An improved multi-object instance segmentation based on deep learning

Nawaf Farhan Fankur Alshdaifat*, Mohd Azam Osman, Abdullah Zawawi Talib
School of Computer Sciences
Universiti Sains Malaysia,11800, Pulau Pinang, Malaysia
**Corresponding author: nawaf@student.usm.my*

Abstract

Deep Learning (DL) networks have attracted growing interest and attention by researchers and scholars alike due to the growing importance of detecting and instance segmentation of objects in an image. Instance segmentation is a critical issue that requires further improvement due to the difficulties in adapting object detection and instance segmentation approaches. This paper presents an approach that overcome these issues by proposing a new approach based on the recent DL approach in addition to developing an approach for multi-object instance segmentation. The improved multi-object segmentation approach presented in this paper consists of three stages. Firstly, it improves the RestNet-101 (Residual Neural Network) backbone by connecting it to the convolution layer for each ResNet block. Secondly, the localization of multiple objects is improved by enhancing the Region Proposal Network (RPN), and thirdly, a complex instance segmentation approach is utilized. The result of this study based on a standard dataset, called the Common Object in Context (COCO) dataset, reveals that the suggested approach compared to other well-known segmentation approaches, has improved the instance segmentation process in terms of precision and training time.

Keywords: COCO dataset; deep-learning; instance-segmentation; localization; multi-object

1. Introduction

Deep learning (DL) is a branch of machine learning which has an ability to learn from unstructured data. It uses a convolution neural network (CNN) because of its ability to interact with complex data and it has a high degree of dimension for depth estimation from a single image (Abuowaida and Chan, 2020), speech recognition (Alkhawaldeh, 2019), object recognition (Al-Hmouz, 2020), object detection (Pathak *et al.*, 2018), semantic segmentation (Lateef and Ruichek, 2019) and instance segmentation (Toda *et al.*, 2020; Alshdaifat *et al.*, 2020). Also, it has the ability to handle a wide variety of data in many different ways by optimizing the deep learning approach throughout the training process. CNN that is used to extract information from the objects to resolve multi-object detection problems has two primary problems (Wang *et al.*, 2020). Firstly, the conventional approaches fail to effectively solve object identification and recognition problems. In particular, they are not able to resolve the recognition problems namely in distinguishing the object from the background and addressing the problem in labeling the object class. Secondly, there is a problem in dealing with the location of an object based on the bounding box. Recent advances in this area are normally driven by a powerful basic approach such as Region-Bounding Box (R-CNN) which draws a bounding box on each object in an image. This approach then uses AlexNet with a modified fully connected layer using Support Vector Machine (SVM) to extract a feature map for each detected image. R-CNN has several disadvantages which include a longer detection process, the need for several steps and a high computation time. Fast R-CNN is an approach that enhanced RCNN by sharing the convolution layers of various proposals to handle the low accuracy and slow detection problems

(Girshick, 2015; BackProp *et al.*, 1998). It uses selective search to produce a region of the field. The proposal region is then submitted to the Regions of Interest (RoI), and the pooling process transforms the feature within each area using max-pooling to create small feature maps. The key limitation of this approach is poor detection which is due to the bottlenecks created by the selective search. Several other researchers (Wan and Goudos, 2020; Sun *et al.*, 2018; Li *et al.*, 2017) have utilized Faster R-CNN to solve the instance segmentation problems by adding a head for the mask of each object (Xu *et al.*, 2020; Jiang, Li, *et al.*, 2020). Finally, every feature map extracted from the convolution layer supported a smaller network with two tasks: classification and regression. However, they are consuming time in the training stage. Meanwhile, the speedup of Faster R-CNN produced the idea of the Region Proposal Network (RPN) approach. Instance segmentation is an important task in object recognition methods which defines any object based on its image (abuowaida). However, there are challenges in the instance segmentation process due to several difficulties such as the variety of images, difference in the colors and size of objects, and contrast between objects. (Khened *et al.*, 2019) suggested an approach called "fully convolutional instance segmentation" (FCIS), for instance, segmentation (Ganesh *et al.*, 2019). However, FCIS suffers from overlapping instances, and errors in the predicted edges. (He *et al.*, 2017) introduced Mask Region-Convolution Neural Network (Mask R-CNN) to predict segmentation at the instance level. The suggested approach uses Faster R-CNN to forecast the mask for each component by adding a branch for every bounding box after Faster R-CNN. R-CNN mask functions concurrently to minimize the time for preparation and testing. In comparison, Mask R-CNN key achievement is in using ROI Align and producing extremely detailed performance. Moreover, another study by (Wen *et al.*, 2020) proposed three stages for instance segmentation whereby each stage has a particular task to predict the instance level for each object. This approach is time consuming as each stage does not work in parallel. Moreover, (Bolya *et al.*, 2019; Li and He, 2018) proposed a real-time approach for instance segmentation called YOLACT which utilizes the parallel idea. However, the approach has not received good feedback since the accuracy of the instance segmentation was not taken into account when solving real-time problems. Accordingly, this paper intends to create a new approach for segmentation. In order to do this, accurate detection of all objects within an image are required by using an accurate instance segmentation that combines the elements or properties from traditional digital or computer vision such as multi-object detection to classify/identify object individually, localize each image via a bounding box and perform semantic segmentation. In other words, in this paper, each pixel is categorized into a series of classifications negating the need to distinguish the object instances. In doing so, it can be assumed that a complex approach or technique is needed to achieve good results. However, on the contrary, this paper demonstrates that a reasonably straightforward and adaptive approach can outperform the previous best-in-class instance segmentation results by addressing many of the above-mentioned issues associated with instance segmentation. Therefore, a novel approach known as multi-object instance segmentation consisting of three steps is proposed. The first step involves the novel backbone architecture which extracts the feature map of each object with a higher accuracy and shorter computation time. The second step involves enhancing the Region Proposal Network (RPN) to localize/locate multiple objects. The third stage involves adopting the Fully Convolution Network (FCN) to produce an instance segmentation that overcomes the object overlapping problem. In this paper, we attempt to enhance multi-object instance segmentation. The remainder of this paper is organized as follows. Section 2 describes the existing approaches and key protocols for image segmentation. Section 3 presents the proposed instance segmentation method, and Section 4 presents the experiments, results and discussion. Finally, the conclusion is presented in Section 5.

