Deep Learning-Based Ant Detection from Images

RAJAT
Murdoch University
33778025@student.murdoch.edu.au

***Abstract*— Invasive ants have been a constant biosecurity threat to Australia. They spread rapidly, threatening the environment, agriculture, and way of life. They eat the plants, sting humans and not only damage machinery and buildings including electrical insulations, but also threaten native plants and insects. To protect the country from invasive ants, biosecurity agents often visually screen incoming vessels, planes and even people. This manual process, which is often done by well-trained experts, is very time consuming and subject to human errors. Automating this process will help stop the spread of invasive ants and increase the efficiency of biosecurity screening. This paper only studies the ant detection from images and is primarily focuses on the detection part. It implements and evaluate the baseline MASK R-CNN algorithm and then fine-tune it to improve its performance. This study uses the limited dataset and only studied the MASK R-CNN for the ResNet101 backbone network. For future work, ants invasive and non-invasive classification can be done using this model with different backbone networks, including a large ant dataset.**

***Index Terms*— Machine Learning, Deep Learning, Ant Detection, MaskRCNN, Neural Networks**

I. INTRODUCTION

There is a number of invasive ant species that are amongst the most serious global invasive pests. Australia's environmental, economic, and social wellbeing is also threatened by these ants. Invasive ants spread rapidly and are a threat to the environment, agriculture, and way of life. Economically, invasive ants impact the primary production through seed consumption or animal attack and biting or stinging farmworkers. They are also impacting electrical infrastructure in buildings and homes. They have the potential to negatively impact the Australian environment, infrastructure and human health and amenity. For instance, the Bulldog ants, one of the world's most dangerous ants, are found in coastal regions of Australia. People generally find ants damaging their plants, gardens and they do not have any effective way to determine their species to use the specific bait to stop them. Additionally, increasing global trade has led to the unintended transport of ants across the world. Airports and ports have been one of the primary sources of ant invasion through luggage and shipment containers. Many ant species have invaded other parts of the world but are yet to reach Australia, and we have no automation method which can be used in the detection and classification of ants to counter the ant invasion problem.

As normal humans cannot differentiate between invasive and native ants, there is a growing and urgent need to automate the detection and recognition process. The purpose of this study is to develop an ant detection method using deep learning and then investigate whether we can automatically classify the detected ants into invasive or non-invasive. The developed method can be used at different places such as airports, ports, and crop fields to protect crops from invasive ant infections. This paper primarily focuses on the detection part only.

Deep learning has become quite popular in recent years because of its application and performance in various fields. It is a branch of machine learning that uses convolution neural networks (CNN) for object detection and recognition, including feature extraction, region proposals, and classification. It has been successfully used for object detection and classification in various fields. However, there is no specific literature or study on ant detection and classification using deep learning. Therefore, this study will focus on that topic and try to solve invasive ant issues by detection and classification of ants.

There have been many neural network methods for a variety of purposes. Convolutional neural network (CNN) is a class of deep learning methods that have become dominant in various computer vision tasks and can be used across multiple domains, including audio and speech classification. CNN is composed of various building blocks, such as convolution layers, pooling layers, and fully connected layers where the network is designed to learn spatial hierarchies of features automatically and adaptively through a backpropagation algorithm [1]. For object detection application, CNN [2, 3] is a deep learning algorithm that takes an input image, assigns importance to various objects in the image, and differentiates one from the other. Convolution layers in CNN are used to extract features such as colour, size, edges, corners in the images [4, 5]. Different networks were introduced to overcome the drawbacks of the predecessors and provide better results, accuracy, speed, and performance [6]. Examples of such networks include R-CNN [7], FAST R-CNN [8, 9], R-FCN [10], FASTER R-CNN [11], and Mask R-CNN [12, 13]. This project will use Mask R-CNN because it localises objects and segments them properly from their background. Such segmentation is crucial for classification of ants. This paper will also discuss the model's architecture and its application to ant detection and how the challenges faced in small object detection can be overcome.

II. Literature review

Ant detection and classification has not been studied enough earlier on small objects and was not explicitly implemented. However, there are some prior studies related to ants and deep learning-based object detection. This part of the paper reviews those previous studies.

