A deep learning based approach for detecting panels in photovoltaic plants

Antonio Greco
University of Salerno
Salerno, Italy
agreco@unisa.it

Christopher Pironti
University of Salerno
Salerno, Italy
cpironti@unisa.it

Alessia Saggese
University of Salerno
Salerno, Italy
asaggese@unisa.it

Mario Vento
University of Salerno
Salerno, Italy
mvento@unisa.it

Vincenzo Vigilante
University of Salerno
Salerno, Italy
vvigilante@unisa.it

ABSTRACT

Photovoltaic (PV) panels are a clean and widespread way to produce renewable energy from sunlight; at the same time, such plants require maintenance, since solar panels can be affected by many types of damaging factors and have a limited yet variable lifespan. With the impressive growth of such PV installations, it is in the public eye the need of a cheap and effective way to continuously monitor the state of the plants and a standard technique designed to promptly replace broken modules, in order to prevent drops in the energy production. Since the faults mainly appear as Hot Spots on the surface of the PV panels, aerial thermal imaging can be used to diagnose such problems and also locate them in huge plants. To this aim, dedicated automatic Computer Vision methods are able to automatically find hot spots from thermal images, where they appear as white stains. In these methods a fundamental step is the segmentation of the PV panels, which allows to automatically detect each module.

In this paper, we address the problem of PV Panel Detection using a Convolutional Neural Network framework called YOLO. We demonstrate that it is able to effectively and efficiently segment panels from an image. The method is quantitatively evaluated and compared to existing PV panel detection approaches on the biggest publicly available benchmark dataset; the experimental results confirm its robustness.

Thanks to Topview s.r.l. (www.topview.it) for providing the videos for the dataset. Also thanks to A.I. Tech s.r.l. (www.aitech.vision) for partially supporting this research.

APPIS 2020, January 7–9, 2020, Las Palmas de Gran Canaria, Spain

ACM ISBN 978-1-4503-7630-3/20/01.
https://doi.org/10.1145/3378184.3378185

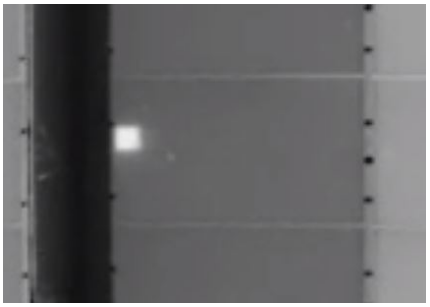

Figure 1: A thermal image of a damaged panel. The bright square is a so called "hot spot", a damaged solar cell, where the power efficiency drops and the energy is dissipated as heat.

CCS CONCEPTS

• **Computing methodologies** → **Visual inspection**; *Object detection*; • **Hardware** → *Defect-based test*;

KEYWORDS

UAV, photovoltaic, deep learning, object detection

ACM Reference Format:
Antonio Greco, Christopher Pironti, Alessia Saggese, Mario Vento, and Vincenzo Vigilante. 2020. A deep learning based approach for detecting panels in photovoltaic plants. In *Proceedings of the 3rd International Conference on Applications of Intelligent Systems (APPIS 2020), January 7–9, 2020, Las Palmas de Gran Canaria, Spain.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3378184.3378185

1 INTRODUCTION

According to the data reported by the European Photovoltaic Industry Association, photovoltaic plants are becoming more and more diffused every year in the world. Indeed, the global photovoltaic energy production is expected to grow from the 99 GW calculated in 2017 to the 157 GW estimated in 2022[1].

[1] http://www.solarpowereurope.org/wp-content/uploads/2018/09/Global-Market-Outlook-2018-2022.pdf, visited 2019-02-25

In such a notable growth of the PV market, there is a corresponding need to increase the reliability of the monitoring systems devoted to detect faults and to ensure that the energy is produced with maximum efficiency. It is well known, in fact, that the PV modules are subject to faults and need periodic maintenance to prevent efficiency drops [1]. A frequent and effective monitoring allows for faults to be detected as soon as possible and thus to overcome losses in the energy production.

