University of South Brittany, Faculty of Science, Department of Geoinformatics

Paris Lodron University Salzburg, Faculty of Natural Sciences, Department of Geoinformatics

HOW RELIABLE ARE SENTINEL-2 CLOUD DETECTION ALGORITHMS?: GLOBAL UNCERTAINTY ESTIMATION WITH GAUSSIAN PROCESSES.

Diploma thesis

Author
Cesar Luis Aybar Camacho

Supervisor (University of South Brittany)
Prof. Francois Septier

Co-supervisor (Paris Lodron University Salzburg)
Prof. Dirk Tiede

Erasmus Mundus Joint Master Degree Programme
Copernicus Master in Digital Earth

Specialization track GeoData Science
Vannes, France, 2022

COPERNICUS MASTER IN DIGITAL EARTH

With the support of the Erasmus+ Programme of the European Union

Prof. Bram Leo Willems, who without expecting anything in return gifted me my first computer ten years ago.

Abstract

Cloud detection (CD) is one of the most critical metadata filters for searching, selecting, and accessing imagery in Earth Observation (EO) platforms. In recent years, the extensive archive of EO datasets has boosted the use of data-driven algorithms to improve cloud and cloud-shadow detection. However, data-driven algorithms require large manually annotated datasets, which are expensive and time-consuming to collect. The first chapter of this thesis introduce CloudSEN12, a new multi-temporal global dataset created exploiting different EO datasets offered by the Copernicus program. CloudSEN12 has 49,400 image patches, including (1) Sentinel-2 level-1C and level-2A multi-spectral data, (2) Sentinel-1 synthetic aperture radar data, (3) auxiliary remote sensing products, (4) different hand-crafted annotations to label the presence of clouds and cloud shadows, and (5) the results from eight state-of-the-art cloud detection algorithms. At present, CloudSEN12 exceeds all previous efforts in terms of annotation richness, scene variability, metadata complexity, control quality, data distribution and size. In the second part, cloudSEN12 is used to establish the current state of the art in cloud detection and cloud cover estimation for Sentinel-2 imagery. Furthermore, we proposed directly estimating cloud cover using both a simple ResNet-18 and a single forward pass uncertainty model. The results show that cloudSEN12 increases the efficiency of data-driven algorithms by at least 20%. In addition, for the first time we analyze how credible is the estimation of uncertainty in cloud detection models.

KEYWORDS

cloud detection, deep learning, U-Net, gaussian process, non-stationary.
Number of pages: 56
Number of appendices: 5

Declaration

This thesis has been composed by Cesar Luis Aybar Camacho for the Erasmus Mundus Joint Master's Degree Program in Copernicus Master in Digital Earth for the academic year 2021/2022 at the Department of Geoinformatics, Faculty of Natural Sciences, Paris Lodron University Salzburg, and Department of Geoinformatics, Faculty of Science, Southern Brittany University.

Hereby, I declare that this piece of work is entirely my own, the references cited have been acknowledged and the thesis has not been previously submitted to the fulfillment of the higher degree.

Cesar Aybar
Southern Brittany, France
28 March 2022

Acknowledgements

I would like to express my gratitude to my supervisors, Prof. Francois Septier and Prof. Dirk Tiede, for their constant help and encouragement. From the start of the project to the end, your guidance, mentoring, and support cleared the way for me to successfully complete this dissertation. The computational requirements for this research were partially covered by the Google Cloud Credits Research Grant Program. Besides, the Radiant Earth Foundation provides us with a space to store the dataset. This work was also partially supported by the Spanish Ministry of Science and Innovation (project PID2019-109026RB-I00, ERDF), the Austrian Space Applications Programme within the SemantiX project (#878939, ASAP 16), and the Linux Foundation Grant projects (project 21-ISC-1-1). The following R and Python packages were used in the course of this investigation and I would like to acknowledge their developers: rgee (Aybar et al. 2020), sf (Pebesma 2018), raster (Hijmans et al. 2015), stars (Pebesma 2020), numpy (Harris et al. 2020), lubridate (Grolemund and Wickham 2011), reticulate (Ushey et al. 2020), dplyr (Wickham et al. 2014), tmap (Tennekes 2018), magick(Ooms 2020), rgeos (Bivand et al. 2017) and ggplot2(Wickham 2011). Finally, I would like to thank B.S. Joselyn Inga and Wendy Espinoza for their work reporting manual labeling errors in the quality control phase of the dataset.

Cesar Aybar
Southern Brittany, France
28 March 2022

Contents

List of Figures

List of Tables

List of Abbreviations

1-D, 2-D . . . . One- or two-dimensional, referring **in this thesis** to spatial dimensions in an image.

Otter . . . . . One of the finest of water mammals.

Hedgehog . . . Quite a nice prickly friend.

What am I in the eyes of most people - a nonentity, an eccentric or an unpleasant person - somebody who has no position in society and never will have, in short, the lowest of the low. All right, then - even if that were absolutely true, then I should one day like to show by my work what such an eccentric, such a nobody, has in his heart.

— Vincent Van Gogh - 1882

1

CloudSEN12 - a global benchmark dataset for cloud semantic understanding

Contents

1.1 Introduction

We are in the midst of an exciting new era of Earth observation (EO), wherein Analysis Ready Data (ARD) (Mahecha et al. 2020; Giuliani et al. 2019; Gomes et al. 2020) products derived from optical satellite big imagery catalogs permit direct analyses without laborious pre-processing. Unfortunately, much of these

products are contaminated by clouds (Wilson and Jetz 2016) and their associated shadows, altering the surface reflectance values and hampering their operational exploitation at large scales. For most of the applications exploiting ARD, cloud and cloud-shadow pixels need to be removed prior to further analyses, i.e. masked out, to avoid distortions in the results.

Improving the accuracy of existing cloud detection (CD) algorithms is a pressing need for the EO community regarding optical sensors such as Sentinel-2. Ideally, CD algorithms classify pixels into clear, cloud shadow, thin cloud, and thick cloud. Splitting clouds into two subclasses allows downstream applications to design different strategies to treat cloud contamination. On the one hand, thick clouds entirely block the surface's view, reflecting most of the light coming from the sun and generating gaps impossible to retrieve using optical sensor data (Ebel et al. 2020). On the other hand, thin clouds do not reflect all the sunlight allowing to observe a distorted view of the surface (Lynch et al. 2002; Chen et al. 2017). For some applications, such as object detection or disaster response (Mateo-Garcia et al. 2021), images contaminated with thin clouds are still helpful. Therefore, distinguishing between thick and thin clouds is a critical first step towards optical data exploitation. Nevertheless, it is worth noting that there is no overall consensus on quantitative approaches delimiting when one class begins and the other ends; thus, it is so far inherently subjective to the image interpreter (Qiu, Zhu, and Woodcock 2020; Foga et al. 2017).

Methodologies for CD can be classified into two main categories: knowledge-driven (KD) and data-driven (DD). KD emphasizes the logical sense connected with physical foundations. For instance, the Function of mask (Fmask, Qiu, Zhu, and He 2019) and Sen2Cor (Qiu, Zhu, and He 2019) use a set of physical rules formulated on spectral and contextual features to distinguish clouds against water or land. Overall, KD algorithms achieve accurate results and good generalization (Sanchez et al. 2020; Zekoll et al. 2021; Cilli et al. 2020). However, it is well-known that they have problems associated with thin cloud omission and non-cloud object commission, frequently at cloud edges and under surfaces with a smooth texture or high reflectance (Melchiorre et al. 2020; Stillinger et al. 2019).

In recent years, supervised data-driven strategies, trained in large manually annotated datasets, have grown notoriety in remote sensing thanks to the success of classical machine learning (ML) and deep learning (DL) techniques (Zhu, Tuia, et al. 2017). Among multiple noteworthy ML precedents (Wei et al. 2020; Bai et al. 2016; Ghasemian and Akhoondzadeh 2018), Sentinel Hub's s2cloudless (Zupanc 2017) is the most extensively used due to its low computational requirements and lightweight design. Nonetheless, when evaluated in certain particular regions, such as tropical forests, s2cloudless falls short of *state-of-the-art* KD cloud detectors (Sanchez et al. 2020; López-Puigdollers et al. 2021; Skakun et al. 2022). Meanwhile, DL has proven to be more effective on CD compared to more classical ML (Li, Li, et al. 2021; Mahajan and Fataniya 2020), although it is subjected to the exigency of pixel-level annotation.

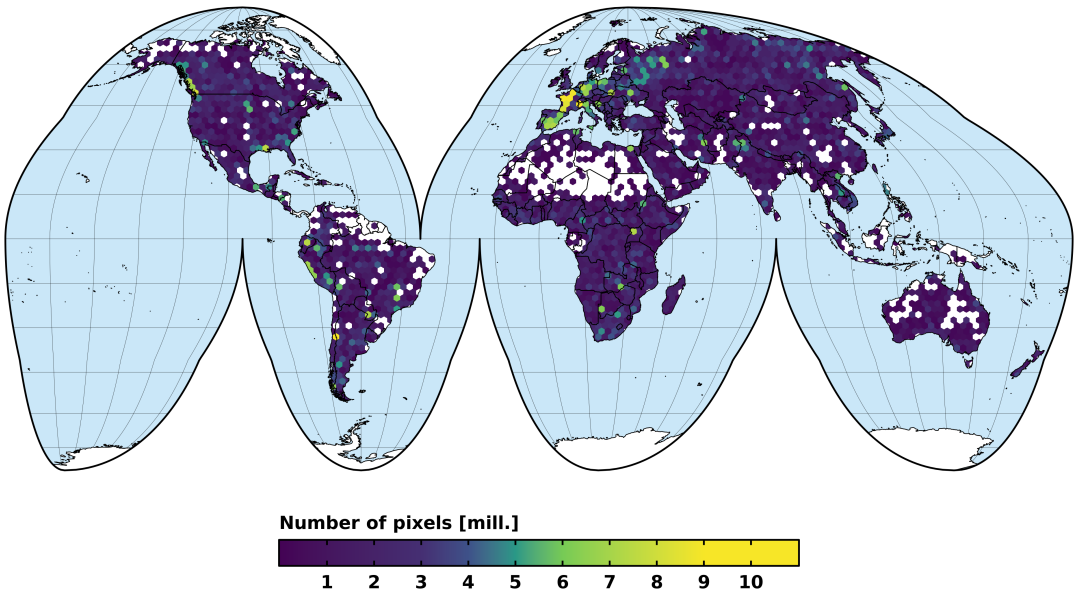

Figure 1.1: CloudSEN12 spatial coverage, purple-to-yellow color gradient represents the amount of hand-crafted annotated pixels. In total in cloudSEN12 there are image patches from 10,000 different locations with five different images for each of them.

The recent progress in DL-based cloud semantic segmentation can be attributed to the proliferation of public cloud semantic segmentation datasets such as SPARCS (Hughes and Kennedy 2019, S2-Hollstein Hollstein et al. 2016, Biome 8Foga et al. 2017), 38-cloud (Mohajerani and Saeedi 2019), BaetensHagolle (Baetens et al. 2019), 95-Cloud (Mohajerani and Saeedi 2020), and CloudCatalogue (Francis et al. 2020). Nonetheless, these datasets have some well-known shortcomings, including the absence of temporal features, a lack of thin clouds or cloud shadows

labels, a high degree of class imbalance, and a relatively small size joined with geographical bias (see Table 1.1 for the current characteristics/limitations of each of those datasets). Furthermore, their quality control process is not always properly described and their development remains unclear. These flaws hinder the natural transition to global DL cloud classifiers and the application of new-fashioned strategies such as few-shot learning, where model parameters can be adapted across geographies (Rußwurm et al. 2020).

Table 1.1: Summary of publicly available CD datasets in comparison to CloudSEN12. An asterisk represents that the dataset does not distinguish the specific class.

Name	Main region	Labels	# of Scenes	Temporal	# of Pixels (10^9)	Thick Clouds %	Thin Clouds %	Cloud Shadows %	Clear %
L8-SPARCS	worldwide	full-scene	80	No	0.080	19.37	*	7.37	73.26
S2-Hollstein	Europe	polygons	59	No	0.003	16.06	16.49	4.53	62.92
L8-Biome8	worldwide	full-scene	96	No	3.964	33.19	14.71	1.55	50.55
L8-38Cloud	USA	full-scene	38	No	1.494	52.36	*	*	47.64
S2-BaetensHagolle	Europe	full-scene	38	No	0.109	22.77+	*	2.71	74.52
L8-95Cloud	USA	full-scene	95	No	3.737	49.27	*	*	50.73
S2-cloudCatalog	worldwide	partial scene	513	No	0.535	52.58	*	1.47	45.95
CloudSEN12	**worldwide**	**partial scene**	**46697**	**Yes**	**4.5**	**xxx**	**xxx**	**xxx**	**xxx**

+ Low and high cloud classes were aggregated.

Inspired by the CityScapes dataset (Cordts et al. 2016), we created and release CloudSEN12, a large and globally distributed dataset (Figure 1.1) for cloud semantic understanding. CloudSEN12 surpasses all previous efforts in size and variability (Figure 1.2) offering 49,250 image patches (IPs) with different annotation types: (i) 10,000 IPs with high-quality pixel-level annotation, (ii) 10,000 IPs with scribble annotation, and (iii) 29,250 unlabeled IPs. The labeling phase was conducted by 14 domain experts using a supervised active learning system. To guarantee high quality in manual annotation, we designed a rigorous four-step quality control protocol based on Zhu, Hu, et al. 2019. Furthermore, CloudSEN12 ensures that for the same geographical location, users can obtain multiple IPs with different cloud coverage: cloud-free (0%), almost-clear (0-25%), low-cloudy (25-45%), mid-cloudy (45-65%), and cloudy (>65%), which ensures scene variability in the temporal domain. Finally, in order to support multi-modal cloud removalMeraner et al. 2020 and data fusionSingh and Komodakis 2018 approaches, each CloudSEN12 IP includes data from a variety of remote sensing sources that have already shown their usefulness in cloud and cloud shadow masking. See Table 1.2 for a full list of assets available for each image patch.

Table 1.2: List of assets available for each image patch.

File / Folder	Name	Scale	Wavelength	Description
S2L1C & S2L2A	B1	0.0001	443.9nm (S2A) / 442.3nm (S2B)	Aerosols.
	B2	0.0001	496.6nm (S2A) / 492.1nm (S2B)	Blue.
	B3	0.0001	560nm (S2A) / 559nm (S2B)	Green.
	B4	0.0001	664.5nm (S2A) / 665nm (S2B)	Red.
	B5	0.0001	703.9nm (S2A) / 703.8nm (S2B)	Red Edge 1.
	B6	0.0001	740.2nm (S2A) / 739.1nm (S2B)	Red Edge 2.
	B7	0.0001	782.5nm (S2A) / 779.7nm (S2B)	Red Edge 3.
	B8	0.0001	835.1nm (S2A) / 833nm (S2B)	NIR.
	B8A	0.0001	864.8nm (S2A) / 864nm (S2B)	Red Edge 4.
	B9	0.0001	945nm (S2A) / 943.2nm (S2B)	Water vapor.
	B11	0.0001	1613.7nm (S2A) / 1610.4nm (S2B)	SWIR 1.
	B12	0.0001	2202.4nm (S2A) / 2185.7nm (S2B)	SWIR 2.
S2L1C	B10	0.0001	1373.5nm (S2A) / 1376.9nm (S2B)	Cirrus.
S2L2A	AOT	0.001	-	Aerosol Optical Thickness.
	WVP	0.001	-	Water Vapor Pressure.
	TCI_R	1	-	True Color Image, Red.
	TCI_G	1	-	True Color Image, Green.
	TCI_B	1	-	True Color Image, Blue.
S1	VV	1	5.405GHz	Dual-band cross-polarization, vertical transmit/horizontal receive.
	VH	1	5.405GHz	Single co-polarization, vertical transmit/vertical receive.
	angle	1	-	Incidence angle generated by interpolating the 'incidenceAngle' property.
extra/	CDI	0.0001	-	Cloud Displacement Index.
	Shwdirection	0.01	-	Direction of cloud shadows. Values range from 0°- 360°.
	elevation	1	-	Elevation in meters. Obtained from MERIT Hydro datasets.
	ocurrence	1	-	JRC Global Surface Water. The frequency with which water was present.
	LC100	1	-	Copernicus land cover product. CGLS-LC100 Collection 3.
	LC10	1	-	ESA WorldCover 10m v100 product.
labels/	fmask	1	-	Fmask4.0 cloud masking.
	QA60	1	-	SEN2 Level-1C cloud mask.
	s2cloudless	1	-	sen2cloudless results.
	sen2cor	1	-	Scene Classification band. Obtained from SEN2 level 2A.
	DL_L8S2_UV_rgbi	1	-	López-Puigdollers et al. 2021 results based on RGBI bands.
	DL_L8S2_UV_rbgiswir	1	-	López-Puigdollers et al. 2021 results based on RGBISWIR bands.
	kappamask_L1C	1	-	KappaMask results using SEN2 level L1C as input.
	kappamask_L2A	1	-	KappaMask results using SEN2 level L2A as input.
	manual_hq	1		High-quality pixel-wise manual annotation.
	manual_sc	1		Scribble manual annotation.