There are few studies in relation to ant detection, which involves using some deep neural networks. In [14], the challenge of tracking the movement of insects in a social group has been addressed by developing an online multi-object tracking framework. The ResNet model is used to obtain ant appearance features with just 15 layers. The study provided an open dataset of detected and tracked ants and capture moving ants in video sequences and prepare a dataset of detected and tracked ants as the benchmark. The network was trained for just 50 ant images which might not be enough. Still, it has been stated that it is advantageous for the study because it eliminates the demand of manually constructing a large dataset. Furthermore, the paper experimented with two ways, namely ant video capture in the indoor lab setup and the other is in the outdoor environment. However, due to the use of a low frame-rate camera, some motion blurring of ant bodies and overexposure in the videos have been noticed. Due to this, the framework failed to associate the labelled bounding box with the trajectory of some fast-moving ants because the framework of the video was not enough and increased tracking difficulties. The study in [15] is about AntVis, a web-based visual analytics tool built on FCN (Fully convolution network) to explore ant movement data. The data is collected from the seven sequences of the video recording of ants moving on tree branches where some of the video clips data set have been removed and reduced to 5 video clips. Also, some of the significant miss-segmented components from the background have been eliminated, which could also mean that the tracking system might not perform well in challenging backgrounds or might be only limited to a given dataset. However, the paper successfully enabled their user to gain insights about an overview of the movement data, detailed explore ant attributes, identify common patterns, and detect abnormal movements using five coordinated views. They are namely the movement, similarity, timeline, statistical, and attribute. Both papers focused on tracking ant movement and exploring the challenges of differentiating each ant individuals but with different neural networks. However, both studies believed that the future extended studies on more dataset and considering more video sequences will enable findings of more interesting features and ant movements.

Some papers also implemented deep neural networks in medical applications. Mask R-CNN has been used in detecting different objects, and some improvements were made in the model to improve detection and classification [16, 17]. It has a good scope in medical applications because of its mask segmentation branch, which can detect micro size objects and segment them. Paper [16] demonstrated the use of Mask R-CNN to perform highly effective and efficient automatic segmentation of the wide range of microscopy images of the cell nucleus. The study uses the Mask R-CNN model-based COCO (Common Objects in Context) pre-trained weights as the baseline for training the model. It is called transfer learning, where a model is not created from scratch. Instead, build upon pre-trained weights and uses the parameters for training. The dataset was augmented using random crops, random rotations, gaussian blurring, and random horizontal and vertical flips to deal with the overfitting issue. It is an excellent way to deal with the problem. Mask R-CNN has also been used in paper [17] for oral disease detection. The dataset for the study is collected from open source and might not be legitimate. Likewise, the paper has also used two networks, ResNet50 and ResNet 101, in their studies and compared the results of both to see how they performed on medical images. However, in some cases for ResNet50, the fingers and teeth had also been masked. Whereas ResNet101 had rare difficulties in segmenting the sores from the images. Moreover, both papers acknowledged the implementation of Mask-RCNN and improved the model to get satisfactory results and explore the model's efficacy and performance for the range of such tasks by using different backbone networks.

The Mask R-CNN model has been studied in some more papers for the detection, classification, and segmentation tasks in the automobile field. These studies explored Improved Mask R-CNN. Paper [18] proposed an improved Mask R-CNN model based on the MS COCO dataset as the baseline to build vehicle damage detection and segmentation algorithm. In contrast, in paper [19], the same model has been used for an anti-collision warning system in intelligent driving. The vehicle damage detection model enables the user to upload photos for assessment, and insurance companies can use the model to process claims quickly. It also compared the results of the Mask R-CNN model with the improved Mask-RCNN model and recorded the improved accuracy by 2.15%, the mask accuracy by 1.89%. On the other side, the anti-collision warning system compared the different deep learning model based on vehicle detection with the new network algorithm designed in the paper. They filtered and supplemented the dataset to increase the accuracy of the training effect of the network. For real-time requirements of smart driving, designed the ResNet86 network to use it as the backbone. All modifications to the network resulted in improved detection speed with a detection accuracy value of mAP (mean average precision) by 17.53 points over the original Mask R-CNN model. Both papers managed to increase the accuracy by using the improved Mask R-CNN. Also, improving and designing the feature extraction algorithm makes the semantic feature information more abundant for different weather conditions. Finally, these papers presented some subtle ways to improve the detection, classification and segmentation process for the Mask R-CNN model and could be used for an extension to other studies.

The method used for the ant detection in this paper is Mask R-CNN, an extension to Faster R-CNN by adding a branch for predicting segmentation masks on each Region of Interest in parallel with the existing branch for classification and bounding box regression [15]. Mask R-CNN relies on region proposals generated via a region proposal network. It follows the Faster R-CNN model by having a feature extractor followed by this region proposal network. Then there is ROI align operation, which allows a very accurate instance segmentation mask to be constructed, and the algorithm adds a network head to produce the desired instance segmentation. The mask and class predictions are decoupled; the mask network head predicts the

Mask independently from the network head predicting the class [12, 16].

