IDENTIFYING OIL PADS IN HIGH SPATIAL RESOLUTION AERIAL IMAGES USING FASTER R-CNN

Aswin Sunil[1, *], Sajithvariyar[1], V.V., Sowmya[1], V., Ramesh Sivanpillai[2], Soman, K.P[1]

[1] Center for Computational Engineering and Networking (CEN), Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, India - aswinkisalayam@gmail.com, (ORCID: 0000-0003-3944-8155) – v v_sajithvariyar@cb.amrita.edu, v_sowmya@cb.amrita.edu

[2] Wyoming GIS Center, University of Wyoming, Laramie, WY 82072, USA (ORCID: 0000-0003-3547-9464) – sivan@uwyo.edu

KEYWORDS: Deep learning, Object detection, Faster R-CNN, Computer vision

ABSTRACT:

Deep learning (DL) methods are used for identifying objects in aerial and ground-based images. Detecting vehicles, roads, buildings, and crops are examples of object identification applications using DL methods. Identifying complex natural and man-made features continues to be a challenge. Oil pads are an example of complex built features due to their shape, size, and presence of other structures like sheds. This work applies Faster Region-based Convolutional Neural Network (R-CNN), a DL-based object recognition method, for identifying oil pads in high spatial resolution (1m), true-color aerial images. Faster R-CNN is a region-based object identification method, consisting of Regional Proposal Network (RPN) that helps to find the area where the target can be possibly present in the images. If the target is present in the images, the Faster R-CNN algorithm will identify the area in an image as foreground and the rest as background. The algorithm was trained with oil pad locations that were manually annotated from orthorectified imagery acquired in 2017. Eighty percent of the annotated images were used for training and the number of epochs was increased from 100 to 1000 in increments of 100 with a fixed length of 1000. After determining the optimal number of epochs the performance of the algorithm was evaluated with an independent set of validation images consisting of frames with and without oil pads. Results indicate that the Faster R-CNN algorithm can be used for identifying oil pads in aerial images.

1. INTRODUCTION

Recent advances in Artificial Intelligence (AI) techniques have led to relatively accurate and rapid detection of features or objects in aerial photographs. Deep learning (DL) methods, a part of AI techniques, are used for identifying objects in aerial and ground-based images (Ayyad et al., 2020). Automatic vehicle detection (Sowmya et al.,2018), straight road detection, building detection (Kavin Kumar D et al., 2018), crop monitoring (M. Ho et al., 2019) are few examples of object identification using DL methods. Identification of complex features such as biological or man-made materials of different shapes with poor contrast from their background continue to be a challenge in this field.

Oil pads are examples of complex features in aerial images. They are distributed across large geographic areas and their number will change depending upon the availability of the oil and its market price. Crude oil is one of the sources for big revenue generation in states like Texas, North Dakota and Wyoming (Baumeister et al., 2016). Wyoming ranked eighth in crude oil output in the United States in 2019. There are a lot of open spaces and wildlife migrate throughout the year in these spaces.

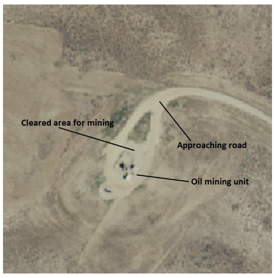


Figure 1. An example of an oil pad

[*] Corresponding author

Oil pads established in these open spaces can impact the movement of wildlife. Therefore the ecologists, land use planners, and policy makers need up-to-date and accurate information about the geographic location of these oil pads. Figure1 shows a portion of an aerial photograph containing an oil pad. The area of land is cleared for a drilling unit and there is an access road connecting the oil pad to the main road. These oil pads are constructed in various sizes, generally spanning across four or five acres. The mining unit looks like a cylindrical well in an aerial view.

Manual identification and mapping of oil pads in aerial photos are time consuming and prone to error. It is challenging to identify oil pads using classical pixel-based classification methods since they appear similar to the background soil (Moser et al., 2017). AI algorithms can be used for identifying oil pads in true color aerial images.