1.2 Methods

This study starts by collecting and combining several public data sources that potentially may help us better comprehend cloud and cloud shadow semantics. Based on this information, semantic classes (Table~1.3) are created using an active system that blends human photo-interpretation and machine learning. Finally, a strict quality control protocol is carried out to ensure the highest quality on the manual labels and to standardize human-level performance. Figure 1.3 depicts the workflow followed to create the dataset.

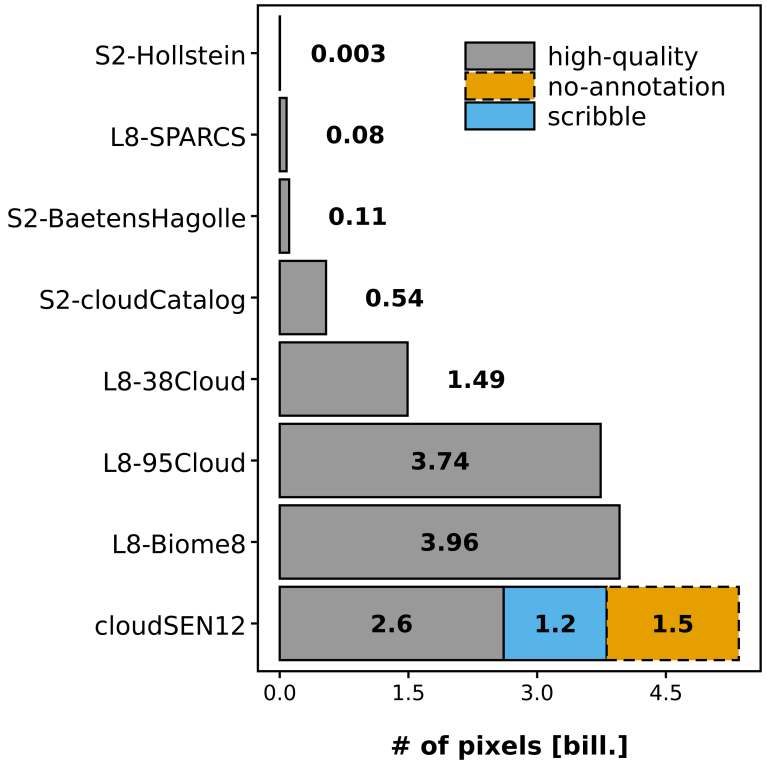

Figure 1.2: Number of hand-crafted pixel annotations between different cloud detection datasets. All the labeled pixels in the CloudSEN12 no-annotation group come from cloud-free IPs.

1.2.1 Data preparation

CloudSEN12 comprises different free and open datasets provided by several public institutions and made accessible by the Google Earth Engine (GEE) platformGorelick et al. 2017. These include Sentinel-2A/B (SEN2), Sentinel-1 (SEN1), Multi-Error-Removed Improved-Terrain (MERIT) DEMYamazaki et al. 2019, Global Surface WaterPekel et al. 2016 (GSW), and Global Land Cover mapsBuchhorn et al. 2020 at 10 and 100 meters. The SEN2 multi-spectral image data corresponds to the 2018–2020 period. We included all the bands from both SEN2 top-of-atmosphere (TOA) reflectance (Level-1C) and SEN2 surface reflectance (SR) values (Level-2A) derived from the Sen2Cor processor, which can be useful to analyze the impact of CD algorithms on atmospherically corrected derived products. See *S2L1C* and *S2L2A* in Table 1.2 for band description. Previous studies have proven the reliability of TOAZekoll et al. 2021 and SR values for cloud detection, but SR data revealed a more plausible differentiation between cloud shadows and clear pixelsDomnich et al. 2021. On the other hand, SEN1 acquires data with a revisit

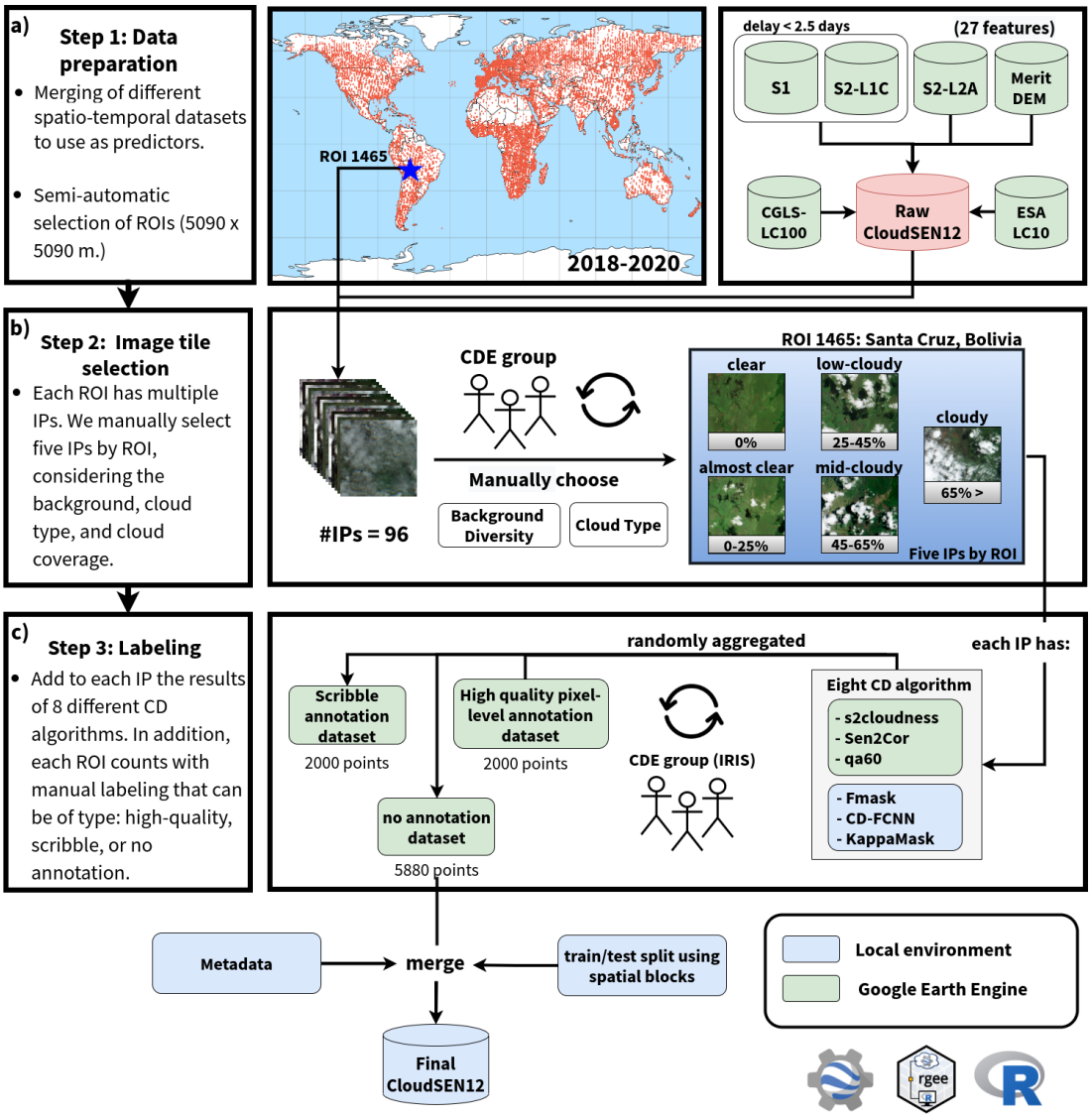

Figure 1.3: A high-level summary of our workflow to generate IPs. a) Satellite imagery datasets that comprises CloudSEN12 assets. b) IP selection by the CDE group. c) Generation of manual and automatic cloud masking.

cycle between 6-12 days according to four standard operational modes: Stripmap (SM), Extra Wide Swath (EW), Wave (WV), and Interferometric Wide Swath (IW). In CloudSEN12, we collect IW data with two polarization channels (VV and VH) from the high-resolution Level-1 Ground Range Detected (GRD) product. Furthermore, we saved the approximate angle between the incident SAR beam and the reference ellipsoid (see *S1* in Table 1.2). Lastly, our dataset also includes previously proposed features for cloud semantic segmentation such as (1) Cloud Displacement IndexFrantz et al. 2018, (2) the direction of cloud shadow (0 - 360°) calculated using the solar azimuth and zenith anglesFernandez-Moran et al. 2021 from SEN2 metadata, (3) elevation from MERIT dataset, (4) land cover

maps from the Copernicus Global Land Service (CGLS) version 3, and the ESA WorldCover 10m v100, and (5) water occurrence from the GSW dataset (see *extra/* in Table 1.2). All the previous features constitute the raw CloudSEN12 imagery dataset (Figure 1.3a). In raw CloudSEN12, full image scenes were resampled to 10 meters using local SEN2 UTM coordinates.

Table 1.3: Cloud semantic categories considered CloudSEN12

Code	Class	Superclass	Description	Priority
0	Clear	Valid	Pixels without cloud and cloud shadow contamination.	4
1	Thick Cloud	Invalid	Opaque clouds that block all the reflectance from the Earth's surface.	1
2	Thin Cloud	Invalid	Semitransparent cloud that modifies the background signal.	3
3	Cloud Shadow	Invalid	Dark pixels thrown by a thick or thin cloud.	2

1.2.2 Image patches selection

In order to gather the raw CloudSEN12 data, we sampled 20,000 random regions of interest (ROIs) distributed worldwide. Each ROI has a dimension of 5,090x5,090 square meters. Besides, we carefully added 5,000 manual selected ROIs to guarantee high scene diversity on complicated surfaces such as snow and built-up areas. After that, a ROI is retained in the dataset if all three of the following requirements are met: (1) SEN2 Level-1C IP does not include saturated, defective, or no-data pixel values, (2) the time difference between SEN1 and SEN2 acquisitions is not higher than 2.5 days, and (3) there are more than 15 SEN2 Level-1C image scenes for the given ROI after applying (2). The total number of ROIs decreased from 25,000 to 12,121 as a result of this filtering. Despite this reduction, CloudSEN12 still manages to reach a full global representation (see Figure 1.1). However, a high number of ROIs does not necessarily imply a consistent distribution among cloud types and cover. Unfortunately, automated image selection based on automatic cloud masking or cloud cover metadata tends to produce misleading results, especially under high-altitude areasTiede et al. 2021, intricate backgrounds Rittger et al. 2020, and mixed cloud types scenes. Hence, to guarantee unbiased distribution between clear, cloud and cloud shadow pixels, 14 cloud detection experts manually selected IPs (hereafter referred to as CDE group, Figure 1.3b). For each ROI, we pick five IPs with different cloud

coverage: cloud-free (0%), almost-clear (0-25%), low-cloudy (25-65%), mid-cloudy (45-65%), and cloudy image (>65%). Atypical clouds such as contrails, ice clouds, and haze/fog had a higher priority than common clouds (i.e., cumulus and stratus). After eliminating ROIs that did not count with at least one IP for each cloud coverage class, the total number of ROIs was reduced from 12,121 to 9,880, resulting in the final CloudSEN12 spatial coverage (Figure 1.1).

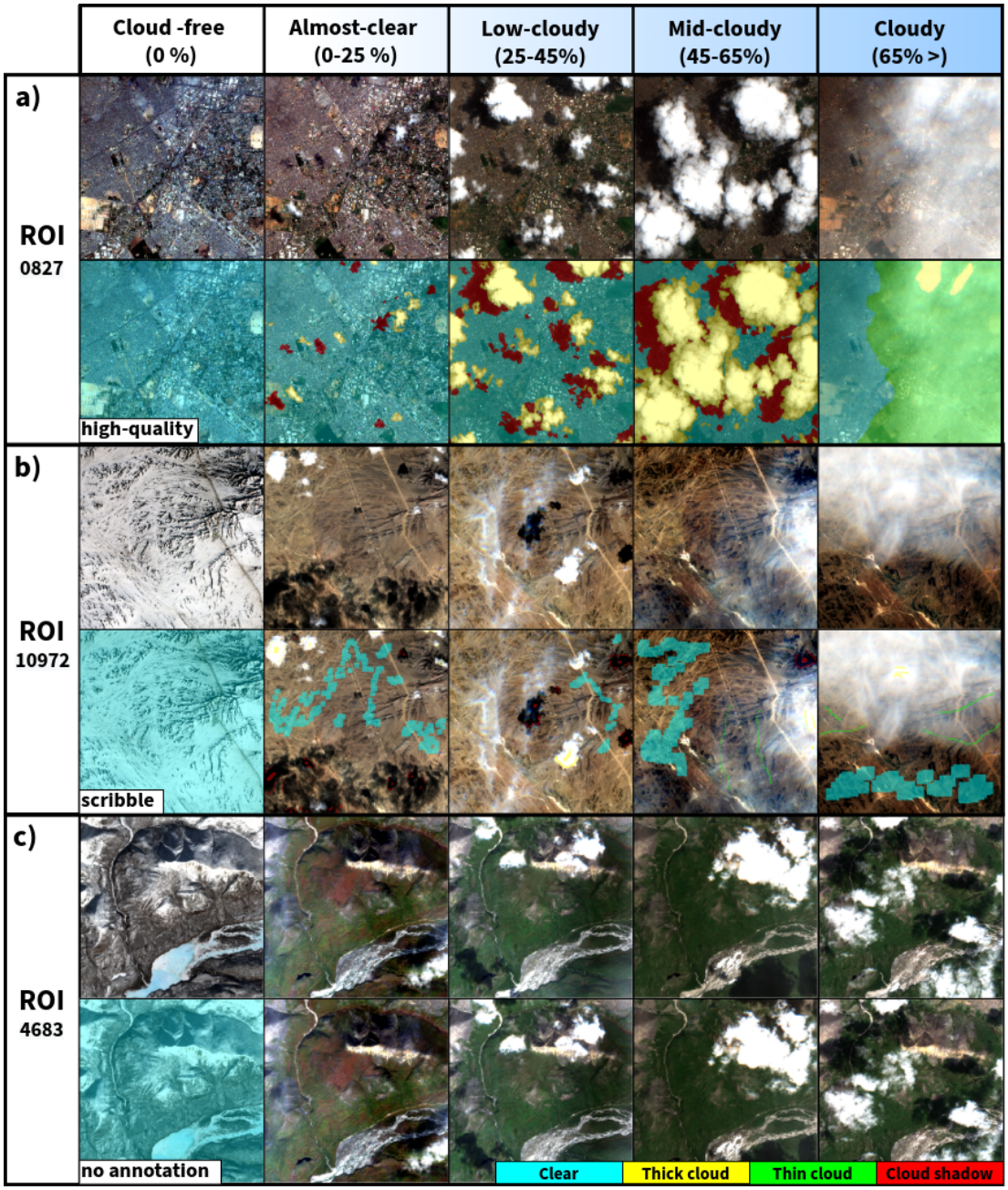

Figure 1.4: The three primary forms of hand-crafted labeling data in CloudSEN12. a) high-quality, the rows depict: SEN2 level 1C data in RGB, manual_hq (high-quality), manual_sc (high-quality).

1.2.3 Annotation strategy

New trends in computer vision shows that reformulating the standard supervised learning scheme can alleviate the huge demands of hand-crafted labeled data. Semi-supervised learning, for instance, can produce more detailed and uniform predictions in semantic segmentation Castillo-Navarro et al. 2020. While weakly-supervised learning suggests a more cost-effective option to pixel-wise annotation, users might utilize scribble labels to train a learning signal for coarse-to-fine enrichmentLi, Chen, et al. 2020. Aware of these manual labeling requirements, CloudSEN12 also supports weakly and self-/semi-supervised learning strategies by including three distinct forms of labeling data: high-quality, scribble, and no-annotation. Consequently, each ROI is randomly assigned to a different annotation group:

- 2,000 ROIs with pixel level annotation, where the average annotation time is 150 minutes (high-quality group, Figure 1.4a).
- 2,000 ROIs with scribble level annotation, where the annotation time is 15 minutes (scribble group, Figure 1.4b).
- 5,880 ROIs with annotation only in the cloud-free (0%) image (no annotation group, Figure 1.4c).