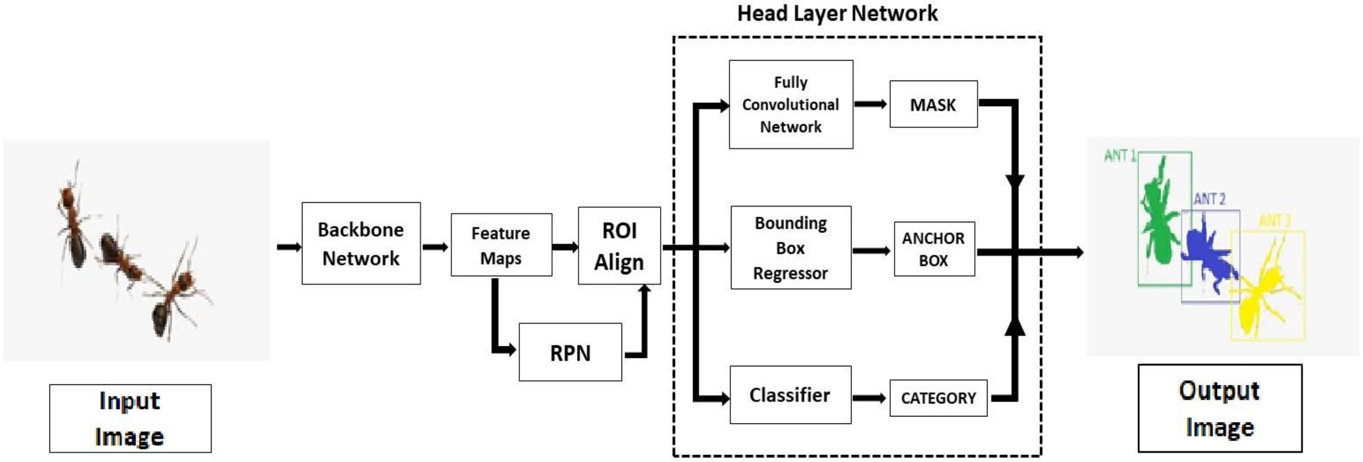


Figure 1: Mask-RCNN framework demonstration for ant detection.

III. Research Question

After studying the previous research related to deep learning-based detection for small objects and considering this work, this paper will try to answer the following questions:

- How does MRCNN model perform for ant detection ?
- What improvements are needed in Mask R-CNN to have a satisfactory result with good accuracy and performance in ant detection?

IV. Methodology

To implement Mask R-CNN for ant detection, we use the transfer learning method of machine learning. Mask-RCNN is already trained with the MS COCO dataset, which has a total of 2.5 million labelled instances in 328,000 images for 91 object categories [20]. That gives us pre-trained weights on which we can create a new model for a custom dataset. This training method has been used before and performs very well with neglecting the burden of making the model from scratch. The base Mask-RCNN repository [21] contains the essential model files for training and testing the model. The code [22] implemented in this study have been used on horse and human detection and showed satisfactory results. We only trained the head network because the pre-trained weights have the other networks well trained, and those parameters set the base for the training of the new model. To better detect the ant images, we created a new category for ants. Therefore, we have two classes, namely background and ant

In this section, we first describe the network architecture (Section IV.A). Then we describe the training process that has been done for model training with the dataset information (Section IV.B). After that, results have been shared for the evaluation and discussion (Section IV.C).

A. Network architecture

The Mask R-CNN framework is divided into two stages. The first stage scans the images and generates proposal regions, i.e., regions that are likely to contain the target objects. The second classifies the proposal regions and generates the bounding boxes and the masks [12]. Figure 1 illustrates this structure.

Backbone network structure: Generally, it is the backbone of the Mask R-CNN based on convolutional neural network architecture; adopts Feature Pyramid Network (FPN) and a residual network (ResNet101) network where ResNet has 101 layers. The large number of layers will significantly reduce the rate of the network structure [18, 21, 23]. Both FPN and ResNet work together on the input image and extract different sized feature maps from the image for objects. The deep residual network (ResNet) overcomes the problem that the learning efficiency becomes lower due to the network's dependence and the inability to improve training effectively [19]. The FPN takes advantage of the inherently hierarchical and multi-scale nature of convolutional neural networks to derive useful features for object detection, semantic and instance segmentation at many different scales [16]. For our study, we used ResNet101 network as backbone because it has much number of layers which can extracts more features from the training dataset.

1) Region Proposal Network (RPN): The feature map generated from the backbone network are processed by the RPN, where the Region of Interest (ROI) features are extracted from the different sized feature maps according to the size of the target object [19]. RPN can divide the feature layer into n x n regions and obtain the feature regions of various scales and aspect ratios [8]. It can predict both boundary position and object scores at each location. Moreover, simple network structure changes without increasing the calculation amount improve the detection performance of small objects and achieve excellent accuracy and speed [19].