There are various DL-based object detection algorithms (Jiao et al., 2019) like YOLO (You Only Look Once) (Redmon et al., 2017), SSD (Single Shot MultiBox Detector) (Liu et al., 2019), Faster R-CNN (S. Ren et al., 2017). Faster R-CNN is used for detecting objects with the help of region proposal networks (S. Ren et al., 2017). Faster R-CNN requires less time for processing data, and it is also more accurate than its previous versions Fast R-CNN (Girshick et al., 2015) and R-CNN.

Xia et al., (2018) used Faster R-CNN to identify 15 objects in 2806 images and achieved an average accuracy of 60.5%. In the aerial images collected from different platforms and sensors, the objects were in a variety of shapes, scales, and orientations. In this study, the algorithm was trained with horizontal bounding box ground truth. Faster R-CNN was used to identify objects in complex battlefield environments and achieved an average accuracy of 94.2% (Xu et al., 2020). Faster R-CNN achieved an accuracy of 94.9% and recall of 95.4% while identifying self-blast glass insulator's location in aerial images (Ling et al., 2019). Ho et al., 2019 successfully identified watermelons present with other complex objects in aerial images with Faster R-CNN. In this study, we tested Faster Faster R-CNN to identify oil pads in aerial images.

2. MATERIALS AND METHODS

2.1 The oil pad aerial image dataset

The images used in this study were downloaded from Wyoming Geographic Information Science Center at University of Wyoming (USA).

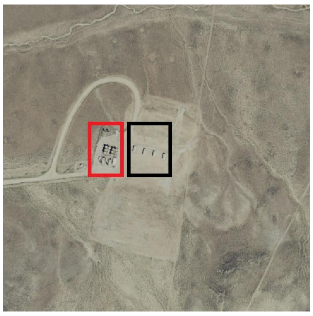
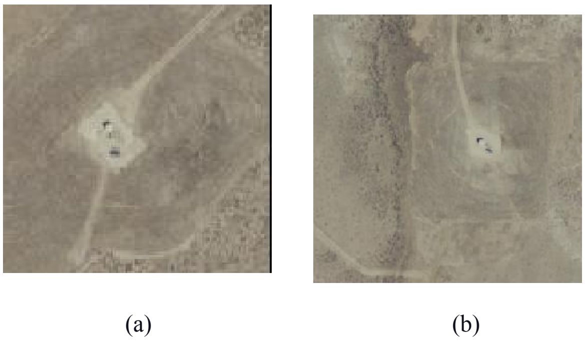


Figure 2. Oil pad used in the present work were parking area in red colour (left) and buildings in black colour (right).

These high spatial resolution (1m x 1m), true colour images were acquired in 2017. Each image contained several oil pads with one or more mining units. In addition to the oil pad other man-made structures such as storing units, vehicles, approaching roads were also present (Figure 2).

2.2 Dataset preparation and preprocessing

The original oil pad image contains multiple oil pads with different numbers of oil mining units, storing units, parking areas, other buildings, roads, and transporting vehicles (Figure 1). From the larger image that contained multiple oil pads, we cropped subset containing only one oil pad and a nearby storage unit. Next, the images were prepared for training and validation. The size of the cropped image is fixed as 256 x 256. The cropped images with a single oil pad (Figure 3) were used to train the deep learning algorithm.



(a) (b)

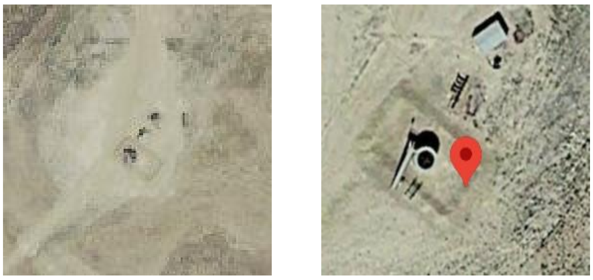
Figure 3. Sample data used for training in the present work.