1.2.4 Human calibration phase

Human-made photo interpretation is not a faultless procedure. It might easily be skewed by an individual's basis, overconfidence, tiredness, or ostrich-effectValdez et al. 2017 proclivity. Hence, to lessen this concern, the CDE group refined their criteria using a 'calibration' dataset composed of 35 manually selected challenging IPs. In this stage, all the labelers can consult each other. As a result, they reached an agreement about the SEN2 band compositions to be used and how to deal with complicated scenarios such as cloud boundaries, thin cloud shadows, and high-reflectance background. A labeler is considered fully trained if its overall accuracy in the calibration dataset surpasses 90%. Then, a 'validation' dataset formed of ten IPs is used to assess individual performance; labelers

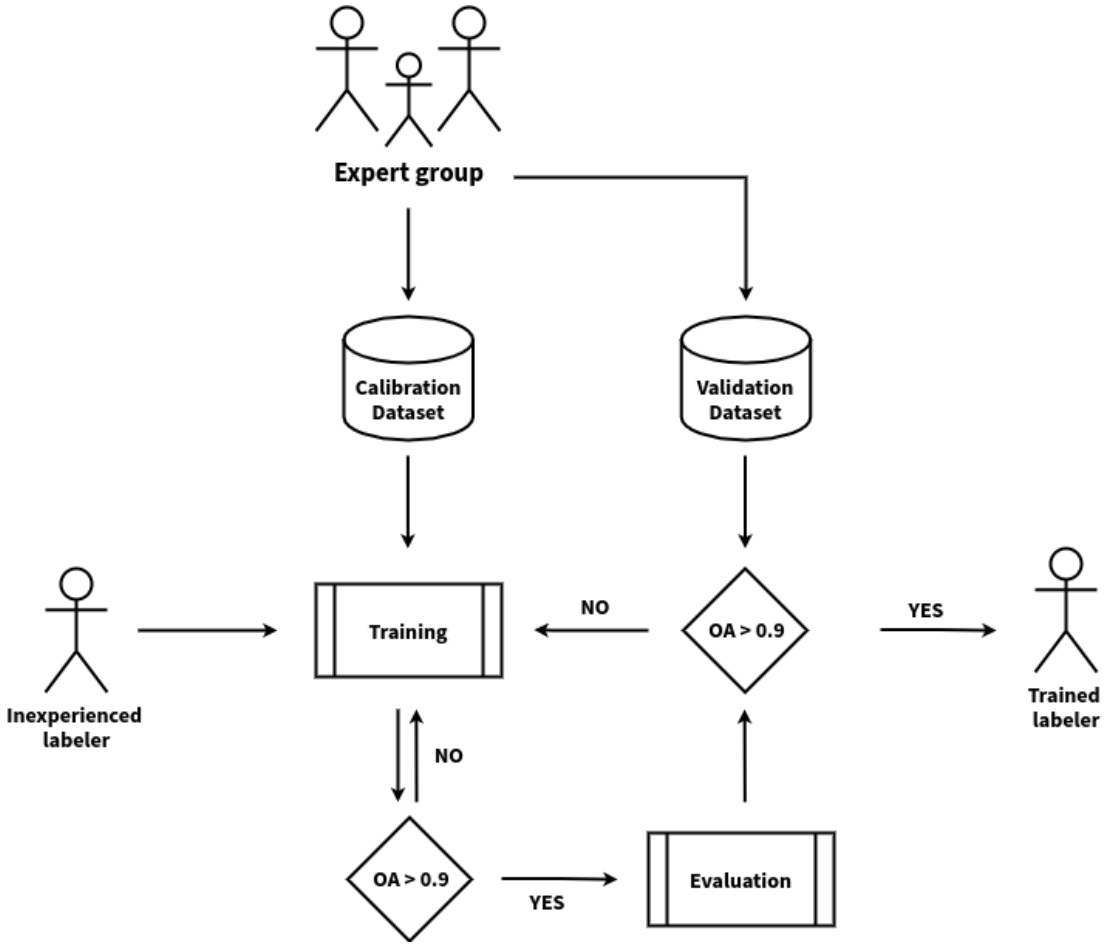

Figure 1.5: Human calibration phase diagram. The overall accuracy (OA) is measured comparing the inexperienced labeler against the expert group results.

are not permitted to confer with one another during this step. If the labeler's overall accuracy drops below 90%, it will return to the calibration phase (Figure 1.5). The human-level performance is measured by comparing the individual labeler's result before and after our four-step control quality procedure (see quality control section). As shown in Figure 1.6, CloudSEN12 set the human-level performance at 95% confidence, varying according to the IP difficulty metadata (see Table 1.4) from 98 to 85%.

1.2.5 Labeling phase

The Intelligence foR Image Segmentation (IRIS) active learning softwareMrziglod 2019 was used in the manual labeling annotation process (Supplementary Figure S1). IRIS allowed CDE members to train a model (learner) with a small set of labeled samples that is iteratively reinforced by acquiring new samples provided by a labeler (oracle). As a result, it dramatically decreases the time spent creating

Table 1.4: Metadata associated to each image patch.

Metadata name	Description
annotator_name	The labeler's name.
roi_id	The region of interest ID.
s2_id_gee	Sentinel-2 GEE ID.
s2_id	Sentinel-2 product ID.
s2_date	Sentinel-2 acquisition date in ISO format.
s2_sen2cor_version	Sen2Cor configuration baseline used at the time of the product generation.
s2_fmask_version	Fmask version.
s2_s2cloudless_version	s2cloudless version.
s2_reflectance_conversion_correction	Earth-Sun distance correction factor.
s2_aot_retrieval_accuracy	Accuracy of aerosol optical thickness model.
s2_water_vapour_retrieval_accuracy	Declared accuracy of the Water Vapor model.
s2_view_off_nadir	The angle from the SEN2 sensor between nadir (straight down) and the scene center.
s2_view_sun_azimuth	SEN2 sun azimuth angle.
s2_view_sun_elevation	SEN2 sun elevation angle.
s1_id	SEN1 product ID.
s1_date	SEN1 acquisition date in ISO format.
s1_grd_post_processing_software_name	Name of the software to pre-processing SEN1.
s1_grd_post_processing_software_version	SEN1 software pre-processing version.
s1_slc_processing_facility_name	Name of the facility where the processing step was performed.
s1_slc_processing_software_version	Software version identification.
s1_radar_coverage	percentage of valid SEN1 pixels contained in this IP.
land_cover	Predominant land use.
label_type	Manual labeling type (i.e., scribble, high-quality or no-annotation).
cloud_coverage	Cloud coverage estimated using photo-interpretation. (see section: Image patches selection).
test	Whether the IP is part of training (train) or testing (test) dataset.
difficulty	Labeler's confidence (from 1 to 5) of the manual annotation.Where one indicates near-perfect and five denotes potentially significant mistakes.
proj:epsg	EPSG code.
proj:geometry	Footprint of this IP.
proj:shape	Number of pixels for the default IP.
proj:centroid	Centroid coordinates of the IP in latitude and longitude.
proj:transform	The affine transformation coefficients.

hand-crafted labels but maintaining the labeler's capacity to make final manual revisions if necessary. For high-quality labeling generation (Figure 1.7a), IRIS starts training a gradient boosting decision tree (GBDT) with s2cloudless cloud probability values greater than 0.7 as thick cloud and less than 0.3 as clear. Next, the labelers make adjustments to the prior results and, if necessary, add other cloud semantic classes such as cloud shadow and thin cloud. Using this new sample set, the GBDT model is re-trained. The two previous steps are repeated several times until the pixel-wise annotation passes the labeler's visual inspection filter. The final high-quality annotation results are then obtained by applying extra manual fine-tuning. Since there are no quantitative criteria to distinguish between the semantic classes, the labelers always attempt to maximize the sensitivity score under ambiguous pixels.

On the other hand, for scribble labeling (Figure 1.7b), the CDE group also used IRIS but without the ML assistance. Labelers spend one-minute adding

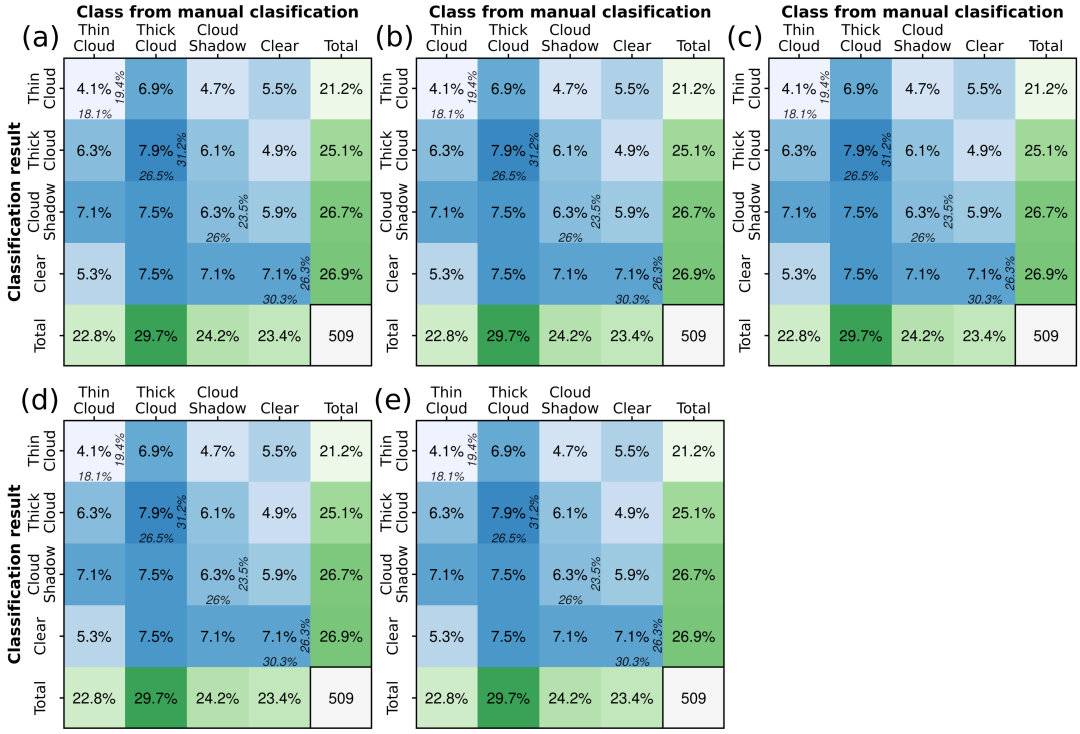

Figure 1.6: Confusion matrices (values in percent) between the high-quality manual labels cast by the CDE group after and before the quality control process. See the sections human calibration and quality control. The original labels are divided based on the difficulty IP property (See Table 1.4).

annotation around centroids of the semantic classes. Usually, pixels adjacent to the centroids are more straightforward to classify automatically. Then, to produce balanced annotations, the CDE group added additional samples at cloud and cloud shadow edges for three minutes.

1.2.6 Quality control phase

Despite the human calibration phase, errors are still common in hand-operated labels. Therefore, statistic and visual inspections were implemented before admitting a manual annotation in CloudSEN12 (Figure 1.8). First, an automatic check is set only for high-quality labels. It proposes that the GBDT accuracy during training must be higher than 0.95. This simple threshold pushes the CDE memsbers to set more samples and care more about labeling correctness. Later, two sequential visual inspection rounds are carried out for scribble and high-quality labels. The evaluators are two other CDE members than the one who labeled the IP. If a mistake is found, it is notified using GitHub DiscussionsHata et al. 2022. Finally, we discern the most challenging IPs (difficulty

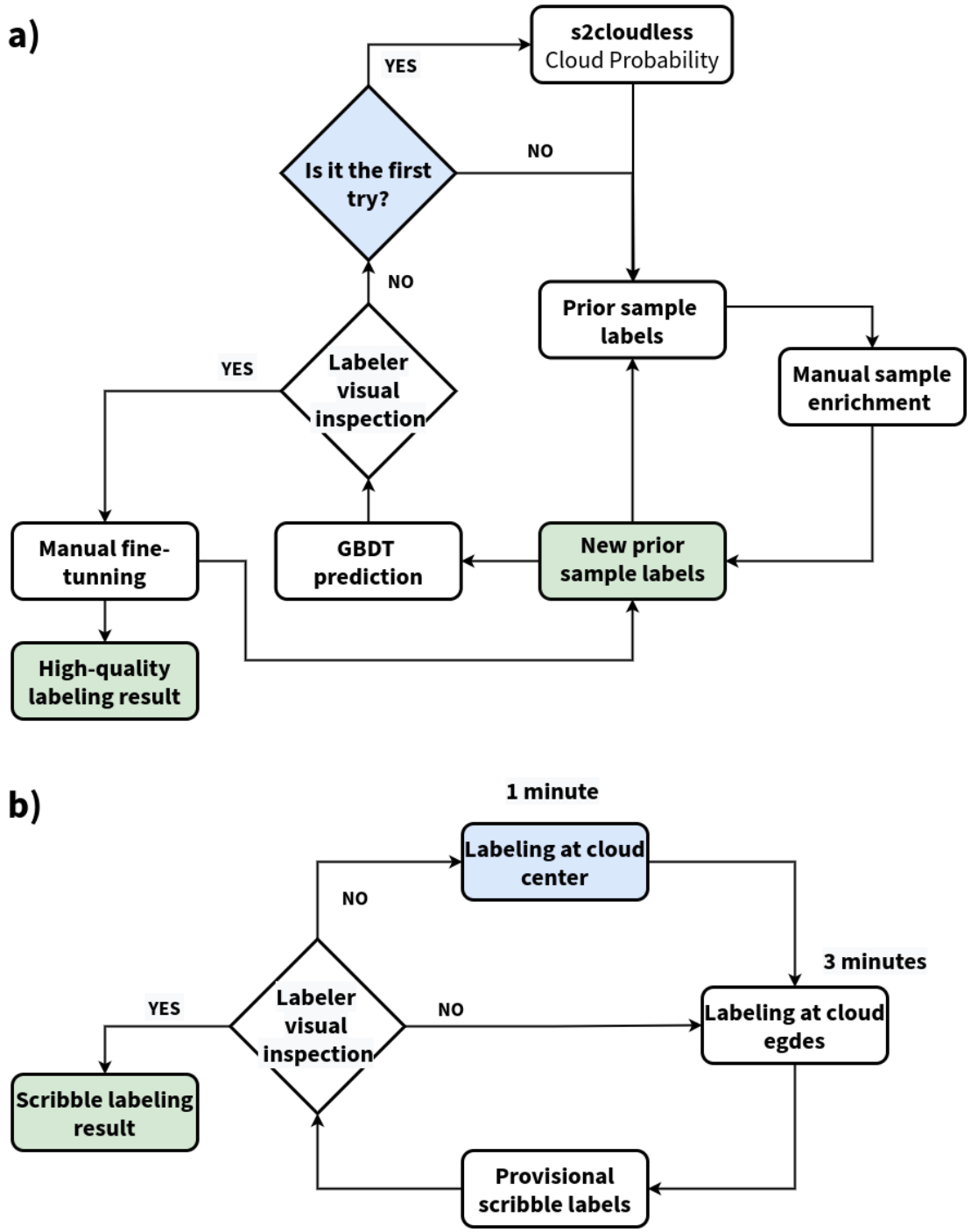

Figure 1.7: a) High-quality labeling phase diagram. The model is set up using s2cloudless priors (blue). Annotations made by labelers with and without ML assistance are saved (green). b) Scribble labeling phase diagram. The labelers starts adding samples near to the centroids (blue), only the annotation cast by the labelers is preserved.

level greater than 4, see Table 1.4) and consult all CDE members to reaffirm or change a semantic class. The deliberations were supported by cloudApp (https://csaybar.users.earthengine.app/view/cloudapp), which is a GEE web application that displays SEN2 image time series from any location on the earth (Figure S2).

