Mask R-CNN Deep Learning-based Approach to Detect Construction Machinery on Jobsites

H. Raoofi[a] and A. Motamedi[b]

[a]Department of Construction, École de Technologie Supérieure, Canada
[b]Department of Construction, École de Technologie Supérieure, Canada
E-mail: Ali.Motamedi@etsmtl.ca

Abstract –

In the construction industry, there is often a need to identify and localize assets and activities on the jobsite to assess and improve the performance of their associated processes. Traditional methods for monitoring construction activities are costly and time-consuming. Excavators and dump trucks are among the most common assets used in the construction industry. Consequently, accurately monitoring their activities can reduce time and increase the efficiency of progress monitoring.

With the presence of cameras on jobsites and the advancement of methods based on artificial intelligence and computer vision, progress monitoring activities can be automated. Furthermore, by using techniques such as deep learning, a wider range of data resources can be processed, and oftentimes more accurate results can be produced for the purpose of object detection.

This research proposes a computer-vision approach that utilizes a Mask Region Based Convolutional Neural Network (Mask-RCNN) to detect excavators and dump trucks in a construction site. This research investigates an innovative technique to achieve high accuracy object detection using relatively small datasets. To overcome the problem of overfitting and improve generalization, a pre-trained model based on a Microsoft COCO dataset is used as a network that presumably has already been trained to distinguish basic features. Finally, the model is further fine-tuned to minimize validation loss.

Keywords –

Computer Vision; Artificial Intelligence; Deep Learning; Mask R-CNN; Construction Monitoring; Progress Management

1 Introduction

An efficient and effective workforce improves the time and cost performance of construction projects [1, 2]. Accurate progress monitoring, safety management, and quality control activities require skilled labor with adequate supervision, which in turn increases time and project costs.

With the advancement of methods based on artificial-intelligence, computer vision and deep learning, the abovementioned activities could be automated, leading to time and cost reductions. Specifically, there is a growing trend to use computer vision approaches to detect construction machinery from video outputs. These technologies could help project managers to access more accurate data in order to monitor construction assets, facilitate progress management and manage site safety.

To address this need, a region-based deep learning architecture called Mask R-CNN is utilized in this study to detect and segment excavators and dump trucks in the images from jobsites using a relatively small dataset.

The main objective of this research is to develop an improved deep-learning-based network to enhance the accuracy of predictions and decrease the processing time. The study's sub-objectives are to:

1. Develop a network for automatic detection of machinery (i.e. excavators and dump trucks) on construction sites from captured videos based on machine learning algorithms.
2. Train and validate the network on a small dataset.
3. Fine-tune the network's parameters to further increase its performance.

2 Related Works

AI techniques can assist project managers in monitoring and analyzing job site activities [2, 3, 4, 5, 6, 7]. Furthermore, AI approaches can be used for safety management to monitor and reduce risks [8, 9, 10, 11, 12].

Machine learning techniques can be deployed to detect construction machinery on jobsites. Project managers can utilize the information about these assets to increase the efficiency and safety of the projects.

Deploying an Unmanned Aerial Vehicle (UAV), Kim et al. [13] presented a visual monitoring method that

could automatically measure proximities among construction vehicles and workers. They localized objects using a deep neural network, YOLOv3, and developed an image rectification method that facilitates the measurement of actual distances from a 2D image. Struck-by hazards around workers could be detected with this method, making timely intervention possible. YOLOv3 provides bounding boxes for detected objects, but it is unable to generate pixel-level masks.

In another study, Kim et al. [14] developed a vision-based method to classify equipment operations in video data. The framework consists of four stages: equipment tracking, individual action recognition, interaction analysis and post-processing. The hybrid detector used in this study consists of ferns and a random forest classification algorithm, which uses a sliding window to propose bounding boxes and cannot provide shape data.

To detect dense multiple vehicles from UAV, Guo et al. [15] presents a deep learning approach that uses a single-stage detection (SSD) algorithm with orientation-awareness and integrates it with a developed feature fusion module. Similar to YOLOv3, the SSD algorithm is unable to provide pixel-level mask data for the detected objects.