In certain images there were multiple oil pads, and it was difficult to crop individual oil pads and surrounding features. This was due to the presence of other features like buildings, parking areas, more than one storage unit either by or very close to the target oil pad. In addition to this, in aerial images, it is very tough to manually differentiate the storing buildings, parking areas, oil pads containing more than one oil mining unit with oil pads containing single mining units. These images were excluded from the study.

2.3 Image annotation

The images used in this study (Figure 2) were geographically referenced i.e., each pixel on the image was encoded with its latitude and longitude coordinates (Geo-tagged images). ERDAS Viewfinder application (Saxena et al., 2015) is used to validate the cropped images created for the present work.

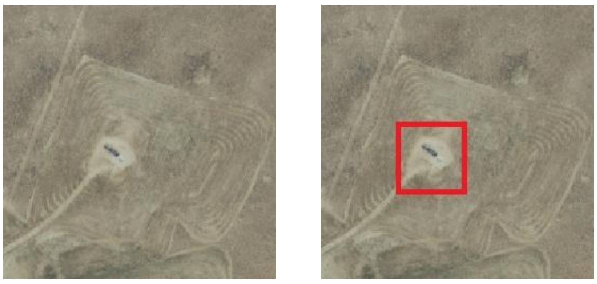
The latitude and longitude values corresponding to each oil pad were entered in Google Earth to confirm the interpretation. The spatial resolution of the Google Earth images was higher than the ones used in this study (Figure 4 a and b). Using this method, we selected images with a single oil pad from other buildings, which appeared like oil pads. As the original images have low resolution, it was challenging to identify the oil pad. Therefore, the bounding box was manually drawn while visually comparing with the high-resolution Google Earth images.



(a) (b)

Figure 4. Sample data for image validation (a) cropped image (b) corresponding google earth image.

The task of attaching metadata to a dataset is referred to as data annotation. Tags, images, and videos are typical examples of metadata. Including specific and precise tags is an important step in creating a dataset for the training of the supervised machine learning models. After sufficient annotated data were trained by a machine learning model, the algorithm may begin to recognize the same patterns which are present in new unannotated data. The most commonly used image annotations are bounding box, Polygon, Keypoint, Lines, and splines annotation (Padilla et al., 2020).



(a) (b)

Figure 5. Sample data for image annotation (a) cropped images using photoshop (b) the corresponding annotated images using VGG image annotator.

In the proposed work we use bounding box annotation. Bounding box annotation consists of square boxes, which were drawn on the images. The bounding box generally aids a machine learning model to identify the area inside the annotated box (Region of Interest (ROI) as a discrete or target object form.

Figure 5(a) shows a sub-set image that displays one oil-pad. The bounding box annotation includes the ROI in our input images. In this example, we included a mining unit, cleared the area, and a road approaching near to the oil pad as ROI inside our bounding box (Figure 5b). The bounding box annotation for the oil pad used in this study is done by VGG annotation tool (Dutta et al., 2019), which is a free online image annotation software.

2.4 Faster R-CNN

Faster R-CNN is the modern technique from the family of R-CNN-based object detection algorithms. Faster R-CNN is less time-consuming than its previous versions Fast R-CNN and RCNN. The main difference between Faster R-CNN with its previous versions is the presence of a region proposal network (RPN) for generating the region of interest instead of selective search. In Faster R-CNN, the input image is fed into the convolutional neural network and then we get a corresponding feature map as the output of the convolutional neural network. Then we pass these feature maps through the Region Proposal Network (RPN) which gives the corresponding object proposals and finally using an object classification and bounding box regression, the feature maps are classified and the bounding boxes are predicted. The object detection of Faster R-CNN is divided into two stages. A region proposal algorithm using Region Proposal Network (RPN) and a Fast R-CNN as a detector network (Figure 6).

