# Instance Segmentation based Object Detection with Enhanced Path Aggregation Network

Ade Indra Onthoni, Prasan Kumar Sahoo
*Dept. of Computer Science and Information Engineering*
*Chang Gung University, Guishan, Taoyuan 33302, Taiwan*
E-mail: m0929021@cgu.edu.tw, pksahoo@mail.cgu.edu.tw

2023 IEEE Latin American Conference on Computational Intelligence (LA-CCI) | 979-8-3503-4807-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/LA-CCI58595.2023.10409479

***Abstract*—Multi-scale networks rely heavily on the aggregation process to aggregate feature information into other feature maps. However, it has disadvantages in aggregating information, where the localization information becomes inconsistent. An Enhancing Path Aggregation Network through concatenation and attention mechanism is proposed here to improve the aggregation path for generating the feature maps in multi-scale network. This approach helps to keep information consistently during the process of aggregation. The proposed model is evaluated using the COCO dataset, which achieves significant improvements on several metrics.**

***Index Terms*—Object Detection, Multi-Scale Feature Map, Computer Vision**

## I. Introduction

Object detection is one of the computer vision tasks, where the goal of this task is to classify and locate the object inside the input image or video. This object detection task can be implemented in different domains such as kidney detection [1] in the medical field or food calorie estimation in health monitoring systems [2]. Object detection can be done using DL (Deep Learning) which is a part of ML (Machine Learning). ML is more of a classic technique where the extracted features that need to be used for object detection task are predefined [3], [4], whereas the DL approach is a more modern technique that teaches the computer to learn the important features and automatically extract the information that is needed for the object detection task. In DL, feature patterns are extracted and learned through the training process [5], [6].

Objects in images can appear in different scale sizes, which is one of the most challenging problems in object detection. To solve this problem, FPN (Feature Pyramid Network) [7] and PANet (Path Aggregation Network) [8] is proposed as a multi-scale network , which can generate feature maps with different scale sizes. The networks will aggregate previous feature map information from the backbone (FPN) or top-down path (PANet) to generate other feature maps that can help detect objects with different scale sizes, and the network shows a huge improvement in detection accuracy. However, the network PANet which an improved architecture after FPN, there is a problem in the way PANet aggregates feature information. The bottom-up augmentation path is an idea proposed by PANet to solve the problem of aggregating low-level features in FPN. However, this path proposed from PANet still leads to inconsistencies in localization information during the process of generating multi-scale feature maps, because in the process of generating new feature maps, it will go through the downsampling process and then merge the downsampled Feature maps and feature maps in the top-down path with the same size, and this process will cause information inconsistency that can lead to false positive detection result.

Motivated by the problem of bottom-up augmentation paths in PANet, we propose a method to enhance the aggregation of feature maps with accurate localization information in PANet by using concatenation, and by adding modified spatial and channel attention mechanism to improve the performance of feature maps. Our research goal is to overcome the limitations of the current bottom-up enhancement path in PANet, so we aim that our designed network can generate multi-scale feature maps with consistent localization information throughout the process, thereby improving the performance to identify the objects by accurately locate and recognize the objects. Our main contribution is proposed a method by modifying the network using PANet as a baseline model, and our method can maintain information consistency during feature information aggregation, thereby improving the ability of the model to accurately locate and classify objects. Furthermore, we propose an improved attention mechanism that enhances features in the internal feature map for improved object recognition. Our contributions are summarized as follows:

- We propose a modified PANet network that uses concatenation to ensure consistent localization information.
- We propose a modified attention mechanism that enhances the features inside feature map.