3 Methodology

In this research, two common classes of construction vehicles, namely excavators and dump trucks, are studied. Due to the unavailability of open datasets, a dataset of 341 annotated images of excavators and dump trucks is created to train the proposed network. In order to develop a high-quality dataset of these heavy machineries in construction sites, public domain images were gathered through Google Image® and Flicker®. Some 269 images are used for the network training, while 72 images are assigned to the evaluation process. VGG Image Annotator (VIA) [16] is used to annotate the images of the dataset to provide the ground truth for the training process (Figure 1).

Figure 1. Data annotation using VIA [16]

Since the size of the dataset is relatively small, pre-trained weights for the MS COCO dataset are used as a transfer learning technique to overcome overfitting problem and increase the accuracy.

3.1 Instance Segmentation

In this study, an instance segmentation technique called Mask R-CNN is used, which can provide pixel-level boundaries for each detected object. Mask R-CNN has a new ability to segment objects in addition to classification and detection, compared with its predecessor Faster R-CNN [17].

As illustrated in Figure 2, first, the feature map of the entire image is extracted using a ResNet-101 architecture as a convolutional backbone. Then, a Region Proposal Network (RPN) analyzes the developed feature map and proposes candidates for object bounding boxes. To resolve Faster R-CNN's problem regarding the pixel-to-pixel misalignment between network inputs and outputs, a quantization-free layer called Region of Interest (RoI) Align is utilized, which preserves spatial locations. After fixing the misalignment problem of the bounding-box candidates, using fully connected (FC) layers, the network can classify objects and recognize bounding boxes and in parallel, a convolutional layer unit predicts masks that are applied separately to each RoI [17].

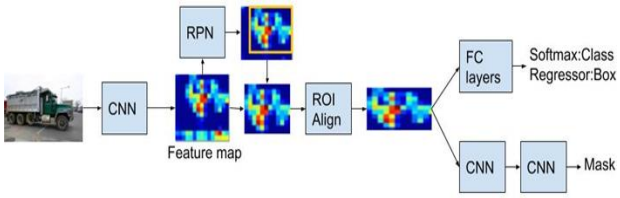

Figure 2. Mask R-CNN architecture

3.1.1 Loss Function

A multi-task loss is defined on each RoI as the sum of the classification loss (L_{cls}), the bounding-box loss (L_{box}), and the mask loss (L_{mask}) [17].

$$L = L_{cls} + L_{box} + L_{mask} \quad (1)$$

3.2 Training Method

As shown in Table 1, the hardware setting is Laptop ROG Strix GL502VS with a NVIDIA GTX 1070 8GB GPU. It has an Intel Core i7 7700HQ as CPU processor and 16GB of RAM. Concerning the limitation of GPU memory, we use 1 image per GPU for each mini-batch and each epoch consists of 100 steps.

Table 1. Hardware setting

ROG Strix GL502VS	
CPU	Intel Core i7 7700HQ

GPU	NVIDIA GTX 1070 8GB
RAM	16GB

Matterport's implementation [18] of Mask R-CNN on Python 3, Keras, and TensorFlow is used. The backbone of our network is ResNet-101 and we utilize only one class of data augmentation in our training phase, which consists of horizontally flipping 50 percent of the images. As described in section 3, pre-trained weights on MS COCO are used as our initial network weights. We adapted a multi-phase training strategy to fine-tune our results. In the first phase (first 5 epochs), only the top layer (heads) of our developed network was trained with the learning rate (lr) of 0.001. Next, for epoch 6 to 15, ResNet stage 4 and up were trained with lr of 0.001, while the rest of the layers are frozen. In the third step, for epoch 16 to 30, ResNet stage 3 and up were trained and lr decreased to 0.0001. In the final phase (epoch 31 to 40), all layers of ResNet were trained with lr of 0.00001. Table 2 presents the specifications of the developed multi-phase training.

Table 2. Multi-phase training specification

Phase	Epochs	Training layers	lr
I	1-5	Only top layer	0.001
II	6-15	ResNet stage 4 and up	0.001
III	16-30	ResNet stage 3 and up	0.0001
IV	31-40	ResNet all layers	0.00001

3.3 Metrics for Evaluating Performance

The network's performance is evaluated and quantized using two metrics: Average precision (AP) and inference time, which is the amount of time the network requires to do the prediction.

