IEEE, 2018).
21. Esposito, M. *et al.* Hyperscout 2 highly integration of hyperspectral and thermal infrared technologies for a miniaturized eo imager. In *Living Planet Symposium*, https://doi.org/10.13140/RG.2.2.25659.67367 (2019).

22. Giuffrida, G. *et al.* Cloudscout: a deep neural network for on-board cloud detection on hyperspectral images. *Remote Sens.*https://doi.org/10.3390/rs12142205 *(2020).*
23. Smith, S. W. *The Scientist and Engineer's Guide to Digital Signal Processing (Chapter 27)* (California Technical Publishing, California, 1997).
24. McFeeters, S. K. The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. *Int. J. Remote Sens.* **17**, 1425–1432. https://doi.org/10.1080/01431169608948714 (1996).
25. Memon, A. A., Muhammad, S., Rahman, S. & Haq, M. Flood monitoring and damage assessment using water indices: a case study of pakistan flood-2012. *Egypt. J. Remote Sens. Space Sci.* **18**, 99–106. https://doi.org/10.1016/j.ejrs.2015.03.003 (2015).
26. Obserstadler, R., Hönsch, H. & Huth, D. Assessment of the mapping capabilities of ers-1 sar data for flood mapping: a case study in germany. *Hydrol. Process.* **11**, 1415–1425 (1997).
27. Twele, A., Cao, W., Plank, S. & Martinis, S. Sentinel-1-based flood mapping: a fully automated processing chain. *Int. J. Remote Sens.* **37**, 2990–3004 (2016).
28. Martinis, S. *et al.* Comparing four operational sar-based water and flood detection approaches. *Int. J. Remote Sens.* **36**, 3519–3543 (2015).
29. Stringham, C. *et al.* The capella x-band sar constellation for rapid imaging. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2019)*, 9248–9251, https://doi.org/10.1109/IGARSS.2019.8900410 (2019).
30. Isikdogan, F., Bovik, A. C. & Passalacqua, P. Surface water mapping by deep learning. *IEEE J. Select. Topics Appl. Earth Obser. Remote Sens.* **10**, 4909–4918 (2017).
31. Rudner, T. *et al.* Multi3net: segmenting flooded buildings via fusion of multiresolution, multisensor, and multitemporal satellite imagery. *Proc. AAAI Conf. Artif. Intell.* **33**, 702–709 (2019).
32. Isikdogan, L. F., Bovik, A. & Passalacqua, P. Seeing through the clouds with DeepWaterMap. *IEEE Geosci. Remote Sens. Lett.*https://doi.org/10.1109/LGRS.2019.2953261 *(2019).*
33. Wieland, M. & Martinis, S. A modular processing chain for automated flood monitoring from multi-spectral satellite data. *Remote Sens.* **11**, 2330. https://doi.org/10.3390/rs11192330 (2019).
34. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F. & Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. *Proceedings of the European conference on computer vision (ECCV)* **801–818**, (2018).
35. Mandanici, E. & Bitelli, G. Preliminary comparison of sentinel-2 and landsat 8 imagery for a combined use. *Remote Sens.* **8**, 1014 (2016).
36. Green, R. O. *et al.* Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS). *Remote Sens. Environ.* **65**, 227–248 (1998).
37. Pearlman, J. S. *et al.* Hyperion, a space-based imaging spectrometer. *IEEE Trans. Geosci. Remote Sens.* **41**, 1160–1173 (2003).
38. Esposito, M. & Marchi, A. Z. In-orbit demonstration of the first hyperspectral imager for nanosatellites. In *International Conference on Space Optics-ICSO 2018*, vol. 11180, 1118020 (International Society for Optics and Photonics, 2019).
39. Vane, G. *et al.* The airborne visible/infrared imaging spectrometer (AVIRIS). *Remote Sens. Environ.* **44**, 127–143 (1993).
40. Wieland, M., Li, Y. & Martinis, S. Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network. *Remote Sens. Environ.* **230**, 111203. https://doi.org/10.1016/j.rse.2019.05.022 (2019).
41. Mateo-García, G., Laparra, V., López-Puigdollers, D. & Gómez-Chova, L. Transferring deep learning models for cloud detection between Landsat-8 and Proba-V. *ISPRS J. Photogr. Remote Sens.* **160**, 1–17. https://doi.org/10.1016/j.isprsjprs.2019.11.024 (2020).