2. Existing approaches and key protocols for image segmentation

This section describes the enhanced approaches/methods for multi-object instance segmentation through the identification of multiple object items and localization along with segmentation of each object item based on the input image. This is a challenging task due to the difference in colors and sizes of the objects. The proposed enhanced approach is established by combining the CNN network, RPN and segmentation to obtain a better result for multi-object instance segmentation. Figure 1 shows the overall architecture of the multi-object instance segmentation. The overall loss function of the multi-object instance segmentation for each object item is represented by the loss function of the instance segmentation identified using the per-pixel sigmoid and average binary cross-entropy to generate boundaries for each class. As a result, the loss function can be found in the input image.

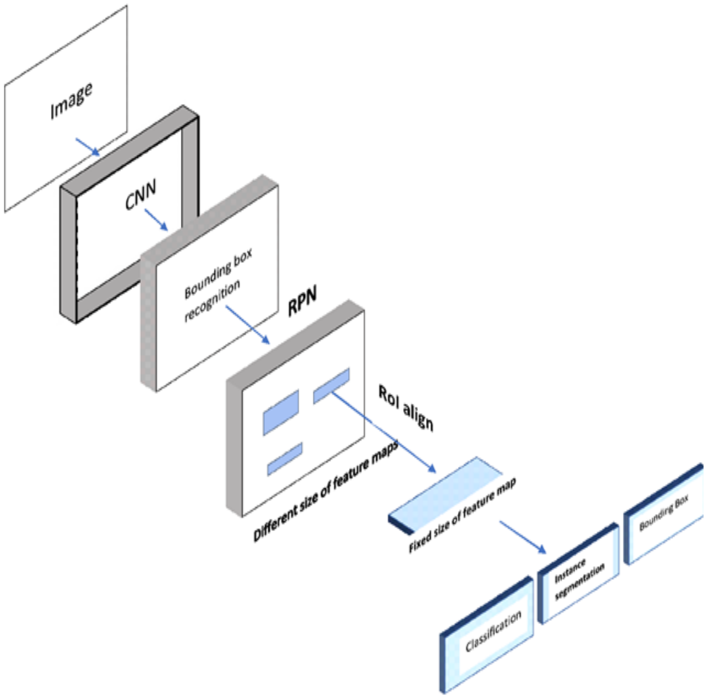

Fig. 1. Overall architecture of multi-object instance segmentation

In this paper, the proposed approach for multi object instance segmentation aims to detect and recognize multiple objects within an image while producing high-quality localization and instance segmentation. This approach is accomplished through the following steps: improving the architecture to extract the object features of an image; enhancing RPN by suggesting recommendations for multiple objects; and instance segmentation using bounding-box detection and categorization of multiple objects together with head architecture and design.

2.1 Proposed enhanced ResNet backbone

ResNet is a DL network containing a series of blocks used to address the gradient vanishing problems to enhance the residual neural network's performance. To overcome the problems, alternate or bypass connections having different dimensions, are identified as shown in Figure 2 (a) shows the existing ResNet backbone, and Figure 2(b) shows the process of combining each of the convolutional layers to incorporate all features from multiple spatial levels by summation of the properties of feature. In the proposed enhanced ResNet backbone, the first convolution layer uses a 3x3 filter size with 3x3 matrix max-pooling, as

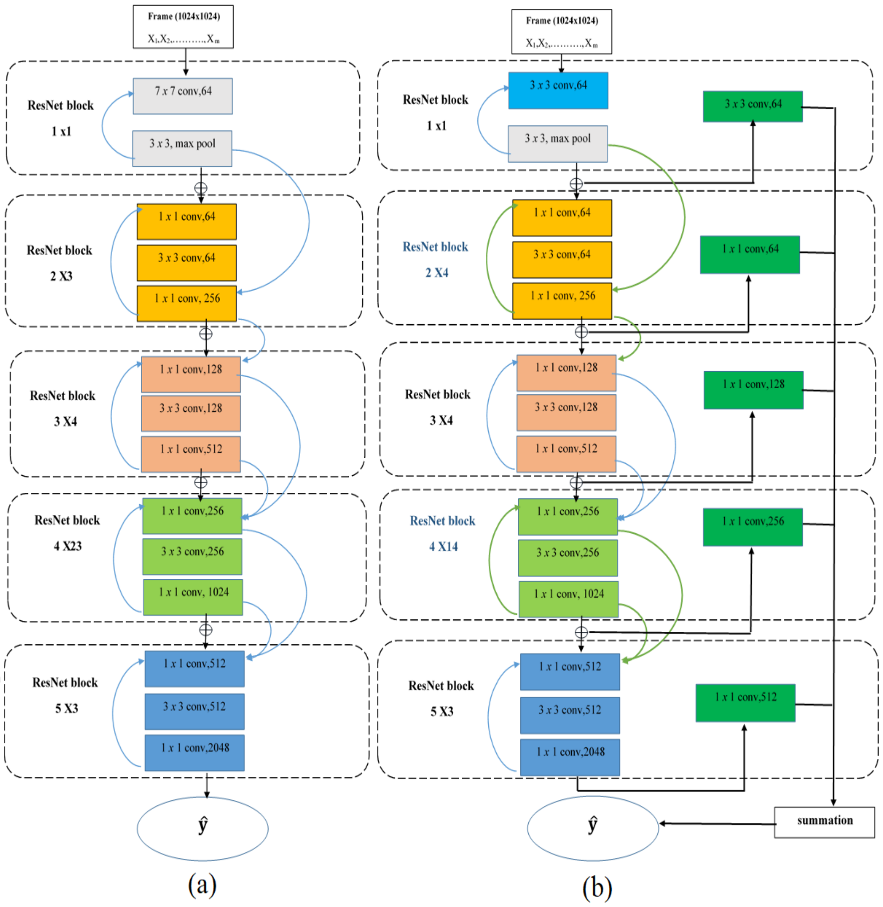

Fig. 2. (a) Existing 13 (b) Proposed (Improved ResNet backbone).