3.3.1 Average precision (AP)

According to the definition of Pascal VOC 2010 [19], for a specific value of Intersection over Union (IoU), the AP measures the precision/recall curve at recall values (r_1, r_2, etc.) when the maximum precision value drops. The AP is then computed as the area under the curve by numerical integration [20].

$$AP = \sum (r_{n+1} - r_n)\, p_{interp}(r_{n+1})$$
$$p_{interp}(r_{n+1}) = \max_{\tilde{r} \geq r_{n+1}} p(\tilde{r}) \quad (2)$$

The metric mAP is the average of AP over a range of IoU from 0.5 to 0.95 at intervals of 0.05 (AP@ [.5:.95]) [20].

3.3.2 Detecting Threshold

To eliminate network predictions having a low confidence score, only detected instances above the threshold level of 0.9 are considered in the final results.

4 Results

Over the 40 epochs of training the network, the minimum validation loss function took place at epoch 38 with a value of 0.1889. Figure 3 illustrates the loss function of the network at each epoch. As shown in Table 3, the total training time was 68 minutes on 1 GPU.

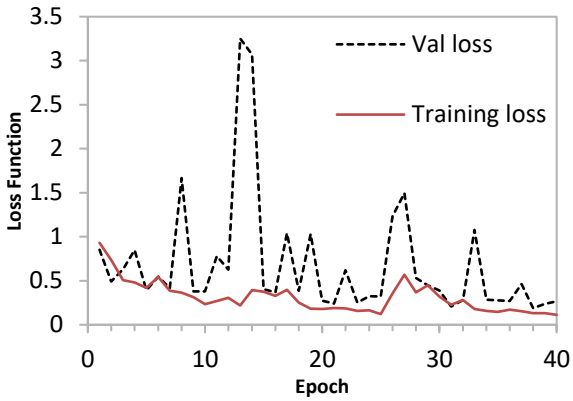

Figure 3. Loss function at each epoch

Table 3. Training time

Phase	Training time
I	7 min
II	24 min
III	48 min
IV	68 min

4.1 Metrics Results

The results of the average precision of the network predictions are reported in Table 4.

Table 4. Average precision results

Average precision	
AP_{50}	0.8792
AP_{75}	0.7438
mAP	0.5984

The inference time is calculated by averaging the time required to segment 10 images, which amounted to 3173 ms with the current hardware setting.

4.2 Examples of network predictions

As illustrated in Figure 4 and Figure 5, the network's performance was excellent for the classification task with a confidence score nearly equal to 1, while keeping a reasonably high-performance level on segmentation with

an IoU measured above 0.85.

Figure 4. Excavator example (Confidence score/IoU), network prediction (red line) vs. ground truth (green line)

Figure 5. Truck example (Confidence score/IoU), network prediction (red line) vs. ground truth (green line)

Although the developed network performs with a good level of accuracy on most testing images, there are some situations in which the network segments the instances weakly. For example, if there is an occlusion in the image, as illustrated in Figure 6, the network has difficulty recognizing the proper boundaries of the occluded objects. For example, in the case of Figure 6, the IoU for the excavator was as low as 0.485.

Another condition that dramatically affects the network's performance is low illumination. As shown in Figure 7, the overall lighting in the image is low. The network confidence score associated with the truck was consequently below the detecting threshold of 0.90, which means the truck cannot be detected by the network.

Figure 6. Example of Occlusion (Confidence score/IoU), network prediction (red line) vs. ground truth (green line)

Figure 7. Poor illumination example

5 Discussion

A deep learning model is developed to segment excavators and trucks utilizing a relatively small dataset of public domain images.

By using a small dataset, complications arise in the training process as the network faces challenges such as overfitting. Transfer learning, data augmentation, and fine-tuning techniques were used to decrease the effect of overfitting and increase the accuracy of results.

To deal with the challenges associated with occlusion and low lighting, it is proposed that a larger training dataset should be created to enhance the network's performance. In this study, the use of data augmentation is limited to flipping, yet the use of a conclusive data augmentation that deals with occlusion and lighting should be considered in future studies.

With the current hardware setting, the inference time was measured as 3173 ms, which is high for real-time applications. By using a more powerful hardware setting, the inference time could be decreased. Additionally, other implementations of Mask R-CNN should be tested to avoid slow performance related to weak network implementation.