In [1] the authors perform a study of many solar modules in Australian plants, with the aim to analyze the typical factors that cause loss of efficiency in PV modules. All the considered events cause visible effects on the electric characteristics (e.g. changes in the resistance), on the visual appearance of the panels (e.g. cracks, dirt are easily visible) and on the temperature. Many defects that jeopardize power efficiency of a solar cell, in particular, cause a significant increase of temperature in the precise spot that a cell occupies. Those *Hot Spots* (HS) are thus observed very frequently because of the joule effect: whenever a PV cell is damaged, the resistance of that cell becomes higher and the power is no more conveyed in form of harvested energy but it is dissipated as heat.

Among the techniques developed to overcome the limitations of manual visual inspection, one of the most important is surely the aerial solar thermography [2]. Indeed, Unmanned Aerial Vehicles (UAVs) provided with infrared or thermal cameras, can be used to acquire a large amount of images related to the plants and then to determine the presence and the position of hot spots[3]; such anomalies appear on a thermal image as bright spots (see Fig. 1). The main advantage is that UAVs are cheap, they can bring significant payloads (for transporting the camera) and can be programmed for predefined flights that typically last up to 30 minutes.

The effectiveness of the hot spot detection for PV plants monitoring, fault diagnosis and quality assessment has been proved and experimented extensively [3] [4] [5].

In the state of the art, however, only a few methods focus on the processing time. In a common architecture, the video footage of the plant is acquired in a flight session, then the analysis is performed offline in a successive moment.

When the algorithm can run in real time, though, the pilot can use algorithm feedback for adjusting the flight parameters as needed and improve the accuracy of the analysis.

A last aspect concerns the localization of the detections. GPS is suggested as a means to locate faults [6], but the current precision of the GPS technology is not sufficient with respect to the size of the solar panels: with a 95% of probability the expected precision ranges from several tenths of centimeters to few meters [2]. This is an essential step: apart from detecting the fault itself, a real system should be able to provide the exact information about which panel exhibits the detected defect and, thus, lead a maintenance team to the position of the proper module to be replaced. To achieve this goal, the system can jointly use GPS and visual sensors:

[2]https://www.gps.gov/systems/gps/performance/accuracy/, visited 2019-02-25

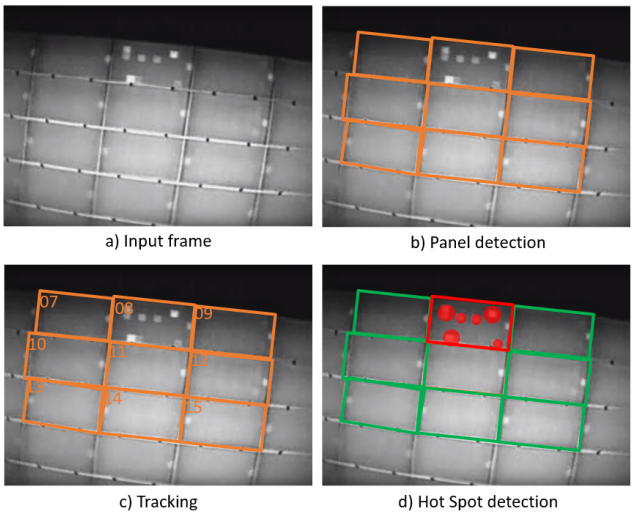

Figure 2: The phases of a typical PV monitoring system. a) the input image, b) the single panels are detected, c) each panel is tracked and associated to the ones in the previous frame by means of an id number d) Hot Spots are detected in every single panel to distinguish faulty units.

tracking algorithms are employed for accurate identification from the video itself [7].

1.1 Related work

A method for automatically assessing the state of photovoltaic plants has to implement a specific pipeline[7] : the single modules are segmented (also known as "detection" phase), then tracked (each panel is uniquely identified), the hot spots are detected and finally the state of each PV module is the output of the system. The output of each phase is represented in Fig. 2.