To evaluate our proposed method, we use the COCO detection benchmark [9]. Our results show that the modified PANet network and attention mechanism outperform the FPN and PANet in terms of mAP, mAR, and F1-score, demonstrating the effectiveness of our proposed method.‘

## II. Related Work

### A. CNN (Convolutional Neural Network)

Object detection networks have grown rapidly over the years, from extracting features using handcrafted techniques

This work is funded by the National Science and Technology Council (NSTC), Taiwan under the grant number 110-2221-E-182-008-MY3.

using ML to DL methods using CNN, that can automatically extract features from images using the basic components in CNN such as convolution layers. This layer will slide over all pixels in the input image to obtain feature information. After extracting the feature information, these feature values need to be normalized to improve performance and help stabilize the neural network, using the activation function which is also one of the basic components of CNN. Another important basic component in CNN is pooling, this is a common operational component in CNNs, usually applied after convolution layers and activation functions. The main function of pooling is to reduce the space or dimension size of the feature map, which can reduce the training time and can be computationally efficient. Another important function of pooling is to help summarize the features present in the feature map region of the convolution layer, it will also help making the network more robust to the spatial variation of the input data.

### B. FPN (Feature Pyramid Network)

FPN is one of the earliest multi-scale networks that can generate feature maps with different scale sizes for solving the challenge in different scale sizes object detection task. Typically, the detection model will only use the single feature map generated from the final stage that contains the richest semantic information feature map. In contrast, FPN reuses the feature maps from each stage in bottom-up path or backbone (CNN) and combine them with the feature maps from the final stage to generate another feature map that contains high semantic information and has different scale sizes according to their stages. Multi-scale feature maps has improved the performance of object detection model using multi-scale feature maps for better capturing objects at different scales instead of using single-scale feature maps.

In CNN early stage, the feature map contains low-level features with accurate localization information, which helps recognizing large-sized objects in the image. However, the path to generate multi-scale feature map in FPN is long, from low-level features to top-level features, making it difficult to obtain accurate positioning information from low-level features, which has become the drawback of FPN.

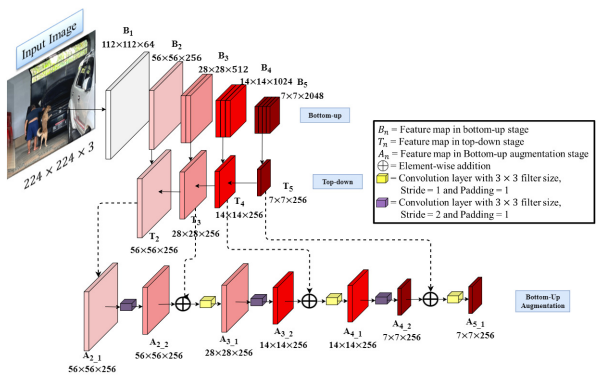


Fig. 1. PANet (Path Aggregation Network) Framework.

### C. PANet (Path Aggregation Network)

PANet is an instance segmentation network that was developed to solve the FPN drawback during the process of aggregation, where the localization information cannot be correctly aggregated to all feature maps in a multi-scale network. PANet designs the network to shorten the path from low-level features to top-most features by adding another bottom-up path to generate multi-scale feature maps with better localization information. In PANet, by adding an additional downsampling path called bottom-up augmentation path, as shown in Fig. 1, low-level features can be aggregated into other feature maps to improve localization information that helps large instance recognition. However, in the additional path, the feature map will go through several layers, and merge to generates a feature map with accurate localization information, and this process will cause the information changes the consistency of the information in $T_2$. This information inconsistency will affect the detection performance negatively.

### D. Attention Mechanism

The attention mechanism is a learning model that selectively focuses on important parts of the extracted features, and unimportant features are suppressed. The first attention mechanism is used in sequence-to-sequence models, such as language translation, speech recognition, and image recognition. This mechanism is first introduced on the encoder decoder based NMT (Neural Machine Translation) [10] in NLP (Natural Language Processing), and then applied in object detection. The used of attention mechanism in object detection model is mostly applied in spatial and channel level of the feature map and this application has shown an improvement in detection model. In order to improve and enhance the feature map in our network, we apply spatial and channel attention mechanism to our feature map that are being generated. We modify the spatial attention mechanism by adding few layers with different size so it can get the information with different receptive field.