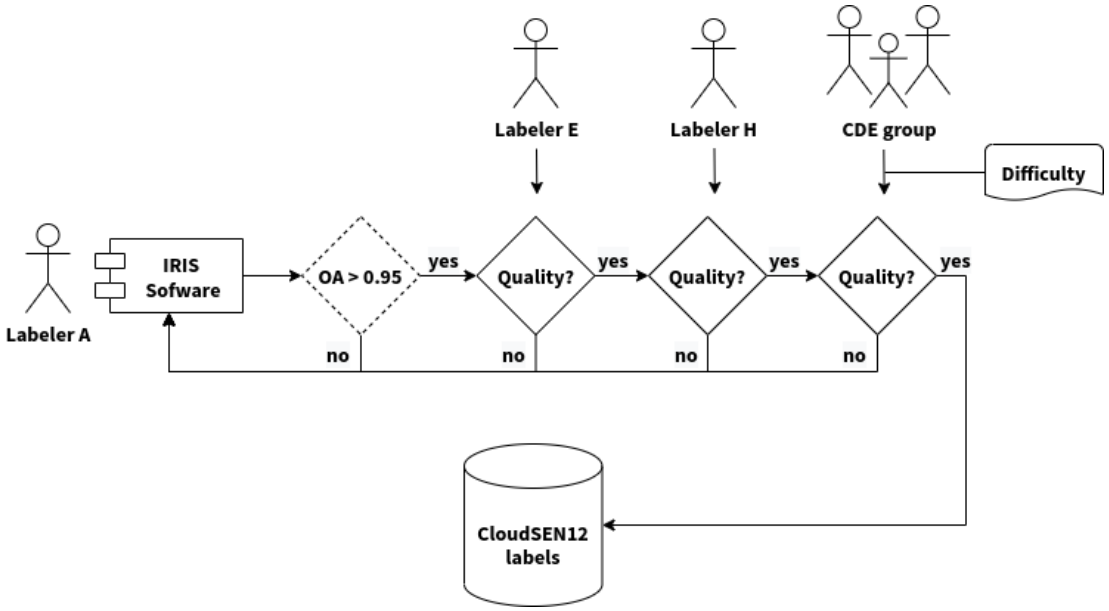

Figure 1.8: Flowchart overview of the entire QC process.

1.2.7 Benchmarking cloud detection models

The variety of ways in which an EO model can be used makes benchmarking a difficult procedureSkakun et al. 2022. For example, assessing CD model performance by seasonality rather than interannual or decadal variation may be more relevant in certain circumstances. Another example is that some data users may want to compare model performance geographically across different biomes or land-cover classes. A typical approach in EO is to benchmark models like traditional computer vision algorithms generating unique metrics for the entire datasets. However, this widespread practice can lead to biased conclusions. We argue that an appropriate model in EO must be capable of obtaining an adequate global metric while being consistent in space across multiple timescales. Furthermore, the observed patterns must be aligned with our physical understanding of the phenomena. In order to cover all the possible EO benchmarking user requirements, we added to each IP the results of eight of the most popular CD algorithms (see *labels/* in Table∼1.2). This provides CloudSEN12 users more flexibility to choose a comparison strategy that is better tailored to their requirements. Next, we detail the CD algorithms available in CloudSEN12:

- Fmask4: Function of Mask Qiu, Zhu, and He 2019 cloud detection algorithm for Landsat and Sentinel-2. We use the MATLAB implementation code via

Linux Docker containers (https://github.com/cloudsen12/models. We set the dilatation parameter for cloud, cloud shadow, and snow to 3, 3, and 0 pixels, respectively. The erosion radius (dilation) is set to 0 (90) meters, while the cloud probability threshold is fixed to 20%.

- Sen2Cor: Software that performs atmospheric, terrain, and cirrus correction to SEN2 Level-1C input data. We store the Scene Classification (SC), which provides a semantic pixel-level classification map. The SC maps are obtained from the "COPERNICUS/S2_SR" GEE dataset.

- s2cloudless: Single-scene CD algorithm created by Sentinel-Hub using a LightGBM decision tree model"LightGBM: A highly efficient gradient boosting decision tree" 2017. The cloud probability values are collected without applying neither a threshold nor dilation. This resource is available in the "COPERNICUS/S2_CLOUD_PROBABILITY" GEE dataset.

- DL_L8S2_UV López-Puigdollers et al. 2021: U-Net with two different SEN2 band combinations: RGBI (B2, B3, B4, and B8) and RGBISWIR (B2, B3, B4, B8, B11, and B12) trained on the Landsat Biome-8 dataset (transfer learningMateo-García et al. 2020; Mateo-García et al. 2021 from Landsat 8 to Sentinel-2).

- KappaMask Domnich et al. 2021: U-Net with two distinct settings: all Sentinel-2 L1C bands and all Sentinel-2 L2A bands except the Red Edge 3 band. It was trained in an extension of the Sentinel-2 Cloud Mask Catalogue.

- QA60: Cloud mask embedded in the Quality assurance band of SEN2 Level-1C products.

Table~1.5 shows the cloud semantic categories for the different CD techniques available in CloudSEN12. It should be noted that only four CD algorithms provide the cloud shadow category.

Table 1.5: Output correspondence for different CD algorithms. Sen2Cor, Fmask, KappaMask, DL_L8S2_UV and S2cloudless are mapped respectively to CloudSEN12 cloud semantic categories. Adapted from Domnich et al.Domnich et al. 2021

Sen2cor	KappaMask	CloudSEN12	Fmask	S2Cloudless	DL_L8S2_UV	QA60
0 No data	0 Missing	-				
1 Saturated or defective		-				
2 Dark area pixels		0 Clear				
3 Cloud shadows	2 Cloud shadows	3 Cloud shadows	2 Cloud shadows			
4 Vegetation	1 Clear	0 Clear	0 Clear	0 Clear	0 Clear	0 Clear
5 Bare Soils		0 Clear				
6 Water		0 Clear	1 Water			
7 Cloud Low probability/ Unclassified	5 Undefined	-				
8 Cloud medium probability		1 Thick cloud				
9 Cloud high probability		1 Thick cloud	4 Cloud	1 Cloud	1 Cloud	1 Cloud
10 Thin cirrus	3 Semi-transparent cloud	2 Thin cloud				
11 Snow		0 Clear	3 Snow			

1.2.8 Preparing CloudSEN12 for machine learning

Splitting our densely annotated dataset into train and test sets is critical to ensure that ML practitioners are always using the same samples when providing results and to ensure that the tested algorithms provide a good generalization. Since cloud formation tends to fluctuate smoothly throughout space, a simple random split is suspicious to violate the assumption of test independence, especially under highly clustered labeled areas, such as the green and yellow regions shown in Figure∼1.1. Therefore, we carry out a spatially stratified block split strategyValavi et al. 2019, based on Roberts et al. 2017 Roberts et al. 2017, to limit the risk of overfitting induced by spatial autocorrelation. First, we divided the Earth's surface into regular hexagons of 100 km^2. Then, the initial hexagons are filtered, retaining only those that intersect with the high-quality dataset (Figure∼**??**). Finally, using the difficulty IP property (see Table 1.4), we randomly stratified the remained blocks using 90% (1827 ROIs) and 10% (173 ROIs) for training and testing, respectively (Figure∼1.9). The unlabeled and scribble datasets might be used as additional inputs for the training phase.

1.3 Data Record

The dataset is available via Radiant MLHub**dsds** data repository at: https://doi.org/10.34911/rdnt.y3xeg3. We defined an IP as the primary atomic unit, representing a single spatio-temporal component. Each IP has 49 assets (see Table 1.2) and 31 properties (see Table 1.4). All of the assets are delivered in the form of

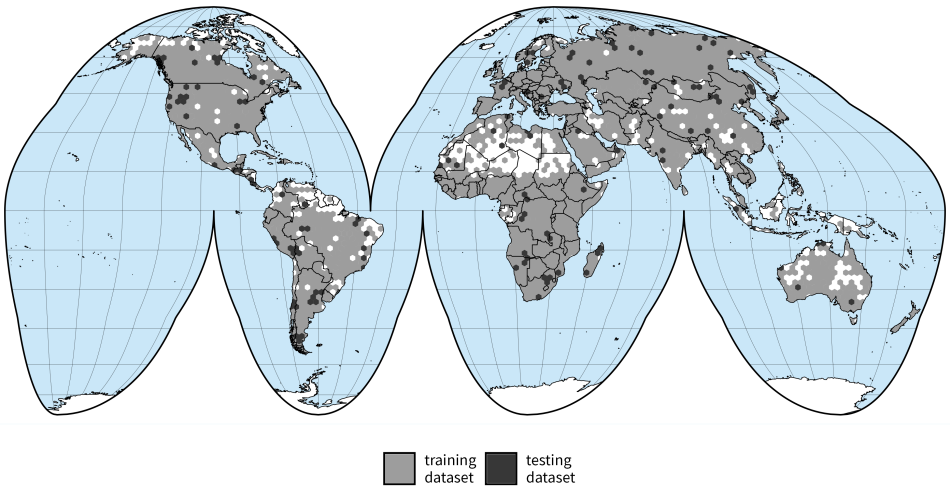

Figure 1.9: Location of the training (blue) and testing (yellow) regions.

LZW-compressed COG (Cloud Optimized GeoTIFF) files. COG is an imagery format for web-optimized access to raster data that has a specific internal pixel structure that allows clients to request just specified areas of a large image by submitting HTTP range requests **IosifescuEnescu2021**. The IP properties are shared using the SpatioTemporal Asset Catalog (STAC) specification. STAC provides a straightforward architecture for reading metadata and assets in JSON format, providing users with a sophisticated browsing experience that seamlessly integrates with modern scripting languages and front-end web technologies.

CloudSEN12 assets, as seen in Figure~**??**, are organized into four levels. The top-level includes three folders: high, scribble, and nolabel. These folders correspond to the annotation categories high-quality (2000 ROIs), scribble (2000 ROIs), and no annotation (5880 ROIs), respectively. In the second level, the folders included data pertaining to a given geographic location (ROI). The folder name is the ROI ID (Figure~**??**b). Since an ROI consists of five IPs at different dates, each ROI folder is subdivided into five folders whose names match the GEE Sentinel-2 product ID of the specific IP (Figure~**??**c). Finally, each IP folder stores the information detailed in Table~1.2 (Figure **??**d).

1.4 Technical Validation

This section reports CloudSEN12's suitability in multi-class semantic segmentation tasks. Based on Domnich et al. Domnich et al. 2021 configuration, we trained a

U-Net Ronneberger et al. 2015 model using all SEN2 L1C bands in the high-quality training dataset as input data. The high-quality test dataset (Figure **??**) is used to evaluate the algorithm performance by overall accuracy. A U-Net architecture is composed of three parts: (1) the encoder, which utilizes convolutional and max-pooling layers to extract features at multi-resolution level, (2) the bottleneck, that force to the model learns a compression of the input data, and (3) the decoder, which builds a segmentation map by combining the bottleneck results, deconvolutional layers for up-sampling and feature maps from the equivalent level in the encoder. In our U-Net implementation, we use MobileNetv2 **Sandler2018** as encoder. The dice coefficient loss and the Adam optimizer are used for fitting the model. Besides, batch normalization layers are added after each convolutional layer for feature normalization throughout the network.

1.5 Usage Notes

This paper introduces CloudSEN12, a new large dataset for cloud semantic understanding, comprising 49,400 image patches distributed across all continents except Antarctica. The dataset has a total size of up to 1 TB. Nevertheless, we assume that most users experiments will only need a fraction of CloudSEN12. Therefore, to simplify its use, we developed a Python package called CloudSEN12. This Python package aims to help machine learning and remote sensing practitioners to:

- Query and download the Radiant MLHub datasets using a user-friendly interface.
- Transform datasets to make them compatible with PyTorch DataLoader class.
- Provide pre-trained models based on the CloudSEN12 dataset.

The CloudSEN12 website www.cloudsen12.github.io includes tutorials for querying and downloading the dataset using the CloudSEN12 package. Besides, there are examples of how to train DL models using PyTorch. Finally, although CloudSEN12 was initially designed for semantic segmentation, it can be easily adapted to tackle other remote sensing problems like SAR-sharpeningSchmitt and Wendleder 2018, colorizing SAR imagesSchmitt, Hughes, et al. 2018, SAR-optical

image matchingHughes, Schmitt, et al. 2018, and land cover mappingKarra et al. 2021.

1.6 Code availability

The code to (1) create the raw CloudSEN12 imagery dataset, (2) download assets associated to each ROI, (3) create the manual annotations, (4) display cloudApp, (5) automatic perform cloud masking, (6) reproduce all the figures, (7) replicate the technical validation, (8) install CloudSEN12 Python package, and (9) deploy CloudSEN12 website is available in our Github organization https://github.com/cloudsen12/.

It is a miracle that curiosity survives formal education.

— Albert Einstein

2 Cloud cover estimation

Contents

2.1 Introduction

Clouds contaminate nearly 0.65% of the earth's surface regardless of time, with hotspots in tropics and subtropics regions (Sassen and Wang 2008; Winker et al. 2010; Wilson and Jetz 2016). Their effects on electromagnetic radiation signals vary according to different types of clouds. Clouds can be characterized in general based on their top pressure (CTP) and optical thickness (COT) properties (Figure 2.3). High COT clouds are easily spotted by Sentinel-2 imagery due to their white

color in all-optical frequency bands. Low COT and high CTP clouds, for example, cirrus clouds, can be featured using the cirrus band (1.36–1.39 μm), which is sensible to water vapor absorption. Lastly, clouds with low COT and low CTP, like haze and fog, can be found using spectral indices like HOT, which take advantage of the high correlation between blue and red bands on land surfaces under clear skies (HOT, Zhang et al. 2002). In general, when the COT property has low values, sunlight is partially reflected, allowing a distorted view of the surface to be observed (Lynch et al. 2002; Chen et al. 2017). For some applications, such as object detection or disaster response (Mateo-Garcia et al. 2021), images contaminated with haze and fog are still helpful. Therefore, users must have control over which contaminated pixels to mask out.

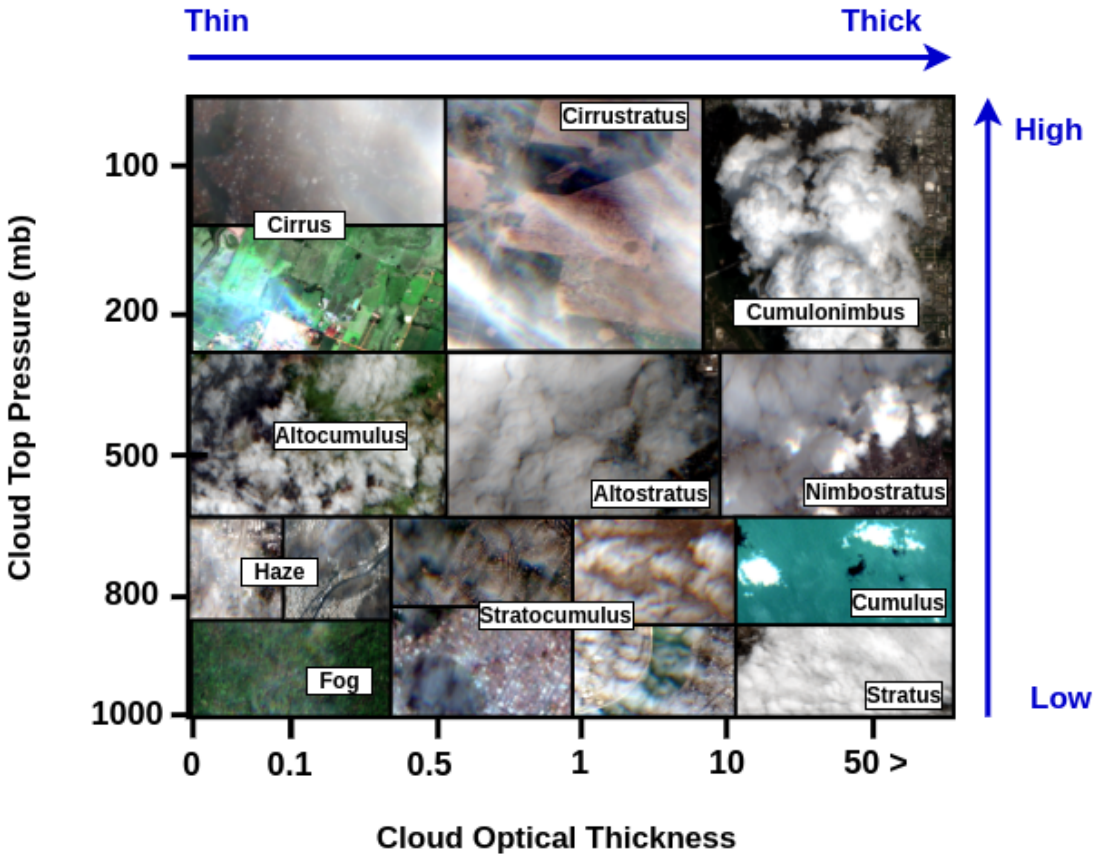

Figure 2.1: Different cloud types depict in Sentinel-2 imagery.