The first phase, namely the detection of the single PV modules, is the fundamental step in the pipeline: every error in this phase will inevitably affect the performance of any subsequent operation. Most of the works in the state of the art of PV fault detection by aerial imagery focus on the qualitative aspects of the task and its result rather than on quantitative performance. Quantitative evaluation is instead essential to ensure that the computer vision algorithm is accurate enough to take on the task of plant monitoring and be helpful in real environments.

Aghaei et al. [8] achieve localization of faults with a mosaicing technique: different images are matched with the Harris corner detector and stitched together in a bigger mosaic picture. The mosaic is then analyzed to detect anomalies; the authors do not report any quantitative result. In a previous work [9] the authors describe a procedure for defect detection, which adopts image binarization to find hot spots; the segmentation of different panels is performed using a Laplace operator. Even in this case, no quantitative results are reported.

In [10] and [11] an auxiliary RGB camera is used to segment panels on the basis of color information and a separate thermal camera to detect faults. This requires matching the images of the two cameras and setting a color threshold. Again, the quantitative results of the tests are not reported.

In [12] the panels are detected through a template matching technique combined with the GNSS data. F-score of 0.83 is the best achieved result for the detection of the panels, even if it is worth mentioning that the analysis has been carried out on a very small dataset.

The method proposed in [13] exploits thresholding to segment panels, where the threshold depends on the median value. This method expects that the temperature is lower at panel borders: while it is a reasonable assumption when panels are sufficiently far apart, in our dataset it fails because the panels are often adjacent and the junctions becomes hot on a regular basis.

In [7] and [14] a model based approach is used to detect panels. In [7] the grid structure of a large PV field is identified by using the Hough transform and a line selection algorithm, then a model of the single panel is used to actually find the modules in the grid structure and to correct detection errors made in previous phases; the resulting F-score is 0.87 on the same wide dataset we will use for our experiments. Arenella et al. [14] leverage similar heuristics, but the detection accuracy is not evaluated alone, only the overall system performance is reported. Both the algorithms require some thresholds to be set for each step, that depend on the flight setting and the plant structure itself.

1.2 Contribution

As shown in the previous Section 1.1, the detection of PV panels is often neglected in the state of the art methods and its performance are not thoroughly measured and evaluated; given the not negligible effect that a wrong detection can have on the subsequent operations, we argue that a PV panel detection method must be reliable enough to realize a system able to properly operate in real environment on a daily basis.

Looking at the state of the art just outlined, there is a clear need of a detection algorithm that is accurate enough, with a measurable and measured accuracy in real conditions, sufficiently high for practical application. Providing a quantitative measure is crucial to decide whether a given detection method would be suited for an application. Furthermore, some other qualities are desirable, even if not strictly needed, such as the capability to run in real time, in order to allow the pilot for real time correction of the flight parameters and the possibility to analyze directly the thermal images, with no need of auxiliary RGB images, simplifying the hardware setup and the need of calibration between the two cameras. Finally, most of the algorithms available nowadays for PV detection require to setup many thresholds and parameters that are specific for a plant and may slightly vary for every flight: for those algorithms one or more calibration flights are needed, and a re-calibration may be periodically needed as well. For example, the detection part of the method in [7], which achieves state of the art performance on the considered benchmark, requires setting at least three different thresholds, one for Canny edge detection, one for the Hough transform and one for the line selection algorithm. We want to stress the need for a method that does not need any plant-dependent configuration, usable as-it-is, in a plug-and-play fashion.

Given those considerations, we propose a method for the detection of PV panels based on a modern deep object detection framework, namely YOLO [15]. YOLO is an efficient network which leverages Deep Convolutional Neural Networks (DCNNs) to simultaneously perform region proposals and classification of patches. It is proven to be a versatile and effective method, yielding an high detection accuracy while having a low computational impact, i.e. it runs in real time on commonly available GPUs.

