International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 5 Issue 1, November-December 2020 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Using Mask R-CNN to Isolate PV Panels from Background Object in Images

Muhammet Sait, Atilla Erguzen, Erdal Erdal

Department of Computer Engineering, Kirikkale University, Kirikkale, Turkey

ABSTRACT

Identifying foreground objects in an image is one of the most common operations used in image processing. In this work, Mask R-CNN algorithm is used to identify solar photovoltaic (PV) panels in aerial images and create a mask that can be used to remove the background from the images. This allows processing the PV panels separately. Using ML to solve this problem can generate more accurate results in comparison to more traditional image processing techniques like using edge detection or Gaussian filtering especially in images where the view might not be easily separable from the objects of interest. The trained model was found to be successful in detecting the PV panels and selecting the pixels that belong to them while ignoring the background pixels. This kind of work can be useful in collecting information about PV installation present in aerial or satellite imagery, or in analyzing the health and integrity of PV modules in large-scale installations e.g., in a solar power plant. The results show that this method is effective with a high potential for improved results if the model is trained using larger and more diverse datasets.

KEYWORD: *Machine learning, Mask R-CNN, detection, image segmentation, object recognition, solar energy, photovoltaics*

How to cite this paper: Muhammet Sait | Atilla Erguzen | Erdal Erdal "Using Mask R-CNN to Isolate PV Panels from Background Object in Images" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-5 | Issue-1, December 2020, pp.1191-1195, URL: www.ijtsrd.com/papers/ijtsrd38173.pdf

IJTSRD38173

INTRODUCTION

There is a growing interest in high quality information about small scale solar power (Photovoltaic, PV) installations among governments, agencies, and decision makers in order to provide better estimates of the growth in power demands and trends in renewable energy use. Currently, statistics on the use of solar energy are based on data from the importation and sales of PV panels. This methodology can only give rough estimates, and cannot keep track of quick local transitions. On the other hand, detecting PV panels in imagery collected by drones in larger scale installations helps process the images to find any faults or damages in the system.

The problem of identifying objects of interest in an image and isolating them from the background can be solved using numerous methods. The most obvious route would be to try using an edge detection algorithm in combination with some pre-processing for noise reduction and follow that with some steps to separate the foreground from the rest of the image. Processing certain types of images using filters and other mathematical edge detection and pixel manipulation techniques can be tricky due to the nature of the scene and what kind of imaging conditions and equipment are used.

The popularity of deep learning use in image processing has been growing and it has been applied in solving problems like road detection[1–3], scene labeling [4], vehicle detection[5], detection of people[6], and detection of buildings[7]. Convolutional Neural Networks (CNNs) were used in previous works for the task of classification and detection of solar panels, where CNNs produced the best results [8].

ML algorithms can provide crisp edges and locate the object of interest in various views often resulting in much better outcomes compared to other image processing methods. Most notably, they surpass the traditional techniques when the objects of interest are surrounded by other background objects in different types of environments, i.e., if the detected objects are small compared to other objects in the view, or if the studied images have different attributes like wide difference levels of brightness and contrast, different imaging resolution, etc.

In this work, one of the ML algorithms, Mask R-CNN[9], is investigated to determine its suitability and effectiveness in identifying photovoltaic modules in aerial photographs taken by a drone flying over a power plant installation. The dataset of images used in this study was collected and made available online by SenseFly systems, using their drone eBee Classic[10]. A number of other available datasets are reviewed in a report by Curier et al., which can be useful for developing studies in this area [11].

Table 1, Technical data about the studied dataset

Groundresolution	14.23 cm/px (5.6 in/px)
Coverage	0.08 square km (0.03 sq. mi)
Flight height	70 m (229.6 ft)
Number of images	1075
Image format	TIFF

Applying Mask R-CNN

During our work on processing a dataset of PV panel images for fault detection, there was a need for treating the PV panel areas in isolation from the rest of the image pixels. This allowed better results and less noise in the output since all interactions with background objects are eliminated. Mask R-CNN algorithm was selected for this application and its results were evaluated.