The first step of the algorithm in the region proposal network is also known as RPN. The task of the Region Proposal Network (RPN) is to predict the area where the object is possibly present. In this work, the object is an oil pad. The RPN is a sub-CNN network because we are performing some convolutional neural network tasks; here in this work we are using resnet50 (He et al., 2016). The output which is getting from this particular network is a feature and then from RPN we get a feature map with corresponding anchor boxes.

The region proposal network (RPN) starts with the input image. The input image of the oil pad is fed into a backbone convolutional neural network. Reset 50 is used here as a backbone convolutional neural network. With the help of RPN, the algorithm finds out the area in which the oil pad is possibly present in the image. Finally, once you get the area where the object is present, the algorithm labels that area as foreground class. In this case, wherever the oil pad is present in an image, the algorithm labels it as foreground class and the area where the oil pad is not present is labeled as background class. Then the area where the oil pad is present and which is labeled as foreground class moves forward to the next step of the algorithm. The region proposal network is to find out the area where the object is present and the area where the object is absent with the help of anchor boxes. In the first step, it generates anchor boxes.

Anchor boxes are a set of predefined bounding boxes with some height and width. Here, we are using different sizes of anchor boxes because we want to capture objects of every size. The size of the oil pad varies with respect to the whole image. In our problem, we are using anchor boxes of different sizes, therefore we can detect oil pads of any size and shape. The anchor boxes are generated during convolution. From the feature map that we get after convolution, the network learns whether the oil pad is present in the input image or not, and also it learns the corresponding location and size of the oil pad in the image. The algorithm places a set of anchor boxes on the input images based on the location present in the output feature map from the backbone convolutional neural network. Once it generates anchor boxes the next step is to calculate the intersection over union (IoU).

IoU simply means the overlapped area of the ground truth box and the predicted box. If the overlapped area is more than 70%, then that object is detected by the algorithm. Otherwise, if the overlapping is less than 70% then the algorithm does not learn from that particular input. The anchor box with the higher IoU labeled as foreground class because the final task of RPN is to find out the area where the object is present and label it as foreground class.

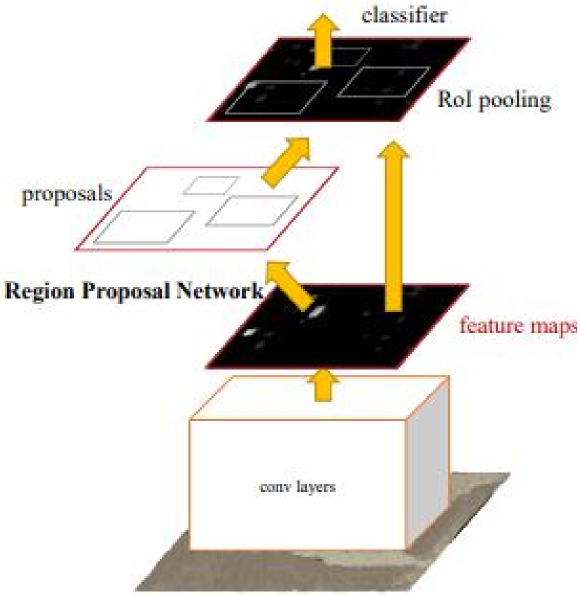
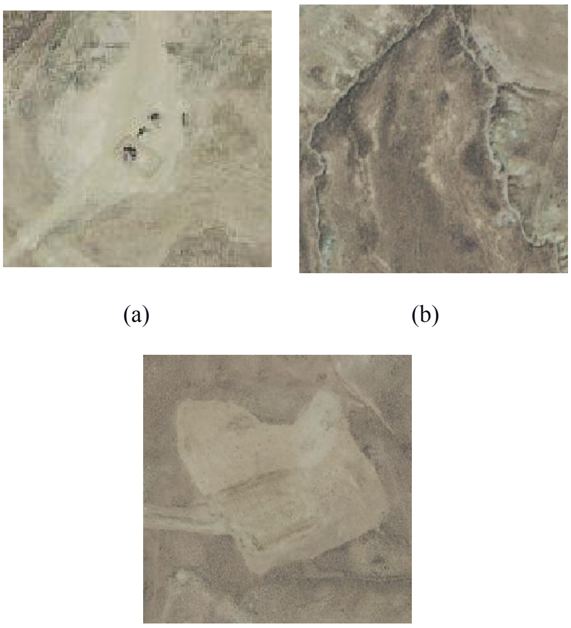