2) ROI Align: The additional part in this model, i.e. mask branch, must determine whether a given pixel is part of the target, and the accuracy must be at the pixel level [18]. Since the input image went through various stages, such as ROI pooling, it first quantises a floating number RoI to the feature map's discrete granularity and changes the image's size. The quantisation leads to misalignments between the RoI and the extracted features resulting in the negatively impacting prediction of pixel-accurate masks [12]. To overcome the issue, the RoI Align layer removes the harsh quantisation of RoIPool and adequately aligns the extracted features with the input, keeping preserve the spatial information feature maps. Therefore, the regional mismatch problem gets solved, and pixel-level detection segmentation can be achieved [18].

3) Bounding box regression and classification: This layer is included in the network head layer in which the region of interest from the RoI Align layer assigned the bounding boxes for the objects. A classifier is doing their classification parallel to mask segmentation of the region of interest [18]. The result provides the input image having bounding boxes and masks on the objects. We only trained the head layer instead of training whole model because we used transfer learning where the model is already trained well and we uses that pretrained model knowledge.

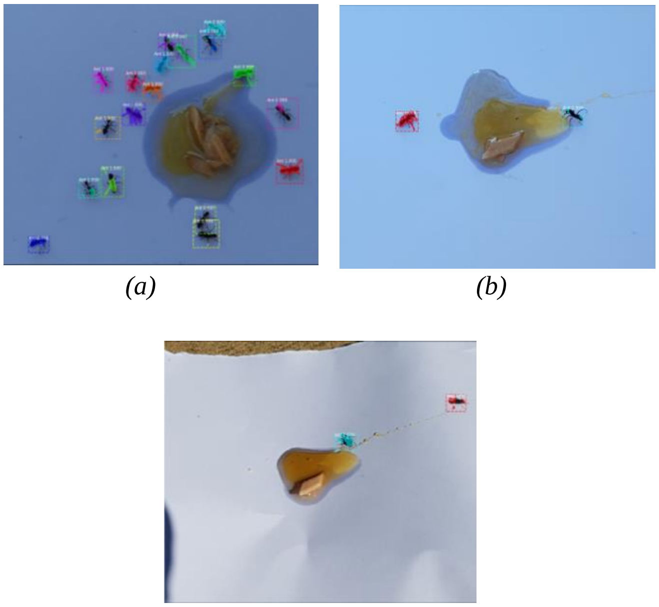
4) Fully Convolutional Network: This network is included in network head layer and is used for mask segmentation in the images. The most significant change introduced by Mask R-CNN is the classification of images by using Masks through Instance Segmentation and masks used for processing at the pixel level comparison, which allows the model to generate the results with very high precision [13].

B. *Training and Dataset:*

The Mask R-CNN repository was pre-trained on the MS COCO dataset [21], and with the help of transfer learning, the same model trained for ant detection. Model training and testing process are done with the help of Google Collaboratory because of the availability of GPU, which is required for image processing. Moreover, to avoid any error in running the code, Keras version 2.2.5 and TensorFlow version 1.x have been used. These versions have been supported earlier as well for error-free execution of codes. They are open-source machine learning libraries that are required for deep learning modelling.

The ant dataset for training and testing has been collected from university campus (Murdoch University). It consists of 26 ant images with two short videos sequences of 13 seconds and 18 seconds. Moreover, 26 ant images are randomly distributed into the training dataset with 17 images and a validation dataset with 9 images. Training dataset is used for model training, whereas validation dataset is used for only validation and evaluation. For the video frame dataset, 14 screenshots from video sequences are taken and converted into jpeg format. The annotation of the dataset needed to be done manually. Therefore, VGG Image Annotator Tool is used. The reason for choosing this tool is because it is easy to use and have been used before in various studies and proved effective. This tool exports the training and validation datasets in the JSON file format used in the training and validation of the model. For initial training and results, the model has been trained on 10 epochs, where epoch means the number of times the learning algorithm works through the entire training dataset. This value has been picked generally; however, we again trained the model for 5 epochs and used that for study. The trained weights obtained from training is then used for testing, and the results have been obtained.