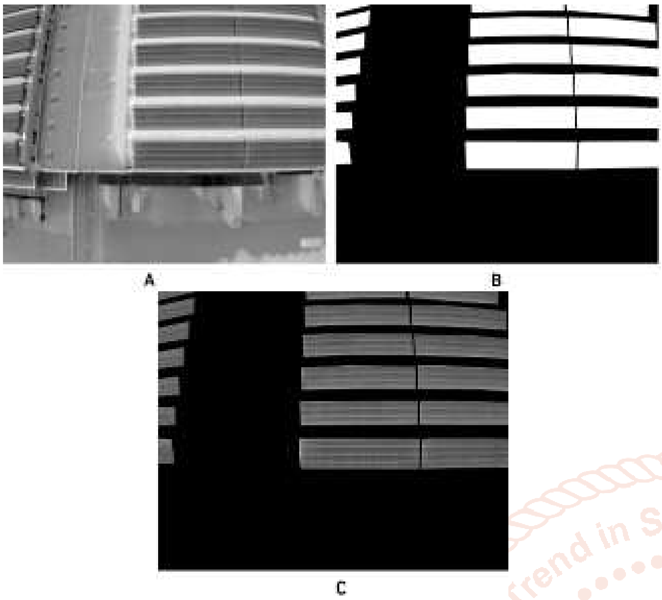
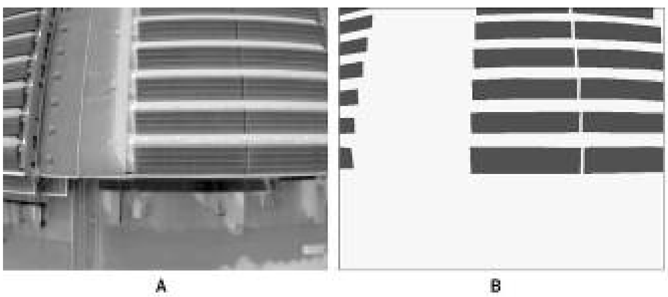


Figure 1 One of the processed images, the respective generated mask, result of multiplication

Mask R-CNN is a deep learning algorithm designed to detect objects in an image and create a segmentation mask for each identified object. The algorithm uses Convolutional Neural Networks (CNNs) as a backbone. Such networks are widely used to perform image classification and recognition, such as face recognition or medical diagnosis.

Some of the computer vision tasks that can be solved using CNNs:
- Classification: does the object of interest appear in the image?
- Object detection: how many objects are there and what are their positions?
- Semantic segmentation: which pixels belong to objects of interest.
- Instance segmentation: determines the pixels for each of the object instances.



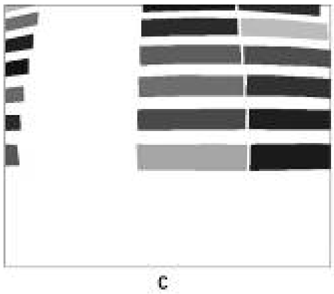


Figure 2 Comparison between semantic and instance segmentation. Original image (A), semantic segmentation result (B), instance segmentation result (C)

The problem of removing the background can be solved using semantic segmentation or sample segmentation methods. By using deep learning for this step, more reliable results compared to traditional image processing techniques were achieved. It also made the software more useful when processing image datasets captured under different conditions, or when using different types of imaging equipment, as this causes image properties to change. CNNs, on the other hand, tend to give better results and are better equipped to solve this problem.

Various algorithms have been considered during the development of our application. Among the algorithms investigated: Fast R-CNN [12], deep image matting[13], [14] and other background removal studies [15], [16]. Canny edge detection and Sobel filter-based methods were also considered.

Mask R-CNN [9]was preferred over other algorithms for the following reasons:
- Multiple open-source applications are available and ready to use;
- Ease of use as the algorithm is well explained and documented;
- Training time is short;
- Its results are superior to other algorithms;

A subset of the complete dataset was selected from the original dataset to train the segmentation model. This new dataset is split into two groups: a training set used to train the model, and a validation set to adjust the model.

