CI-Dataset and DetDSCI Methodology for Detecting Too Small and Too Large Critical Infrastructures in Satellite Images: Airports and Electrical Substations as Case Study

Francisco Pérez-Hernández, José Rodríguez-Ortega, Yassir Benhammou, Francisco Herrera, *Senior Member, IEEE*, and Siham Tabik

***Abstract*—The detection of critical infrastructures in large territories represented by aerial and satellite images is of high importance in several fields such as in security, anomaly detection, land use planning, and land use change detection. However, the detection of such infrastructures is complex as they have highly variable shapes and sizes, i.e., some infrastructures, such as electrical substations, are too small while others, such as airports, are too large. Besides, airports can have a surface area either small or too large with completely different shapes, which makes its correct detection challenging. As far as we know, these limitations have not been tackled yet in previous works. This article presents 1) a smart critical infrastructure (CI) dataset, named CI-dataset, organized into two scales, small and large scales critical infrastructures and 2) a two-level resolution-independent critical infrastructure detection (DetDSCI) methodology that first determines the spatial resolution of the input image using a classification model, then analyses the image using the appropriate detector for that spatial resolution. The present study targets two representative classes, airports and electrical substations. Our experiments show that DetDSCI methodology achieves up to 37.53% F1 improvement with respect to Faster R-CNN, one of the most influential detection models.**

***Index Terms*—Convolutional neuronal networks, detection, ortho-images, remote sensing images.**

I. Introduction

CRITICAL infrastructures are a type of human land use that are essential for the functioning of a society and economy [26], [31], [33]. Any threat to these facilities can cause severe problems. Examples of critical infrastructures include airports, electrical substations, and harbors among others. The detection of this type of infrastructures in high resolution ortho-images is of paramount importance in several fields such as security, land use planning, and change detection [5], [14], [23], [34].

Currently, deep CNNs have been largely used in the classification of high resolution ortho-images [6], [12], [33] as they achieve good accuracies specially in distinguishing objects of similar scales in images of the same size and same spatial resolution. Nevertheless, the detection of critical infrastructures with dissimilar sizes and scales, e.g., electrical substations that cover a surface area of the order of hundreds m^2 versus airports that can cover up to hundreds km^2, is still challenging. Besides, unlike bridges or motorways, infrastructures such as airports and electrical substations have large intraclass and interclass scale variations. Each airport has a completely different structure and shape when seen from space.

Detection task is addressed using remote sensing data and deep convolutional neural networks (CNNs). Remote sensing data are high resolution ortho-images that can be obtained from unmanned aerial vehicle (captured at height < 30 km and covers from 0,1 to $100 km^2$), planes (at height < 30 km and covers from 10 to $100 km^2$) or satellites (> 150 km 10–1000 Km^2) [30]. Obtaining large amounts of this type of data are expensive. Fortunately, few sources, such as Google Earth[1] and Bing Maps,[2] allow to download aerial and satellite images freely for the academic community. Nevertheless, most existing land use datasets are prepared only for training classification models, do not include neither annotations for training detection models nor information about the scale or zoom level of the images. As far as we know, none of the public databases prepared for training detection models provide images of some critical infrastructures like electrical substations.

This article presents two-level deep learning detection for different scale critical infrastructures (DetDSCI) methodology in ortho-images. We reformulate the problem of detecting critical infrastructures in ortho-images into two subproblems, the detection of small and large scale critical infrastructures. DetDSCI methodology consists of two stages as follows:

[1][Online]. Available: Google Earth: https://earth.google.com/web
[2]Bing Maps: https://www.bing.com/maps

Manuscript received September 20, 2021; revised November 5, 2021 and November 15, 2021; accepted November 16, 2021. Date of publication November 18, 2021; date of current version December 8, 2021. This work was partially supported by Trust-ReDaS (PID2020-119478GB-I00 Ministerio de Ciencia e Innovación), DeepL-ISCO (A-TIC-458-UGR18 Universidad de Granada/FEDER) and (B-TIC-722-UGR20 Consejería de Economía, Conocimiento, y Universidad from the Junta de Andalucía). *(Corresponding author: Francisco Pérez-Hernández.)*

The authors are with the Andalusian Research Institute in Data Science, and Computational Intelligence, University of Granada, 18071 Granada, Spain (e-mail: fperezhernandez@ugr.es; jrodriguez98@ugr.es; benhammouyassir2@gmail.com; herrera@decsai.ugr.es; siham@ugr.es).

Digital Object Identifier 10.1109/JSTARS.2021.3128994

1) The first stage is based on a spatial resolution classification model that analyses the 2000×2000 pixels input image to estimate its zoom level and hence determine the detector to be used in the next stage.
2) The second stage includes two expert detectors, one of them for small and the other for large critical infrastructures. Once the zoom level of the input image is determined by the first stage, the selected detector will analyze that input image according to its spatial resolution. Middle scale infrastructures can be detected by both detectors.

Addressing the detection of too small and too large scale critical infrastructures in remote sensing images independently on the spatial resolution can offer better performance. Our study targets two representative critical infrastructures, namely airports and electrical substations. As there are no public detection datasets that include both categories of critical infrastructures, we carefully built a specialized dataset, critical infrastructures dataset (CI-dataset). CI-dataset is organized into two subsets, small scale critical infrastructure (CI-SS) dataset with electrical substation class and large scale critical infrastructure (CI-LS) dataset with airport class.

The main contributions of this article can be summarized as follows:

1) Unlike the traditional process adopted to build most datasets, we followed a dynamic process to construct the high quality CI-dataset organised into two scales, CI-SS for small scale critical infrastructures and CI-LS for large scale critical infrastructures. This process can be used to include more types of infrastructures. CI-dataset is available through this link.[3]
2) We present DetDSCI methodology, a two-stages deep learning detection for dissimilar scale critical infrastructures in ortho-images. DetDSCI methodology first determines the spatial resolution of the input image then analyses it according to its spatial resolution using the appropriate expert detector. This methodology overcomes the baseline detectors trained on our high quality dataset. Code of DetDSCI methodology is available through this link.[4]

The rest of this article is organized as follows. First, a comprehensive review of related works is provided in Section II. Our DetDSCI methodology is presented in Section III. The dynamic process of building our CI-dataset is provided in Section IV. The experimental analysis carried out for the construction of CI-dataset and the evaluation of DetDSCI methodology are given in Section V. Finally, Section VI concludes this article.

II. Related Works

Related works that apply deep learning on remote sensing data can be broadly divided into two types, top-down and bottom-up works:

1) Top–down works, first build a large dataset with an important number of object-classes, mainly objects that can be recognized from remote sensing images, e.g., vehicles or soccer stadiums. Then, the studies analyze these images using a deep learning classification or detection models [6], [7], [10], [12], [19], [20], [28], [29], [33].
2) Bottom-up works focus on solving a specific problem that involves one or few object classes, e.g., airports [3], [4], [21], [32], [35], trees [2], [13], [15], [27], clouds [17], and whales [16]. Besides, some works [8], [9], [17], [24], [36] focus on designing new methods to further improve the detection, in general, in satellite images.

Our work belongs to the second category as our final objective is to build a good detector of two specific critical infrastructures, namely, airports and electrical substations. This section provides a brief summary of the current general datasets that include some critical infrastructures, the so-called top–down works (see Section II-A) then reviews the deep learning approaches used in bottom-up works (see Section II-B).

A. Top–Down Works

Most databases provided by top–down works are multiclass datasets that include some critical infrastructures, annotated for the task of image classification, which limits their usefulness. See summary in Table I where only a few datasets are prepared for the task of detection.

For example, in [33], the authors created LULC dataset organized into 21 classes. Each class contains 100 images of size 256×256 pixels. The authors in [6] provide a dataset named NWPU-RESISC45. This dataset is composed of 31.500 images of 256×256 pixels, in 45 classes with 700 images in each class. NWPU-RESISC45 includes images with a large variation in translation, spatial resolution, viewpoint, object pose, illumination, background, and occlusion. Besides, it has high within-class diversity and between-class similarity. Functional Map of the World (fMoW) [12] is a dataset containing a total of 523.846 images with a spatial resolution of 0.31 and 1.60 meters per pixel. It includes 62 classes with 132.716 instances from OpenStreetMap. These datasets are prepared for the image classification task and hence they are not useful for the detection task.