In this work we show how such a framework can be fit in a PV fault monitoring pipeline. Furthermore, we give a quantitative evaluation of the results on a publicly available dataset of thermal videos that were acquired in real plants.

The proposed approach has all the four required features: it is accurate (the quantitative evaluation will follow in Section 3) and it is already proven to be fast and able to run in real time even on lightweight embedded devices [16]. If properly trained, this approach is able to directly analyze thermal images, without the need for auxiliary footage from a calibrated RGB camera. Finally, no mandatory plant-dependent configuration is needed, so that the approach can be used as a plug-and-play method as desired.

To the best of our knowledge, this is the first time that a Deep Learning method is applied for the segmentation of modules in a PV plant.

2 PROPOSED SOLUTION

In order to detect PV panels from aerial thermal images, we propose to use YOLO, a widely adopted object detection framework based on deep learning [15].

The main advantage in using YOLO with respect to the older detectors based on Deep Convolutional Neural Networks (DCNNs), is that a single pass in the network will give all the information required to find the objects in the image, with no further computationally expensive steps. Older architectures require instead two separate steps: they first identify several "proposal" regions, where an object is most likely to be found, then separately classify each region to confirm or discard the proposal. To this aim, the YOLO architecture is simpler, yet faster and comparably accurate, since running a classifier on multiple proposal regions is more computationally expensive.

The YOLO framework was originally designed for the problem of generic object recognition, with the aim of automatically detecting different classes of objects and simultanouesly classify them. In our setting we do not need to distinguish different classes of objects, but just one, namely, PV modules from background. This makes the problem simpler allowing the architecture to be trained with significantly less data (only thousands of images, with respect to the hundreds of thousands or even more typically required for other tasks).

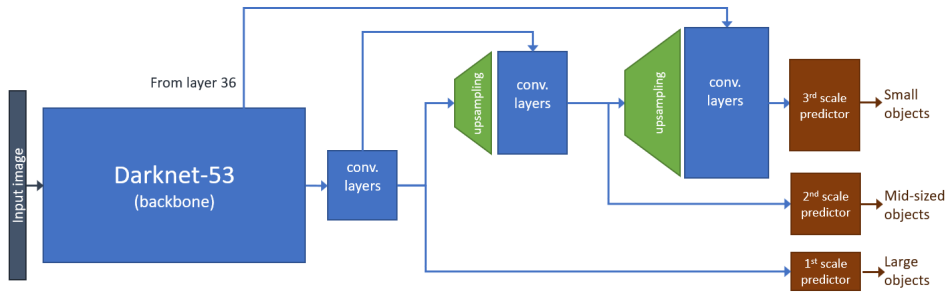

Figure 3: The YOLOv3 network architecture. Multiple skip connections are used to concatenate features extracted by early layers with more refined ones from last layers, in order to have effective detection on objects of all sizes.

In more details, YOLO works as follows: in a single stage detection architecture the whole image is used as input for the first layer. Then the network generates a list of bounding boxes, each with a measure of confidence.

The main problem in training a network for a detection task is the codification of the desired effect in the form of a loss function. The YOLO framework divides the image in a $S \times S$ grid. Each grid cell predicts B bounding boxes along with the confidence of that detection, namely a measure of "objectness". The loss function is designed to teach each cell only to detect objects that are centered in that cell.

At inference time, the objectness measure for each box is used to distinguish the correct detections from the non-detections. In the setup proposed in the original experimentation ($S = 7, B = 2$) it is only possible to detect 98 boxes, but usually only a few of them have an objectness value significantly greater than zero, due to the training procedure described above. The cell-based structure limits the possibility of detecting lots of small objects very close together, but for the considered application this problem has limited effect, since the PV modules are large enough and their centers always fall into the receptive fields of different cells.