Both the training and validation set consist of:
1. Real images themselves (in original condition).
2. PV module mask corresponding to the fields of PVMs.

Masks used in the training and validation of the model were created manually by a human annotator and stored as PNG images where pixels have only two values: 0 for background pixels, and 1 for PV pixels. The masks are later converted to JSON annotations that can be loaded to be used to train the Mask R-CNN model.

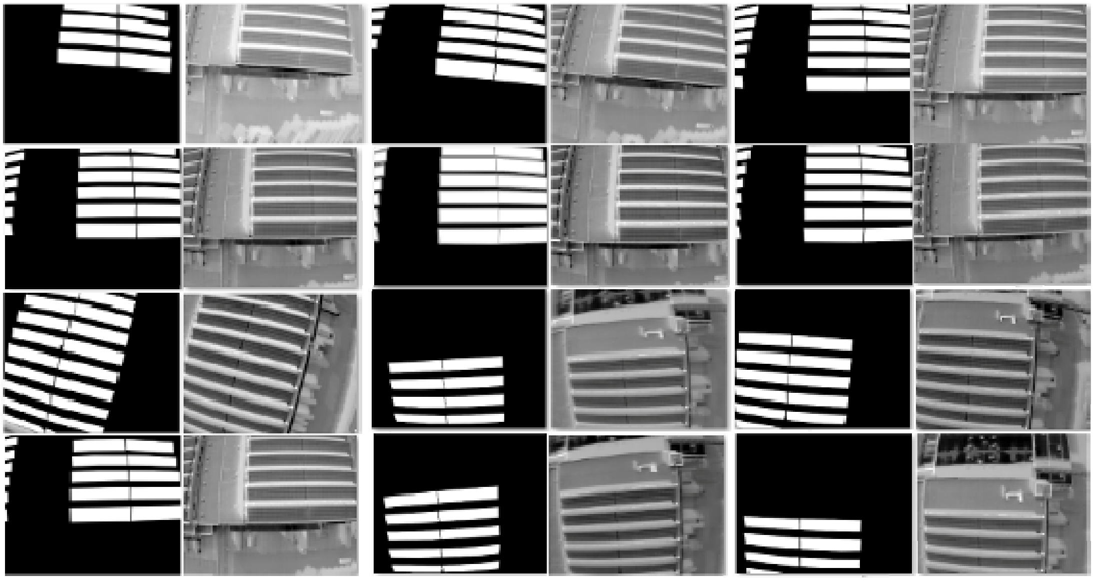


Figure 3 Example images from the studied dataset along with their masks

Training the model

To train and evaluate the ML model, Detectron2 [17] was chosen as an implementation of Mask R-CNN in this study. It is a modern open-source software system that enhances previous deep learning models and provides a large number of modern ML algorithms implementations ready for training and use.

To prepare the dataset for training and evaluation, masks have been transformed into a custom JSON annotation that can be read by Detectron2 and used directly. This annotation was developed as part of the Microsoft COCO dataset [18]. This format is supported by many deep learning libraries, making it the current de facto standard for image segmentation datasets. JSON annotations were generated from mask files stored as PNG images using a tool called pycococreator[19].

Figure 4 Sample of the training input data

After preparing the training images and JSON annotations, the training is executed on the GPU to minimize training time. Google Colab[20]is used for this step. Colab is a free service that allows running machine learning code written in Python on servers equipped with NVIDIA Tesla K80 or P100 GPUs.

To achieve higher accuracy from the model using a small training set, transfer learning[21] was used. This means that instead of starting the training process from scratch, the training starts from a pre-trained model on a different dataset. This is useful because the model will know what to look for even if it cannot yet define our custom object classes.

The starting model selected is the R50-FPN Mask R-CNN model that was pre-trained on COCO dataset. A new "pvm" class was added and the model was tweaked so that it can recognize these objects and create segmentation masks for them.