In recent years, a plethora of cloud masking methods have been presented (Hagolle et al. 2017; Domnich et al. 2021; Louis et al. 2016; Qiu, Zhu, and He 2019; Richter and Schläpfer 2019; Wevers et al. 2021; López-Puigdollers et al. 2021; Frantz 2019). On the basis of cloud masking results, cloud cover metadata (Figure 2.2) is generated for searching, selecting, and accessing imagery

datasets (**tiede2021investigating**). Cloud cover prediction can be interpreted as a statistical regression problem. Positive residuals are the result from cloud comission errors (non-cloud as cloud), whereas negative residuals derive from cloud omission errors (cloud as non-cloud).

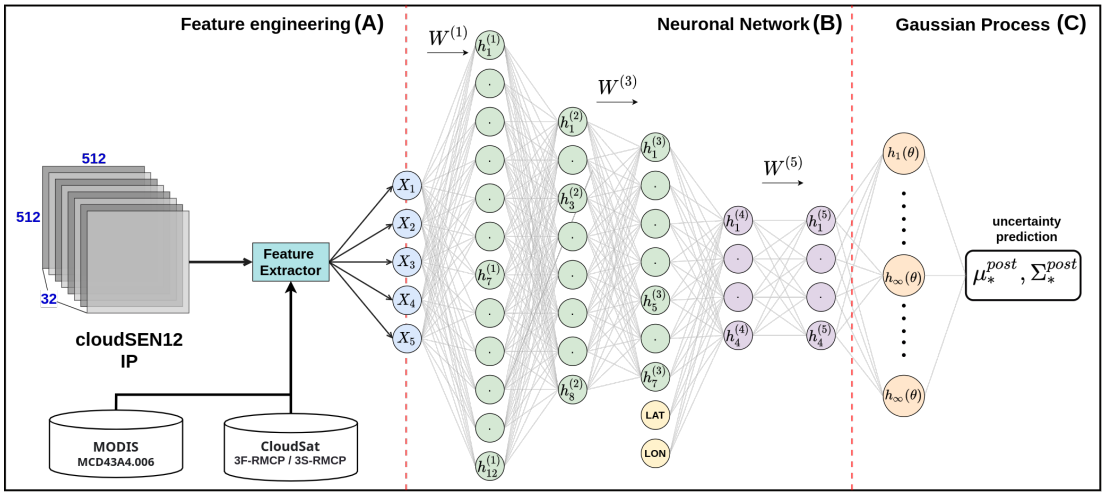

Figure 2.2: A simplified diagram that illustrates the regression problem.

Over the last years, only a few studies have attempted to benchmark the various Sentinel-2 cloud masking methods. For instance, Cilli et al. 2020 compare DD with KD methods by analyzing 135 Sentinel-2 images distributed worldwide. They concluded that DD methods outperform KD methods. According to their experiments, 10^4 manually labeled pixels are sufficient for train machine learning algorithms to operate accurately cloud masking. Nonetheless, it is well established that DD models are highly dependent on the training dataset (López-Puigdollers et al. 2021). As a result, the comparison could be unfair, especially if the KD methods have not been as well calibrated to the dataset. Zekoll et al. 2021 compare three KD threshold-based: FMask, ATCOR, and Sen2Cor using a sample-based dataset. The results show that Sen2Cor outperforms the other methods. However, human-made datasets, especially those created by sampling, can be positively skewed if we consider that humans tend to overlook unpleasant information such as cloud borders (ostrich-effect, Valdez et al. 2017). Using four different datasets, the Cloud Mask Intercomparison eXercise (CMIX) recently compared ten cloud masking algorithms. They suggested that no single algorithm performed better than the others (Skakun et al. 2022). Similar conclusions are found in Tarrio et al. 2020 by analyzing 28 images over six Sentinel-2 tiles in Africa and Europe.

This chapter presents a new DD technique based on state-of-the-art probabilistic deep neural networks that generate cloud cover estimation together with well-calibrated uncertainty. A cloud clover product with uncertainty values will give final users more leeway in balancing commission and omission errors. In short, uncertainty can be a result of two distinct sources: aleatoric or epistemic (Hora 1996, Der Kiureghian and Ditlevsen 2009). While aleatoric measures the data's stochasticity, epistemic measures the models' structure and parameters' uncertainty. We propose estimate the epistemic uncertainty with a variational deep kernel learning (VDKL). The following sections provide a quick overview of the data used in this research (section 2), details about the implementation of VDKL (section 3), discuss the results (section 4), and finally reach some conclusions.

2.2 Data

2.2.1 Sentinel-2

Sentinel-2 (SEN2) is a Copernicus EO mission that consists of two satellites: SEN2A (launched on June 23, 2015) and SEN2B (launched on: March 7, 2017). At the equator, each Sentinel-2 satellite has a ten-day repeat cycle. However, both satellites present the same orbit with a 180° phase delay, shortening to five-day revisiting time. The cloudSEN12 dataset acquired 10 000 image patches of 509×509 pixels of SEN2 A/B Multi-Spectral Instrument (MSI) top-of-atmosphere (TOA) reflectance images (Level-1C) from 2018 to 2020. The thirteen spectral bands available in SEN2 A/B (Table 1.2) constitute the input data from the SDKL model. For computational efficiency in training, the SEN2 image patches are resized to 128 x 128 pixels, while keeping the original aspect ratio.

2.2.2 Reference data

The proposed probabilistic neural network requires a diverse ground-truth dataset to create robust representations that reflect cloud type and landscape heterogeneity. Fortunately, high-quality cloudSEN12 has a large number of image patches with cloud and cloud shadow semantics worlwide. Apart from the hand-crafted data at the pixel level, the automatic cloud masking techniques Fmask4, Sen2Cor,

s2cloudless, DL L8S2 UV, KappaMask, and QA60 are used to compare VDKL results. In order to maintain fairness, we only consider cloudSEN12 high-quality IPs because of the risk of bias due to scene incompleteness in scribble and no-annotation. Besides, IPs with cloud coverage lower than 5 % are not considered to avoid positive bias in the results. For both human and automatic pixel-level products, the cloud cover percentages were computed by dividing the number of cloud pixels by the total number of pixels (Figure 2.2). See chapter one or Skakun et al. 2022 for a more detailed overview of the prior cloud masking algorithms.

2.3 Methodology

This study aims to create a regression model that predicts cloud cover y given SEN2 imagery X. The regression model f is trained using a dataset, $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$ with x_i and $y_i \in \mathbb{R}$. The segment x_i represents an array of $509 \times 509 \times 13$, and y_i is the cloud cover value (see section 2.2.2), and N is the number of observations set as 10000. We propose the use of a variational stocastic deep kernel learning (VSDKL) regression. It combines standard deep neural networks (DNN) with gaussian process regression (GP). While DNN captures the non-stationary and hierarchical structure, GP permits the estimation of their uncertainty considering the local autocorrelation of the observations in latent space. Rasmussen 2003 provide a detail description of GP. The standard GP, VSDKL, the model setup and validation are briefly reviewed in this following section.

2.3.1 Gaussian process regresion

Standard Gaussian Process Regression (GP) models is a expressive probabilistic model in which both training and testing data points are regarded samples of a joint multivariate normal distribution (Williams and Rasmussen 2006). As other regression models, a GP model is formed by noisy variables of the true underlying function f that projects the vector space X into real-valued targets y, i.e. $y = f(\mathbf{x}) + \epsilon$. The element ϵ represents the noise variables with $\mathcal{N}(0, \sigma^2)$. In a GP model we assume that all the finite dimensional distributions $f(\mathbf{x})$ are normally distributed with μ as a mean and $K_{XX}|\gamma$ as the prior covariance matrix. The covariance (kernel) matrix regulates the smoothness of GPs and its values

are implicitly dependent on the kernel hyperparameters γ. In this specific case, the estimation of $f(\mathbf{x})$ can be expressed given by:

$$\mathbf{f} = f(\mathbf{x}) = [f(x_1), ..., f(x_m)]^\top \sim \mathcal{GP}\left(\mu, K_{XX}|\gamma\right) \tag{2.1}$$

Conditioning the joint normal distribution by the training points, the posterior distribution of the output values $f(\mathbf{x}_*)$ at the test data point X_* can be inferred as:

$$\begin{aligned} f(\mathbf{x}_*) \mid X_*, X, \mathbf{y}, \boldsymbol{\gamma}, \sigma^2 &\sim \mathcal{N}(\mu^*, \Sigma^*), \\ \mu^* &= \mu_{X_*} + K_{X_*X}\widehat{K}_{XX}^{-1}\mathbf{y}, \\ \Sigma^* &= K_{X_*X_*} - K_{X_*X}\widehat{K}_{XX}^{-1}K_{XX_*} \end{aligned} \tag{2.2}$$

A hat denotes an added diagonal, i.e. $\widehat{K}_{XX} = K_{XX} + +\sigma^2 I$. μ^* and Σ^* are the posterior mean and covariance matrix respectively. The matrices of the form $K_{X_iX_j}$ denote cross-covariances between the train (X) and test (X_*) vector spaces. The hyperparameters λ of the kernel are usually learned directly by minimizing the negative log marginal likelihood $\mathcal{L}(\theta)$ with respect to training observations:

$$\begin{aligned} \mathcal{L} &= -\log p(\mathbf{y} \mid \gamma, X) \propto \mathbf{y}^\top \widehat{K}_{XX}^{-1}\mathbf{y} + \log\left|\widehat{K}_{XX}\right|, \\ \frac{\partial \mathcal{L}}{\partial \theta} &= \mathbf{y}^\top \widehat{K}_{XX}\frac{\partial \widehat{K}_{XX}^{-1}}{\partial \theta}\widehat{K}_{XX}\mathbf{y} - \operatorname{tr}\left\{\widehat{K}_{XX}^{-1}\frac{\partial \widehat{K}_{XX}}{\partial \theta}\right\} \end{aligned} \tag{2.3}$$

The main bottleneck for kernel learning is solve the linear system $\widehat{K}_{XX}^{-1}y$ in equation 2.3. The standard approach is to compute the Cholesky decomposition of the matrix $\widehat{K}_{XX}^{-1}$. The Cholesky decomposition's core algorithm uses a divide-and-conquer approach that is inefficient on GPU acceleration (Krishnamoorthy and Menon 2013). Furthermore, it requires $\mathcal{O}(n^3)$ computation and $\mathcal{O}(n2)$ storage for GP inference and kernel learning (Rasmussen 2003). To address the above challenges, several approaches to scaling up GP inference have been proposed (Gardner et al. 2018; Cunningham et al. 2008; Dong et al. 2017; Bach 2013; Wilson, Dann, et al. 2015). In this paper, we address the GP inference issue by using the Blackbox Matrix-Matrix multiplication inference (BBMM, Gardner et al. 2018). BBMM use preconditioned batched conjugate gradients to solve linear systems, reducing the asymptotic time complexity of GP inference from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$. Besides, it overcomes memory constraints by divvying the kernel matrix to perform matrix-vector multiplication (MVM, Demmel 1997) without having

to explicitly construct the kernel matrix, reducing the memory requirement to $\mathcal{O}(n)$. Finally, BBMM parallelize partitioned MVMs across multiple core, enabling a better use of GPU hardware in comparison to the Cholesky factorization.

2.3.2 KISS-GP

Structured kernel interpolation (SKI) or KISS-GP is a scalable Gaussian process variant that combine the use of inducing point (**williams2000using**), structure exploiting (**wilson2014covariance**), and sparse interpolation. The inducing point technique states that we can approximate the exact GP inference using a low-rank kernel given a set of $m \times n$ data points Z.

$$K_{XX} \approx K_{XZ} K_{ZZ}^{-1} K_{ZX} \tag{2.4}$$

The inducing points Z are set on a grid in KISS GP, and the linear systems K_{ZZ}^{-1} are solved efficiently using either Kronecker or Toeplitz algebra. The K_{XZ} component which represent the cross covariances for the kernel evaluated at the training X and inducing inputs points Z is approximated by interpolating on the covariance matrix K_{ZZ}.

$$K_{X,Z} \approx W K_{Z,Z} \tag{2.5}$$

where W is an $n \times m$ matrix of interpolation weights that can be extremely sparse and values are determined using a deterministic interpolation approach, for instance inverse distance weighting. By substituting $K_{X,Z}$ in Eq. 2.4, we get:

$$K_{X,X} \approx K_{XZ} K_{ZZ}^{-1} K_{ZX} \approx W K_{Z,Z} K_{ZZ}^{-1} K_{ZZ} W^T = W K_{Z,Z} W^T = K_{SKI} \tag{2.6}$$

2.3.3 Variational Stocastic Deep kernel learning

Variational Stocastic Deep kernel learning (VSDKL) is a probabilistic deep network that simultaneously learns a feature extractor and GP parameters. Since run an exact GP with $509 \times 509 \times 13$ (SEN2 dimensions) is computational intractable. The variational approach directly approximates the mean and covariance in Eq. 2.2 to determinate the posterior GP parameters by inducing points

(**titsias2009variational**). We apply the same sampling approach as in KISS-GP to select the inducing points. The VSDKL network's structure is depicted in Figure 1.2. The deep non-linear feature extractor $\mathbf{h}(\mathbf{x}, \mathbf{w})$, parametrized by weights $\mathbf{w}$, is applied to the observed input variable $\mathbf{X}$. Next, the DNN outputs are modeled using a variational $\mathcal{GP}$ by:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{h}_{\mathbf{w}}(\mathbf{x})), k_{\gamma}(\mathbf{h}_{\mathbf{w}}(\mathbf{x}), \mathbf{h}_{\mathbf{w}}(\mathbf{x}'))) \tag{2.7}$$

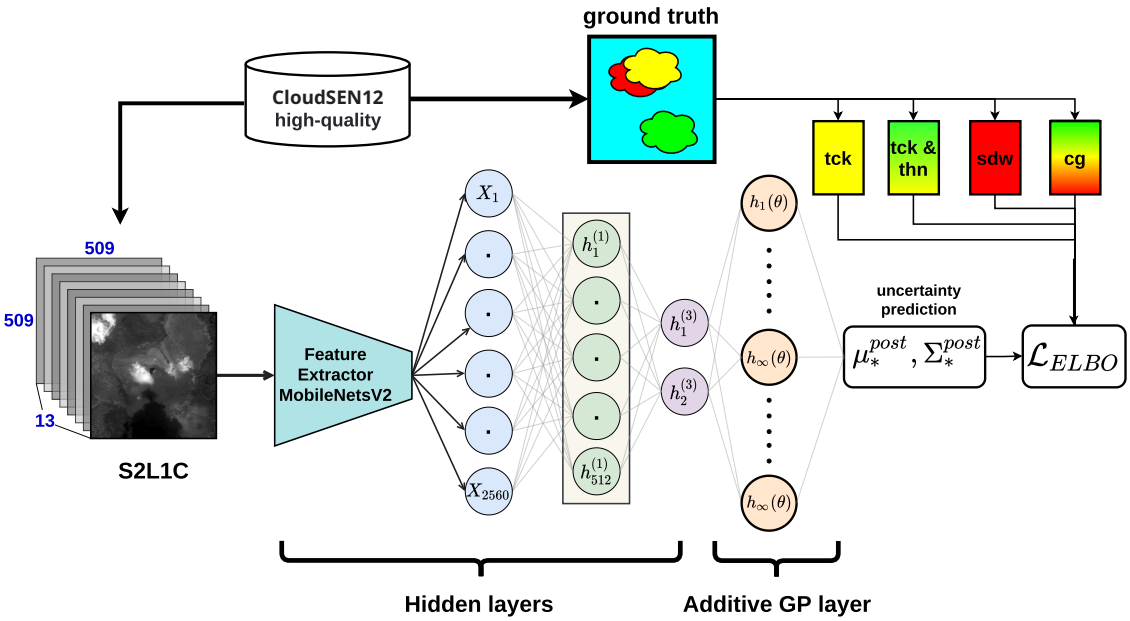

Figure 2.3: Different cloud types depict in Sentinel-2 imagery.

2.3.4 Model training

The complete training procedure is described in Algorithm 1. First, we use a geographical block-by-block sampling method to split the dataset into two parts: training and testing. As explained in section 2.3.3, a DKL has two parts: DNN (ResNet-18) and a GP. Following the recommendation of Wilson, we train from scratch first the feature extractor (DNN). The DNN (ResNet-18) was trained to minimize the L1 loss between the cloud cover predictions and the cloud cover obtained derived from human-photo interpretation. The number of minibatch size, momentum, and total iteration is 64, 0.9, and 10^6, respectively. Instead of adding additional dimensions which increases the risk of feature collapse, the variational GP module is placed directly on top of the last convolutional layer, which is 512

dimensional in the case of the ResNet-18. All parameters of the model, including neural network weights w and kernel parameters γ, are optimized end-to-end via backpropagation to minimize the ELBO negative log marginal likelihood.