## III. System Model

The framework of our proposed method to solve the inconsistent feature map localization information is shown in Fig. 2. The process of generating multi-scale feature maps needs to go through multiple convolution layers and merge feature maps, which will change the information inside the feature maps and become inconsistent which will negatively affect the detection performance, resulting in false detection. In our proposed method, we modify the process of generating feature maps in the bottom-up augmentation path to keep the localization information consistent through process aggregation, we add a concatenation process, and we also apply our proposed method called ISC-AM method (improved Spatial Channel Attention Mechanism) to enhance the performance of feature maps, as shown in Fig. 3. For bottom-up and top-down paths, there are no changes, it still uses the same as the original model [8], as shown in Fig. 1 for bottom-up and top-down paths.

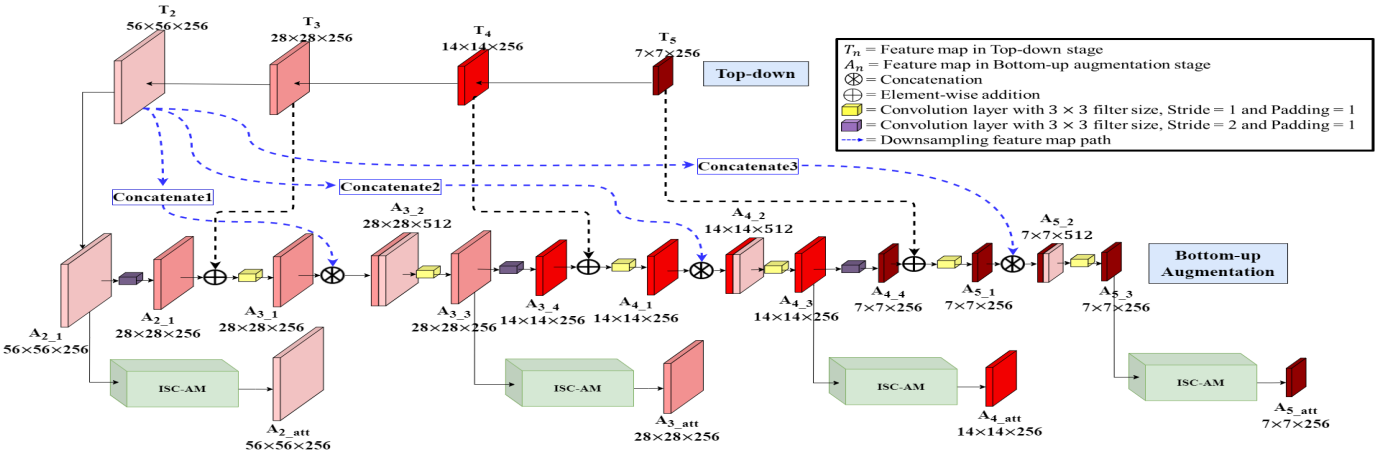


Fig. 2. The proposed model framework. The multi-scale feature map in top-down stage and bottom-up augmentation are denoted by *Tn* and *An*, respectively. The concatenation and element-wise addition processes are denoted as ⊗and ⊕, respectively. The yellow cube box denotes 3x3 convolution layer with stride = 2 and the purple cube with stride = 1. The blue dotted line denotes the downsampling process for feature map with larger dimension and the black dotted line is the element-wise addition for feature map that has the same dimension to merge together.

### *A. Proposed Bottom-Up Path Augmentation*

The bottom-up augmentation path is where we modify the path and implement our proposed method using concatenation, this is to the keep accurate localization information consistent during generating feature maps. The original bottom-up augmentation [8] has some drawback. The generated feature maps contain inconsistent localization information that happens in the bottom-up augmentation path. Localization information is very important as it enables detection models to recognize large instances in images and has better performance to localize object locations in feature maps. The reason why this issue happens is because that the feature map with accurate localization information $T_2$ is only directly aggregated once in multi-scale feature map, as shown in Fig. 1, only $A_2$ directly aggregated from $T_2$, that's why when it generates the next feature map in the bottom-up augmentation path, $T_2$ information becomes inconsistent. So in order to solve this problem and keep the localization information consistent during aggregation and merging process, we modify the path by adding a concatenation process to directly aggregate the localization information.