Experimental results

The performance of the machine learning algorithm is evaluated using a test dataset during the training phase. The resulting Average Precision value is available in table 2. This metric shows how good the algorithm is when finding masks for PVM class objects. More performance metrics results are provided in table 3, which indicate the accuracy, sensitivity and recall of the trained model.

The images used for both training and testing the models were selected from the original dataset so that blurry images and identical frames or almost identical frames were eliminated. Images that did not contain PV panels at all were also not used in the training or validation of the model.

Table 2, Bounding box Average Precision per category

Class	AP
pvm	75.534

Table 3, Performance measurements of the trained model

Accuracy	Sensitivity	Recall
0.774	0.75	0.877

Two examples of pictures used to evaluate our educated model are shown in.

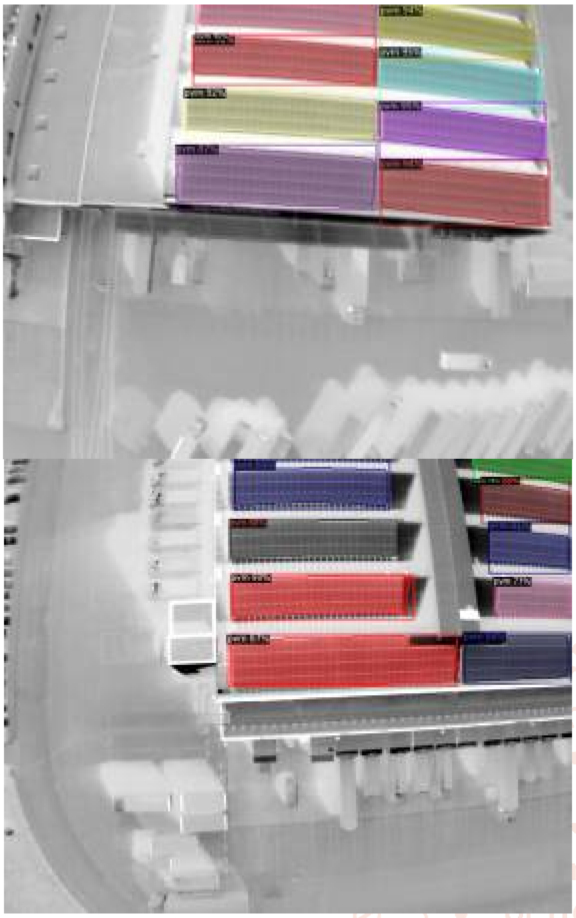


Figure 5 Sample result of Mask R-CNN model validation

Conclusion

Using ML algorithms in image segmentation and object isolation from the background can be more useful and accurate than traditional algorithms in manyuse cases depending on the type and nature of the processed images. The advances in both hardware and ML algorithms and libraries allow applying these novel techniques in solving older problems resulting in more accurate outputs. Trained models on large training datasets can identify objects of interest in different environments and under various imaging conditions, compared to traditional image processing which usually assumes certain conditions that must be met for the algorithm to give optimal results. The results show that the selected algorithm is effective and accurate for this task. The performance measures demonstrate that the algorithm could detect most of the true PV pixels while avoiding background pixels successfully. The model was only trained using a small dataset which contained images taken using the same equipment under the same environmental conditions. Better results are expected to be achieved when using larger datasets with more diverse images to train and validate the model. This work was used in part to help assess large-scale PV installations and detect faults and malfunctions. It can also be useful in information gathering applications where PV panels are detected in satellite and aerial images.

References

[1] V. Mnih, G. E. Hinton, Learning to detect roads in high-resolution aerial images, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), Springer Verlag, 2010: pp. 210–223. https://doi.org/10.1007/978-3-642-15567-3_16.

[2] J. Fritsch, T. Kuhnl, A. Geiger, A new performance measure and evaluation benchmark for road detection algorithms, in: IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC, 2013: pp. 1693–1700. https://doi.org/10.1109/ITSC.2013.6728473.