The full network architecture provides 24 convolutional layers ("backbone" classifier network) with fully connected layers on top. The convolutional part is trained on the ImageNet dataset for object recognition. The final fully connected layer predicts the $S \times S \times 5B$ tensor, containing all the boxes from all the cells and their confidence values. When trained on a dataset with multiple objects, for each cell one "neuron" for each class is added, in order to assign a class label to the bounding boxes detected by that cell; in this work we do not need such feature, since we have just one "class" of objects namely PV panels.

Several versions of YOLO have been proposed in the last years; in more details, we will refer in the following to versions v2 [17] (hereinafter YOLOv2) and v3 [18] (hereinafter YOLOv3), which are respectively the fastest and the most accurate one. In more details, with respect to the original framework one of the crucial improvements in YOLOv2 is the use of the *anchor boxes*. In the original version of YOLO, each cell of the grid naively predicts the coordinates of the bounding box it is responsible for; in the improved v2 version, the network predicts the offsets from a set of specific *anchors*, that are predefined boxes with a given aspect ratio and size. Furthermore, the predicted offsets are expressed as relative with respect to the prior size: such mechanism allows the usage of the logistic activation function and greatly improves the stability of the training process. The anchor boxes are chosen running a k-means clustering on the training set, in order to choose priors that are appropriate for the considered dataset. This step is crucial in our work, because the particular shapes of the PV modules dictate particular shapes of the anchor boxes.

To improve the performance when dealing with small objects, YOLOv2 uses a skip connection (pass-through layer) to take advantage of the fine grained features when actual detections are predicted in the final layers. Finally the batch normalization is used, and the backbone network architecture is updated: instead of the 24 convolutional layers, a newer network called Darknet-19 is used. Darknet-19 leverages more modern architectural features yielding better accuracy at a significantly lower computational cost. It is worth noting that the YOLOv2 network is fully convolutional, and is trained with images of different sizes: this allows the detector to be used at various resolutions, and the authors show that bigger is better.

YOLOv3 introduces yet a new backbone classifier (Darknet-53) with more modern devices from the DCNN literature. Moreover, YOLOv3 explicitly predicts three different scales, using features taken from intermediate stages, as shown in Fig. 3. The resulting accuracy is significantly improved with arguably marginal effects on the inference time.

3 EXPERIMENTATION

In this section we will provide a quantitative evaluation of the proposed method. In Subsection 3.1 we describe the dataset used in our experimentation. In Subsection 3.2 we describe the metrics adopted to assess the performance of the methods and the experimental protocol used for the evaluation. In Subsection 3.3 we finally show the results, compare them with other approaches available in the literature and discuss the experimental findings.

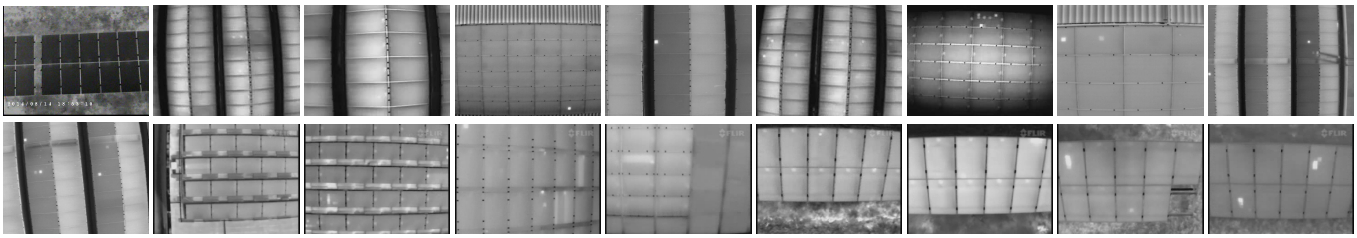

Figure 4: One frame from each video in the dataset. Video 1 to 18, from left to right, top to bottom.

3.1 Dataset

To the best of our knowledge, the only dataset available in the literature and provided with the annotations has been proposed by our research group and is available upon request[3].