As shown in Fig. 1, the original method of generating feature maps in bottom-up path augmentation, $T_2$ already has the same dimension size and the number of channels as $A_2$, we can just directly use $T_2$ as the new feature map $A_2$ in the bottom-up augmentation path. To generate feature map $A_3$, it will use $A_2$ and $T_3$ as input, first we need to match the dimension size between $A_2$ with $T_3$ by using convolution layer of size 3x3 with stride = 2, it will halve the dimension size of $A_2$ and will have similar dimension size with $T_3$. Afterward, we merge the two feature maps using the element-wise addition operation to generate a new feature map $A_3$. This process will continue until we obtain the last feature map ($A_5$) in the bottom-up augmentation path. From this bottom-up augmentation path, we can see that the localization information from the feature map $T_2$, as the path goes deeper, it becomes inconsistent with each feature map generated.

To recover the inconsistent information that occurs throughout the process in the bottom-up augmentation path, we modify the path to add an additional process to aggregate localization information directly from feature maps $T_2$. The concatenation operation is used after the merging process is complete, as shown in Fig. 2. After feature maps concatenate, the number of channels needs to be reduced from 512 to 256, so the concatenated feature map will go through a convolution layer with a size of 3x3 and a channel number of 256.

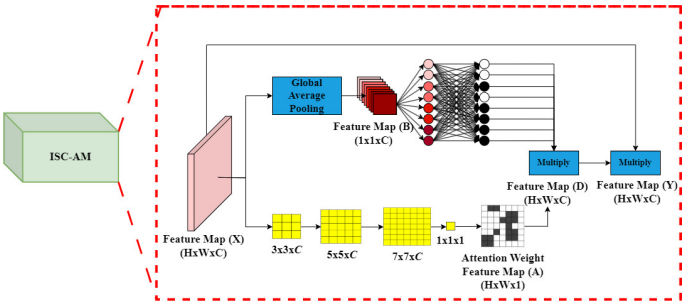


Fig. 3. The ISC-AM framework.

*1) **Downsampling Accurate Localization Feature Map**:* Since the feature maps $T_2$ will be directly aggregated to other feature maps, we have to match the dimension sizes before concatenating to all feature maps in the bottom-up augmentation path. So to concatenate it directly, we need to downsample the feature map to match the dimension size, and to downsample we use a 3x3 convolution with stride = 2, this will halve the dimension size. To concatenate $T_2$ with a smaller feature map, for example, $A_5$, then we must downsample $T_2$ 4 times to match the dimension size with $A_5$.

*2) **Modified Model to Generate Multi-scale Feature Map**:* The bottom-up path to generate multi-scale feature maps is to

take multiple feature maps as input, first step is to match the dimension sizes of the feature maps to make them uniform, and then use the element-wise addition operation to merge and generate a new feature map. As shown in Fig. 1, to generate feature map $A_3$ it needs $A_2$ and $T_3$ as input, the first step is to make feature map $A_2$ dimension size match with $T_3$, so it needs to be upsampled by factor of 2. The modified model also goes through a similar process, but it requires another feature map as input using a concatenation process compared to the original detection model, as shown in Fig. 2, we concatenate the merged feature map $A_3$ with the feature map that contain accurate localization information ($T_2$).

### B. Modified Spatial Channel Attention Mechanism

After the feature map has been generated with consistent accurate localization information, in order to enhance the feature map to improve the detection performance, we apply spatial and channel attention mechanism that has been modified to enhance the feature information in our modified model. For the improved attention mechanism, we can see the framework shown in Fig. 3, our attention mechanism is named ISC-AM. The model consists of 2 attention mechanisms, spatial and channel attention mechanism, for the channel attention mechanism, we adopt the Squeeze-and-Excitation network [11] method.