Examples of datasets prepared for the task of object detection are NWPU VHR-10, xView, DIOR, and DOTA. NWPU VHR-10 dataset [7] is organized into 10 classes, each class contains 800 images of width 1000 pixels. It contains mainly small scale objects such as airplane, ship, storage tank, baseball diamond, tennis court, basketball court, ground track field, harbor, bridge, and vehicle. Authors on [19] presented xView dataset for detecting 60 object-classes with over 1 million instances. These classes are focused on vehicles and small scale objects and the images have a width of 3000 pixels. DIOR, a new dataset was published on [20], where 23 463 images and 192 472 instances covered 20 object classes. DIOR dataset has a large range of object size variations and is focused on detection with a width on the images of 800 pixels. DOTA dataset [29] is composed of 15 classes of small scale objects with 2.806 images from

[3][Online]. Available: CI-dataset: https://dasci.es/transferencia/open-data/ci-dataset/

[4]DetDSCI methodology: https://github.com/FPerezHernandez92/DetDSCI-Methodology

TABLE I
CHARACTERISTICS OF GENERAL DATASETS THAT INCLUDE SOME CRITICAL INFRASTRUCTURES

Dataset	#Classes (#Infrastructure)	#Images (#Instances)	#Image width	Source	Resolution	Annotation
LULC[33]	21 (7)	2100 (2100)	256	National Map	30cm	Classification
NWPU RESISC45[6]	45 (13)	31500 (31500)	256	Google Earth	20cm-30cm	Classification
fMoW[12]	62 (25)	523846 (132716)	N/A	OpenStreetMap	31cm-1.6m	Classification
NWPU VHR-10[7]	10 (4)	800 (3651)	$\sim$1000	Google Earth	15cm-12m	Horizontal BB
xView[19]	60 (9)	1400 (1000000)	3000	DigitalGlobe	31cm	Horizontal BB
DIOR[20]	20 (11)	23463 (192472)	800	Google Earth	30cm-50cm	Horizontal BB
DOTA[29]	15 (6)	2806 (188282)	800$\sim$4000	Google Earth	15cm-12m	Oriented BB

Google Earth, where the total instances are 188.282. The size of the images is between 800 and 4.000 pixels, and they are labeled with oriented bounding boxes. Although the last four datasets are prepared for the task of object detection, they do not focus on any specific problem as they are all types of visible objects from space. In addition, none of these datasets includes electrical substations and only DIOR includes the airport category.

B. Bottom-Up Works

A large number of bottom-up works focus on improving the detection of airports. In [35], the authors propose a method using CNNs for airport detection on optical satellite images. The proposed method consists mainly of three steps, namely, region proposal, CNN identification, and localization optimization. The model was tested on an image data set, including 170 different airports and 30 nonairports. All the tested optical satellite images were collected from Google Earth with a resolution of 8m $\times$ 8m and a size of about 3000 $\times$ 3000 pixels. The method proposed in [3] first detects various regions on RSIs, then uses these candidate regions to train a CNN architecture. The sizes of the airport images were 3000 $\times$ 2000 pixels with a resolution of 1 m. A total of 92 images were collected. In [4], the authors developed a hard example mining and weight-balanced strategy to construct a novel end-to-end CNN for airport detection. They designed a hard example mining layer to automatically select hard examples by their losses and implement a new weight-balanced loss function to optimise CNN. The authors in [32] proposed an end-to-end airport detection method based on CNNs. Additionally, a cross-optimization strategy has been employed to achieve convolution layer sharing between the cascade region proposal networks and the subsequent multithreshold detection networks, and this approach significantly decreased the detection time. Once the airport is detected, they use an airplane detector to obtain these instances. To address the insufficiency of traditional models in detecting airports under complicated backgrounds from remote sensing images, authors in [21] proposed an end-to-end remote sensing airport hierarchical expression and detection model based on deep transferable CNNs.

In addition, several studies focus on improving the detection of general objects in remote sensing images. For example, in [36], the authors provided a remote sensing dataset called HRRSD and designed a CNN called HRCNN based on deformable proposal technique for improving the detection of these classes. In [8], the authors first, applied a dual attention feature enhancement (DAFE) module to selectively emphasise informative features from multiple resolutions. Then, introduced a context feature enhancement (CFE) module to fully leverage the abundant information emerged in remote sensing objects. The authors in [9] proposed a discriminate CNN (D-CNN) to classify remote sensing scenes. They demonstrated that D-CNN maps the images of the same scene close to each other, while images of different scenes are mapped very far from each other. In [24], the authors presented the GACL Net to improve the detection of small-scale objects in remote sensing images. The model uses the global features to guide the channel attention of the local convolutional features, and the axis-concentrated prediction process takes the single-axis pooling process to avoid coordinate prediction disturbance. The authors in [17] developed a CNN specially designed for cloud detection in optical remote sensing images.

TABLE II
CORRESPONDENCE BETWEEN SPATIAL RESOLUTION AND ZOOM LEVEL

Large critical infrastructures		Small critical infrastructures	
Zoom level	Spatial resolution (m^2/pixel)	Zoom level	Spatial resolution (m^2/pixel)
14	6.2	18	0.39
15	3.1	19	0.19
16	1.55	20	0.10
17	0.78	21	0.05
		22	0.02
		23	0.01

III. DetDSCI Methodology: Two-Level Deep Learning Detection for Different Scale Critical Infrastructure Methodology in Ortho-Images

This section presents DetDSCI methodology, which aims at addressing the detection of airports and electrical substations of very dissimilar sizes and shapes in large areas represented by satellite images, see illustration in Fig. 1. We define two broad ranges of spatial resolutions also called zoom levels, see correspondence between zoom level and spatial resolution in Table II. The first range includes zoom levels in [14,17] and the second range includes zoom levels in [18,23]. These intervals have been selected experimentally as described in the next section.

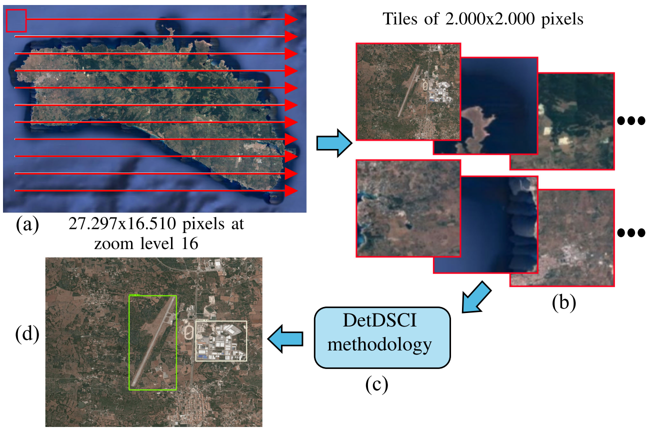


Fig. 1. DetDSCI methodology detection applied to the island of Menorca (Spain). (a) A sliding window processing approach. (b) Obtained 2000 × 2000 pixels crops. (c) DetDSCI methodology applied to each crop. (d) Output image with detection results.

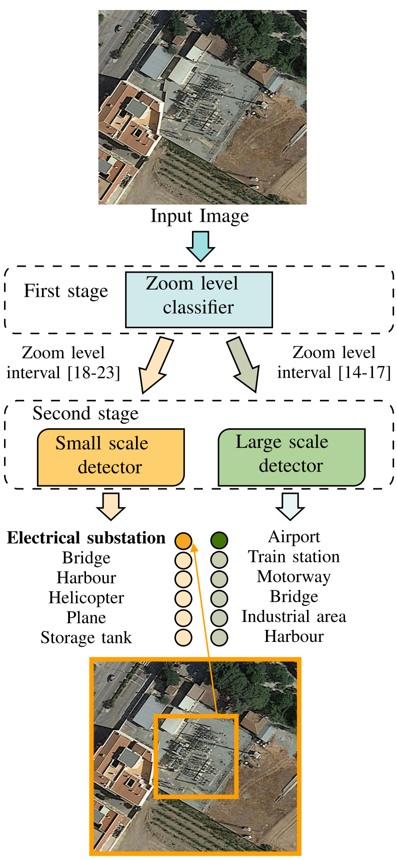


Fig. 2. DetDSCI methodology.

To reduce the number of FP due to the differences in different zoom levels, DetDSCI methodology first distinguishes between the two zoom level ranges and then applies the corresponding detector according to the spatial resolution of each input image. In particular, DetDSCI is actually a two stages pipeline as illustrated in Fig. 2. The first stage determines whether the input image belongs to the first or second zoom levels interval. Depending on the selected zoom level interval, the second stage analyses that image using the specialised detector on that specific group of critical infrastructures.

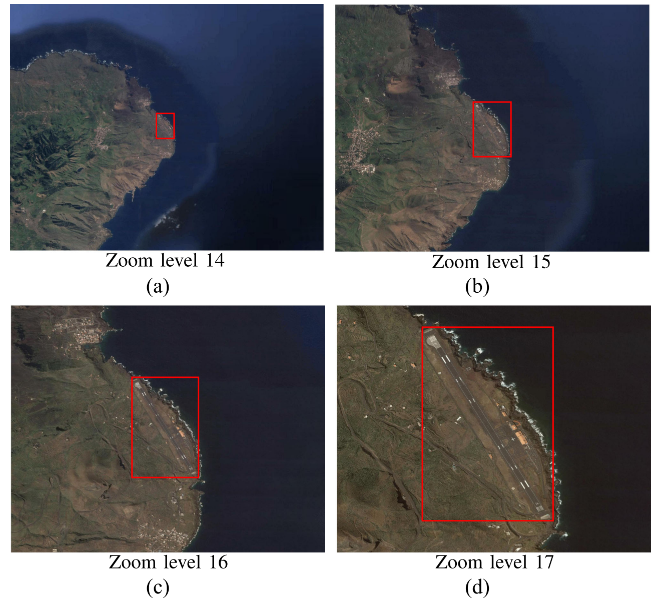


Fig. 3. Four images of El Hierro airport (latitude: 27.81402°N, longitude: -17.88518°W, Canary Islands, Spain) with zoom levels 14(a), 15(b), 16(c) and 17(d), obtained from Google Maps.