42. Simard, P. Y., Steinkraus, D. & Platt, J. C. Best practices for convolutional neural networks applied to visual document analysis. *Proceedings of the Seventh International Conference on Document Analysis and Recognition -* **2**, (2003).
43. Ding, J., Chen, B., Liu, H. & Huang, M. Convolutional neural network with data augmentation for sar target recognition. *IEEE Geosci. Remote Sens. Lett.* **13**, 364–368 (2016).
44. Pekel, J.-F., Cottam, A., Gorelick, N. & Belward, A. S. High-resolution mapping of global surface water and its long-term changes. *Nature* **540**, 418–422. https://doi.org/10.1038/nature20584 (2016).
45. Schumann, G.J.-P. The need for scientific rigour and accountability in flood mapping to better support disaster response. *Hydrol. Process.* **1**, (2019).
46. Copernicus Emergency Management System. https://emergency.copernicus.eu/. Accessed: 2019-09-15.
47. UNOSAT. http://floods.unosat.org/geoportal/catalog/main/home.page. Accessed: 2019-09-15.
48. Global Flood Inundation Map Repository. https://sdml.ua.edu/glofimr/. Accessed: 2019-09-15.
49. s2cloudless: Sentinel Hub's cloud detector for Sentinel-2 imagery. https://github.com/sentinel-hub/sentinel2-cloud-detector. Accessed: 2019-09-15.
50. Gorelick, N. *et al.* Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sens. Environ.* **202**, 18–27. https://doi.org/10.1016/j.rse.2017.06.031 (2017).
51. Schmitt, M., Hughes, L. H., Qiu, C. & Zhu, X. X. SEN12MS-a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion. *ISPRS Ann. Photogr. Remote Sens. Spat. Inf. Sci.***IV–2/W7**, 153–160. https://doi.org/10.5194/isprs-annals-IV-2-W7-153-2019 (2019).
52. JRC Yearly Water Classification. https://developers.google.com/earth-engine/datasets/catalog/JRC_GSW1_1_YearlyHistory. Accessed: 2021-01-31.
53. McFeeters, S. K. Using the normalized difference water index (NDWI) within a geographic information system to detect swimming pools for mosquito abatement: a practical approach. *Remote Sens.* **5**, 3544–3561. https://doi.org/10.3390/rs5073544 (2013).
54. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234–241 (Springer, 2015).
55. Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S. & Cardoso, M. J. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 240–248 (Springer, 2017).
56. Huang, B., Reichman, D., Collins, L. M., Bradbury, K. & Malof, J. M. Tiling and Stitching Segmentation Output for Remote Sensing: Basic Challenges and Recommendations. arXiv:1805.12219 [cs] (2019).
57. Jones, J. W. Improved automated detection of subpixel-scale inundation-revised dynamic surface water extent (DSWE) partial surface water tests. *Remote Sens.* **11**, 374. https://doi.org/10.3390/rs11040374 (2019).
58. Ahmad, S. K., Hossain, F., Eldardiry, H. & Pavelsky, T. M. A fusion approach for water area classification using visible, near infrared and synthetic aperture radar for south asian conditions. *IEEE Trans. Geosci. Remote Sens.* **58**, 2471–2480. https://doi.org/10.1109/TGRS.2019.2950705 (2020).
59. Cooley, S. W., Smith, L. C., Stepan, L. & Mascaro, J. tracking dynamic northern surface water changes with high-frequency planet CubeSat imagery. *Remote Sens.* **9**, 1306. https://doi.org/10.3390/rs9121306 (2017).
60. Ploton, P. *et al.* Spatial validation reveals poor predictive performance of large-scale ecological mapping models. *Nat. Commun.* **11**, 4540. https://doi.org/10.1038/s41467-020-18321-y (2020).
61. Mateo-Garcia, G., Laparra, V., Lopez-Puigdollers, D. & Gomez-Chova, L. Cross-sensor adversarial domain adaptation of Landsat-8 and Proba-V images for cloud detection. *IEEE J. Selected Top. Appl. Earth Obser. Remote Sens.*https://doi.org/10.1109/JSTARS.2020.3031741 *(2020).*
62. Rambour, C. *et al.* Flood detection in time series of optical and sar images. *Int. Arch. Photogr. Remote Sens. Spat. Inf. Sci.* **43**, 1343–1346 (2020).