Algorithm 1 Algorithm for training SVDKL

1: ResNet-18 (NN) pretrained using cloud cover obtained from the cloudSEN12 high-quality dataset.
2: Residual NN $f(\theta) : x \rightarrow \mathbb{R}^J$ with feature space dimensionality J and parameters θ.
3: Approximate GP with parameters $\varphi = \{l, s, \omega\}$, where l length scale and s output scale of covariance kernel, ω GP variational parameters.
4: Set initial inducing points using KISS-GP approach.
5: **for** minibatch $x_b, y_b \subset X, Y$ **do**
6: $\quad p(y'_b \mid x_b) \leftarrow \mathcal{GP}(f_{\theta'}(x_b))$
7: $\quad \mathcal{L} \leftarrow ELBO_{\phi}(p(y'_b \mid x_b), y_b)$
8: $\quad (\phi, \theta) \leftarrow (\phi, \theta) + \eta * \nabla_{\phi,\theta}\mathcal{L}$
9: **end for**

2.3.5 Model evaluation

Before estimating the effectiveness of the SVDKL in cloud cover estimates, we establish the current state of cloud masking methods by analyzing the similarity between cloud semantic categories (Table 1.3) for cloudSEN12. We established the "cloud" and "non-cloud" superclasses (Table 1.3) that aggregate thick and thin cloud and clear and cloud shadows classes, respectively. We report the producer's accuracy (PA) as the key error metric to assess the disparities between predicted and expected pixels. Furthermore, we complement this metric with the user's accuracy (UA) and balanced overall accuracy (BOA).

$$\text{PA} = \frac{TP}{TP + FN}, \text{UA} = \frac{TP}{TP + FP}, \text{BOA} = 0.5 \left(PA + \frac{TN}{TN + FP} \right) \tag{2.8}$$

where TP, TN, FP, and FN are the true positive, true negative, false positive, and false negative, respectively. High PA values show that cloud pixels have been effectively masked out, whereas high UA values indicate that the algorithm is cautious to exclude non-cloud pixels. High BOA values are related to a good balance of false positives and false negatives.

For cloud cover prediction, the SVDKL and the other reference models available in cloudSEN12 were evaluated using two popular continuous metrics: root mean

squared error (RMSE), and mean absolute error (MAE). The RMSE is the most important metric to examine. It is a quadratic scoring that report the relative deviations in absolute terms.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{2.9}$$

With $\hat{y}_i$ representing the prediction of the DKL and y_i the cloud error ground truth at each IP i. The MAE, on the other hand, is a linear score, meaning that all individual differences contribute equally to the average.

$$MAE = \frac{1}{N}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{n}\right| \tag{2.10}$$

Since the cloud masking error spans from 0 to 1, the RMSE and MAE will not exceed 1, where values closer to 0 suggesting better model fitting. Additionally, we report the efficiency of the algorithms to correctly classify cloud-free images using filters of 1, 5, 10, and 20%. Finally, the

Lastly, we assess the validity of the SVDKL probabilistic estimates using the Continuous Ranked Probability Score (CRPS). Continuous ranked probability score (CRPS) contrasts cumulative distribution functions (CDFs) of predicted probabilistic distributions with the ground truth (citar). The CRPS has the same dimension as the ground truth. The formula is as follows:

$$\mathrm{CRPS} = \frac{1}{N}\sum_{t}^{N}\int (P(\widehat{B}(t) \leq x) - P(B(t) \leq x))^2 \, \mathrm{d}x \tag{2.11}$$

where $P(\widehat{B}(t) \leq x)$ are the CDFs of the probabilistic forecasts, and $P(B(t) \leq x)$ are the "CDFs" of the observations. Since SVDKL is gaussian processes based model, the definition of CRPS assume a normally distribution in $(P(\widehat{B}(t) \leq x)$ estimates. The $P(B(t) \leq x)$ is estimated by the empirical CDF, which is regarded as a step function because the data are discrete values. A lower CRPS score suggests improved uncertainty estimation performance. A lower CRPS values suggest that the uncertainty estimation performed better.

2.4 Results

2.4.1 Cloud masking

The Figure 2.4 and Table 2.1 show BOA, PA, and UA density error curves and summary statistics for the first experiment. For all algorithms, BOA and PA values exhibited a well-defined binomial error distribution with peak modes of different intensities. Taking only into account the three algorithms with the highest BOA (KappaMask L2A, Fmask, and KappaMask L1C), we found that the mode of the secondary peak is close to 0.5 and 0 for BOA and PA, respectively. At least 5 % of the total IPs are contained by this secondary distribution (see PA_{low} in Table 2.1). On the other side, the major peak's mode is close to 0.90 and 0.95 for BOA and PA, holding the 66 % of the IPs (see PA_{high} in Table 2.1). These results indicate that more than half of cloud pixels are easily recognizable. A simple visual examination reveals that semitransparent clouds are the primary cause of the secondary peak's formation. Low-thickness clouds, such as cirrus and haze tend to produce more omission errors independent of the cloud detection algorithm. This can be explained because modules for semitransparent clouds are simply a conservative threshold in the cirrus band (B10). Besides, semitransparent clouds are either ignored or unfairly represented in most datasets European Space Agency 2019. This particular flaw does not occur in cloudSEN12. Therefore, a simple regional adjustment of semitransparent cloud module parameters using this dataset should bring a significant improvement. Figure 2.4 demonstrates furthermore that not all algorithms exhibit the same behaviour. On the basis of the PA and UA metrics, we may differentiate between two types of algorithms: cloud conservative (CD-FCNN, QA60, and Sen2Cor) and non-cloud conservative (KappaMask, Fmask, and s2cloudless).

The first group exhibits high UA values at the expense of worsening PA. As observed in the PA heatline plot, these algorithms show a pronounced bimodal distribution and a wide interquartile range, with more than half of the IPs exhibiting PA values below 0.5. Considering the high temporal resolution of SEN2 imagery, it seems unsuitable to utilize cloud-conservative techniques, except for extremely cloudy regions where each clear pixel is crucial European Space

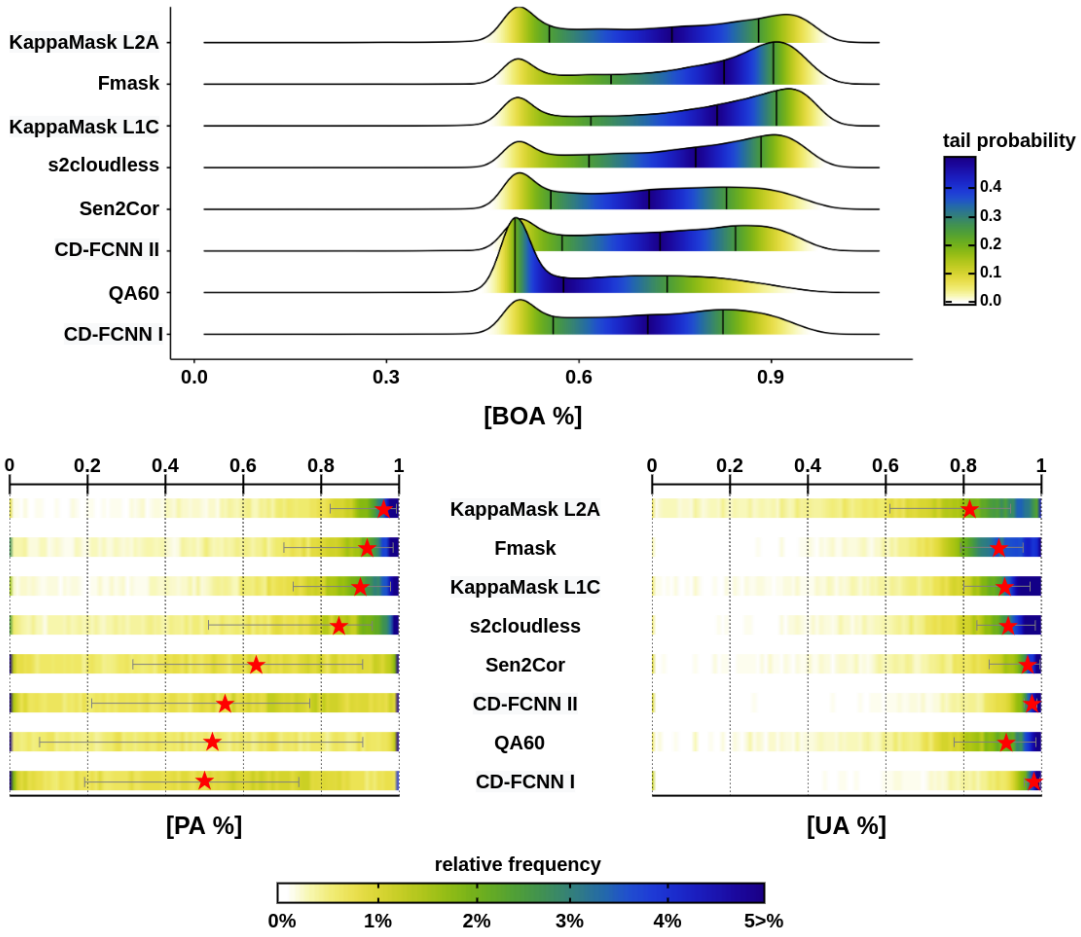

Figure 2.4: BOA, PA, and UA comparison for the cloudSEN12 dataset. The upper figure depicts BOA density estimations for all cloudSEN12 IPs high-quality. The colors reflect the tail probability estimated by $0.5 - abs(0.5 - ecdf)$. The vertical black lines drawn represent the first, second, and third quartiles, respectively. The heatlines in the lower figure shows the PA and UA value distribution. The red stars shows the median and the gray lines the 25th and 75th percentiles.

Agency 2019. On the other hand, in non-cloud conservative algorithms, over half of all IPs have PA values greater than 0.9 (see column PA_{high} in Table 2.1), but as a result, the UA_{high} metric decrease significantly. Based on BOA estimates, we may conclude that QA60 is the most unreliable algorithm, failing to distinguish both cloud and non-cloud pixels. Whereas, KappaMask level 2A is clearly the best at detecting clouds, even semitransparent clouds that other algorithms usually overlook. However, the main drawback of KappaMask level 2A is that it quite overestimates clouds under specific land cover types, such as mountains, open/enclosed water bodies, and coastal environments. It explains why almost 70 % of all IPs present a UA metric below 0.9 (see Table 2.1 and Figure 2.4). Considering that the L1C and L2A versions of KappaMask are trained

on a relatively small dataset from Northern Europe, it is expected that utilizing a larger dataset should lead to better results. Finally, Fmask, KappaMask level 1C, and s2cloudlless provide a more balanced and stable solution, with inaccuracies evenly distributed across different cloud types and land covers. Hence it makes them suitable for creating cloud-free composites over broad areas.

Table 2.1: Metrics based on the percentage of IPs with PA/UA values less than 0.1 (low), 0.1 to 0.9 (middle), and more than 0.9 (high). Values closest to one in the "high" group are better, whereas values close to zero in the other two groups are the ideal. The best values for each metric have been highlighted in bold.

Experiment	CD algorithm	PA_{low}%	PA_{int} %	PA_{high} %	UA_{low} %	UA_{int} %	UA_{high} %
First	**KappaMask L2A**	**2.22**	**29.87**	**67.91**	0.96	67.76	31.28
	Fmask	5.22	38.37	56.41	**0.19**	53.56	46.25
	KappaMask L1C	3.49	43.24	53.27	0.48	41.82	57.7
	s2cloudless	5.41	49.75	44.84	0.22	37.94	61.85
	Sen2Cor	10.21	62.76	27.02	0.62	26.72	72.65
	CD-FCNN II	15.54	69.8	14.66	0.58	13.8	85.62
	QA60	21.36	50.31	28.33	0.58	43.25	56.17
	CD-FCNN I	17.71	71.34	10.95	1.04	**12.88**	**86.08**
Second	**KappaMask L2A**	40.04	57.06	2.9	14.71	38.86	46.43
	KappaMask L1C	29.93	58.59	**11.48**	21.89	61.27	16.85
	Sen2Cor	63.76	**35.88**	0.36	**9.27**	**18.64**	**72.09**
	Fmask	**22.56**	74.84	2.59	17.57	76.48	5.95

2.4.2 Cloud cover

The Table 2.2 and Figure 2.5

Table 2.2: Benchmarking of cloud cover methods. The best two values for each metric have been highlighted in bold. CF means cloud-free.

	Model	**MAE**	**RMSE**	**CF 1> PA**	**CF 5> PA**	**CF 10> PA**	**CF >1 UA**	**CF >5 UA**	**CF >10 UA**
CloudSEN12	ResNet-18	**0.069**	**0.123**	**0.918**	**0.935**	**0.950**	0.749	**0.697**	**0.631**
	DKL	**0.080**	**0.147**	0.019	0.479	0.763	0.636	0.776	0.676
	DKL + sigma	0.145	0.184	0	0	0.202	0	0	0.766
	DKL - sigma	0.145	0.184	0.907	0.924	0.940	0.480	0.438	0.396
Knowledge Driven	Fmask	0.115	0.208	**0.853**	**0.901**	0.901	0.789	0.633	0.537
	Sen2Cor	0.161	0.251	0.763	0.830	0.862	0.653	0.488	0.411
	QA60	0.194	0.295	0.825	0.851	0.859	0.52	0.436	0.380
Data Driven	KappaMask L1C	0.131	0.241	0.721	0.772	0.808	**0.815**	0.630	0.528
	KappaMask L2A	0.199	0.317	0.501	0.603	0.648	**0.813**	**0.66**	**0.568**
	s2cloudless	0.116	0.194	0.792	**0.901**	**0.930**	0.753	0.593	0.509
	DL-L8S2-UV RGB	0.211	0.311	0.361	0.507	0.583	0.800	0.629	0.551
	DL-L8S2-UV RGBSWIR	0.195	0.288	0.402	0.521	0.597	0.812	0.625	0.538

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

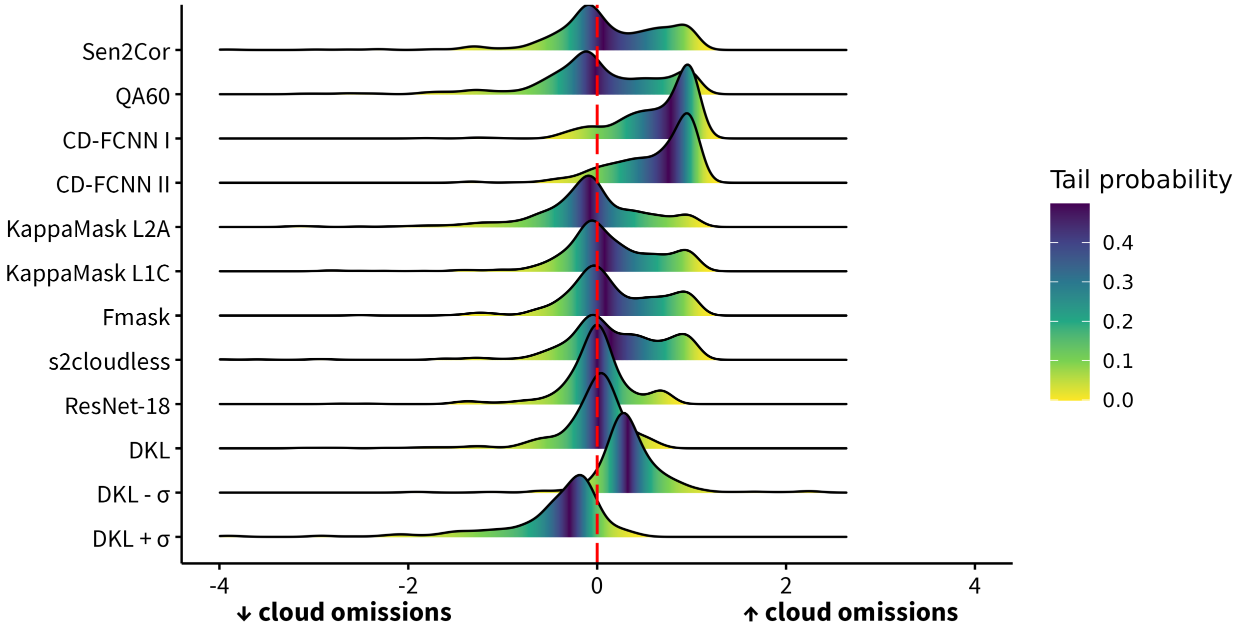

Figure 2.5: Residual density estimations for all cloudSEN12 IPs high-quality in the test dataset. y. The colors reflect the tail probability estimated by $0.5 - abs(0.5 - ecdf)$.