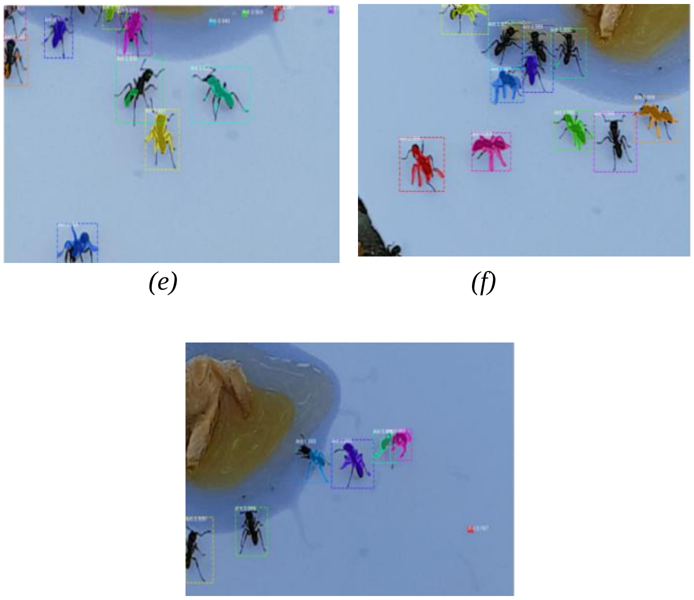
C. *Results and evaluation:*



(a) *(b)*

(c)

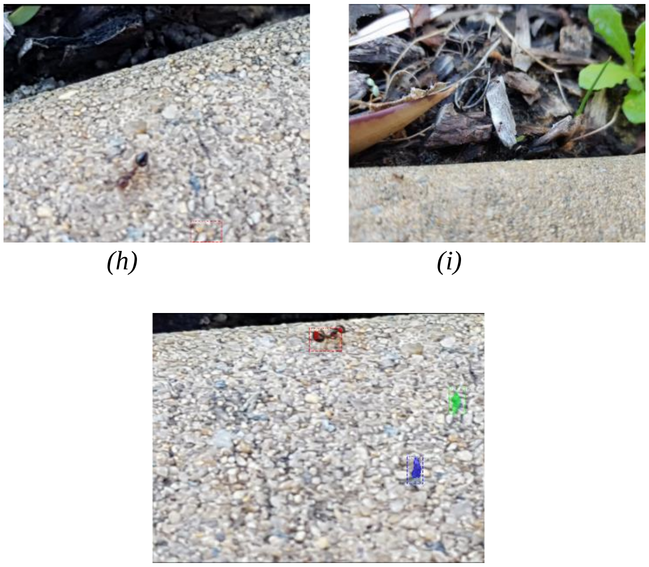
Figure 2.1 (a), (b), (c): Examples of detection results on validation images.



(e) *(f)*

(g)

Figure 2.2. Examples of ant detection results on validation images. None of the text images has been seen by the network during the training phase.



(h) (i)

(j)

Figure 2.3 (h), (i), (j): Examples of detection results for complex background in which either the model failed to detect the ants or predicted background as ants.

The preliminary results for ant detection are shown above. The results observed from the initial training are based on 10 epochs. The evaluation is done for intersection over union (IOU) values equals to 0.5 which is the percentage for prediction over ground truth. However, the mean average precision (mAP) for detection is not good enough, and the model misses some mask. In complex background situation, either the model failed to detect the ant in the images or predicted background as the ants. The evaluation of the model is done by determining the precision, recall and F-score values.

Table 1
Evaluation metrics for 10 epochs with IOU = 0.5

Evaluation Metrics	Training Dataset	Validation Dataset
mAP	0.281	0.216
mAR	0.881	0.772
F1 Score	0.426	0.337

The model mAP, which is the performance accuracy of the model obtained for validation is 0.216, and the mAR, which is the accuracy for all positives obtained for validation is 0.772. We also trained the model for 5 epoch value and used it again to predict the training and validation dataset to see how the model performs then. The results for 5 epochs are much better than the 10 epochs result, and there is slight improvement in the detection.

Table 2
Evaluation metrics for 5 epochs with IOU = 0.5

Evaluation Metrics	Training Dataset	Validation Dataset
mAP	0.338	0.318
mAR	0.959	0.794
F1 Score	0.500	0.454

The mAP score for 5 epochs slightly improved which is 0.318 and the mAR achieved is 0.794. Moreover, the F1-score values for the validation dataset is 0.454. We have the dataset images with two types of backgrounds i.e. one with normal white background and one with complex background.