The videos in the dataset include different PV plants, acquired from thermal cameras mounted on UAVs. The dataset consists of 18 videos and is an extended version of the one mentioned in [7]. The dataset contains a total of 50,449 panels and 4,939 hot spots.

In the dataset, different (non-optimal) flight conditions are reproduced: the altitude varies even in the same video; multiple paths are depicted (East to West, North to South and so on); the orientation and the inclination of the PV panels is variable as well; different cameras with different resolutions are used. Furthermore, the panels belong to different real PV plants and vary in size, shape, color and placement; the flights were performed under different weather, illumination and shading conditions. We believe that the dataset contains sufficient variations to reasonably represent many challenging situations faced in common applications that concern the automatic aerial thermal inspection of PV plants. A method trained and tested on this dataset is expected to be robust to variations of altitude, orientation and flight style as well as different manufacturers and models of panels, dispositions in the plant and various climatic conditions and illuminations. It is evident from Fig. 4, which shows one sample frame from each video, that the available video clips are very heterogeneous.

Videos 11 to 15 were used for testing, while all the others were used for training, with about 36,000 panels contained in training set and the remaining 14,500 used for testing. The same experimental setting is also used in previous works [7, 14]. Videos 16 to 18 are introduced in this paper, so we include them in the training set and leave the test set untouched to preserve the comparability with the approaches available in the state of the art.

3.2 Experimental protocol

We evaluate the performance in terms of Precision (P), Recall (R) and F-Score (F), following the same experimental protocol proposed in [14] and [7]. Those values are calculated in terms of the number of True Positives (TP), False Positives (FP) and False Negatives (FN), with the following formulas:

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$F = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$

More specifically, in order to discern if a detection should be considered a true positive or a false positive, we compute the overlap between the boxes associated to the detected panel at the frame i, O_i, and the ground truth GT_i at the same frame. We then consider the Pascal Criterion [19]: O_i is a true positive if the following condition is satisfied:

$$\frac{area(O_i \cap GT_i)}{area(O_i \cup GT_i)} \geq 0.4 \tag{4}$$

otherwise O_i can be considered a false positive. On the contrary, if an object GT_i exists in the groundtruth at frame i that has no matching detection O_i according to the Pascal Criterion, we consider it a false negative.

Thanks to the use of the Pascal Criterion we take into account missed and spurious detection but we also consider significant localization errors, that will count both as a false positive and a false negative.

In our evaluation protocol we use Videos 11 to 15 as a test set.

3.3 Results and discussion

In order to prove the effectiveness of the proposed approach, we present the results in two main settings, namely *parameter-free* and *fine-tuned*. In case of a "parameter-free" experimentation, the test videos are not used to tune the parameters. This experimentation aims to simulate the case in which the human operator does not need to perform any kind of configuration on the specific environment. This kind of "parameter-free" protocol allows also to test the generalization capability of the method over never seen environments.

On the other hand, the fine-tuned setting allows to show the effect of a plant-dependent fine-tuning, in which the first 30% of each test video is used for fine-tuning the network, while the remaining part is the effective test set. In other words, we are verifying how a small effort by the human operator (namely a fine tuning of the network) before the operating phase could improve the performance of the system. Fine tuning consists in training the Neural Network on the

[3] https://mivia.unisa.it

specific data from the fine-tuning set (first 30% of each test video) but starting from a pre-trained model, whose weights were learned from the training set. Such a kind of training converges faster than a more generic training from random or non domain-specific weights, and uses a lower learning-rate.

Moreover, we compare the performance achieved using two different versions of the detection framework, YOLOv2 and YOLOv3.

The experimental results are presented in Fig. 5. Here we also report the best performance available in the scientific literature using the same dataset (Previous work), proposed in [7]. It is important to highlight that such method requires parameter tuning on each video, so as to adapt to the various plants and flight settings. It implies that a heavy configuration setup from the human operator is mandatory. Viceversa, one of the main advantages deriving from the proposed approach is that the configuration procedure is not time-consuming (in the *parameter-free* setup this can be completely avoided), thus confirming that our solution is ready to be used in commercial applications.