Figure 6 Architecture of Faster R-CNN (S. Ren et al., 2017)

Hence, the algorithm labels that area as a foreground class where the value of IoU greater than 70% and the area where other anchor boxes which have IoU less than 70% should label as background class. Therefore the algorithm tells the foreground class and background class based on IoU value. Finally, the anchor boxes which are labeled as foreground class go to the next step. The classifier finds out whether there is an object in the image and the task of the regressor is to draw a bounding box in the object which we classified and it refine the bounding box.

Faster R-CNN uses the object detection network which is almost the same as the one which is used in the Fast R-CNN (Girshick et al., 2015). That is why it is called a Fast R-CNN detector. Fast R-CNN detector also shares the same convolutional network as the backbone that is resnet50. In the ROI pooling the output of RPN are anchor boxes where the objects are present. In this step, we are using anchor boxes of different sizes. That means the output anchor boxes from RPN would be different in size because we are dealing with different sizes of oil pads. Here comes the significance of ROI. During the training of the region proposal network, it assigns two class labels for each anchor box either 0 or 1. By using IoU value it provides a binary class label for anchor box with oil pad and anchor box without oil pad. In the case of positive anchors, the algorithm considers two conditions, first one is the anchor boxes having the highest IoU value. Second is anchor boxes have more than 70 % IoU value concerning the ground truth box. (Ming-Jui Ho et al., 2019)The oil pad image and corresponding annotation information were passed as input data to the algorithm. The training dataset contains positive samples, i.e., images that contain oil pad. We adopted Resnet50 as backbone CNN which was pre-trained on image net. Stochastic gradient descent (SGD) (Qian et al., 2015) was used as the optimizer and Mean Absolute Error (MEA) (Myttenaere et al., 2015) was used as the loss function. The algorithm was trained with the number of epochs initially set at 100 and increased by increments of 100 until the final value of 1000. We fixed the epoch length as 1000 for all epochs. The train dataset contains 200 images and the validation images consist of 30 images with a lot of variation. Validation set contains 10 oil pad images, 10 non-oil pad images, and 10 abandoned oil pad images.



(c)

Figure 7. Validation data used in the present work (a) Image with an oil pad (b) Image without an oil pad (c) Abandoned oil pad

We used more variety of data for the validation set. The validation set consists of images with oil pad (Figure 7a), without oil pads (Figure 7b) and abandoned oil pads (Figure 7c). Abandoned oil pads appear similar to active ones (Figure 7a) but there is a little difference. The area of the land should be cleared. The shape of cleared land is almost the same as active oil pads. The main difference from the active oil pad is that the abandoned oil pads do not contain any kind of oil mining unit (Figure 7c).

Faster R-CNN is trained for different epochs. The performance of the trained model is evaluated using standard metrics such as accuracy (1), precision (2), recall (3), F1 score (4), and specificity (5). In the present work, we have two target classes: oil pad and non-oil pad.

Where the true positive (TP) Indicates the number of oil pads successfully detected by the algorithm. True negative (TN) indicates the number of non-oil pad objects that are correctly predicted as non-oil pads. False-positive (FP) means the number of non-oil pad objects that are wrongly detected as oil pads. False-negative (FN) means the number of oil pads that the algorithm did not recognize as an oil pad.