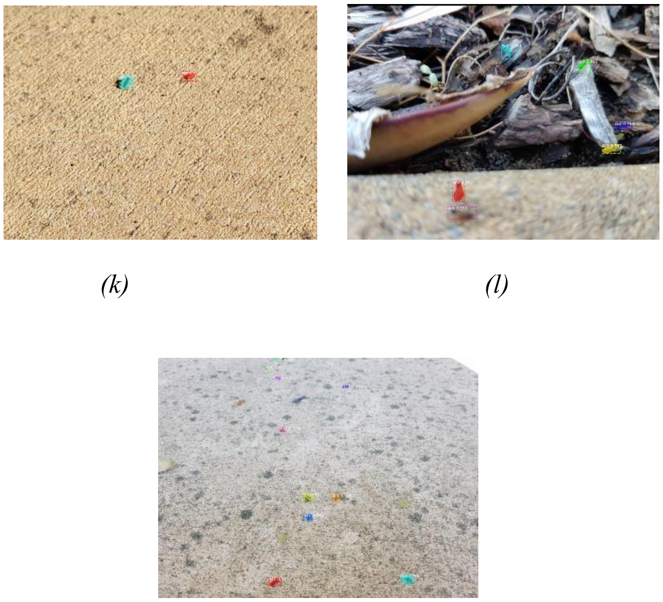
Table 3
Model evaluation of validation dataset images with normal and complex background

Image ID	Background Type	Average Precision (AP)	Average Recall (AR)
0	Normal	0.25	1.0
1	Normal	0.5	1.0
2	Normal	0.666	1.0
3	Normal	0.5	1.0
4	Normal	0.332	0.944
5	Complex	0.100	0.2
6	Complex	0.0	0.0
7	Complex	0.0	1.0
8	Normal	0.0952	1.0

From table 3, we see that model perform well for normal background with no complexity at the back. But for the challenging background, the model fails to predict the ants in the image or find it hard to predict the ants. Also, the model can predict the bounding box over the ants quite well for the normal background but cannot mask them properly and faces problem in masking each ant in the model. However, for complex images in the training dataset, the model performs quite well and can detect ants with challenging background. But it also overpredicts and gives some false positive predictions as well. A more critical analysis and discussion on the results is being done in the discussion part.

V. Discussion

The model can predict ants quite well, considering the small training data being served to the model. It can predict ants quite well for the normal background. We also had some blur images for ants, and the model could predict those blur ants as well in some cases. Furthermore, we tried to run the model on the training dataset and study the results. The model can detect ants and classify ants with bounding boxes effectively but cannot mask the ants properly or miss the mask. This incomplete mask has been the issue for the validation dataset as well. The issue could be less training and could be resolved by increasing training time by serving the model with more dataset having good annotation labelling. Besides that, it performs quite well for the challenging background in the training dataset and predicts the ants. Some of the training dataset results for the complex background are shown below.

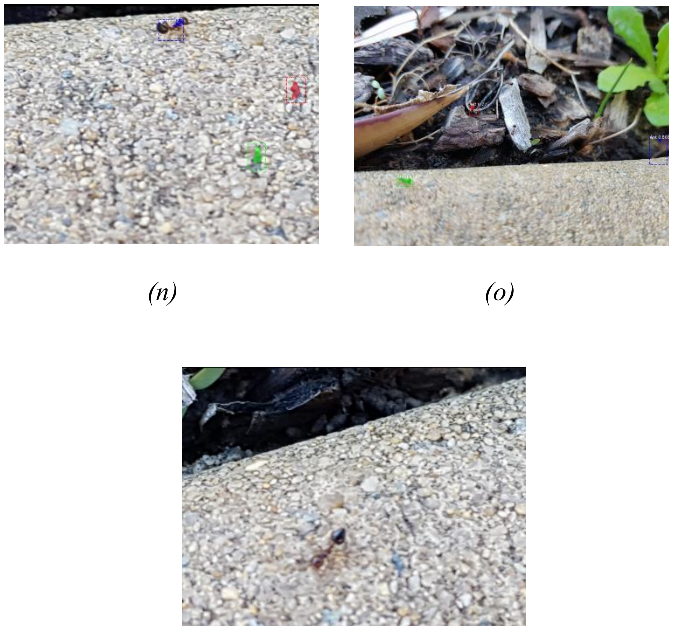


(k) *(l)*

(m)

Figure 3.1 (k), (l), (m): Examples of detection results on training dataset with the complex background where the model can predict ants quite well.

The detection result of the training dataset with the complex background is good. But this is not the case with the validation dataset. The model either overpredicts the ants or fails to detect ants in the complex background for validation images. It is because the model doesn’t have enough parameters knowledge about the ants for differentiating them in situations when there is complex background. We can fix this issue by serving those images to the model training again where the model failed to learn and differentiate between ants and background, resulting in good detection for complex background. Some of the results where the model failed or performed badly are shown below.