The next code summarizes the DetDSCI methodology:

```
 DetDSCI(image):
   zoom_level_image = ZoomLevelClassi-
fier(image)
   if zoom_level_image <= 17:
       class = LargeScaleDetec-
tor(image)
   elif zoom_level_image >= 18:
       class = SmallScaleDetec-
tor(image)
   return class
```

A. Stage 1: Estimating the Spatial Resolution of the Input Image

To distinguish between too large and too small critical infrastructures, we consider two zoom levels intervals, [14,17] and [18,23]. Too large infrastructures can be visually recognised in 2000 × 2000 pixels images of zoom levels 14, 15, 16, and 17. See an example in Fig. 3. While, too small scale infrastructures can be visually recognised in 2000 × 2000 pixels images of zoom levels 18, 19, 20, 21, 22, and 23. See an example in Fig. 4. Medium size infrastructure, such as bridges, can be included in both small and large groups.

In Figs. 3 and 4, we found out that electrical substations are difficult to recognize by the human eye in zoom level 18 and airports are difficult to recognize in zoom level 14. This is because the provided number of pixels with these zoom levels do not give enough information about the target objects. In parallel, some 2000 × 2000 pixel images cannot contain the entire airport

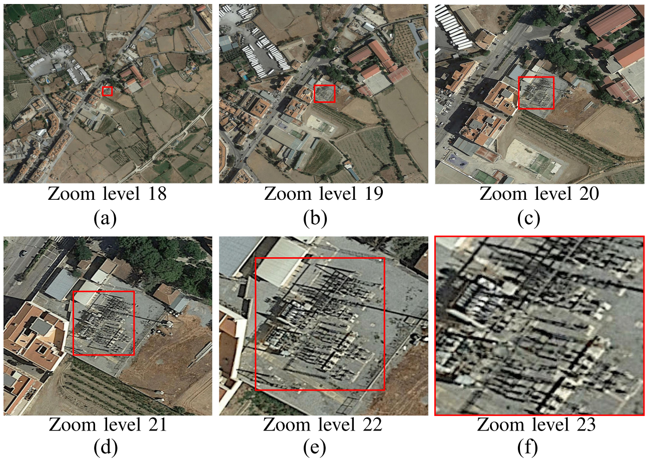


Fig. 4. Six images of Guadix electrical substation (latitude: 37.30853^{o}N, longitude: -3.12997^{o}W, Granada, Spain) with zoom levels 18(a), 19(b), 20(c), 21(d), 22(e) and 23(f), obtained from Google Maps.

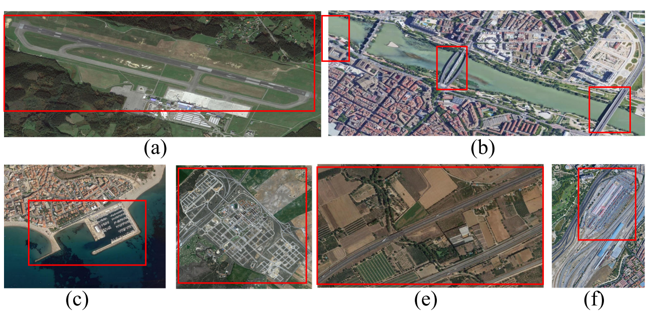


Fig. 5. Examples of the classes considered by the large infrastructure detection model, left to right: airport (a), bridges (b), harbor (c), industrial area (d), motorway (e), and train station(f).

at zoom level 17. Similarly, some 2000×2000 pixel images cannot contain the entire electrical substation in zoom level 23. In spite of all this, including these zoom levels in the training dataset improves the robustness of the detector as it can be seen in Section V-B1.

The first stage of DetDSCI distinguishes between these two intervals, large [14,17] and small [18,23] zoom levels interval. This stage is based on a binary classification model that analyses the input image to determine its zoom level interval and hence determines the most appropriate detector to be used in the second stage.

B. Stage 2: Detection of Critical Infrastructures

The zoom level interval estimated in the first stage will be used to guide the selection of the detector in the second stage. In particular, this stage is based on following two detection models.

1) The first detection model is applied to large scale infrastructures. It considers six infrastructure classes, namely airport, bridge, harbor, industrial area, motorway, and train station. Fig. 5 shows examples of these classes.
2) The second detection model is applied to small scale infrastructures. It considers six classes, namely electrical substation, bridge, plane, harbor, storage tank, and helicopter. Fig. 6 shows examples of these classes.

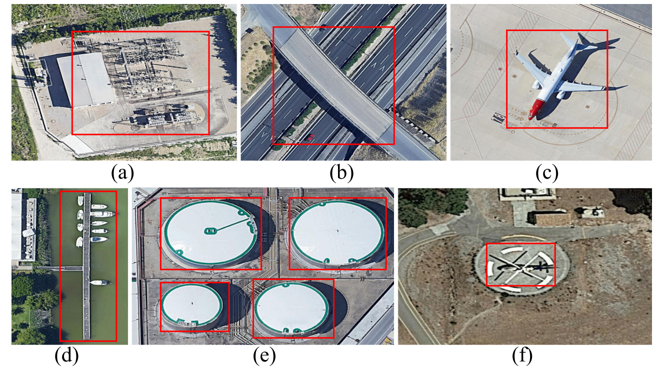


Fig. 6. Examples of the classes considered in the small infrastructure detection model, left to right: electrical substation (a), bridge (b), plane (c), harbor (d), storage tanks (e), and helicopter (f).

It is worth mentioning that the inclusion of new classes in both detectors was based on the preliminary experimental study explained in the next section.

IV. CI-Dataset Construction Guided by the Performance of Faster R-CNN

It is well known that building good quality models requires good quality datasets, also called smart data [25]. The concept of smart data includes all preprocessing methods that improve value and veracity of data. In the context of object detection, usually training datasets are first built then analyzed using machine learning models. This classical procedure is suitable only when the involved objects are of similar sizes and can be correctly identified at the same spatial resolution.

To overcome these limitations, we built the critical infrastructures dataset, CI-dataset, guided by the performance of one of the most robust detectors, namely Faster R-CNN. We organized CI-dataset into two subsets, one for small scale, CI-SS, and the other one for large scale, CI-LS critical infrastructures. The construction process of both subsets is dynamic and guided by the performance of Faster R-CNN detection model on the electrical substation class for CI-SS and the airport class for CI-LS. This section describes the construction process used to obtain the final high-quality CI-dataset for detecting electrical substations and airports.

The dynamic process guided by the detection model is based on three main steps:

1) *Step 1: Constructing the initial set for each target class:* First, we selected the combination of zoom levels at which the airports and the electrical substations can be recognized by the human eye. Then, we downloaded images for each one of these two classes with different zoom levels. Afterward, we selected the most suitable zoom levels combination guided by the performance of Faster R-CNN.
2) *Step 2: Extending the dataset with more object classes:* We analyzed all the object classes that can be confused with the target class and hence can cause false positives (FP). All these potential FP are obtained from public datasets and included in our CI-dataset. Then the performance of

TABLE III
NAMES OF THE TRAINING AND TEST SUBSETS OF THE CI-DATASET AND THE CORRESPONDING DETECTION MODEL CREATED AT EACH STEP OF THE PROCESS

	Train	Test	Detection model
Step 1	CI-SS_train_alpha	CI-SS_test_alpha	CI-SS_Det_alpha
Step 2	CI-SS_train_beta	CI-SS_test_stable	CI-SS_Det_beta
Step 3	CI-SS_train_stable	CI-SS_test_stable	CI-SS_Det_stable
Step 1	CI-LS_train_alpha	CI-LS_test_alpha	CI-LS_Det_alpha
Step 2	CI-LS_train_beta	CI-LS_test_stable	CI-LS_Det_beta
Step 3	CI-LS_train_stable	CI-LS_test_stable	CI-LS_Det_stable

the model is analyzed to select the final object classes to be included.

3) *Step 3: Further increasing the size of the training set:* We increased the number of instances of the final classes in the training set using new images from Google Maps.

For simplicity, we named the three different versions of the training, test datasets and detection model according to the construction step as described in Table III. At the end of this process, we obtained the final CI training and test datasets.

A. Step 1: Constructing the Initial Set for Each Target Class

The first process is to carefully select the zoom levels at which the considered objects fit in a 2000 $\times$ 2000 pixels image and can be recognised by the human eye. Ortho-images of this size can capture small scale critical infrastructures within 18–23 zoom levels (see Fig. 6) and large scale critical infrastructures within 14–17 zoom levels (see Fig. 5). For building CI-dataset, we used two services to visualize then download images from Google Maps, namely, SAS Planet[5] and Google Maps API.[6]

Although all selected zoom levels provide useful information for training the detection model, the lowest, 14 and 18, and highest zoom levels, 17 and 22 and 23, require specific manual preprocessing to fit 2000 $\times$ 2000 pixels[7] so that they can be used for training the detection model. For the test process, no preprocessing is applied and zoom levels 14 and 17 for large scale [see Fig. 7(a)] and 18, 22, and 23 for small scale [see Fig. 7(b)] infrastructures are discarded. That is, we consider zoom levels in [19,21] for the electrical substation and in [15,16] for the airport class, in the test set. Once the zoom levels are selected for the training process, the images of the target class are downloaded to build subsets CI-SS and CI-LS.

Finally, once the target class dataset is constructed, we analyzed all the combinations of zoom levels to determine, which one improves the learning process of the detection models. Guided by the performance of the Faster R-CNN on the target class, we discarded the zoom levels that did not help in the learning process of the detector.