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

2.4.3 Model uncertainty

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

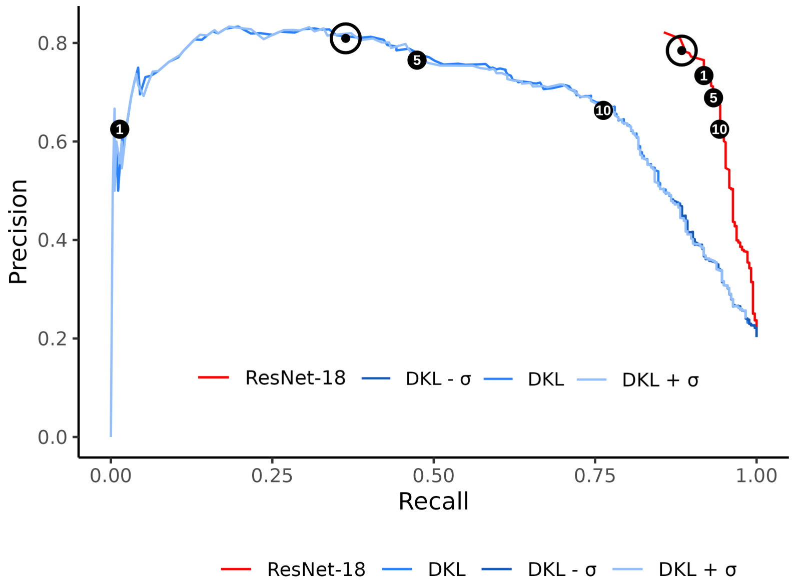

Figure 2.6: PA vs UA curves for the SVDKL and ResNet-18. The 1, 5, and 10 thresholds are represented by black spots on the plot. The double points represent the best value for UA and PA, based on a rule of two and one, respectively.

2.5 Discussions

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

2.6 Conclusions

TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO TODO

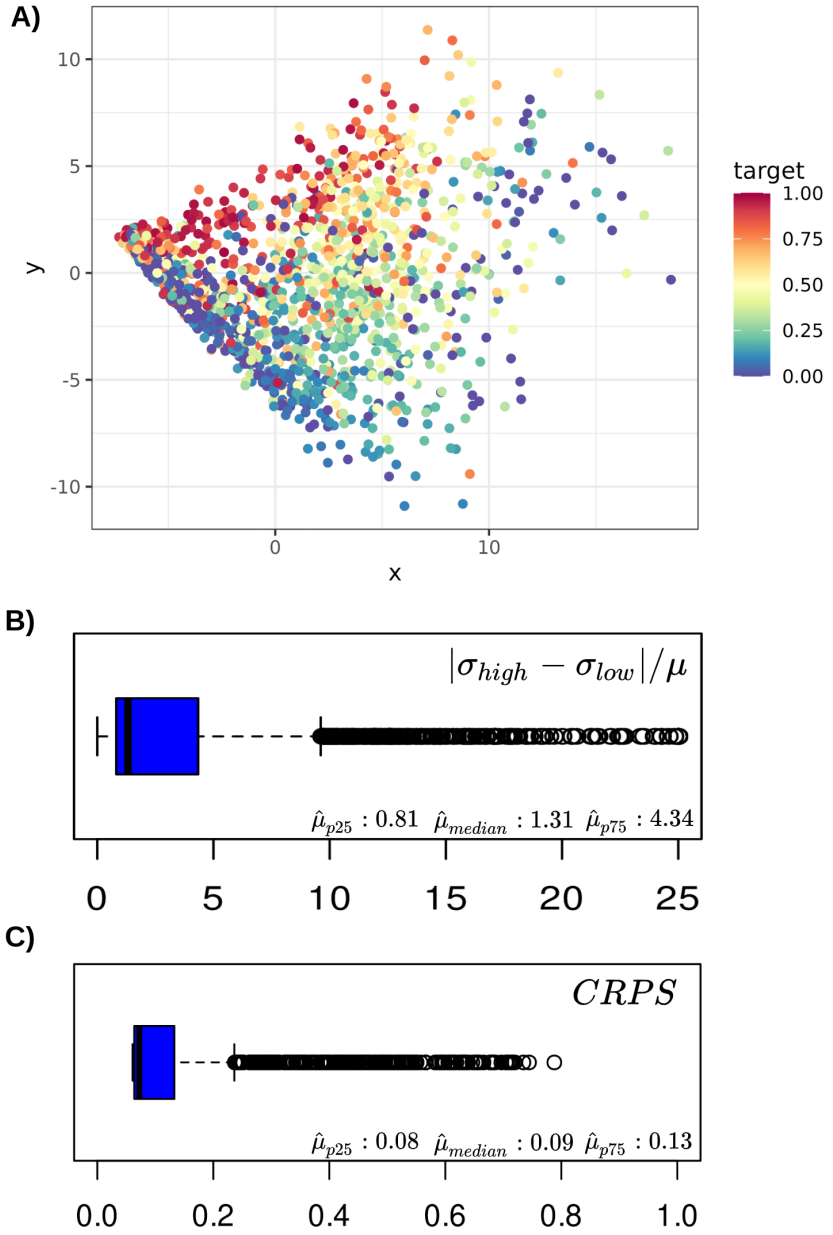

Figure 2.7: Summary of the urcentainty results for the SVDKL. A) A principal component analysis (PCA) on the last layer of ResNet-18, colors represents the cloud coverage. B) Coefficient of variation. C) CRPS.

I may not have lived long, but I am certain of one thing. If there is a type of person capable of changing something, it is someone who is willing to sacrifice what he values most!. He is the type of person who, in order to confront a monster, is capable of losing his own humanity. A person who is unable to make a sacrifice may be unable to change anything!

— Armin Arlelt

Conclusion

If we don't want Conclusion to have a chapter number next to it, we can add the `{-}` attribute.

More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

This paragraph, by contrast, *will* be indented as it should because it is not the first one after the 'More info' heading. All hail LaTeX. (If you're reading the HTML version, you won't see any indentation - have a look at the PDF version to understand what in the earth this section is babbling on about).

Appendices

A
Appendix - Code

This first appendix includes the non-stationary gaussian processes in Gpytorch:

In 02-rmd-basics-code.Rmd

And here's another one from the same chapter, i.e. Chapter ??:

B

Appendix - Figures

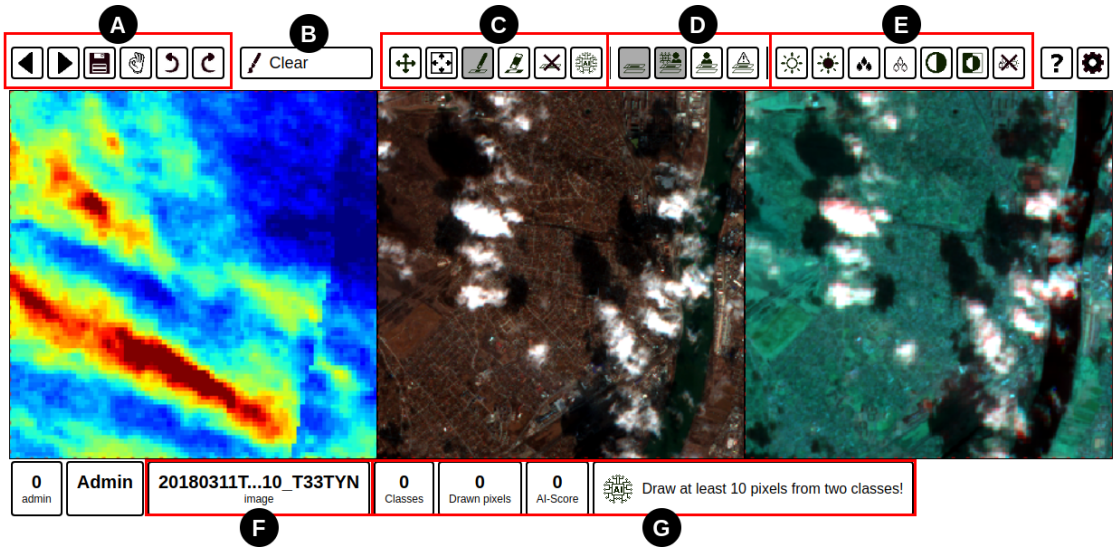

Figure S1: IRIS (Intelligently Reinforced Image Segmentation) graphical user interface. There are seven feature bars in it. A) Edit and navigation bar. B) Select drawing semantic classes. C) Draw bar; the last bottom executes the GBDT algorithm that filling out the mask using prior manual annotations, D) Testing bar, it helps to compare human and AI annotations. F) Image metadata, it display image thumbnail and location using Google maps. E) Image contrast bar which change image brightness and saturation. G) Machine learning summary support, that shows GBDT performance metrics.

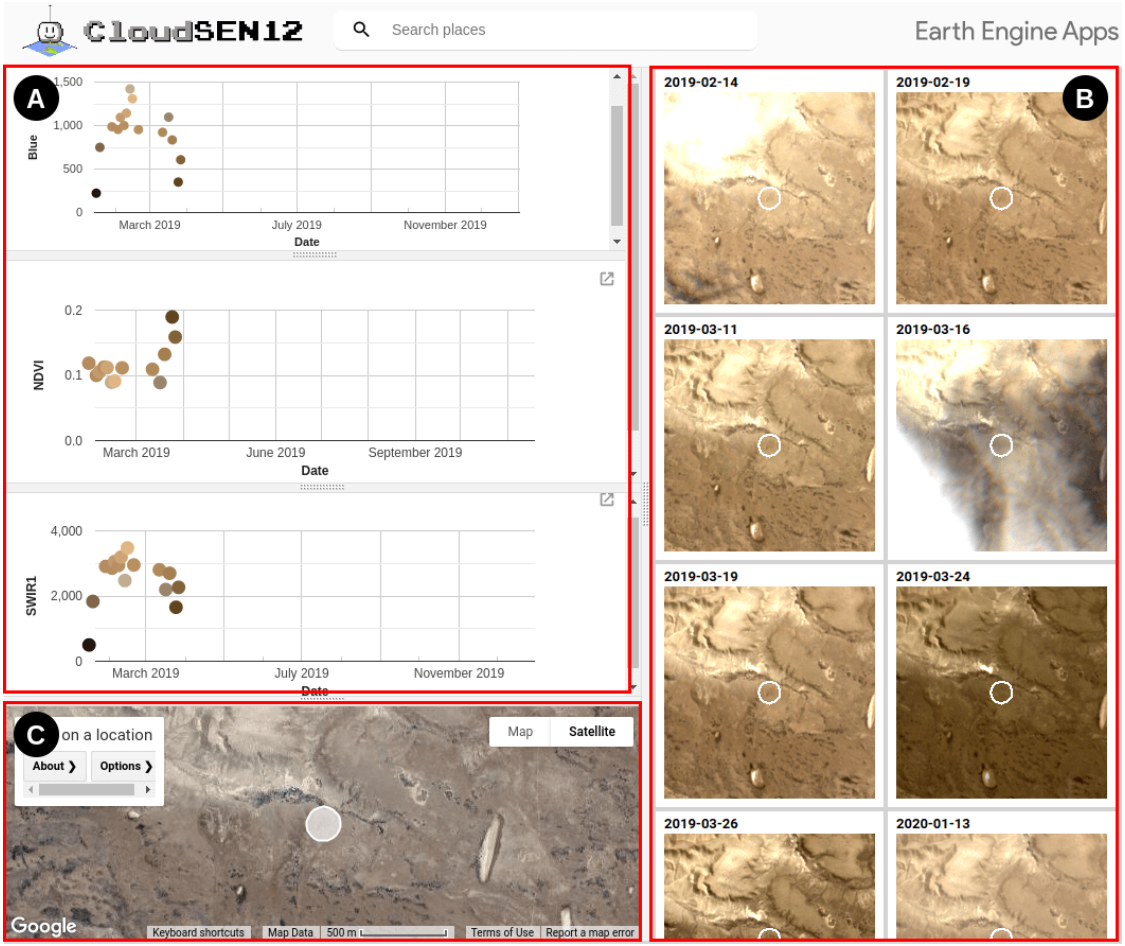

Figure S2: Three main cloudApp panels. A) Display time series for Blue, SWIR1 bands and NDVI for all images in a one-year moving window. B) Inspect image thumbnails; the white circle's values are averaged and displayed in panel A. C) Map display for showing the image patch's centroid.

Bibliography

Aybar, Cesar et al. (2020). “rgee: An R package for interacting with Google Earth Engine”. In: *Journal of Open Source Software* 5.51, p. 2272. DOI: 10.21105/joss.02272.

Bach, Francis (2013). “Sharp analysis of low-rank kernel matrix approximations”. In: *Conference on Learning Theory*. PMLR, pp. 185–209.

Baetens, Louis, Camille Desjardins, and Olivier Hagolle (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”. In: *Remote Sensing* 11.4, pp. 1–25. DOI: 10.3390/rs11040433.

Bai, Ting et al. (2016). “Cloud detection for high-resolution satellite imagery using machine learning and multi-feature fusion”. In: *Remote Sensing* 8.9, pp. 1–21. DOI: 10.3390/rs8090715.

Bivand, Roger et al. (2017). “Package ‘rgeos’”. In: *The Comprehensive R Archive Network (CRAN)*.

Buchhorn, Marcel et al. (2020). “Copernicus Global Land Service: Land Cover 100m: Collection 3: epoch 2015: Globe (Version V3.0.1)”. In: *Zenodo*, pp. 1–14.

Castillo-Navarro, Javiera et al. (2020). “Semi-Supervised Semantic Segmentation in Earth Observation: The MiniFrance Suite, Dataset Analysis and Multi-task Network Study”. In: arXiv: 2010.07830. URL: http://arxiv.org/abs/2010.07830.

Chen, Bin et al. (2017). “Spatially and Temporally Weighted Regression: A Novel Method to Produce Continuous Cloud-Free Landsat Imagery”. In: *IEEE Transactions on Geoscience and Remote Sensing* 55.1, pp. 27–37. DOI: 10.1109/TGRS.2016.2580576.

Cilli, Roberto et al. (2020). “Machine Learning for Cloud Detection of Globally Distributed Sentinel-2 Images”. In: ii, pp. 1–17. DOI: 10.3390/rs12152355.

Cordts, Marius et al. (2016). “The Cityscapes Dataset for Semantic Urban Scene Understanding”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2016-Decem, pp. 3213–3223. DOI: 10.1109/CVPR.2016.350. arXiv: 1604.01685.

Cunningham, John P, Krishna V Shenoy, and Maneesh Sahani (2008). “Fast Gaussian process methods for point process intensity estimation”. In: *Proceedings of the 25th international conference on Machine learning*, pp. 192–199.

Demmel, James W (1997). *Applied numerical linear algebra*. SIAM.

Der Kiureghian, Armen and Ove Ditlevsen (2009). “Aleatory or epistemic? Does it matter?” In: *Structural safety* 31.2, pp. 105–112.

Domnich, Marharyta et al. (2021). “KappaMask: Ai-based cloudmask processor for sentinel-2”. In: *Remote Sensing* 13.20. DOI: 10.3390/rs13204100.

Dong, Kun et al. (2017). “Scalable log determinants for Gaussian process kernel learning”. In: *Advances in Neural Information Processing Systems* 30.

Ebel, Patrick et al. (2020). “Multi-sensor data fusion for cloud removal in global and all-season sentinel-2 imagery”. In: *arXiv*, pp. 1–13. DOI: 10.1109/tgrs.2020.3024744. arXiv: 2009.07683.

European Space Agency (2019). *CEOS-WGCV ACIX II CMIX Atmospheric Correction Inter-comparison Exercise Cloud Masking Inter-comparison Exercise 2nd workshop*. 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.

Fernandez-Moran, Roberto et al. (2021). "Towards a novel approach for Sentinel-3 synergistic OLCI/SLSTR cloud and cloud shadow detection based on stereo cloud-top height estimation". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 181, pp. 238–253. DOI: https://doi.org/10.1016/j.isprsjprs.2021.09.013.