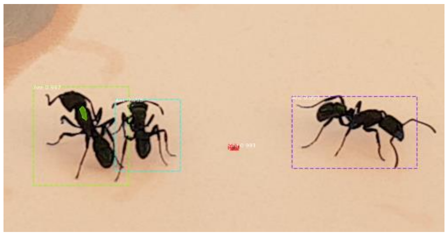


(n) *(o)*

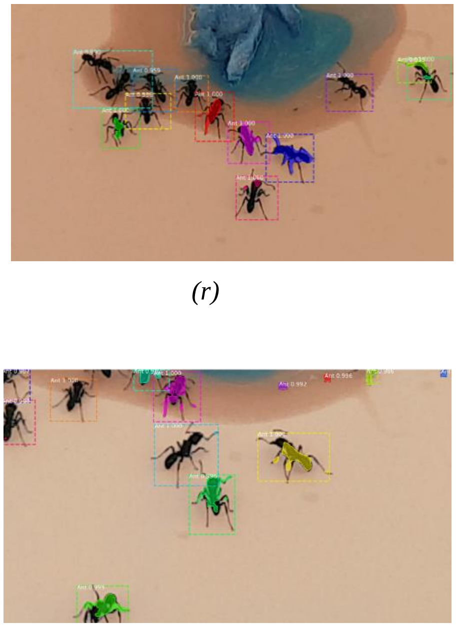
(p)

Figure 3.2 (n), (o), (p): Examples of detection results on validation dataset with the complex background where either model struggled to predict ants or overpredicted the ants.

To understand model predictions, we also used the model on several video frame images. The model can predict the ants with bounding box over them very well. It can also predict the ants which are not properly present in the image frame. But the mask segmentation over the ants is poor and misses for some ants as well. There were some cases where model wrongly predicted the objects as ants. These results show that model needs more training on ant images so that the model can perform well for mask segmentation as well. The fully convolutional network proposes the mask over images. We might need to need to look at that separately to improve the mask segmentation. The prediction results are shown below for video frame images.



(q)



(r)

(s)

Figure 3.2 (q), (r), (s): Prediction results for the video sequence frames taken as the screenshots from the video dataset.

Overall, on the basis of results from the training and validation datasets, we could say that if the model is trained with the more dataset including more challenging dataset, it can perform well as it did for the complex training images. However, the improper mask or missing mask could partially be addressed by training the model with more dataset. But it might be needed to study the fully convolutional network layer which produces the mask over images about how good this layer is getting trained from the training dataset and determining the loss values as well.

VI. LIMITATIONS

The model is trained on a minimal dataset which could be increased, including the different ant categories. The model doesn't perform well for challenging image where it finds hard to differentiate between background and ant. We used the ResNet101 network as the backbone to extract more feature from the dataset, but it might be increasing the loss for the small dataset. We didn't try other backbone networks for the study. Besides, we only trained the network head because the pre-trained weights are used. So, we didn't train all the model layers. Furthermore, we didn't find the optimal value of epochs where the model should have the minimum loss so that the trained weights for the optimal epochs could have been used for better detection. Also, the training doesn't show the accuracy for each epoch. We only tried running the model on some of the video frame images and haven't used the model on actual videos to see how the model performs in following ants movement as if it keeps tracking them or not.

VII. CONCLUSION

This paper demonstrates the Mask R-CNN model implementation for ant detection and achieved the preliminary results. This study's main research content is about how to make the Mask R-CNN algorithm detect ants. The model based on the MS COCO dataset is used as the baseline, and the transfer learning feature is used for ant detection as the standard for pre-trained weights. The model performs well for the training images served for training but fails to perform in complex validation images. That's because the model doesn't have enough parameters and is not trained on enough complex images.

To improve the detection, we could determine the optimal epochs number where the model has the minimum loss. Furthermore, we could try different backbone networks to see which network could extract more features with less noise and loss. We can also serve the same dataset to the model with different angles so that the model learns more parameters for ants, including the complex images where the model performs poorly or fails to perform. Also, the model training could be increased until there is very little mask loss and good predictions for ants in every situation. At last, the M-RCNN model can detect ants, and improvements can be made to increase detection and accuracy. However, a more extensive study could be done to explore the gaps this study hasn't considered.

VIII. FUTURE WORK

For future work, different backbone network could be used for the model, and a comparison could be made between them to determine which backbone is most suitable for the model. Moreover, invasive and non-invasive ant labelled datasets can be used to train the model and classify the ants using this research as the baseline for ant classification. Additionally, the model's ant tracking capabilities could also be studied for the video dataset. Instead of training the network head, all layers could be trained with the dataset to evaluate the model performance.

ACKNOWLEDGEMENT

The python notebook code implemented in the study has been created by Dhruvil Shah [22], and some suggestion has been taken from him for executing the program and dataset annotation. The dataset has been provided by the project supervisor Dr Hamid Laga, and all the progress have been made under his supervision.