[5][Online]. Available: SAS Planet: //www.sasgis.org/

[6]Google Maps API: //https://cloud.google.com/maps-platform

[7]Preprocessing includes fusing multiple tiles, cropping a tile and/or resizing the obtained image to 2000 $\times$ 2000 pixels. Notice that this size corresponds to the the input layer of the detection model.

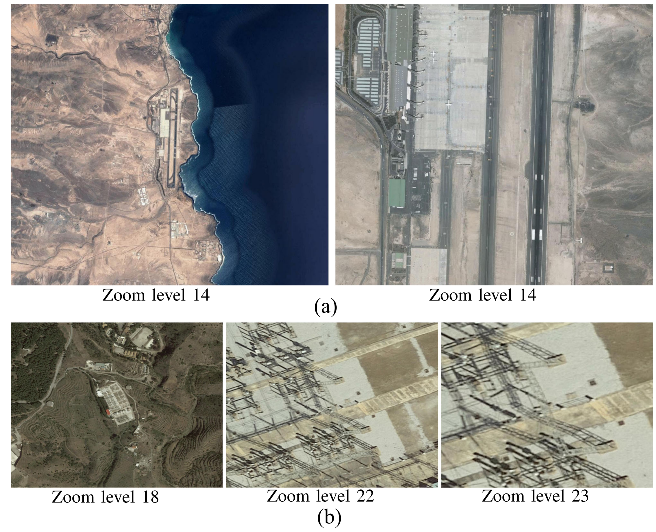


Fig. 7. Zoom levels discarded for the test. a) Large scale discard 14 for having the objects too far away and 17 for occupying more of the image. b) Small scale discard 18 for having the objects too far away and 22 and 23 for occupying more of the image.

TABLE IV
NUMBER OF INSTANCES IN THE ELECTRICAL SUBSTATION CLASS, A) CI-SS_TRAIN_ALPHA, B) CI-SS_TEST_ALPHA

(a)

Zoom level	Electrical substation
18	103
19	103
20	103
21	103
22	103
23	103
Total	618

(b)

Zoom level	Electrical substation
19	27
20	27
21	27
Total	81

Small Scale: The initial CI-SS dataset, CI-SS_train_alpha, is built using the electrical substation images with zoom levels from 18 to 23. We downloaded 550 images with different zoom levels, as shown in Table IV(a). For building the test set, CI-SS_test_alpha, we downloaded 75 images of the electrical substation class with zoom levels from 19 to 21, as shown in Table IV(b).

Large Scale: The initial version of CI-LS dataset, CI-LS_train_alpha, is built using only airport images with zoom levels from 14 to 17. We downloaded 160 images of airports from Spain and 80 airports from France, as shown in Table V(a). To build the initial test set, CI-LS_test_alpha, we downloaded 32 images of Spanish airports with two zoom levels 15 and 16, as shown in Table V(b).

B. Step 2: Extending the Dataset With More Object Classes

After a careful analysis of the FP committed by the detection model when trained on the initial dataset, we determined all potential object classes that make the detector confuse the target class with other different objects. At this stage, we analyzed the impact of each one of these potential FP on the learning of the

TABLE V
NUMBER OF INSTANCES IN THE AIRPORT CLASS, A) CI-LS_TRAIN_ALPHA, B) CI-LS_TEST_ALPHA

(a)

Zoom level	Airport
14	60
15	69
16	251
17	124
Total	504

(b)

Zoom level	Airport
15	17
16	16
Total	33

TABLE VI
NUMBER OF INSTANCES IN THE SMALL SCALE CRITICAL INFRASTRUCTURES, CI-SS_TRAIN_BETA

	Google Maps Zoom level 18	19	20	21	22	23	DOTA	Total
Electrical substation	103	103	103	103	103	103	-	618
Large vehicle	0	3	26	5	3	0	16923	16960
Swimming pool	111	104	62	11	2	0	1732	2022
Helicopter	0	0	0	0	0	0	630	630
Bridge	19	18	5	0	0	0	2041	2083
Plane	0	0	0	0	0	0	7944	7944
Ship	0	0	0	0	0	0	28033	28033
Soccer ball field	4	4	1	0	0	0	311	320
Basketball court	0	0	0	0	0	0	509	509
Ground track field	0	0	0	0	0	0	307	307
Small vehicle	0	0	141	234	68	5	26099	26547
Harbour	0	0	0	0	1	0	5937	5938
Baseball diamond	0	0	0	0	0	0	412	412
Tennis court	6	6	1	0	0	0	2325	2338
Roundabout	25	26	13	1	0	0	385	450
Storage tank	23	39	36	12	0	0	5024	5134

TABLE VII
NUMBER OF INSTANCES IN THE FINAL TEST VERSION OF SMALL SCALE CRITICAL INFRASTRUCTURES, CI-SS_TEST_STABLE DATASET

Zoom level	Electrical substation	Helicopter	Bridge	Plane	Harbour	Storage tank
19	27	8	21	68	57	136
20	27	8	15	35	27	50
21	27	6	13	17	12	24
Total	81	22	49	120	96	210

TABLE VIII
NUMBER OF INSTANCES IN THE LARGE SCALE CRITICAL INFRASTRUCTURES, CI-LS_TRAIN_BETA DATASET

	Airport	Train station	Motorway	Bridge	Industrial	Harbour
Google Maps zoom level 14	60	1	566	1	11	1
Google Maps zoom level 15	69	2	819	1	14	1
Google Maps zoom level 16	251	2	3207	8	34	1
Google Maps zoom level 17	124	19	2859	4	50	1
DIOR	1327	1011	-	3967	-	5509
Total	1831	1035	7451	3981	109	5513

TABLE IX
NUMBER OF INSTANCES IN THE FINAL TEST VERSION OF LARGE SCALE CRITICAL INFRASTRUCTURES, CI-LS_TEST_STABLE DATASET

Zoom level	Airport	Train station	Motorway	Bridge	Industrial	Harbour
15	17	25	518	115	59	32
16	16	22	303	55	27	20
Total	33	47	821	170	86	52

detector and extended the dataset with more object classes from public datasets and sources. If the performance improves, that potential FP class is maintained in the dataset, otherwise it is eliminated from the dataset.

For small scale infrastructure, the DOTA dataset will be added since their objects are of similar scales. For large scale infrastructures, the DIOR dataset will be used as it contains infrastructures of similar sizes. For both small and large scale datasets, we also included a large number of images of the same classes downloaded from Google Maps and annotated manually for detection.

Small Scale: We included in CI-SS_train_beta all DOTA classes listed in Table VI, in addition to a large number of images downloaded from Google Maps. Then, we eliminated each DOTA class one by one and evaluated its impact on the detector performance.

In addition, as we found that the most relevant new classes are bridge, harbor, storage tank, plane, and helicopter, the detector is trained to discriminate these classes too. For building CI-SS_test_stable, we included 132 images of the five new classes, as summarized in Table VII.

Large Scale: After analyzing the FP with Faster R-CNN, we included three object classes from DIOR dataset together with a large number of images of the same classes downloaded form Google Maps into CI-LS_train_beta, namely train station, bridge and harbor, and built the motorway and industrial area class (see Table VIII). We built a test set, CI-LS_test_stable, by including 114 new images of the five classes as it can be seen in Table IX.

C. Step 3: Further Increasing the Size of the Training Set

In this stage, we further increase the number of all the new object classes added to both training subsets using new images from Google Maps.

Small Scale: As the CI-SS_Det_beta trained model confuses electrical substation with several elements from urban areas, we included urban areas as context in the new training images in the rest of the classes. Namely, we downloaded a total of 1173 new images. The characteristics of the resulting CI-SS_train_stable are shown in Table X.

Large Scale: We further increased the size of CI-LS_train_beta dataset by including 768 new images. The characteristics of the resulting CI-LS_train_stable are shown in Table XI.

V. EXPERIMENTAL STUDY

This section provides all the performed experimental analysis to obtain CI-dataset and the evaluation of DetDSCI methodology. Section V-A summaries the experimental setup for the analysis. Section V-B provides all the detection model results obtained during the CI-dataset construction process. Finally,

TABLE X
NUMBER OF INSTANCES IN THE FINAL TRAIN SMALL SCALE CRITICAL INFRASTRUCTURES, CI-SS_TRAIN_STABLE DATASET

	Google Maps Zoom level 18	19	20	21	22	23	DOTA	Total
Electrical substation	103	278	267	247	103	103	-	1101
Swimming pool	111	911	370	141	2	0	1732	3267
Helicopter	0	20	17	17	0	0	630	684
Bridge	19	88	39	19	0	0	2041	2206
Plane	0	13	8	2	0	0	7944	7967
Soccer ball field	4	146	65	40	0	0	311	566
Basketball court	0	91	49	35	0	0	509	684
Ground track field	0	4	0	0	0	0	307	311
Harbour	0	1	0	0	1	0	5937	5939
Baseball diamond	0	2	0	0	0	0	412	414
Tennis court	6	126	46	27	0	0	2325	2530
Roundabout	25	103	38	8	0	0	385	559
Storage tank	23	538	249	73	0	0	5024	5907

TABLE XI
NUMBER OF INSTANCES IN THE FINAL TRAIN LARGE SCALE CRITICAL INFRASTRUCTURES, CI-LS_TRAIN_STABLE DATASET

		Airport	Train station	Motorway	Bridge	Industrial	Harbour
Google Maps zoom level	14	60	5	1012	37	69	17
	15	69	6	1280	37	71	17
	16	251	6	3947	57	116	27
	17	124	27	4805	168	291	23
DIOR		1327	1011	-	3967	-	5509
Total		1831	1055	11044	4266	547	5593

Section V-C provides the analysis and comparison of the proposed DetDSCI methodology.