.

$$Acuracy = \frac{TP + TN}{Total\ number\ of\ \ test\ data} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = Sensitivity = \frac{TP}{TP + FN} \quad (3)$$

$$F1\ Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (4)$$

$$Specif\,icity = \frac{TN}{TN + FP} \quad (5)$$

3. RESULT AND DISCUSSION

Table 1 shows the performance of the Faster R-CNN on high-resolution aerial images of oil pads and non-oil pads. From the table, we can understand that Faster R-CNN performs very well on 500 epochs. At the 500th, epoch algorithm obtains an accuracy of 0.90 and precision of 0.90. This indicates that 90% of oil pads are correctly classified as oil pads. The ability of Faster R-CNN to detect a true oil pad is very high at 500 epochs. At the same time, the probability that the Faster R-CNN classifies a non-oil pad as an oil pad is very rare. We tested our model using different conditions and the results values are getting exactly the same as shown in the table 1.

Figure 8 shows a few examples of test results of oil pad images on the 500 epochs. The red box represents the oil pad identified by the proposed model. Almost in every case, the model performed very well. But, in the case of the image (Figure 8b), the model identified it as a non-oil pad because we couldn't see any box on that image. That peculiar oil pad is slightly different from other oil pads because the approach roads and cleared areas are not visible much when compared with the images used for training the Faster R-CNN.

Figure 9 shows the test result of the Faster R-CNN model of non-oil pad images on the 500th epoch. We obtained the highest metrics for 500 epochs which is shown in table 1. The model predicted correctly in almost every case except in figure 9c. In these cases, the model gives false positives at the 500th epoch because the area inside the red box identified by the model contains ground features almost similar to an oil pad.

Number Of epochs	**Accu racy**	**precision**	**Recall**	**F1 Score**	**Specifi city**
100	0.70	0.70	0.70	0.70	0.70
200	0.85	0.81	0.90	0.85	0.80
300	0.85	0.81	0.90	0.85	0.80
400	0.85	0.81	0.90	0.85	0.80
500	0.90	0.90	0.90	0.90	0.90
600	0.85	0.88	0.80	0.84	0.90
700	0.70	0.83	0.50	0.62	0.90
800	0.75	0.77	0.70	0.73	0.80
900	0.50	0.50	0.40	0.44	0.60
1000	0.70	0.66	0.80	0.72	0.60

Table 1. Performance of Faster R-CNN evaluated using accuracy, precision, recall, F1 score and specificity on test data.

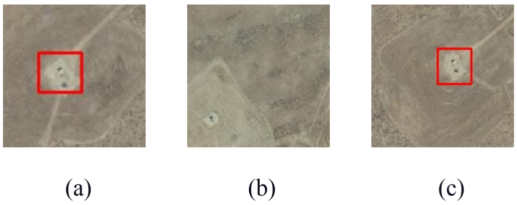

(a) (b) (c)

Figure 8. A sample test result of oil pad images on the 500th epoch (a) Test image contain oil pad and algorithm detected as an oil pad (b) Test image contain oil pad and algorithm not detected as an oil pad (c) Test image contain oil pad and algorithm detected as oil pad

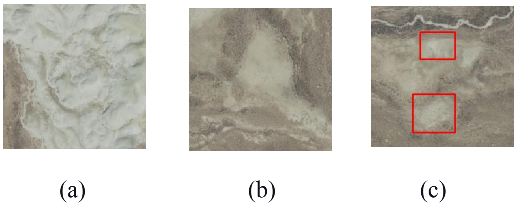

(a) (b) (c)

Figure 9. A sample test result of non-oil pad images on the 500th epoch (a) Test image contains without oil pad and algorithm predicted it correctly (b) Test image contains without oil pad and algorithm predicted it correctly (c) Test image contains without oil pad and algorithm detected as oil pad

Figure 9 shows the test result of the Faster R-CNN model of non-oil pad images on the 500th epoch. We obtained the highest metrics for 500 epochs which is shown in table 1. The model predicted correctly in almost every case except in figure 9c. In these cases, the model gives false positives at the 500th epoch because the area inside the red box identified by the model contains ground features almost similar to an oil pad.