shown in Figure 2 (b). The output of the convolution layer is used to enter the ResNet block of the suggested backbone to improve the flow of knowledge across the layers further along. To find more details, the output of each ResNet block is shared through the convolution layer. Consecutively, five examples of ResNet Blocks and a convolutions layer of the building blocks are produced as shown in Figure 3

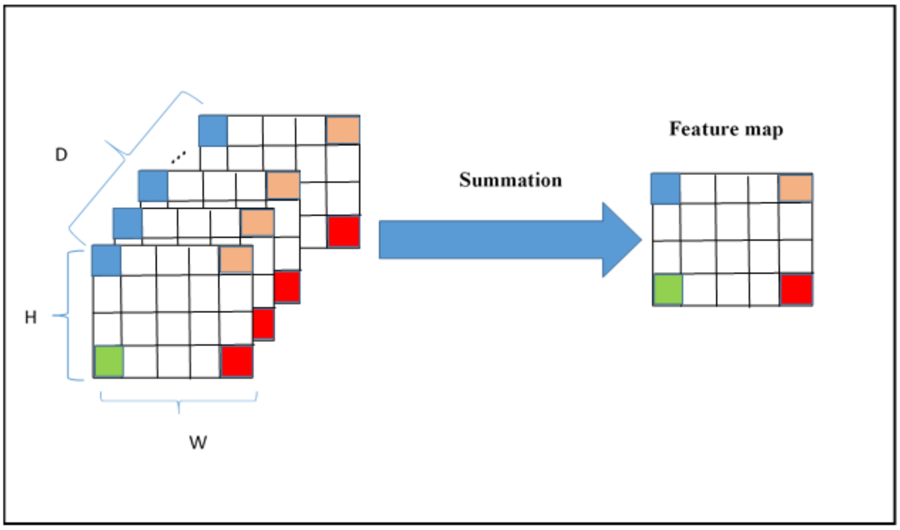

Fig. 3. shared feature maps through convolutional layer.

For retrieving more details, the output from each ResNet block is shared through the convolutional layer. Consecutively, we received five examples of ResNet Blocks and a convolutions layer of the building blocks as shown in Figure 3. Results are combined in each of the convolutional layers to incorporate all features from multiple spatial levels by summation of the properties of feature maps extracted from the ResNet blocks' five originals, as shown in Figure 2. The suggested backbone will be discussed in the following sections.

2.1.1 Proposed improved ResNet architecture

This paper proposes a new architectural design of ResNet to improve the training for each layer. Thus, it will enhance the performance of the ResNet backbone. The shortcut or bypass connection will be described in the following section to address the gradient diminishing problem. However, there are still a number of issues in ResNet such as a) identification of layers that have not received adequate training; b) identification of layers that have received over training; c) understanding the reason for selecting a large filter for the initial convolution layer, as suggested in Figure 3. As shown in the following formula, the repeated example, is paired with the proposed ResNet formulation:

$$y = \sum_{r=1}^{n} F(x, \{W_{i,r}\}) + x, \tag{1}$$

where r for each ResNet building block is the number of repetitions. The value of r of the unfittingly qualified network layers should be increased and decreased for properly trained layers. The enhanced reverse propagation of ResNet-101 is based on the following formula:

$$\frac{\partial L}{\partial \omega_i^{conv}} = \delta_i^{conv} \cdot \theta(s_i^{conv-1}) \tag{2}$$

A smaller filter size than ResNet is selected to extract complex and smaller features when fewer parameters are needed. Therefore, this step will increase the efficiency of the computation. Also, adding a convolutional layer for each block increases the performance of the feature selection process through extract several local and accurate feature map on the shallow layers, which can be written as:

$$F_{futuremap} = \sum_{i=1}^{W} \sum_{j=1}^{H} y_q(i,j), \tag{3}$$

where the $F_{futuremap}(.)$ feature maps. y_q is element of the feature map with spatial dimension HXW.

Feature map statistics through sum pooling to generate channel-wise statistics $F_{futuremap}$ ∈R^{HXW} and the q^{th} element of $q = 1, 2, ..., D$ as shown in this Equation.

3. Localisation

RPN is improved by extending the Faster R-CNN architecture for localization of multiple objects of different sizes in each image. The enhanced Faster R-CNN increases the degree of detection of the bounding boxes. The proposed enhanced RPN is described in the following sub-sections.

3.1 Enhancing RPN by adding shortcut identification

The proposed enhanced RPN is achieved by using shortcut identification as follows: RPN produces numerous rectangular object proposals which are added to the feature map via a proposed CNN backbone to predict the existence of the object for each image. A regression model is used to manage multi-scale feature maps and split the feature map into a set of anchors for the image size (n x n) as illustrated in Figure 4.