With reference to the *parameter-free* setting, we see that the performance achieved by YOLOv3 is, as expected, significantly better than the one obtained with YOLOv2. Anyway, this is paid back in terms of time required for the processing, since it is about 2 times slower according to our benchmarks.

When fine-tuning is possible on plant-specific images (i.e. in the *fine-tuned* setting), the performance in terms of F-Score is boosted up to 95%, +4%. In particular the Precision is held constant to 94%, while the Recall raises to 98%, + 7%. This means that the fine-tuning allows to fix many false negatives that occur in the *parameter-free* setting.

It is worth noting that, in this experimentation, the results are presented on a frame-by-frame basis, where each frame is considered as an image by itself. The continuity of the video can be exploited to perform a temporal smoothing, leveraging basic tracking techniques [7]. Tracking consists in re-identifying the same panel frame by frame, so as to track its apparent trajectory in the scene. The same PV panel may be present in many frames and the detection system may miss it in certain cases: a common case is where the panel is at the border of the image; those cases will be considered False Negatives by our experimental protocol, while a protocol that also considers the effects of tracking will neglect those errors since they do not affect the performance of the complete system: the panel is correctly found in previous and/or successive frames, so that no actual miss occurs.

The same reasoning applies in the opposite case, when our system detects a panel that is not present in the groundtruth because it is only partially visible and it was not annotated as PV module. Such situations are considered errors in our experimental protocol, but would be neglected in a complete system since the tracking step will easily fix them.

From the experimental results it is evident that our approach has a much higher precision with respect to [7] (92% vs 83%) on average, while recall is only slightly improved. This means that number of false detections is halved on the target dataset with the most accurate version, YOLOv3.

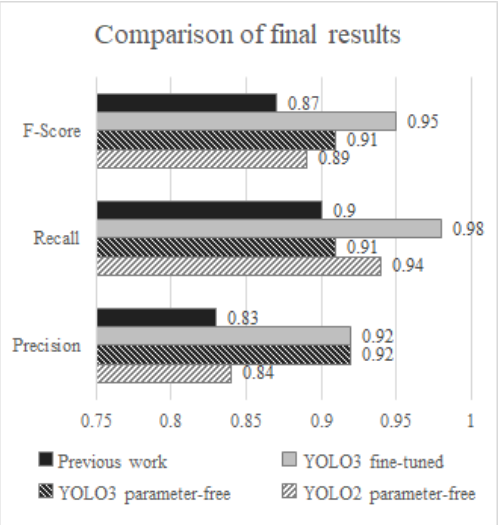

Figure 5: Results on the test set (videos 11 to 15).

4 CONCLUSION

In this work we focused on the problem of detecting PV panels in aerial imagery acquired from thermal cameras on board of UAVs by employing a deep learning framework. To the best of our knowledge, this is the first time a deep learning based methodology has been exploited to detect PV panels. We used the YOLO detection frameworks in its versions 2 and 3. We verticalized the above framework on the problem at hands.

Our method has a particular advantage with respect to state of the art methodologues: indeed, it does not need a heavy fine tuning procedure and the definition of a set of configuration parameters to be manually defined by the human operator for each specific scenario, thus making such approach particularly suited for its use in commercial applications. We conducted a huge experimentation over a dataset containing 18 different videos acquired in very different environments; the achieved results confirm the effectiveness of the proposed approach. In more details, we evaluated the performance in two different settings: YOLOv3 detector on our dataset achieves 91% F-Score when tested on typologies of plants that were never seen before by the system. The result is slightly better than the one achieved with YOLOv2 (89%). The accuracy is improved to 95% when the system is also fed with some plant-specific imagery: the improvement is mainly due to a greatly improved recall (up to 98%).