REFERENCES

[1] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging,* vol. 9, no. 4, pp. 611-629, 2018, doi: 10.1007/s13244-018-0639-9.

[2] M. R. Minar and J. Naher, "Recent advances in deep learning: An overview," *arXiv preprint arXiv:1807.08169,* 2018.

[3] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object Detection with Deep Learning: A Review," *arXiv pre-print server,* 2019-04-16 2019, doi: None arxiv:1807.05511.

[4] T. Liu, S. Fang, Y. Zhao, P. Wang, and J. Zhang, "Implementation of Training Convolutional Neural Networks," *arXiv pre-print server,* 2015-06-04 2015, doi: None arxiv:1506.01195.

[5] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," 2017: IEEE, doi: 10.1109/icengtechnol.2017.8308186. [Online]. Available: https://dx.doi.org/10.1109/icengtechnol.2017.8308186 https://ieeexplore.ieee.org/document/8308186/

[6] K. Tong, Y. Wu, and F. Zhou, "Recent advances in small object detection based on deep learning: A review," *Image and Vision Computing,* vol. 97, p. 103910, 2020, doi: 10.1016/j.imavis.2020.103910.

[7] D. M. Montserrat, Q. Lin, J. Allebach, and E. Delp, "Training Object Detection And Recognition CNN Models Using Data Augmentation," *Electronic Imaging,* vol. 2017, no. 10, pp. 27-36, 2017, doi: 10.2352/issn.2470-1173.2017.10.imawm-163.

[8] C. Tang, Y. Feng, X. Yang, C. Zheng, and Y. Zhou, "The Object Detection Based on Deep Learning," 2017: IEEE, doi: 10.1109/icisce.2017.156. [Online]. Available: https://dx.doi.org/10.1109/icisce.2017.156 https://ieeexplore.ieee.org/document/8110383/

[9] R. Girshick, "Fast R-CNN," *arXiv pre-print server,* 2015-09-27 2015, doi: None arxiv:1504.08083.

[10] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," *arXiv pre-print server,* 2016-06-21 2016, doi: None arxiv:1605.06409.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *arXiv pre-print server,* 2016-01-06 2016, doi: None arxiv:1506.01497.

[12] K. He, G. Gkioxari, P. Doll\'ar, and R. Girshick, "Mask R-CNN," *arXiv pre-print server,* 2018-01-24 2018, doi: None arxiv:1703.06870.

[13] A. S. Paste and S. Chickerur, "Analysis of Instance Segmentation using Mask-RCNN," 2019: IEEE, doi: 10.1109/icicict46008.2019.8993224. [Online]. Available: https://dx.doi.org/10.1109/icicict46008.2019.8993224

[14] X. Cao, S. Guo, J. Lin, W. Zhang, and M. Liao, "Online tracking of ants based on deep association metrics: method, dataset and evaluation," *Pattern Recognition,* vol. 103, p. 107233, 2020, doi: 10.1016/j.patcog.2020.107233.

[15] T. Hu *et al.*, "AntVis: A web-based visual analytics tool for exploring ant movement data," *Visual Informatics*, vol. 4, no. 1, pp. 58-70, 2020, doi: 10.1016/j.visinf.2020.02.001.

[16] J. W. Johnson, "Adapting mask-rcnn for automatic nucleus segmentation," *arXiv preprint arXiv:1805.00500,* 2018.

[17] R. Anantharaman, M. Velazquez, and Y. Lee, "Utilising Mask R-CNN for Detection and Segmentation of Oral Diseases," 2018: IEEE, doi: 10.1109/bibm.2018.8621112. [Online]. Available: https://dx.doi.org/10.1109/bibm.2018.8621112 https://ieeexplore.ieee.org/document/8621112/

[18] Q. Zhang, X. Chang, and S. B. Bian, "Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN," *IEEE Access*, vol. 8, pp. 6997-7004, 2020, doi: 10.1109/access.2020.2964055.

[19] C. Xu *et al.*, "Fast Vehicle and Pedestrian Detection Using Improved Mask R-CNN," *Mathematical Problems in Engineering,* vol. 2020, p. 5761414, 2020/05/31 2020, doi: 10.1155/2020/5761414.

[20] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," *arXiv pre-print server,* 2015-02-21 2015, doi: None arxiv:1405.0312.

[21] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," 2017.

[22] D. Shah. "Mask RCNN implementation on a custom dataset!" [Online]. Available: https://towardsdatascience.com/mask-rcnn-implementation-on-a-custom-dataset-fd9a878123d4. [Accessed: May 10, 2021].

[23] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117-2125.