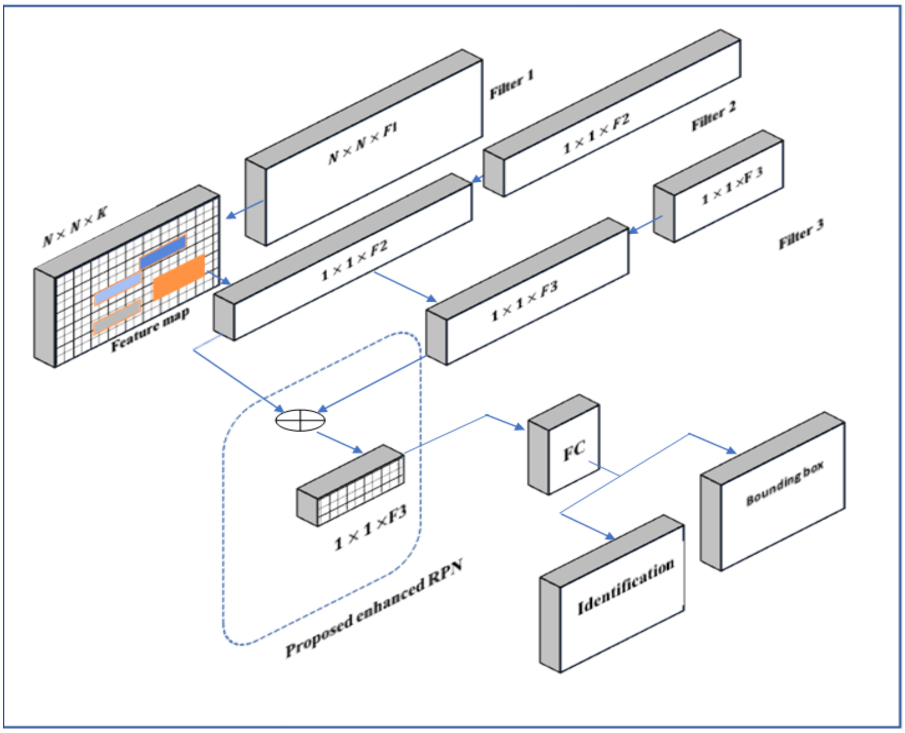

Fig. 4. Improved of Faster R-CNN.

Forward formula for each bounding box have four coordinates of a frame patch x.

$$B = (B_x, B_y, B_{height}, B_{width}) \tag{4}$$

.

$$Loss_l oc = \sum_{j \in (x,y,height,width)} |f(\hat{B}_j) - f(B_j)| \tag{5}$$

Where

$$f(B) = \begin{cases} 0.5B^2 & if|B| < 1 \\ |B| - 0.5 & otherwise \end{cases} \tag{6}$$

B^j= target bounding box
$\hat{B}^j$ = Predicted bounding box.
W= Weights for RPN.

the last layer forward formula as a follow:

$$B_j = \sum_{j=1}^{n} \sum_{i=1}^{m} W_j{}^i \cdot B_j{}^i \tag{7}$$

.

the hidden layer forward formula as a follow:

$$B_j = \sum_{j=1}^{n} \sum_{i=1}^{m} W_j{}^i \cdot B_j{}^i + B_j{}^i \tag{8}$$

.

The back propagation from the last layer :

$$\frac{\partial L}{\partial \omega_k^4} = \frac{\partial L}{\partial B_k^4} \cdot \frac{\partial B_k^4}{\partial \omega_k^4} \tag{9}$$

$$= \delta_k^n \cdot \frac{\partial \left(\sum_{k=1}^n (B_k^3) \cdot \omega_k^4 \right)}{\partial \omega_k^4} = \delta_k^4 \cdot \ (B_k^3) \tag{10}$$

where

$$\delta_k^4 = \frac{\partial L}{\partial B_k^4} \tag{11}$$

$$= \frac{\partial \sum_{j \in (x,y,height,width)} |f({B_j}^4) - f(B_j)|}{\partial B_k^4} \tag{12}$$

$$= \frac{\partial f(B_k^4)}{\partial B_k^4} = f'(B_k^4) \tag{13}$$

where $\hat{B_k}^4 = f(B_k^4)$ is the standard deviation between the expected performance and the final layer performance. Then, weight gradient L is determined using

$$\frac{\partial L}{\partial \omega_k^4} = \frac{\partial L}{\partial B_k^3} \cdot \frac{\partial B_k^3}{\partial \omega_k^3} \tag{14}$$

$$= \delta_k^3 \cdot \frac{\partial \left(\sum_{k=1}^n B_k^2 \cdot \omega_k^3 + B_k^2) \right)}{\partial \omega_k^3} = \delta_k^3 \cdot (B_k^2) \tag{15}$$

where B_k^3 have two parts,$B_k^3 = \sum_{k=1}^n (B_k^2) \cdot \omega_k^3$ is the stander part and (B_k^2) is added through the identity shortcut connections.

δ_k^3 represented as

$$\delta_k^3 = \frac{\partial L}{\partial B_k^3} = \frac{\partial L}{\partial B_k^4} \cdot \frac{\partial B_k^4}{\partial B_k^3} = \sum_{k=1}^n \delta_k^4 \omega_k^4 \tag{16}$$

Then, calculate the gradient of L for the w_k^2

$$\frac{\partial L}{\partial \omega_k^2} = \delta_k^2 \cdot B_k^1 \tag{17}$$

The B_k^2 represented as

$$\delta_k^2 = \sum_{k=1}^n \delta_k^3 (\omega_k^3 + 1) \tag{18}$$

There is the same gradient formula in the remainder of the hidden layer in RPN.

$$\frac{\partial L}{\partial \omega_k^n} = \delta_k^n \cdot B_k^{n-1} \tag{19}$$

and

$$\delta_L^n = \sum_{k=1}^n \delta_k^{n+1} (\omega_k^{n+1} + 1) \tag{20}$$

where n= 2.3....4.

Finally, calculate the gradient of L for w_k^1 as

$$\frac{\partial L}{\partial \omega_k^1} = \delta_k^1 \cdot x_k \tag{21}$$

and

$$\delta_i^1 = \sum_{k=1}^{n} \delta_k^2 (\omega_k^2 + 1) \tag{22}$$

.

The RPN gradient (3 layers) connection weight can therefore be represented as

$$\frac{\partial L}{\partial \omega_k^n} = \delta_k^n \cdot\ B_k^{n-1}, \quad n = 1, 2, 3 \tag{23}$$

.

3.2 Enhancing RPN by repeating the detection model

Enhancing RPN is accomplished by repeating the detection model for all boxes to obtain bounding boxes and achieve better accuracy. Figure 5 (a) shows the proposed enhanced RPN and (b) shows the existing RPN method (Faster RCNN).