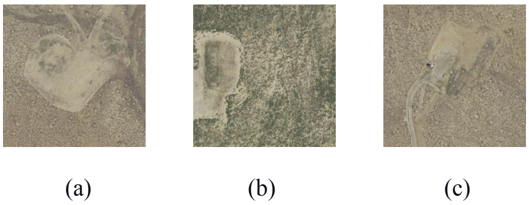


Figure 10. A sample test result of abandoned oil pad images on (a) Test image contains abandoned oil pad and algorithm predicted it correctly (b) Test image contains abandoned oil pad and algorithm predicted it correctly (c) Test image contains abandoned oil pad and algorithm predicted it correctly

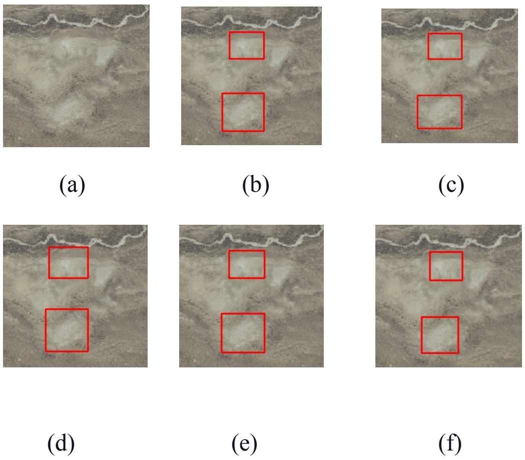


Figure 11. A sample test result of non-oil pad images on different epoch (a) Test image (b) Test result at 200th epoch (c) Test result at 400th epoch (d) Test result at 500th epoch (e) Test result at 600th epoch (f) Test result at 1000th epoch

Figure 10 shows the sample test result of abandoned oil pad images on the 500th epoch. In every case, the Faster R-CNN predicted correctly. Figure 11 shows sample test results of non-oil pad images in different epochs. The test image is shown in Figure 11a representing a non-oil pad image in testing data. The red box is identified by the model during testing. In this particular image, the model gives false-positive results throughout every epoch. Hence, the algorithm fails in this condition, where the ground texture of some areas is the same as an oil pad. When analyzing the area inside the identified box (red box), it is observed that texture features within the detected region match with the oil pad regions given during training.

4. CONCLUSION AND FUTURE WORK

Our results indicate that the algorithm must be trained for 500 epochs to achieve highest detection accuracy. Increasing the number of epochs beyond 500 decreased the detection accuracy. Next, we plan to incorporate aerial images acquired in 2015 and 2019 and simultaneously detect multiple oil pads.

ACKNOWLEDGEMENTS

Authors thank UW Wyoming GIS Center for providing the high spatial resolution images used in this study.

REFERENCES

Ayyad, A., 2020: Real-time system identification using deep learning for linear processes with application to unmanned aerial vehicles, *IEEE Access 8*, 122539-122553.

Baumeister, C., Lutz, K., 2016: Lower oil prices and the US economy: Is this time different, *Brookings Papers on Economic Activity 2016.2,* 287-357. doi.org/10.1353/eca.2016.0029.

Xu, C., Quan, W., Song, Y., Wang, Y., 2020: An Optimal Faster-RCNN Algorithm for Intelligent Battlefield Target Recognition, *IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), China,* 330-335, doi.org/10.1109/ICAICA50127.2020.9181857.

De Myttenaere, A., Golden, B., Le Grand, B., 2015: Using the mean absolute percentage error for regression models, *Proceedings. Presses universitaires de Louvain*.

Dutta, A., Gupta, A., Zissermann, A., 2019: The VIA annotation software for images, audio and video, *Proceedings of the 27th ACM International Conference on Multimedia*, doi.org/10.1145/3343031.3350535.