A. Experimental Setup

The dynamic construction of the dataset requires the use of a good detection model. After a careful experimental analysis, we found that Faster R-CNN is the most suitable for this study as it achieves a good speed accuracy trade-off [18].

For training the detection models, the images were resized to 2000×2000 pixels image, which represents the required size of the input layer of modern detectors. A careful selection of the zoom level is necessary so that the entire object can fit in the image.

In the experiments carried out in the next sections, we used Keras [11] as a deep learning framework for classification and TensorFlow [1] as a deep learning framework for detection.

For evaluating and comparing the performance we will use these metrics: *Precision*, *Recall,* and *F1*((1)).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

where the number of true positives (TP), FP, and false negatives (FN) is computed for each class.

TABLE XII
DA TECHNIQUES BY MODEL

Model name	Data augmentation technique
DA1	Normalize image
DA2	Random image scale
DA3	Random rgb to gray
DA4	Random adjust brightness
DA5	Random adjust contrast
DA6	Random adjust hue
DA7	Random adjust saturation
DA8	Random distort colour

TABLE XIII
CONFIGURATION OF FE FOR DIFFERENT MODELS

Model name	Region Proposal	ResNet model	with DA
FE1	Faster R-CNN	ResNet 101 V1	No
FE2	Faster R-CNN	ResNet 101 V1	Yes
FE3	Faster R-CNN	ResNet 152 V1	No
FE4	Faster R-CNN	ResNet 152 V1	Yes
FE5	Faster R-CNN	Inception ResNet V2	No
FE6	Faster R-CNN	Inception ResNet V2	Yes

The detection performance is evaluated in terms of mAP [(2)] and mAR [(3)] standard metrics for object detection tasks [22] given 100 output regions.

$$mAP = \frac{\sum_{i=1}^{K} AP_i}{K} \qquad AP_i = \frac{1}{10} \sum_{r \in [0.5,\ldots,0.95]} \int_0^1 p(r)dr \tag{2}$$

$$mAR = \frac{\sum_{i=1}^{K} AR_i}{K} \qquad AR_i = 2\int_{0.5}^{1} recall(o)do \tag{3}$$

where given K categories of elements, p represents the precision and r $recall$ defines the area under the interpolated precision-recall curve for each class i. Whereas o is intersection over union (IoU) in recall(o) is the corresponding recall under the recall-IoU curve for each class i.

The performance of the detection models can be improved with the use of several optimization techniques, namely data augmentation (DA) and analyzing different feature extractors (FE). The eight DA techniques used to this task are listed in Table XII and their impact will be study on the performance of each detector.

Besides, we consider six FE listed in Table XIII and train the models with or without the best DA techniques. We will analyze the impact of all these factors on the performance of each detection model.

B. Experimental Study for the Construction of the CI-Dataset

Section IV provided a detailed description of the construction process of CI-dataset. This section provides the experimental results of the detection model at each stage of that process. The performance obtained in steps 1, 2, and 3 are, respectively, analyzed in Sections V-B1, V-B2, and V-B3. Finally, the experimental analysis of the use of DA techniques and different FE is provided in Section V-B4.

TABLE XIV
PERFORMANCE (%) OF CI-SS_DET_ALPHA WHEN TRAINED ON DIFFERENT ZOOM LEVEL COMBINATIONS OF CI-SS_TRAIN_ALPHA AND TESTED ON CI-SS_TEST_ALPHA DATASET

Zoom level combination	Precision	Recall	F1	mAP 0.5 electrical substation	mAP 0.5-0.95 mean	mAR 0.5-0.95 mean
18,19,20,21,22,23	**96.49**	67.90	79.71	87.45	48.30	60.70
19,20,21,22,23	93.44	70.37	**80.28**	86.23	**51.70**	60.40
18,19,20,21,22	91.94	70.37	79.72	**89.90**	48.70	59.00
20,21,22,23	92.31	59.26	72.18	79.35	43.50	55.80
19,20,21,22	89.39	**72.84**	80.27	89.18	51.60	**62.60**
21,22,23	82.76	29.63	43.64	57.90	28.10	38.40
20,21,22	89.29	61.73	72.99	80.55	44.50	54.40
21,22	82.35	17.28	28.57	51.11	24.50	34.70

TABLE XV
PERFORMANCE (%) OF CI-LS_DET_ALPHA WHEN TRAINED ON DIFFERENT ZOOM LEVEL COMBINATIONS OF CI-LS_TRAIN_ALPHA AND TESTED ON CI-LS_TEST_ALPHA DATASET

Zoom level combination	Precision	Recall	F1	mAP 0.5 airport	mAP 0.5-0.95 mean	mAR 0.5-0.95 mean
14,15,16,17	**87.76**	**86.00**	**86.87**	**89.52**	**61.30**	**69.10**
14,15,16	78.85	82.00	80.39	84.67	55.50	62.10
15,16,17	68.42	78.00	72.90	87.89	54.50	64.20
15,16	87.23	82.00	84.54	82.66	51.00	57.90

TABLE XVI
IMPACT OF ELIMINATING EACH INDIVIDUAL DOTA'S CLASS FROM THE CI-SS_TRAIN_BETA ON THE DETECTION PERFORMANCE (%)

Classes deleted	Precision	Recall	F1
None	88.28	58.38	70.22
- Small vehicle	**92.61**	59.64	72.53
- Large vehicle	90.30	62.44	73.81
- Ship	90.67	**67.53**	77.35
- Tennis court	88.09	63.00	73.39
- Baseball diamond	89.97	66.33	76.31
- Ground track field	87.02	65.77	74.84
- Basketball court	91.19	63.80	74.99
- Soccer-ball field	93.47	66.64	**77.74**
- Roundabout	90.48	65.28	75.70
- Swimming pool	90.74	66.55	76.73

TABLE XVII
PERFORMANCE (%) OF CI-LS_DET_BETA WHEN TRAINED ON CI-LS_TRAIN_BETA AND TESTED ON CI-LS_TEST_STABLE

		CI-LS_Det_beta
mAP 0.5	Mean	22.03
	Airport	**85.73**
	Train station	6.98
	Motorway	4.30
	Bridge	31.97
	Industrial	2.87
	Harbour	0.31
mAP 0.5-0.95	Mean	12.20
	Small	2.00
	Medium	4.70
	Large	14.40
mAR 0.5-0.95		22.10

TABLE XVIII
PERFORMANCE (%) OF CI-SS_DET_STABLE AND CI-SS_DET_BETA ON CI-SS_TEST_STABLE AND CI-SS_DET_ALPHA WHEN TRAINED AND TESTED ONLY ON THE ELECTRICAL SUBSTATION CLASS

		CI-SS_Det_alpha (only ele. sub.)	CI-SS_Det_beta (six classes)	**CI-SS_Det_stable (six classes)**
mAP 0.5	Mean	**87.45**	54.21	65.98
	Electrical substation	**87.45**	78.88	85.00
	Plane	0.00	82.94	**85.30**
	Helicopter	0.00	**33.83**	10.39
	Bridge	0.00	18.33	**63.16**
	Storage tank	0.00	83.07	**92.28**
	Harbour	0.00	58.66	**59.75**
mAP 0.5-0.95	Mean	**48.30**	32.30	38.60
	Small	0.00	15.30	**25.90**
	Medium	**31.80**	23.50	27.90
	Large	**49.70**	36.80	43.40
mAR 0.5-0.95		**60.70**	47.80	53.10

TABLE XIX
TP, FP, FN, RECALL (%), PRECISION (%) AND F1 (%) IN CI-SS_TEST_STABLE

	TP	FP	FN	Precision	Recall	F1
CI-SS_Det_alpha (only ele. sub.)	117	449	7	20.67	**94.35**	33.91
CI-SS_Det_beta (six classes)	75	124	49	37.69	60.48	46.44
CI-SS_Det_stable (six classes)}	112	62	12	**64.37**	90.32	**75.17**

CI-SS_Det_stable is trained on CI-SS_train_stable and CI-SS_Det_beta is trained on CI-SS_train_beta. For comparison purposes, CI-SS_Det_alpha is trained only on airports.