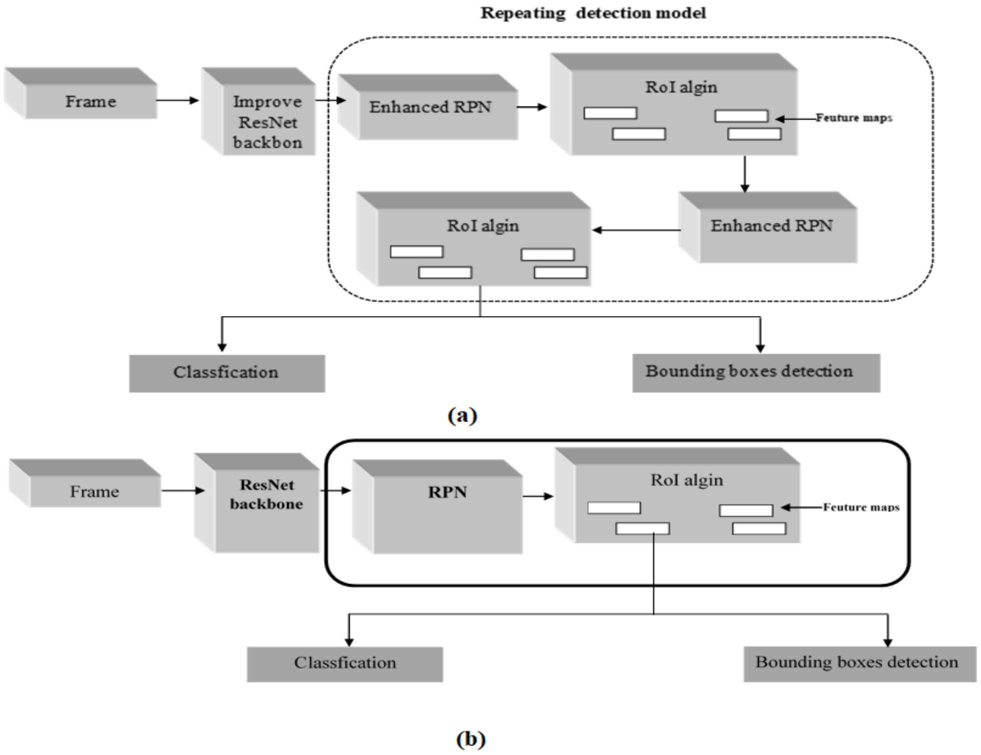

Fig. 5. (a) Proposed enhanced RPN method (a) Existing method (Faster R-CNN)

RPN is used to determine the presence of the object in the frame (Figure 6) depend on the p value, which shall be described as p* equals to:

$$p* = \begin{cases} 1 & if IoU < 0.7 \\ -1 & if IoU < 0.3 \\ 0 & otherwise \end{cases} \tag{24}$$

This formula calculates the degree of overlapping between anchors and bounding boxes of ground-truth, where Intersection over Union (IoU) is described as

$$IoU = \frac{Anchor \cap groundtruthbox}{Anchor \cup groundtruthbox} \tag{25}$$

This formula is used to decide whether the expected box includes an item or it is from the background.

3.3 Proposed multi-object instance segmentation

Instance segmentation is an important stage in object detection (Long *et al.*, 2015). In this stage, images are divided into instances and meaningful segments where each object is classified. The purpose of multi-object instance segmentation is to obtain good quality segmentation by overcoming the object overlapping problems. Since the image contains a set of overlapping objects, the border of each object is detected. Thus, to address the overlapping issues, FCN is added.

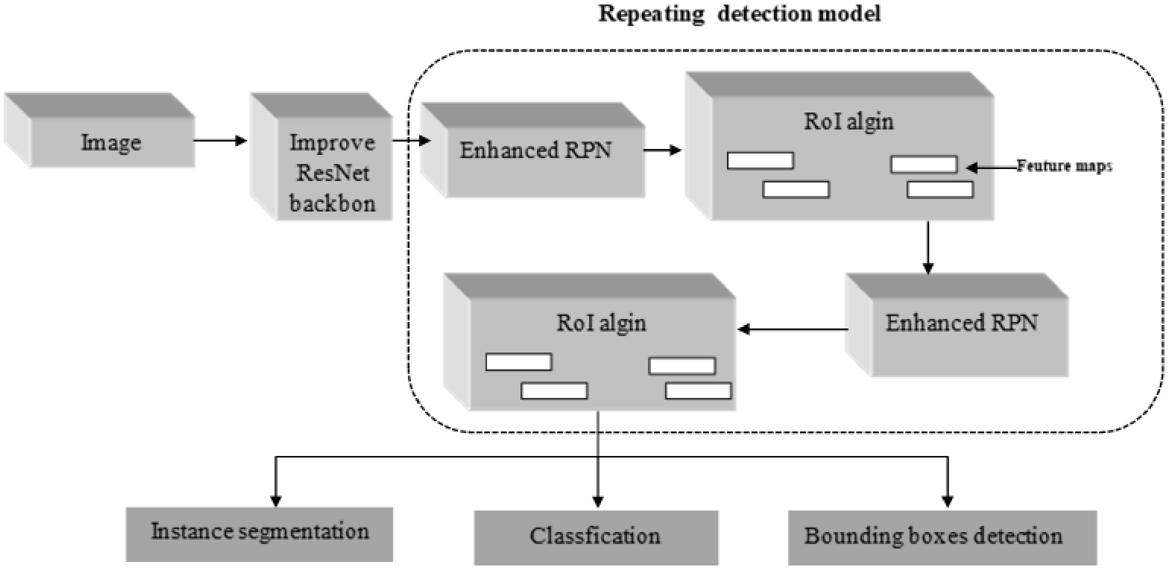

Fig. 6. Architecture of instance segmentation

As shown in Figure 6, a RoI alignment layer is used for a feature map of the same dimension because the bounding box is made up of multiple sizes of the feature map. The segmentation loss feature of the instance is defined as

$$L_{segmentation} = -\frac{1}{s^2} \sum_{1 \leq i,j \leq s} \left[y_{ij} \log y_{ij}^k + (1 - y_{ij} \log(1 - y_{ij}^k) \right] \tag{26}$$

Where:
y_{ij}=the ground truth of boundaries
y_{ij}^k = the predicted value of boundaries.