Girshick, R., Donahue, J., 2014: Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE conference on computer vision and pattern recognition*, doi.org/10.1109/cvpr.2014.81

Girshick, R., 2015: Fast r-cnn, *Proceedings of the IEEE international conference on computer vision (ICCV)*, doi.org/10.1109/iccv.2015.169

He, K., Zhang, X., Ren, S., Sun, J., 2016: Deep residual learning for image recognition, *In Proceedings of the IEEE conference on computer vision and pattern recognitio*n, 770-778, doi.og/10.1109/cvpr.2016.90

Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., Qu, R., 2019: A survey of deep learning-based object detection, *IEEE Access 7,* 128837-128868, doi.org/10.1109/access.2019.2939201

Kumar, D. K., Parameswaran, L., Thangavel, S. K., 2018: A Computer Vision Based Approach for Object Recognition in Smart Buildings, *Proceedings of 2nd International Conference On Computational Vision and Bio Inspired Computing*. doi.org/10.1002/9781119439004.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C., 2016: Ssd: Single shot multibox detector, *European conference on computer vision. Springer, Cham,* doi.org/10.1007/978-3-319-46448-0_2.

Ho, M. J., Lin, Y. C., Hsu, H. C., Sun, T. Y., 2019: An Efficient Recognition Method for Watermelon using Faster R-CNN with Post-Processing, *8th International Conference on Innovation, Communication and Engineering (ICICE), Zhengzhou, Henan Province, China,* 86-89, doi.org/10.1109/ICICE49024.2019.9117374.

Mittal, P., Sharma, A., Singh, R., 2020: Deep learning-based object detection in low-altitude UAV datasets: A survey, *Image and Vision Computing*, doi.org/10.1016/j.imavis.2020.104046.

Moser, B., Hochreiter, B., Herbst, R., Schmid, J. A., 2017: Fluorescence colocalization microscopy analysis can be improved by combining object‐recognition with pixel‐intensity‐correlation. *Biotechnology journal, 12(1)* doi.org/10.1002/biot.201600332.

Padilla, R., Netto, S. L., da Silva, E. A., 2020: A survey on performance metrics for object-detection algorithms, *In 2020 International Conference on Systems, Signals and Image Processing (IWSSIP),* 237-242, doi.org/10.1109/IWSSIP48289.2020.9145130.

Qian, Q., Jin, R., Yi, J., Zhang, L., Zhu, S., 2015: Efficient distance metric learning by adaptive sampling and mini-batch stochastic gradient descent (SGD), *Machine Learning 99.3*, 353-372, doi.org/10.1007/s10994-014-5456-x.

Redmon, J., Farhadi, A., 217: YOLO9000: better, faster, stronger, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263-7271, doi.org/10.1109/cvpr.2017.690

Ren, S., He, K., Girshick, R., Sun, J., 2017: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *In IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137-1149, doi.org/10.1109/TPAMI.2016.2577031.

Saxena, P., 2015: Geo-Location based augmented reality application, *IJRET: International Journal of Research in Engineering and Technology*, 4(7), 495-498.

Ajay, A., Sowmya, V., Soman, K. P., 2018: Vehicle detection in Aerial imagery using Eigen features, *IEEE International Conference on Communication and Signal Processing-ICCSP 17, IEEE, Adhiparasakthi Engineering College, Melmaruvathur*, 1620-1624, doi.org/10.1109/iccsp.2017.8286664.

Xia, G.S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M. and Zhang, L., 2018: DOTA: A large-scale dataset for object detection in aerial images, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3974-3983, doi.org/10.1109/cvpr.2018.00418

Ling, Z., Zhang, D., Qiu, R. C., Jin, Z., Zhang, Y., He, X., Liu, H., 2019: An accurate and real-time method of self-blast glass insulator location based on faster R-CNN and U-net with aerial images, *In CSEE Journal of Power and Energy Systems*, 5(4), 474-482, doi.org/10.17775/CSEE JPES.2019.00460.