We think that the detection system experimented in this work can be used in a complete fault monitoring architecture due to its ease of configuration and its accuracy. The fact that the system can run in real time is a further added value.

REFERENCES

[1] Sinisa Djordjevic, David Parlevliet, and Philip Jennings. Detectable faults on recently installed solar modules in western australia. *Renewable energy*, 67:215–221, 2014.

[2] Harley Denio. Aerial solar thermography and condition monitoring of photovoltaic systems. In *Photovoltaic Specialists Conference (PVSC), 2012 38th IEEE*, pages 000613–000618. IEEE, 2012.

[3] Cl Buerhop, D Schlegel, M Niess, C Vodermayer, R Weißmann, and CJ Brabec. Reliability of ir-imaging of pv-plants under operating conditions. *Solar Energy Materials and Solar Cells*, 107:154–164, 2012.

[4] Yihua Hu, Wenping Cao, Jien Ma, Stephen J Finney, and David Li. Identifying pv module mismatch faults by a thermography-based temperature distribution analysis. *IEEE Transactions on Device and Materials Reliability*, 14(4):951–960, 2014.

[5] C Buerhop and H Scheuerpflug. Field inspection of pv-modules using aerial, drone-mounted thermography. In *Proc. 29th EU-PVSEC*, pages 2975–2979, 2014.

[6] John A Tsanakas, Long D Ha, and F Al Shakarchi. Advanced inspection of photovoltaic installations by aerial triangulation and terrestrial georeferencing of thermal/visual imagery. *Renewable Energy*, 102:224–233, 2017.

[7] Vincenzo Carletti, Antonio Greco, Alessia Saggese, and Mario Vento. An intelligent flying system for automatic detection of faults in photovoltaic plants. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–14, 2019.

[8] Mohammadreza Aghaei, Sonia Leva, and Francesco Grimaccia. Pv power plant inspection by image mosaicing techniques for ir real-time images. In *Photovoltaic Specialists Conference (PVSC), 2016 IEEE 43rd*, pages 3100–3105. IEEE, 2016.

[9] Mohammadreza Aghaei, Francesco Grimaccia, Carlo A Gonano, and Sonia Leva. Innovative automated control system for pv fields inspection and remote control. *IEEE Transactions on Industrial Electronics*, 62(11):7287–7296, 2015.

[10] Sergiu Dotenco, Manuel Dalsass, Ludwig Winkler, Tobias Würzner, Christoph Brabec, Andreas Maier, and Florian Gallwitz. Automatic detection and analysis of photovoltaic modules in aerial infrared imagery. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–9. IEEE, 2016.

[11] Francesco Grimaccia, Sonia Leva, and Alessandro Niccolai. Pv plant digital mapping for modulesâĂŹ defects detection by unmanned aerial vehicles. *IET Renewable Power Generation*, 11 (10):1221–1228, 2017.

[12] Pia Addabbo, Antonio Angrisano, Mario Luca Bernardi, Graziano Gagliarde, Alberto Mennella, Marco Nisi, and Silvia Ullo. A uav infrared measurement approach for defect detection in photovoltaic plants. In *Metrology for AeroSpace (MetroAeroSpace), 2017 IEEE International Workshop on*, pages 345–350. IEEE, 2017.

[13] P Guerriero and S Daliento. Automatic edge identification for accurate analysis of thermographic images of solar panels. In *Clean Electrical Power (ICCEP), 2017 6th International Conference on*, pages 768–772. IEEE, 2017.

[14] Alessandro Arenella, Antonio Greco, Alessia Saggese, and Mario Vento. Real time fault detection in photovoltaic cells by cameras on drones. In *International Conference Image Analysis and Recognition*, pages 617–625. Springer, 2017.

[15] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[16] Nils Tijtgat, Wiebe Van Ranst, Toon Goedeme, Bruno Volckaert, and Filip De Turck. Embedded real-time object detection for a uav warning system. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2110–2118, 2017.

[17] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[18] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[19] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010.