4. Results and Discussion

The experimental findings and the assessment of the approach described in the preceding sections are presented in this section. The assessment of the proposed multi-object instance segmentation is based on the following measures:

1. AP (Averaged Precision across intersection over union (IoU) thresholds) is used to compare the proposed approach with various approaches
2. The average GPU time required for training and testing is used to gauge the efficiency of the enhanced multi-object instance segmentation approach.

The results were compared with other multi-object instance segmentation approaches, namely Mask R-CNN (He *et al.*, 2017), CASCADE R-CNN (Zhong *et al.*, 2020), FCIS (Khened *et al.*, 2019) and MNC (Wen *et al.*, 2020).

4.1 Experimental specifications

he proposed approach was tested using the Amazon Web Services (AWS) and Amazon Machine Image (AMI) with GPU-Us-Tesla V100 16 GB and VCPUs-8 core 61 GB requirements. The optimal approach used in this approach is based on gradient descent (SGD)(Aljarrah *et al.*, 2012). The weight decay was 0.0001, the learning momentum was 0.9 and the learning rate was 0.001 over 12 epochs. The result was then compared with other approaches on the COCO dataset (Lin *et al.*, 2014).

4.2 Datasetes

The performance of our approach on COCO (Lin *et al.*, 2014) was evaluated using 118k images for training, 5k for validation and 20k for testing. COCO's AP measure averages AP across IoU thresholds from 0.5 to 0.95, with an interval of 0.05. It detects the performance at various qualities. All the approaches were trained using the COCO training dataset and evaluated using the validation set. The final results also include the test set for a fair comparison with the state-of-the-art approaches.

4.3 Result of the evaluation of the proposed ResNet backbone

Table 1 presents the performance of various backbone networks and our proposed backbone on COCO dataset (Lin *et al.*, 2014). Our proposed backbone outperformed other backbones in the evaluation in the evaluation. The proposed backbone architecture chooses a specific number of duplicates for each convolution block, as illustrated in Figure 3. The improved backbone enhances the performance by selecting suitable duplicates for each block of convolution layers and decreasing the filter size to extract the features of the input image. Furthermore, adding the convolution layer for each block increases the efficiency of the selection method by collecting several local and extracting function maps from shallow layers. The features are then fed into other layers leading to good results compared to other backbones i.e., the other backbones ResNet and ResNetXt, as illustrated in Table 1.

Table 1. Performance of various backbone networks and our proposed backbone on COCO dataset at training and inference on different thresholds (0.50, 0.75, (S) small, (M) medium, (L))

Backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Our backbone	46.0	67.2	41.8	30.5	48.3	57.1
ResNet-50	37.3	59.2	40.9	21.4	40.8	49.8
Resnet-101	40.0	61.8	43.7	22.5	43.4	52.7
ResNetXt-101	42.1	64.1	45.9	25.6	45.9	54.4
ResNetXt-152	45.2	66.9	49.7	28.5	49.4	56.8

4.4 Result of the evaluation of the proposed enhanced RPN architecture

Table 2 compares the performance of object detection single-model results (Average Precision bounding box APbb) against other state-of-the-art approaches. Based on the accuracies, the proposed approach produces the best-fit solution.

Table 2. Performance of object detection singlemodel results (bounding box APbb) and state of-the-art approaches on COCO dataset based on different thresholds (0.50, 0.75, (S) small, (M) medium, (L) large)

approach	Back-bone	AP$_{bb}$	AP$_{bb}$50	AP$_{bb}$75	AP$_{bb}S$	AP$_{bb}M$	AP$_{bb}L$
Our-Approach	**ResNet-101**	**45.0**	**66.2**	**41.8**	**30.5**	**48.3**	**57**
Our-Approach	**improved ResNet**	**46.0**	**67.0**	**42.5**	**31.2**	**49.2**	**57.3**
Faster R-CNN, RoIAlig	ResNet-101	37.3	59.6	40.3	19.8	40.2	48.8
Mask R-CNN	Resnet 101	38.2	60.3	41.7	21.1	41.1	50.2
Cascade R-CNN	ResNet-101	42.1	64.1	45.9	25.6	45.9	54.4

Henceforth, several strategies are proposed in this paper to address the issues surrounding inadequate training and defining the best possible filter size. Adding a convolutions layer for each block improves the productivity of the selection method by capturing many local and precise characteristics on the shallow layers. Similarly, RPN is enhanced via the expansion of a Faster R-CNN architecture to enable the localization of multiple but different sized objects within each image. It is anticipated that the enhanced Faster R-CNN will have the capacity to increase the extent of regression for the bounding boxes. Likewise, a shortcut to identity and enhanced RPN is also proposed. The anticipated improvement in the proposed approach for RPN aims to lessen the disparity of IoU, decrease false positives and allow proper distribution of training to reduce overfitting and affects concerning instance segmentation of the supposed Cascade R-CNN and Mask RCNN approach utilizing Faster R-CNN with no modifications made beforehand. Therefore, the proposed approach resulted in substantial performance improvement for multi-object instance segmentation.

4.5 Result of the evaluation of the evaluation of the proposed multi-object instance segmentation

Our proposed approach was compared with to other state-of-the-art approaches. All the approaches were trained and tested with several thresholds. At various AP thresholds, the proposed approach worked higher. Thanks to the proposed approach enhancements, the proposed approach's AP performance was 44.2, which was greater than that of R-CNN, R-CNN CASCADE, MNC and FCIS.