63. Bonafilia, D., Tellman, B., Anderson, T. & Issenberg, E. Sen1Floods11: A georeferenced dataset to train and test deep learning flood algorithms for sentinel-1. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* **210–211**, (2020).
64. Nemni, E., Bullock, J., Belabbes, S. & Bromley, L. Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery. *Remote Sens.***12**, 2532, https://doi.org/10.3390/rs12162532 (2020)
65. Gupta, R. *et al.* Creating xBD: A dataset for assessing building damage from satellite imagery. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* **10–17**, (2019).
66. Mateo-Garcia, G. *et al.* Flood detection on low cost orbital hardware. arXiv preprint arXiv:1910.03019 (2019).

Acknowledgements

This research was conducted at the Frontier Development Lab (FDL), Europe, 2019 led by Trillium Technologies. The authors gratefully acknowledge support from European Space Agency (ESA); Pierre-Philippe Mathieu and the team at PhiLab at ESA ESRIN; Google LLC; Intel Corporation; Kellogg College, University of Oxford; UNICEF representative Dohyung Kim.; James Parr, Jodie Hughes, Leo Silverberg and Sarah McGeehan from Trillium Technologies and other organisations and mentors who supported FDL Europe 2019. G.M.-G. has been supported by the Spanish Ministry of Science and Innovation (Projects TEC2016-77741-R and PID2019-109026RB-I00, ERDF). S.O. is supported by the EPSRC Grant EP/L016427/1 and The Financial Times. A.G.B. is supported by EPSRC/MURI Grant EP/N019474/1 and by Lawrence Berkeley National Lab. J.V.M. is supported by STFC/GCRF Grant ST/R002673/1.

Author contributions

G.M.-G., J.V.M. and L.S. equally contributed to the overall research. G.M.-G. was responsible for dataset creation and model performance analysis; S.O., L.S. and J.V.M. developed the model training framework, L.S. also performed testing on embedded hardware. D.B and A.G.B. were the main domain and machine learning supervisors and were responsible for the conception and guidance of the project, supported by G.S. and Y.G.; they provided advice on flooding, remote sensing and machine learning aspects; A.G.B. also contributed to the codebase. All authors reviewed and contributed to the manuscript.

Duplicate publication statement

A short summary of some material in this paper was previously presented at the Humanitarian Aid and Disaster Response (HADR) workshop at NeurIPS 2019 (Vancouver)[66]. The pre-print was peer-reviewed for inclusion in the workshop, which is not archival and does not form part of the NeurIPS conference proceedings. This paper provides substantially more information and addresses several aspects of the dataset that were not discussed in the workshop paper (such as the the effects of training on S2 only, the effect of temporal misalignment and stratification of results on flood and permanent water). We have also extensively curated the training and testing sets and we include a more thorough discussion of the context of our work, flood mapping, and the satellite we have targeted for deployment. Alongside this paper, we also release the *WorldFloods* dataset as well as the code used to train and benchmark our models.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-021-86650-z.

Correspondence and requests for materials should be addressed to G.M.-G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.