Foga, Steve et al. (2017). "Cloud detection algorithm comparison and validation for operational Landsat data products". In: *Remote Sensing of Environment* 194, pp. 379–390. DOI: 10.1016/j.rse.2017.03.026. URL: http://dx.doi.org/10.1016/j.rse.2017.03.026.

Francis, Alistair et al. (Nov. 2020). *Sentinel-2 Cloud Mask Catalogue*. Zenodo. DOI: 10.5281/zenodo.4172871. URL: https://doi.org/10.5281/zenodo.4172871.

Frantz, David (2019). "FORCE—Landsat+ Sentinel-2 analysis ready data and beyond". In: *Remote Sensing* 11.9, p. 1124.

Frantz, David et al. (2018). "Improvement of the Fmask algorithm for Sentinel-2 images: Separating clouds from bright surfaces based on parallax effects". In: *Remote Sensing of Environment* 215.April, pp. 471–481. DOI: 10.1016/j.rse.2018.04.046.

Gardner, Jacob et al. (2018). "Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration". In: *Advances in neural information processing systems* 31.

Ghasemian, Nafiseh and Mehdi Akhoondzadeh (2018). "Introducing two Random Forest based methods for cloud detection in remote sensing images". In: *Advances in Space Research* 62.2, pp. 288–303. DOI: 10.1016/j.asr.2018.04.030. URL: https://doi.org/10.1016/j.asr.2018.04.030.

Giuliani, Gregory et al. (2019). "Earth observation open science: enhancing reproducible science using data cubes". In: *Data* 4.4, pp. 4–9. DOI: 10.3390/data4040147.

Gomes, Vitor C.F., Gilberto R. Queiroz, and Karine R. Ferreira (2020). "An overview of platforms for big earth observation data management and analysis". In: *Remote Sensing* 12.8, pp. 1–25. DOI: 10.3390/RS12081253.

Gorelick, Noel et al. (2017). "Google Earth Engine: Planetary-scale geospatial analysis for everyone". In: *Remote Sensing of Environment* 202, pp. 18–27. DOI: 10.1016/j.rse.2017.06.031. URL: https://doi.org/10.1016/j.rse.2017.06.031.

Grolemund, Garrett and Hadley Wickham (2011). "Dates and times made easy with lubridate". In: *Journal of statistical software* 40, pp. 1–25.

Hagolle, O. et al. (2017). "MAJA ATBD Algorithm Theoretical Basis Document". In: *MAJA-TN-WP2-030 V1.0 2017/Dec/07* 18639, pp. 0–39. URL: http://www.cesbio.ups-tlse.fr/multitemp/?p=12432.

Harris, Charles R et al. (2020). "Array programming with NumPy". In: *Nature* 585.7825, pp. 357–362.

Hata, Hideaki et al. (2022). "GitHub Discussions: An exploratory study of early adoption". In: *Empirical Software Engineering* 27.1. DOI: 10.1007/s10664-021-10058-6. arXiv: 2102.05230.

Hijmans, Robert J et al. (2015). "Package 'raster'". In: *R package* 734.

Hollstein, André et al. (2016). "Ready-to-use methods for the detection of clouds, cirrus, snow, shadow, water and clear sky pixels in Sentinel-2 MSI images". In: *Remote Sensing* 8.8, pp. 1–18. DOI: 10.3390/rs8080666.

Hora, Stephen C (1996). "Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management". In: *Reliability Engineering & System Safety* 54.2-3, pp. 217–223.

Hughes, Lloyd H., Michael Schmitt, et al. (2018). "Identifying Corresponding Patches in SAR and Optical Images with a Pseudo-Siamese CNN". In: *IEEE Geoscience and Remote*

Sensing Letters 15.5, pp. 784–788. DOI: 10.1109/LGRS.2018.2799232. arXiv: 1801.08467.

Hughes, M. Joseph and Robert Kennedy (2019). "High-quality cloud masking of landsat 8 imagery using convolutional neural networks". In: *Remote Sensing* 11.21. DOI: 10.3390/rs11212591.

Karra, Krishna et al. (2021). "Global land use / land cover with Sentinel 2 and deep learning". In: pp. 4704–4707. DOI: 10.1109/igarss47720.2021.9553499.

Krishnamoorthy, Aravindh and Deepak Menon (2013). "Matrix inversion using Cholesky decomposition". In: *2013 signal processing: Algorithms, architectures, arrangements, and applications (SPA)*. IEEE, pp. 70–72.

Li, Liyuan, Xiaoyan Li, et al. (2021). "A review on deep learning techniques for cloud detection methodologies and challenges". In: *Signal, Image and Video Processing*. DOI: 10.1007/s11760-021-01885-7. URL: https://doi.org/10.1007/s11760-021-01885-7.

Li, Yansheng, Wei Chen, et al. (2020). "Accurate cloud detection in high-resolution remote sensing imagery by weakly supervised deep learning". In: *Remote Sensing of Environment* 250.August, p. 112045. DOI: 10.1016/j.rse.2020.112045. URL: https://doi.org/10.1016/j.rse.2020.112045.

"LightGBM: A highly efficient gradient boosting decision tree" (2017). In: *Advances in Neural Information Processing Systems* 2017-Decem.Nips, pp. 3147–3155.

López-Puigdollers, Dan, Gonzalo Mateo-García, and Luis Gómez-Chova (2021). "Benchmarking deep learning models for cloud detection in landsat-8 and sentinel-2 images". In: *Remote Sensing* 13.5, pp. 1–20. DOI: 10.3390/rs13050992.

Louis, Jérôme et al. (2016). "Sentinel-2 SEN2COR: L2A processor for users". In: *European Space Agency, (Special Publication) ESA SP* SP-740.May, pp. 9–13.

Lynch, David K et al. (2002). *Cirrus*. Oxford University Press.

Mahajan, Seema and Bhavin Fataniya (2020). "Cloud detection methodologies: variants and development—a review". In: *Complex & Intelligent Systems* 6.2, pp. 251–261. DOI: 10.1007/s40747-019-00128-0. URL: https://doi.org/10.1007/s40747-019-00128-0.

Mahecha, Miguel D. et al. (2020). "Earth system data cubes unravel global multivariate dynamics". In: *Earth System Dynamics* 11.1, pp. 201–234. DOI: 10.5194/esd-11-201-2020.

Mateo-Garcia, Gonzalo et al. (Mar. 2021). "Towards global flood mapping onboard low cost satellites with machine learning". en. In: *Scientific Reports* 11.1, p. 7249. DOI: 10.1038/s41598-021-86650-z.

Mateo-García, G. et al. (2021). "Cross-Sensor Adversarial Domain Adaptation of Landsat-8 and Proba-V Images for Cloud Detection". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, pp. 747–761. DOI: 10.1109/JSTARS.2020.3031741.

Mateo-García, Gonzalo et al. (2020). "Transferring deep learning models for cloud detection between Landsat-8 and Proba-V". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 160. (DOI:10.1016/j.isprsjprs.2019.11.024), pp. 1–17. DOI: 10.1016/j.isprsjprs.2019.11.024.

Melchiorre, Andrea, Luigi Boschetti, and David P. Roy (2020). "Global evaluation of the suitability of MODIS-Terra detected cloud cover as a proxy for Landsat 7 cloud conditions". In: *Remote Sensing* 12.2, pp. 1–16. DOI: 10.3390/rs12020202.

Meraner, Andrea et al. (2020). "Cloud removal in Sentinel-2 imagery using a deep residual neural network and SAR-optical data fusion". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 166.January, pp. 333–346. DOI: 10.1016/j.isprsjprs.2020.05.013. URL: https://doi.org/10.1016/j.isprsjprs.2020.05.013.

Mohajerani, Sorour and Parvaneh Saeedi (2019). "Cloud-Net: An End-To-End Cloud Detection Algorithm for Landsat 8 Imagery". In: *International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 1029–1032. DOI: 10.1109/IGARSS.2019.8898776. arXiv: 1901.10077.

— (2020). "Cloud-Net+: A cloud segmentation CNN for landsat 8 remote sensing imagery optimized with filtered jaccard loss function". In: *arXiv*, pp. 1–12. arXiv: 2001.08768.

Mrziglod, John (2019). *IRIS - Intelligence foR Image Segmentation*.

Ooms, Jeroen (2020). "magick: Advanced graphics and image-processing in R". In: *R package version* 2.1.

Pebesma, Edzer (2018). "Simple features for R: Standardized support for spatial vector data". In: *R Journal* 10.1, pp. 439–446. DOI: 10.32614/rj-2018-009.

— (2020). "stars: Spatiotemporal arrays, raster and vector data cubes". In: *R package version 0.4–1 ed2020 https://CRAN. R-project. org/package= stars*.

Pekel, Jean François et al. (2016). "High-resolution mapping of global surface water and its long-term changes". In: *Nature* 540.7633, pp. 418–422. DOI: 10.1038/nature20584. URL: http://dx.doi.org/10.1038/nature20584.

Qiu, Shi, Zhe Zhu, and Binbin He (2019). "Remote Sensing of Environment Fmask 4 . 0 : Improved cloud and cloud shadow detection in Landsats 4 – 8 and Sentinel-2 imagery". In: *Remote Sensing of Environment* 231.May, p. 111205. DOI: 10.1016/j.rse.2019.05.024. URL: https://doi.org/10.1016/j.rse.2019.05.024.

Qiu, Shi, Zhe Zhu, and Curtis E. Woodcock (2020). "Cirrus clouds that adversely affect Landsat 8 images: What are they and how to detect them?" In: *Remote Sensing of Environment* 246.September 2019, p. 111884. DOI: 10.1016/j.rse.2020.111884. URL: https://doi.org/10.1016/j.rse.2020.111884.

Rasmussen, Carl Edward (2003). "Gaussian processes in machine learning". In: *Summer school on machine learning*. Springer, pp. 63–71.

Richter, R and D Schläpfer (2019). "Atmospheric and topographic correction (ATCOR theoretical background document)". In: *DLR IB*, pp. 564–03.

Rittger, Karl et al. (2020). "Canopy Adjustment and Improved Cloud Detection for Remotely Sensed Snow Cover Mapping". In: *Water Resources Research* 56.6, pp. 1–20. DOI: 10.1029/2019WR024914.

Roberts, David R. et al. (2017). "Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure". In: *Ecography* 40.8, pp. 913–929. DOI: 10.1111/ecog.02881.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *IEEE Access* 9, pp. 16591–16603. DOI: 10.1109/ACCESS.2021.3053408. arXiv: 1505.04597.

Rußwurm, Marc et al. (2020). "Meta-learning for few-shot land cover classification". In: *Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops*, pp. 200–201.

Sanchez, Alber Hamersson et al. (2020). "Comparison of Cloud Cover Detection Algorithms on Sentinel–2 Images of the Amazon Tropical Forest". In: *Remote Sensing* 12.8, p. 1284. DOI: 10.3390/rs12081284.

Sassen, Kenneth and Zhien Wang (2008). "Classifying clouds around the globe with the CloudSat radar: 1-year of results". In: *Geophysical research letters* 35.4.

Schmitt, A. and A. Wendleder (2018). "SAR-sharpening in the Kennaugh framework applied to the fusion of multi-modal SAR and optical images". In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4.1, pp. 133–140. DOI: 10.5194/isprs-annals-IV-1-133-2018.

Schmitt, M., L. H. Hughes, et al. (2018). "Colorizing sentinel-1 SAR images using a variational autoencoder conditioned on Sentinel-2 imagery". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* 42.2, pp. 1045–1051. DOI: 10.5194/isprs-archives-XLII-2-1045-2018.

Singh, Praveer and Nikos Komodakis (2018). "Cloud-GAN: Cloud removal for sentinel-2 imagery using a cyclic consistent generative adversarial networks". In: *International Geoscience and Remote Sensing Symposium (IGARSS)* 2018-July, pp. 1772–1775. DOI: 10.1109/IGARSS.2018.8519033.

Skakun, Sergii et al. (2022). "Cloud Mask Intercomparison eXercise (CMIX): An evaluation of cloud masking algorithms for Landsat 8 and Sentinel-2". In: *Remote Sensing of Environment*, In press. DOI: 10.1016/j.rse.2022.112990.

Stillinger, Timbo et al. (2019). "Cloud Masking for Landsat 8 and MODIS Terra Over Snow-Covered Terrain: Error Analysis and Spectral Similarity Between Snow and Cloud". In: *Water Resources Research* 55.7, pp. 6169–6184. DOI: 10.1029/2019WR024932.

Tarrio, Katelyn et al. (2020). "Comparison of cloud detection algorithms for Sentinel-2 imagery". In: *Science of Remote Sensing* 2, p. 100010.

Tennekes, Martijn (2018). "tmap: Thematic Maps in R". In: *Journal of Statistical Software* 84, pp. 1–39.

Tiede, Dirk et al. (2021). "Investigating ESA Sentinel-2 products' systematic cloud cover overestimation in very high altitude areas". In: *Remote Sensing of Environment* 252, p. 112163. DOI: 10.1016/j.rse.2020.112163.

Ushey, Kevin et al. (2020). "reticulate: Interface to Python". In: *R package version* 1, p. 16.

Valavi, Roozbeh et al. (2019). "blockCV: An r package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models". In: *Methods in Ecology and Evolution* 10.2, pp. 225–232. DOI: 10.1111/2041-210X.13107.

Valdez, Calero, Martina Ziefle, and Michael Sedlmair (2017). "A Framework for Studying Biases in Visualization Research". In: *VIS 2017: Dealing with Cognitive Biases in Visualisations*. URL: http://eprints.cs.univie.ac.at/5258/1/calero-valdez2017framework.pdf.

Wei, Jing et al. (2020). "Cloud detection for Landsat imagery by combining the random forest and superpixels extracted via energy-driven sampling segmentation approaches". In: *Remote Sensing of Environment* 248.July, p. 112005. DOI: 10.1016/j.rse.2020.112005. URL: https://doi.org/10.1016/j.rse.2020.112005.

Wevers, Jan et al. (Dec. 2021). *IdePix for Sentinel-2 MSI Algorithm Theoretical Basis Document*. Version Version 1.0. DOI: 10.5281/zenodo.5788067. URL: https://doi.org/10.5281/zenodo.5788067.

Wickham, H et al. (2014). "Dplyr: A fast, consistent tool for working with data frame like objects, both in memory and out of memory". In: *R package version 0.7* 6.

Wickham, Hadley (2011). "ggplot2". In: *Wiley interdisciplinary reviews: computational statistics* 3.2, pp. 180–185.

Williams, Christopher K and Carl Edward Rasmussen (2006). *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA.

Wilson, Adam M. and Walter Jetz (2016). "Remotely Sensed High-Resolution Global Cloud Dynamics for Predicting Ecosystem and Biodiversity Distributions". In: *PLoS Biology* 14.3, pp. 1–20. DOI: 10.1371/journal.pbio.1002415.

Wilson, Andrew Gordon, Christoph Dann, and Hannes Nickisch (2015). "Thoughts on massively scalable Gaussian processes". In: *arXiv preprint arXiv:1511.01870*.

Winker, DM et al. (2010). "The CALIPSO mission: A global 3D view of aerosols and clouds". In: *Bulletin of the American Meteorological Society* 91.9, pp. 1211–1230.

Yamazaki, Dai et al. (2019). “MERIT Hydro: A High-Resolution Global Hydrography Map Based on Latest Topography Dataset”. In: *Water Resources Research* 55.6, pp. 5053–5073. DOI: 10.1029/2019WR024873.

Zekoll, Viktoria et al. (2021). “Comparison of masking algorithms for sentinel-2 imagery”. In: *Remote Sensing* 13.1, pp. 1–21. DOI: 10.3390/rs13010137.

Zhang, Y., B. Guindon, and J. Cihlar (2002). “An image transform to characterize and compensate for spatial variations in thin cloud contamination of Landsat images”. In: *Remote Sensing of Environment* 82.2-3, pp. 173–187. DOI: 10.1016/S0034-4257(02)00034-2.

Zhu, Xiao Xiang, Jingliang Hu, et al. (2019). “So2Sat LCZ42: A Benchmark Dataset for Global Local Climate Zones Classification”. In: *arXiv* 14.8, pp. 2–13. arXiv: 1912.12171.

Zhu, Xiao Xiang, Devis Tuia, et al. (2017). “Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources”. In: *IEEE Geoscience and Remote Sensing Magazine* 5.4, pp. 8–36. DOI: 10.1109/MGRS.2017.2762307.

Zupanc, Anze (2017). “Improving Cloud Detection with Machine Learning”. In: URL: https://medium.com/sentinel-hub/improving-cloud-detection-with-machine-learning-c09dc5d7cf13%20(accessed%20on%2001July%202021).