[3] C. César, T. Mendes, V. Frémont, D.F. Wolf, D. Fernando, Exploiting Fully Convolutional Networks for Fast Road Detection, 2016. https://hal.archives-ouvertes.fr/hal-01260697 (accessed December 17, 2020).

[4] N. Audebert, B. Le Saux, S. Lefèvre, Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-scale Deep Networks, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 10111 LNCS (2016) 180–196. http://arxiv.org/abs/1609.06846 (accessed December 17, 2020).

[5] X. Chen, S. Xiang, C. L. Liu, C. H. Pan, Vehicle detection in satellite images by hybrid deep convolutional neural networks, IEEE Geosci. Remote Sens. Lett. 11 (2014) 1797–1801. https://doi.org/10.1109/LGRS.2014.2309695.

[6] D. C. De Oliveira, M. A. Wehrmeister, Towards real-time people recognition on aerial imagery using convolutional neural networks, in: Proc. - 2016 IEEE 19th Int. Symp. Real-Time Distrib. Comput. ISORC 2016, Institute of Electrical and Electronics Engineers Inc., 2016: pp. 27–34. https://doi.org/10.1109/ISORC.2016.14.

[7] O. A. B. Penatti, K. Nogueira, J. A. Dos Santos, Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?, in: IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., IEEE Computer Society, 2015: pp. 44–51. https://doi.org/10.1109/CVPRW.2015.7301382.

[8] J. M. Malof, L. M. Collins, K. Bradbury, A deep convolutional neural network, with pre-training, for solar photovoltaic array detection in aerial imagery, in: Int. Geosci. Remote Sens. Symp., Institute of Electrical and Electronics Engineers Inc., 2017: pp. 874–877. https://doi.org/10.1109/IGARSS.2017.8127092.

[9] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2020) 386–397. https://doi.org/10.1109/TPAMI.2018.2844175.

[10] SenseFly, Solar Panel Installation, (n.d.). https://www.sensefly.com/education/datasets/?dataset=1416 (accessed December 2, 2020).

[11] V. Of, T. H. E. Available, Monitoring spatial sustainable development: Semi-automated analysis of satellite and aerial images for energy transition and sustainability indicators, ArXiv. (2018).

[12] R. Girshick, Fast R-CNN, Proc. IEEE Int. Conf. Comput. Vis. 2015 Inter (2015) 1440–1448. https://doi.org/10.1109/ICCV.2015.169.

[13] N. Xu, B. Price, S. Cohen, T. Huang, Deep image

matting, Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017. 2017-Janua (2017) 311–320. https://doi.org/10.1109/CVPR.2017.41.

[14] D. Swiggett, Image Matting and Applications, (n.d.) 1–17.

[15] A. Nikitakis, A novel Background Subtraction Scheme for in - Camera Acceleration in Thermal Imagery, 2016 Des. Autom. Test Eur. Conf. Exhib. (2016) 1497–1500.

[16] D. Kim, J. Youn, C. Kim, Automatic photovoltaic panel area extraction from UAV thermal infrared images, J. Korean Soc. Surv. Geod. Photogramm. Cartogr. 34 (2016) 559–568. https://doi.org/10.7848/ksgpc.2016.34.6.559.

[17] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, Detectron2, (2019). https://github.com/facebookresearch/detectron2.

[18] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: Common objects in context, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8693 LNCS (2014) 740–755. https://doi.org/10.1007/978-3-319-10602-1_48.

[19] Waspinator, waspinator/pycococreator: Helper functions to create COCO datasets, (2018). https://github.com/waspinator/pycococreator.

[20] Colaboratory, (n.d.). https://colab.research.google.com/notebooks/intro.ipynb (accessed December 17, 2020).

[21] Y. Wei, Y. Zhang, J. Huang, Q. Yang, Transfer Learning via Learning to Transfer, in: 35th Int. Conf. Mach. Learn., PMLR, 2018: pp. 5085–5094. http://proceedings.mlr.press/v80/wei18a.html (accessed December 17, 2020).