Table 3. Results on the evaluation state-of-the-art approaches and our proposed backbone on COCO dataset with different thresholds (0.50, 0.75, (S) small, (M) medium, (L))

Approach	AP	AP$_{0.50}$	AP$_{0.75}$	AP$_S$	AP$_M$	AP$_L$
Mask R-CNN baseline	36.9	55.5	43.8	20.4	41.2	48.3
Cascade mask R-CNN	42.0	57.3	44.8	25.0	45.0	53.3
MNC	24.6	44.3	24.8	40.7	25.9	60.2
FCIS	29.2	49.5	27	41.1	30.0	50.0
ResNet-101 and improved RPN	**42.2**	**59.1**	**45.3**	**25.2**	**43**	**53.4**
Our approach	**44.2**	**63**	**38.5**	**29.3**	**45.0**	**57**

The backbone for improvements solves the problem of disappearing gradients by residual learning by either growing or decreasing the training performance of each convolution block in the case of a certain duplicate. Simultaneously, the filter size is reduced to obtain the required number of features from the image input, which are then fed into other layers to improve results compared with other approaches. Further, in comparison with Mask RCNN and FCIS, which use Faster R-CNN without prior modifications, Faster R-CNN approach established by reusing RPN to achieve best results in the detection phase had a positive impact on instance segmentation since the usage of modified RPN by Cascade R-CNN approach has taken a long time to adjust. In brief, the proposed approach has obtained significant efficiency relative to the other approaches in multi-object instance segmentation

4.6 Training Time

The training time and frame output are essential considerations to assess network efficiency during the testing phase. These considerations have also been tackled by implementing a range of innovations that decreases the time taken during the training phase and improves the pace of the output frame per second during the testing process. The period of preparation and the production of frames per second are illustrated in Table 4.

Table 4. Evaluation of training phase and production of frames per second in the multi-object segmentation testing phase utilizing numerous state of-the-art approaches

Approach	Training Time in second	Frame per second
Proposed approach	2465	**9.05**
Mask R-CNN	2478	6.07
CASCADE R-CNN	9580	8.03
MNS	5177	5.45
FCIS	3256	5.75

Based on table 4,the proposed approach has the shortest training time and the largest number of frames per second in the testing process. Therefore, by choosing the required filter measurements, the issues associated with ResNet-101 are prevented and consideration is given to the layer which requires more training as well as to minimize the iteration of layer training, which does not need more training, achieving better outcomes at the quickest possible period. On either hand, state-of-the-art architectures have become more time-consuming, as the backbone network has been influenced by ResNet-101. As already pointed out, there are several issues with ResNet-101 design, such as a large filter scale, which raises the training times. Training has now been undertaken on several levels, which do not need preparation or increasing the period required to complete the training. Figure 7 shows some visual experimental findings from this paper and the link to the ground truth