TABLE XX
PERFORMANCE (%) OF CI-LS_DET_STABLE AND CI-LS_DET_BETA TESTED ON CI-LS_TEST_STABLE AND CI-LS_DET_ALPHA TRAINED AND TESTED ONLY ON THE AIRPORT CLASS

		CI-LS_Det_alpha (only airports)	CI-LS_Det_beta (six classes)	**CI-LS_Det_stable (six classes)**
mAP 0.5	Mean	**89.52**	22.03	36.48
	Airport	**89.52**	85.73	85.37
	Train station	0.00	6.98	**26.45**
	Motorway	0.00	4.30	**5.16**
	Bridge	0.00	31.97	**40.53**
	Industrial	0.00	2.87	**20.96**
	Harbour	0.00	0.31	**40.40**
mAP 0.5-0.95	Mean	**61.30**	12.20	18.80
	Small	0.00	2.00	**2.40**
	Medium	0.00	4.70	**6.50**
	Large	**61.30**	14.40	23.00
mAR 0.5-0.95		**69.10**	22.10	33.90

TABLE XXI
COMPARISON OF TP, FP, FN, TN, PRECISION (%), RECALL (%) AND F1 (%) OF CI-LS_DET_STABLE TRAINED ON CI-LS_TRAIN_STABLE AND TESTED ON CI-LS_TEST_STABLE WITH CI-LS_DET_BETA AND CI-LS_DET_ALPHA

	TP	FP	FN	Precision	Recall	F1
CI-LS_Det_alpha (only airports)	29	19	1184	60.42	2.39	4.60
CI-LS_Det_beta (six classes)	236	35	977	87.08	19.46	31.81
CI-LS_Det_stable (six classes)	334	39	879	**89.54**	**27.54**	**42.12**

CI-LS_Det_alpha is trained and tested only on the airport class.

TABLE XXII
RESULTS (%) OF THE DIFFERENT MODELS WITH A DA TECHNIQUE IN CI-SS_TRAIN_STABLE AND CI-SS_TEST_STABLE

		DA1	DA2	DA3	DA4	DA5	DA6	DA7	**DA8**
mAP 0.5	Mean	22.26	67.85	66.84	68.07	66.45	64.83	64.67	**69.07**
	Electrical substation	0.01	**84.89**	83.65	83.36	82.35	83.23	82.81	82.30
	Plane	41.34	83.23	**88.72**	88.08	82.35	88.06	85.69	86.70
	Helicopter	0.02	19.82	16.48	14.39	14.99	12.42	10.32	**24.52**
	Bridge	15.83	64.90	61.18	**65.86**	62.84	55.08	60.38	64.96
	Storage tank	64.28	90.25	89.44	**91.66**	91.16	91.29	91.47	89.88
	Harbour	12.11	64.02	61.55	65.05	65.03	58.79	57.32	**66.07**
mAP 0.5-0.95	Mean	12.80	38.70	39.20	39.30	39.30	39.20	38.80	38.40	**39.50**
	Small	0.00	23.30	14.10	24.40	23.80	21.80	**31.00**	13.50
	Medium	2.60	26.50	25.60	27.50	**28.70**	28.20	26.20	26.60
	Large	18.90	43.70	44.90	44.70	44.30	43.60	43.70	**45.60**
mAR 0.5-0.95		23.50	54.20	54.40	53.50	**54.70**	54.10	52.80	54.20

TABLE XXIII
RESULTS (%) OF DIFFERENT FE WITH OR WITHOUT DA TECHNIQUES IN CI-SS_TRAIN_STABLE AND CI-SS_TEST_STABLE

		FE1	**FE2**	FE3	FE4	FE5	FE6
mAP 0.5	Mean	65.98	**68.97**	63.16	65.39	65.83	63.96
	Electrical substation	85.00	85.19	83.05	87.55	82.73	**87.78**
	Plane	85.30	84.43	85.81	80.91	**86.29**	84.96
	Helicopter	10.39	23.14	6.83	12.48	**48.03**	6.23
	Bridge	**63.16**	62.38	48.45	50.31	60.54	39.71
	Storage tank	**92.28**	88.97	91.01	90.89	90.93	91.82
	Harbour	59.75	69.70	63.82	70.22	69.71	**73.29**
mAP 0.5-0.95	Mean	38.60	**40.20**	36.70	37.60	36.50	37.60
	Small	**25.90**	13.30	4.70	3.10	2.70	3.90
	Medium	27.90	**29.90**	23.60	21.50	29.70	28.60
	Large	43.40	**46.30**	42.20	44.50	40.70	42.10
mAR 0.5-0.95		53.10	**54.10**	51.20	53.10	50.70	51.30

TABLE XXIV
RESULTS (%) OF THE DIFFERENT MODELS WITH A DA TECHNIQUE IN CI-LS_TRAIN_STABLE AND CI-LS_TEST_STABLE

		DA1	DA2	**DA3**	DA4	DA5	DA6	DA7	DA8
mAP 0.5	Mean	3.61	35.91	**37.11**	36.98	36.62	35.04	36.34	36.98
	Airport	19.54	85.71	90.31	85.75	90.87	**91.50**	88.18	85.84
	Train station	0.07	20.72	**27.98**	26.12	23.53	15.84	19.50	23.39
	Motorway	0.36	4.89	6.19	5.92	6.36	5.20	5.81	**6.63**
	Bridge	0.35	39.44	37.78	40.44	36.33	35.92	36.35	**45.05**
	Industrial	0.11	17.05	21.02	21.05	15.85	15.53	**22.06**	15.04
	Harbour	1.22	**47.64**	39.37	42.62	46.76	46.24	46.13	45.94
mAP 0.5-0.95	Mean	1.60	18.50	**19.30**	18.20	18.30	18.50	17.90	17.70
	Small	0.10	3.40	3.00	**7.00**	2.20	3.50	2.30	5.20
	Medium	0.00	6.20	**7.30**	6.60	6.30	6.70	6.30	6.00
	Large	3.00	20.70	22.40	21.10	21.70	20.80	21.50	**23.00**
mAR 0.5-0.95		13.10	34.80	34.50	**35.40**	33.40	34.20	34.50	34.70

1) Analysis of Step 1: Construction of the Target Class Dataset: Once the initial CI-dataset of the target class is constructed, we analyzed all the combinations of zoom levels to determine, which one improves the learning process of the detection models. Although the initial number of training images is not too large, the models are learning correctly how to distinguish between the different classes. Guided by the performance of the detection model on the target class, we discarded the zoom levels that did not help in the learning process of the detector.

TABLE XXV
RESULTS (%) OF DIFFERENT FE WITH OR WITHOUT DA TECHNIQUES IN CI-LS_TRAIN_STABLE AND CI-LS_TEST_STABLE

		FE1	FE2	FE3	FE4	**FE5**	FE6
mAP 0.5	Mean	36.48	37.52	37.67	38.05	**42.34**	40.98
	Airport	85.37	86.46	84.03	**87.70**	86.01	87.21
	Train station	26.45	24.17	**34.20**	22.31	27.76	22.43
	Motorway	5.16	5.53	4.80	5.77	5.95	**8.01**
	Bridge	40.53	47.81	36.69	48.86	**57.27**	54.25
	Industrial	20.96	17.43	23.53	17.54	**23.64**	22.38
	Harbour	40.40	43.71	42.78	46.13	**53.41**	51.63
mAP 0.5-0.95	Mean	18.80	18.30	18.80	18.50	**20.30**	20.10
	Small	2.40	5.70	3.20	6.50	**9.70**	7.70
	Medium	6.50	7.30	6.30	6.70	**8.50**	7.20
	Large	**23.00**	21.60	22.00	22.90	22.50	22.40
mAR 0.5-0.95		33.90	36.30	35.10	35.20	35.20	**37.70**

TABLE XXVI
NUMBER OF IMAGES BY ZOOM LEVEL USED FOR TRAINING AND EVALUATE THE CLASSIFIERS

	14	15	16	17	18	19	20	21	22	23
Train	252	400	1256	2984	200	591	1080	2268	6406	663
Test	19	52	52	19	44	304	304	304	19	19

TABLE XXVII
CONFUSION MATRIX FOR THE CLASSIFIER BY ZOOM LEVEL INDIVIDUALLY

Zoom level	14	15	16	17	18	19	20	21	22	23
14	0	13	5	0	0	0	0	0	1	0
15	0	14	34	2	0	0	0	2	0	0
16	0	0	25	26	0	0	1	0	0	0
17	0	0	1	18	0	0	0	0	0	0
18	0	0	0	33	0	8	2	0	1	0
19	1	0	0	9	0	209	69	12	4	0
20	0	0	0	0	0	12	224	57	11	0
21	0	0	0	2	0	1	6	268	25	2
22	0	0	0	0	0	0	0	2	17	0
23	0	0	0	0	0	0	0	1	18	0

TABLE XXVIII
CONFUSION MATRIX FOR THE CLASSIFIER BY ZOOM LEVEL BY GROUP

Zoom level	[14,17]	[18,23]
[14,17]	134	8
[18,23]	28	966

Small Scale: The performance of the first detector, CI-SS_Det_alpha, trained on different zoom level combinations shows similar results as it can be seen from Table XIV. We selected the combination that provides the highest number of images, which is the one that includes all the zoom levels, 18, 19, 20, 21, 22, and 23.

Large Scale: The performance of the detection model, CI-LS_Det_alpha, in different zoom level combinations shows that the best and most stable results are obtained by the combination of these zoom levels, 14, 15, 16, and 17, as it can be seen in Table XV.

2) Analysis of Step 2: Extending the Number of Classes: Once the CI-dataset is extended with new classes from public

TABLE XXIX
PERFORMANCE (%) COMPARISON BETWEEN DETDSCI METHODOLOGY, BASE_DET, CI-LS_DET_STABLE AND CI-SS_DET_STABLE WHEN TESTED ON THE FUSION OF CI-SS_TEST_STABLE AND CI-LS_TEST_STABLE

	TP	FP	FN	Precision	Recall	F1
Base_Det	70	35	44	66.67	61.40	63.93
CI-LS_Det_stable	27	**3**	88	**90.00**	23.48	37.24
CI-SS_Det_stable	71	32	44	68.93	61.74	65.14
DetDSCI methodology	**83**	24	**32**	77.57	**72.17**	**74.77**

datasets, we analyzed whether the new classes improve the performance of the detection models.