6 Conclusion

In this study, a deep learning model was developed to accurately detect and segment two types of construction machineries. The network's performance resulted in an average precision of 0.8792 and inference time of 3173 ms, using a relatively small dataset and a transfer learning technique.

The pixel-level segmentation approach can provide the spatial information about objects. Compared with the previous approach relying on bounding boxes to measure proximities between vehicles, the generated pixel-level masks increase the accuracy of the proximity calculations related to safety control and decrease false safety alarms.

The number of vehicle classes in the proposed model could be increased to include a broader range of machineries and could be used to efficiently manage construction assets and monitor safety on jobsites.

The network performs poorly when objects are occluded or poorly lit. In future studies, it is proposed that a larger and more diverse dataset be used to overcome problems such as occlusion and poor lighting. Further work is also needed to study the effect of data augmentation on the network performance when it faces occlusion or illumination challenges.

References

[1] Ghanem AG, AbdelRazig YA (2012) A Framework for Real-Time Construction Project Progress Tracking. Earth & Space 2006 1–8.

[2] Luo H, Xiong C, Fang W, Love PED, Zhang B, Ouyang X (2018) Convolutional neural networks: Computer vision-based workforce activity assessment in construction. Automation in Construction 94:282–289.

[3] Son H, Choi H, Seong H, Kim C (2019) Detection of construction workers under varying poses and changing background in image sequences via very deep residual networks. Automation in Construction 99:27–38.

[4] Fang Q, Li H, Luo X, Ding L, Rose TM, An W, Yu Y (2018) A deep learning-based method for detecting non-certified work on construction sites. Advanced Engineering Informatics 35:56–68.

[5] Fang W, Ding L, Zhong B, Love PED, Luo H (2018) Automated detection of workers and heavy equipment on construction sites: A convolutional neural network approach. Advanced Engineering Informatics 37:139–149.

[6] Lee Y-J, Park M-W (2019) 3D tracking of multiple onsite workers based on stereo vision. Automation in Construction 98:146–159.

[7] Ayhan BU, Tokdemir OB (2019) Predicting the outcome of construction incidents. Safety Science 113:91–104.

[8] Luo X, Li H, Yang X, Yu Y, Cao D (2018) Capturing and Understanding Workers' Activities in Far‐Field Surveillance Videos with Deep Action Recognition and Bayesian Nonparametric Learning. Computer-Aided Civil and Infrastructure Engineering. 34:333–351.

[9] Ding L, Fang W, Luo H, Love PED, Zhong B, Ouyang X (2018) A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. Automation in Construction 86:118–124.

[10] Mneymneh BE, Abbas M, Khoury H (2017) Automated Hardhat Detection for Construction Safety Applications. Procedia Engineering 196:895–902.

[11] Fang Q, Li H, Luo X, Ding L, Luo H, Rose TM, An W (2018) Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. Automation in Construction 85:1–9.

[12] Fang W, Ding L, Luo H, Love PED (2018) Falls from heights: A computer vision-based approach for safety harness detection. Automation in Construction 91:53–61.

[13] Kim D, Liu M, Lee S, Kamat VR (2019) Remote proximity monitoring between mobile construction resources using camera-mounted UAVs. Automation in Construction 99:168–182.

[14] Kim J, Chi S, Seo J (2018) Interaction analysis for vision-based activity identification of earthmoving excavators and dump trucks. Automation in Construction 87:297–308.

[15] Guo Y, Xu Y, Li S (2020) Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network. Automation in Construction 112:103124.

[16] Dutta A, Zisserman A (2019) The VIA Annotation Software for Images, Audio and Video. In: Proceedings of the 27th ACM International Conference on Multimedia. ACM, New York, NY, USA, pp 2276–2279.

[17] He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp 2980–2988.

[18] Abdulla W (2017) Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow. GitHub repository, https://github.com/matterport/Mask_RCNN

[19] Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A (2010) The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results.

[20] Hui J (2019) mAP (mean Average Precision) for Object Detection. In: Medium. On-line: https://medium.com/@jonathan_hui/map-mean-

average-precision-for-object-detection-
45c121a31173. Accessed 14 Dec 2019.