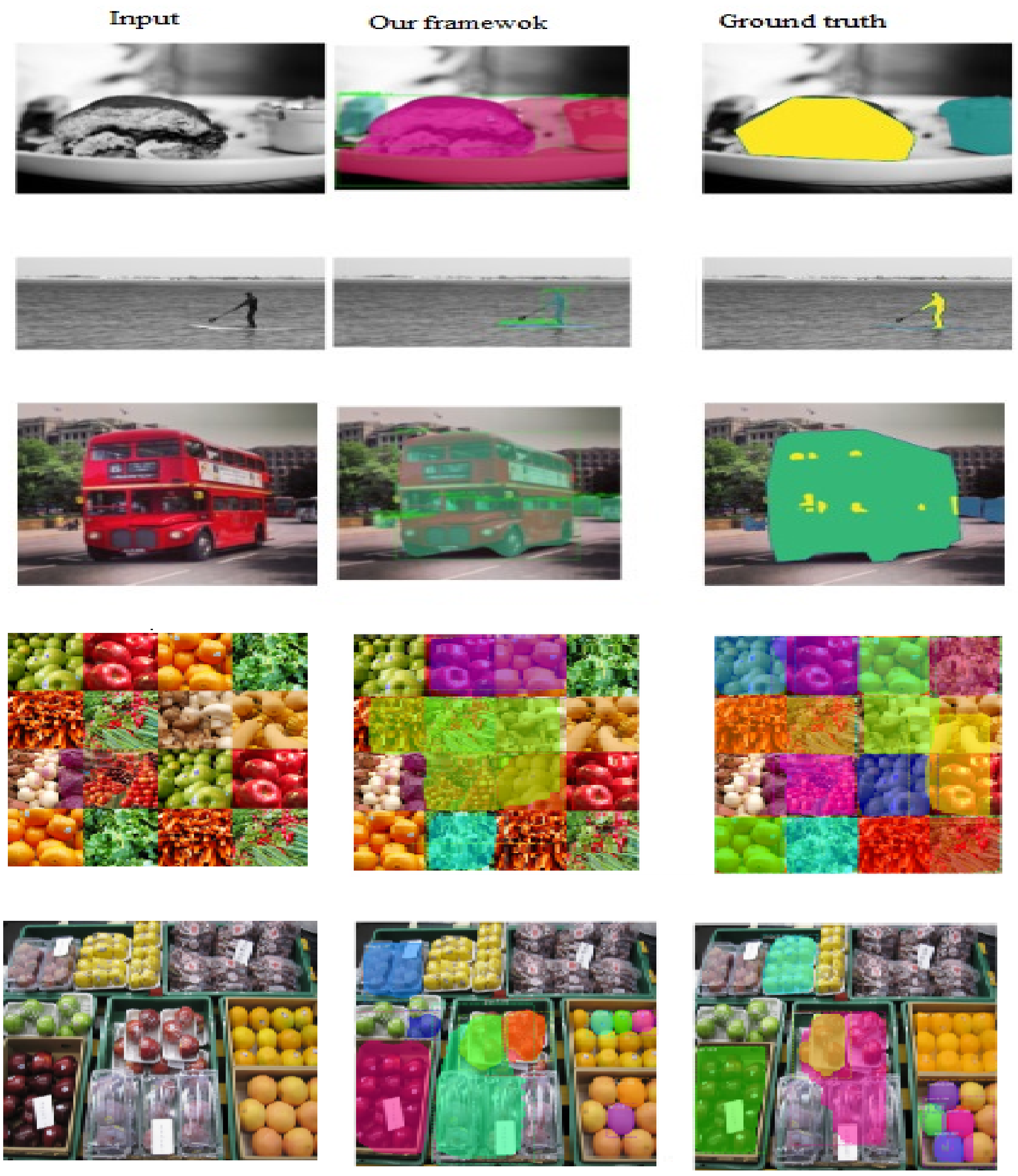

Fig. 7. Visual results from this paper compared to ground truth

5. Conclusion

This paper has proposed a novel approach for multi-object instance segmentation. The instance segmentation process has been enhanced by improving the feature extraction process i.e., by creating a new backbone by connecting multiple copies of the ResNet improvement blocks and linking them to the convolutional layer to gain essential channel features and improve the use of more important images. Further, an updated and optimized identification technique has been introduced to allow better detection of multiple items by utilizing RPN and an instance segmentation process. Better performance in term of the accuracy (AP with different thresholds) has been obtained by the proposed approach compared to other exiting methods. The proposed approach easily and accurately detects, locates and segments multiple objects with a shorter training time for the testing stage.

References

BackProp, E., LeCun, Y., Bottou, L., Orr, G. B., & Muller, K. (1998). Neural networks: Tricks of thetrade.

Aljarrah, I. A., Ghorab, A. S., & Khater, I. M. (2012). Object recognition system using template matching based on signature and principal component analysis. *International Journal of Digital Information andWireless Communications (IJDIWC)*, *2*(2), 156–163.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. Mi- crosoft coco: Common objects in context. In: *European conference on computer vision*. Springer. **2014**,740–755.

Girshick, R. Fast r-cnn. In: *Proceedings of the ieee international conference on computer vision*. **2015**,1440–1448.

Long, J., Shelhamer, E., & Darrell, T. Fully convolutional networks for semantic segmentation. In: *Pro-ceedings of the ieee conference on computer vision and pattern recognition*. **2015**, 3431–3440.

He, K., Gkioxari, G., Dollár, P., & Girshick, R. Mask r-cnn. In: *Proceedings of the ieee international conference on computer vision*. **2017**, 2961–2969.

Li, H., Huang, Y., & Zhang, Z. (2017). An improved faster r-cnn for same object retrieval. *IEEE Access*,*5*, 13665–13676.

Li, B., & He, Y. (2018). An improved resnet based on the adjustable shortcut connections. *IEEE Access*,*6*, 18967–18974.

Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of deep learning for object detection.
Procedia computer science, *132*, 1706–1717.

Sun, X., Wu, P., & Hoi, S. C. (2018). Face detection using deep learning: An improved faster rcnn approach. *Neurocomputing*, *299*, 42–50.

Alkhawaldeh, R. S. (2019). Dgr: Gender recognition of human speech using one-dimensional conven- tional neural network. *Scientific Programming*, *2019*.

Bolya, D., Zhou, C., Xiao, F., & Lee, Y. J. Yolact: Real-time instance segmentation. In: *Proceedings ofthe ieee international conference on computer vision*. **2019**, 9157–9166.

Ganesh, P, Volle, K, Burks, T., & Mehta, S. (2019). Deep orange: Mask r-cnn based orange detection and segmentation. *IFAC-PapersOnLine*, *52*(30), 70–75.

Khened, M., Kollerathu, V. A., & Krishnamurthi, G. (2019). Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Med-ical image analysis*, *51*, 21–45.

Lateef, F., & Ruichek, Y. (2019). Survey on semantic segmentation using deep learning techniques. *Neurocomputing*, *338*, 321–348.

Abuowaida, S. F., & Chan, H. Y. (2020). Improved deep learning architecture for depth estimation from single image. *Jordanian Journal of Computers and Information Technology (JJCIT)*, *6*(04), 434–445.

Al-Hmouz, R. (2020). Deep learning autoencoder approach: Automatic recognition of artistic arabic calligraphy types. *Kuwait Journal of Science*, *47*(3).

Alshdaifat, N. F. F., Talib, A. Z., & Osman, M. A. (2020). Improved deep learning framework for fish segmentation in underwater videos. *Ecological Informatics*, *59*, 101121.

Jiang, Y., Li, C. et al. (2020). Convolutional neural networks for image-based high-throughput plant phenotyping: A review. *Plant Phenomics*, *2020*, 4152816.

Toda, Y., Okura, F., Ito, J., Okada, S., Kinoshita, T., Tsuji, H., & Saisho, D. (2020). Training instance seg- mentation neural network with synthetic datasets for crop seed phenotyping. *Communications biology*,*3*(1), 1–12.

Wan, S., & Goudos, S. (2020). Faster r-cnn for multi-class fruit detection using a robotic vision system. *Computer Networks*, *168*, 107036.

Wang, N., Wang, Y., & Er, M. J. (2020). Review on deep learning techniques for marine object recogni-tion: Architectures and algorithms. *Control Engineering Practice*, 104458.

Wen, Y., Hu, F., Ren, J., Shang, X., Li, L., & Xi, X. (2020). Joint multi-task cascade for instance seg- mentation. *Journal of Real-Time Image Processing*, 1–7.

Xu, B., Wang, W., Falzon, G., Kwan, P., Guo, L., Chen, G., Tait, A., & Schneider, D. (2020). Automated cattle counting using mask r-cnn in quadcopter vision system. *Computers and Electronics in Agriculture*,*171*, 105300.

Zhong, Q., Li, C., Zhang, Y., Xie, D., Yang, S., & Pu, S. (2020). Cascade region proposal and global context for deep object detection. *Neurocomputing*, *395*, 170–177.

Submitted: 31/10/202**0**
Revised: 04/04/2021
Accepted: 12/04/2021
DOI: 10.48129/kjs.10879