Small Scale: First, we trained the model on all DOTA classes and our built electrical substation class. Then, we analyzed the impact of each DOTA's class on the detection model by eliminating that class from the training dataset. As it can be seen from Table XVI, eliminating the three DOTA classes, small vehicle, large vehicle, and ship, improves the F1 of CI-SS_Det_beta detection model. This is due to the fact that the images of these objects provide very few information about their features, i.e., they are represented using very few pixels.

Therefore, the final dataset CI-SS_train_stable contains 13 classes, tennis court, baseball diamond, ground track field, basketball court, soccer-ball field, roundabout, and swimming pool in addition to bridge, harbor, storage tank, helicopter, plane, and electrical substation.

Large Scale: The results of the detection model, CI-LS_Det_beta, trained on CI-LS_train_beta, are shown in Table XVII. As it can be observed from this table, including some DIOR classes increases the mAP of the detection model on the airport class to 85.73%.

3) Analysis of Step 3: Increasing the Size of the Dataset: Once the final classes are determined, new images are included to further improve the performance of the models.

Small Scale: A comparison between CI-SS_Det_beta and the new CI-SS_Det_stable, trained on the CI-SS_train_stable (see Table X), tested on the CI-SS_test_stable (see Table VII) dataset, is shown in Table XVIII. The performance of CI-SS_Det_alpha trained and tested only on the electrical substation is included in the table as reference as well. These results show clearly that the performance of CI-SS_Det_stable improves when increasing the size of the training dataset.

For a further analysis, we observed the TP, FP, FN, Precision, Recall, and F1 as shown in Table XIX. As it can be observed, CI-SS_Det_stable reduces substantially the number of FP and achieves the best F1 value. Therefore, the CI-SS_Det_stable model will be used in the rest of this article as it provides the highest performance on our target class, electrical substation.

Large Scale: A comparison between CI-LS_Det_beta and the new CI-LS_Det_stable, trained on CI-LS_train_stable (see Table XI), tested on CI-LS_test_stable (see Table IX) dataset, is shown in Table XX. The mAP of CI-LS_Det_alpha trained and tested only on the airport class is included in the table as reference as well. As it can be seen from these results, CI-LS_Det_stable shows very similar mAP on airports than CI-LS_Det_beta but much better mAP on the rest of potential FP.

A comparison with CI-LS_Det_stable trained on CI-LS_train_stable and tested on CI-LS_test_stable is provided in Table XXI. In general, CI-LS_Det_stable provides the highest F1.

4) Analysis of the Improvement of the Detection Models: The selection of the right DA techniques and FE can surely further improve the performance of the detection model. We consider eight DA techniques listed in Table XII and study their impact on the performance of each detector. Besides we consider six FE listed in Table XIII and train the models with or without the best DA techniques. We analyze the impact of all these factors on the performance of each detection model.

Small Scale: Table XXII shows the performance of CI-SS_Det_stable when applying individually different DA techniques on CI-SS_train_stable. As it can be observed from this table, applying DA8, random distort color, achieves the best results in this model.

Table XXIII shows the impact of the different FE and DA on the performance of CI-SS_Det_stable. As it can be seen, the best mAP is obtained when using Faster R-CNN ResNet101 V1 with FE2 and DA techniques. This detection model will be the new CI-SS_Det_stable.

Large Scale: Table XXIV shows the performance of CI-LS_Det_stable when applying different DA techniques on CI-LS_train_stable. These results show that applying DA3, random rgb to gray, achieves the best detection results.

Table XXV shows the impact of the different FE and DA on CI-LS_Det_stable. As it can be seen the best performance is obtained with Faster R-CNN Inception ResNet V2 with FE5 and without DA techniques. This model will be the new CI-LS_Det_stable in the rest of this article.

C. *Experimental Study of DetDSCI Methodology*

Once CI-dataset is constructed and the final models are trained on the small and the large scale critical infrastructures, we develop the zoom level classifier for the DetDSCI methodology. The construction of the zoom level classifier is presented in Section V-C1 and the analysis of DetDSCI methodology is shown in Section V-C2.

1) Construction of the Zoom Level Classifier: In the first stage of DetDSCI methodology, a zoom level classifier analyses the input image and determines the scale of this input. This stage can be addressed either by identifying the specific zoom level of each input image or by identifying intervals of zoom levels.

In particular, we developed and analyzed two classification models, the first one was trained on 10 zoom level classes, from 14 to 23, and the second classification model was trained on two zoom level intervals, interval [14,17] and [18,23]. Table XXVI shows the number of images used to train and test these two classification models. The images used were selected from datasets CI-SS_train_stable, CI-SS_test_stable, CI-LS_train_stable, and CI-LS_test_stable.

Fig. 8. Examples of detection obtained by the baseline model, Base_Det (left), and DetDSCI methodology (right).

The confusion matrix for the classification by individual zoom level is shown in Table XXVII. The overall accuracy of this model is 68.31%, which is very low.

The confusion matrix for the classification by interval is shown in Table XXVIII. This model obtains an accuracy of 96.83%, which is substantially higher than the classification by individual zoom level. Therefore, we selected this classifier to be included in our DetDSCI methodology.

2) Analysis of DetDSCI Methodology: In this section, we analyze and compare the performance of DetDSCI methodology against the baseline detectors CI-LS_Det_stable and CI-SS_Det_stable and a baseline detector, Base_Det, trained on all the data and zoom levels.

The characteristic of each model is

1) Base_Det: is a Faster R-CNN ResNet 101 V1 trained on small and large scale classes from CI-SS_train_stable and CI-LS_train_stable without any separation.
2) CI-LS_Det_stable: is a Faster R-CNN Inception ResNet V2 trained on the CI-LS_train_stable dataset.
3) CI-SS_Det_stable: is a Faster R-CNN ResNet 101 V1 with DA techniques trained on the CI-SS_train_stable dataset.
4) DetDSCI Methodology: is the methodology by which each input image is classified by the zoom level classifier and based on the output of this classifier, the detector to be used is selected between CI-LS_Det_stable or CI-SS_Det_stable.

We tested the four models on the images of the target classes, electrical substation from CI-SS_test_stable and airport from CI-LS_test_stable. The results in terms of TP, FP, FN, Precision, Recall, and F1 are shown in Table XXIX.

As it can clearly seen from this table, DetDSCI methodology overcomes Base_Det, CI-SS_Det_stable and CI-LS_Det_stable in all the aspects by achieving the highest performance. In particular, DetDSCI methodology achieves an improvement in F1 of up to 37.53%. Therefore, it can be concluded that the division between small and large scales gives better results. Fig. 8 illustrates the results of the detections obtained by Base_Det and DetDSCI methodology detections.

The inference time of the small scale detector, Faster R-CNN ResNet101 V1, on a NVIDIA Tesla V100 32 GB GPU is 0.076 s, while the large scale detector, Faster R-CNN Inception ResNet V2, takes 0.095 seconds. The ResNet-50 classifier executes in 0.0029 s. In total, the DetDSCI methodology process takes 0.0979 seconds in analyzing an input image.

VI. Conclusion

The detection of critical infrastructures in satellite images is a very challenging task due to the large scale and different shapes, some infrastructures are too small, e.g., electrical substations, while others are too large, i.e., airports. This work addressed this problem by building the high quality dataset, CI-dataset, organised into two subsets, CI-SS and CI-LS and using DetDSCI methodology. The construction process of CI-SS and CI-LS was guided by the performance of the detectors on electrical substations and airports, respectively.

DetDSCI methodology is a two-stage based approach that first identifies the zoom level of the input image using a classifier and then analyses that image with the corresponding detection model, CI-LS_Det_stable or CI-SS_Det_stable. DetDSCI methodology achieves the highest performance with respect to the baseline detectors not only in the target objects, but also in the rest of infrastructure classes included in the dataset.

As conclusions, the proposed datasets and methodology are the best solution for addressing the problem of different and dissimilar scale critical infrastructures detection in remote sensing images. This approach can be easily extended to more critical infrastructures.

As a future work, we will extend the dataset and methodology to more critical infrastructures and design a strategy to group sets of classes according to their zoom level and shared features, with the objective to achieve more robust detection models.

References

[1] M. Abadi *et al.*, “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” 2016, *arXiv:1603.04467*.
[2] K. S. Bjerreskov, T. Nord-Larsen, and R. Fensholt, “Classification of nemoral forests with fusion of multi-temporal sentinel-1 and 2 data,” *Remote Sens.*, vol. 13, no. 5, 2021, Art. no. 950.
[3] Ü. Budak, A. ŞengÜr, and U. Halici, “Deep convolutional neural networks for airport detection in remote sensing images,” in *Proc. IEEE 26th Signal Process. Commun. Appl. Conf.*, 2018, pp. 1–4.
[4] B. Cai, Z. Jiang, H. Zhang, D. Zhao, and Y. Yao, “Airport detection using end-to-end convolutional neural network with hard example mining,” *Remote Sens.*, vol. 9, no. 11, 2017, Art. no. 1198.
[5] M. Carranza-GarcÍa, J. GarcÍa-Gutiérrez, and J. C Riquelme, “A framework for evaluating land use and land cover classification using convolutional neural networks,” *Remote Sens.*, vol. 11, no. 3, 2019, Art. no. 274.
[6] G. Cheng, J. Han, and X. Lu, “Remote sensing image scene classification: Benchmark and state of the art,” *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
[7] G. Cheng, J. Han, P. Zhou, and L. Guo, “Multi-class geospatial object detection and geographic image classification based on collection of part detectors,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 98, pp. 119–132, 2014.
[8] G. Cheng, C. Lang, M. Wu, X. Xie, X. Yao, and J. Han, “Feature enhancement network for object detection in optical remote sensing images,” *J. Remote Sens.*, vol. 2021, 2021.
[9] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, “When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNS,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018, doi: 10.1109/TGRS.2017.2783902.
[10] G. Cheng, P. Zhou, and J. Han, “Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016, doi: 10.1109/TGRS.2016.2601622.
[11] François Chollet *et al.* “Keras: Deep learning library for theano and tensorflow,” vol. 7, no. 8, T1, 2015. [Online]. Available: https://keras.io/k
[12] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee, “Functional map of the world,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6172–6180.
[13] N. Flood, F. Watson, and L. Collett, “Using a U-net convolutional neural network to map woody vegetation extent from high resolution satellite imagery across queensland, Australia,” in *Proc. Int. J. Appl. Earth Observ. Geoinformation*, 2019, vol. 82, paper 101897.
[14] D. Guidici and M. L. Clark, “One-dimensional convolutional neural network land-cover classification of multi-seasonal hyperspectral imagery in the San Francisco Bay area, California,” *Remote Sens.*, vol. 9, no. 6, 2017, Art. no. 629.
[15] E. Guirado, D. Alcaraz-Segura, J. Cabello, S. Puertas-RuÍz, F. Herrera, and S. Tabik, “Tree cover estimation in global drylands from space using deep learning,” *Remote Sens.*, vol. 12, no. 3, 2020, Art. no. 343.
[16] E. Guirado, S. Tabik, Marga L. Rivas, D. Alcaraz-Segura, and F. Herrera, “Whale counting in satellite and aerial images with deep learning,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, 2019.

[17] Q. He, X. Sun, Z. Yan, and K. Fu, "Dabnet: Deformable contextual and boundary-weighted network for cloud detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, doi: 10.1109/TGRS.2020.3045474.

[18] J. Huang *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7310–7311.

[19] D. Lam *et al.*, "xview: Objects in context in overhead imagery," 2018, *arXiv:1802.07856*.

[20] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogrammetry Remote Sens.*, vol. 159, pp. 296–307, 2020.

[21] S. Li, Y. Xu, M. Zhu, S. Ma, and H. Tang, "Remote sensing airport detection based on end-to-end deep transferable convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 10, pp. 1640–1644, Oct. 2019.

[22] Tsung-Yi Lin *et al.*, "Microsoft Coco: Common objects in context," in *Proc. Euro. Conf. Comput. Vis.*, 2014, pp. 740–755.

[23] S. Liu, Z. Qi, X. Li, and A. Gar-On Yeh, "Integration of convolutional neural networks and object-based post-classification refinement for land use and land cover mapping with optical and SAR data," *Remote Sens.*, vol. 11, no. 6, 2019, Art. no. 690.

[24] X. Lu, Y. Zhang, Y. Yuan, and Y. Feng, "Gated and axis-concentrated localization network for remote sensing object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 179–192, Jan. 2020.

[25] J. Luengo, D. García-Gil, S. Ramírez-Gallego, S. García, and F. Herrera. *Big Data Preprocessing*. Cham, Switzerland: Springer, 2020.

[26] B. Oshri *et al.*, "Infrastructure quality assessment in africa using satellite imagery and deep learning," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 616–625.

[27] A. Safonova, E. Guirado, Y. Maglinets, D. Alcaraz-Segura, and S. Tabik, "Olive tree biovolume from UAV multi-resolution image segmentation with mask R-CNN," *Sensors*, vol. 21, no. 5, 2021, Art. no. 1619.

[28] X. Sun, B. Wang, Z. Wang, H. Li, H. Li, and K. Fu, "Research progress on few-shot learning for remote sensing image interpretation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2387–2402, 2021, doi: 10.1109/JSTARS.2021.3052869.

[29] G.-S. Xia *et al.*, "Dota: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3974–3983.

[30] T.-Z. Xiang, G.-S. Xia, and L. Zhang, " Mini-UAV-based remote sensing: Techniques, applications and prospectives," 2018, *arXiv:1812.07770*.

[31] Z. Xiao, Y. Gong, Y. Long, D. Li, X. Wang, and H. Liu, "Airport detection based on a multiscale fusion feature for optical remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 9, pp. 1469–1473, Sep. 2017.

[32] Y. Xu, M. Zhu, S. Li, H. Feng, S. Ma, and J. Che, "End-to-end airport detection in remote sensing images combining cascade region proposal networks and multi-threshold detection networks," *Remote Sens.*, vol. 10, no. 10, 2018, Art. no. 1516.

[33] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. geographic Inf. Syst.*, 2010, pp. 270–279.

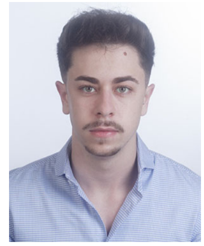
[34] C. Zhang *et al.*, "Joint deep learning for land cover and land use classification," *Remote Sens. Environ.*, vol. 221, pp. 173–187, 2019.

[35] P. Zhang, X. Niu, Y. Dou, and F. Xia, "Airport detection on optical satellite images using deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp 1183–1187, Aug. 2017.

[36] Y. Zhang, Y. Yuan, Y. Feng, and X. Lu, "Hierarchical and robust convolutional neural network for very high-resolution remote sensing object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp 5535–5548, Aug. 2019.

Francisco Pérez-Hernández received the M.Sc. degree in computer science from the University of Granada, Granada, Spain, in 2016, where he is currently working toward the Ph.D. degree with the Department of Computer Science and Artificial Intelligence.

His research interests include object detection and classification with deep learning in different fields.



José Rodríguez-Ortega was born in Granada, Andalusia, Spain in 1998. He received the B.S degree in computer science from the University of Granada, Granada, Spain, in 2020, He is currently studying and the second B.S degree in mathematics from the National University of Distance Education (UNED). He is currently working toward the M.S degree in data science with the University of Granada.

In 2020, he did an internship as a Software Engineer and started as a Research Assistant with the Soft Computing and Intelligent Information Systems research group with the University of Granada. His research interests include deep learning for computer vision and artificial intelligent.

Yassir Benhammou received the computer science engineering degree from The National School of Applied Sciences of Tetouan, AbdelMalek Essaâdi University of Tetouan, Tetouan, Morocco, in 2016. He is currently working toward the Ph.D. degree in a co-tutelle thesis between The National School of Applied Sciences of Berrechid, Hassan 1st University of Settat, Settat, Morocco, and The Andalusian Research Institute in Data Science and Computational Intelligence, University of Granada, Granada, Spain.

His research interests include deep learning for remote sensing and healthcare, data preprocessing, and machine learning.

Francisco Herrera (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in mathematics from the University of Granada, Granada, Spain, in 1988 and 1991, respectively.

He is currently a Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, and the Director of the Andalusian Research Institute in Data Science and Computational Intelligence. He is also an Academician with the Royal Academy of Engineering, Madrid, Spain. He has been the Supervisor of 51 Ph.D. students. He has authored or coauthored more than 500 journal papers, receiving more than 86 000 citations (Scholar Google, H-index 144). His current research interests include among others, Computational Intelligence (including fuzzy modeling, computing with words, evolutionary algorithms and deep learning), information fusion and decision making, and data science (including data preprocessing, prediction, nonstandard classification problems, and Big Data).

Dr. Herrera has been nominated as a highly cited researcher by Clarivate Analytics (in the fields of computer science and engineering, respectively, from 2014 to present). He currently acts as an Editor-in-Chief of *Information Fusion* (Elsevier). He acts as Editorial Member of a dozen of journals. He was the recipient of the several honors and awards, among others: ECCAI Fellow 2009, IFSA Fellow 2013, 2010 Spanish National Award on Computer Science ARITMEL to the "Spanish Engineer on Computer Science," the International Cajastur "Mamdani" Prize for Soft Computing (Fourth Edition, 2010), the IEEE TRANSACTIONS ON FUZZY SYSTEM Outstanding 2008 and 2012 Papers, the 2011 Lotfi A. Zadeh Prize Best Paper Award (IFSA Association), the 2013 AEPIA Award to a scientific career in Artificial Intelligence, the 2014 XV Andalucía Research Prize Maimónides, the 2017 Andalucía Medal (by the regional government of Andalucía), the 2018 "Granada: Science and Innovation City" Award.

Siham Tabik received the B.Sc. degree in physics from University Mohammed V, Rabat, Morocco, in 1998 and the Ph.D. degree in computer science from the University of Almería, Almería, Spain, in 2006.

She is currently an Associate Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain. She is the Coordinator of the area of research, "AI and remote sensing for biodiversity and global change" with DaSCI (The Andalusian Interuniversity Institute on Data Science and Computational Intelligence). Her research interests include artificial intelligence and computer vision for remote sensing.