The spatial attention mechanism, we modified by increasing the number of convolution layer with different sizes. The reason for these multiple convolutions is to expand the receptive field so that it can learn local patch information of different sizes from the input feature map, which will help the attention mechanism to learn the complex pattern in the feature map, so it can recognize objects and put the attention weight better in feature map. As shown in Fig. 3, Given a feature map $X \in \mathbb{R}^{H\times W\times C}$, it will enter 2 different branches and generate 2 feature maps $A$ and $B$, respectively. The first branch is the feature map $A \in \mathbb{R}^{1\times 1\times C}$ for channel attention mechanism using Global Average Pooling , and the second branch is the feature map $B \in \mathbb{R}^{H\times W\times 1}$ for spatial attention mechanism, which is the modified attention mechanism. Both feature maps will have the Sigmoid activation function applied to obtain the attention weight maps for both feature maps, then we perform element-wise multiplication to obtain the feature maps $D \in \mathbb{R}^{H\times W\times C}$. After we get the feature map $D \in \mathbb{R}^{H\times W\times C}$, that is the the attention weight map, and then perform an element-wise multiplication to get the final feature map output $Y \in \mathbb{R}^{H\times W\times C}$. This modified spatial channel attention mechanism is applied to all final feature maps generated in the bottom-up enhancement path.

## IV. Performance Evaluation

In here to evaluate the proposed method, we compare our method using COCO dataset [9]. We present our result using COCO 2017 object detection and instance segmentation challenge.

### A. Implementation Details

For the implementation, we re-implement Mask R-CNN with FPN and Mask R-CNN with PANet, the framework we use is Python, Keras and Tensorflow [12]. We trained the model using pre-trained weights of Mask R-CNN for COCO dataset. There are some differences between the official paper implementation results [7], [8] and the framework we use [12].

For example, in this implementation the learning rates used are different from the original implementation. In the original the learning rate is 0.02, however, in this framework, the learning rate is 0.001, because using 0.02 can be too high, causing the weights to explode, this is might be related to differences how Caffe and Tensorflow compute gradients, in this implementation using smaller learning rates converge faster. For learning momentum and weight decay are still the same as the original implementation, 0.9 and 0.0001 respectively. The model is trained with 160k iteration, at 120k iteration the learning rate is decrease by factor of 10. In the original implementation, 8 GPU were used with 2 images each, resulting in a batch size of 16 images, in this implementation we use only 1 GPU with 2 images, resulting in a batch size of 2 images. This implementation difference causes the results to differ from the original paper.

### B. Experiments on COCO Dataset

*1) **Datasets**:* The implementation model is trained using the COCO dataset 2017 [9]. The dataset contains a total of 118k images for training and 5k images for validation and has 80 classes annotated with pixel-wise instance masks.

*2) **Metrics evaluation**:* To evaluate detection results, we use standard evaluation metrics, *i.e.,*$AP$, $AP_{50}$, $AP_{75}$, $AP_S$, $AP_M$ and $AP_L$. The $AP$ stands for Average Precision, the numbers above the $AP$ represent IoU, and the uppercase letters above $AP$ represent the object scale size, for example, $AP_S$ represents the average precision of small objects. We also evaluate the performance on $AR$ (Average Recall) and F1 score.

*3) **Detection Results Comparison**:* For detection result comparison, because we implement our method using Mask R-CNN as the detection model, we compare the object detection and instance segmentation outputs of Mask R-CNN. We also compare the two outputs using statistical results and visual result comparisons.

### C. Visualization Result Comparison

We also show visual detection results for instance segmentation using Mask R-CNN with the PANet model PANet and our two proposed methods. Visual detection includes 4 parts detection (a, b, c, and d), the first part is a ground-truth image (a), the second (b), third (c), and fourth parts (d) are the detection results of PANet, and methods with and without attention mechanism, respectively. There are 3 visual comparisons, the first part shown in Fig. 4 shows the comparison where our proposed method without the attention mechanism performs better, and the second part shown in Fig. 5 shows our proposed method with attention mechanism

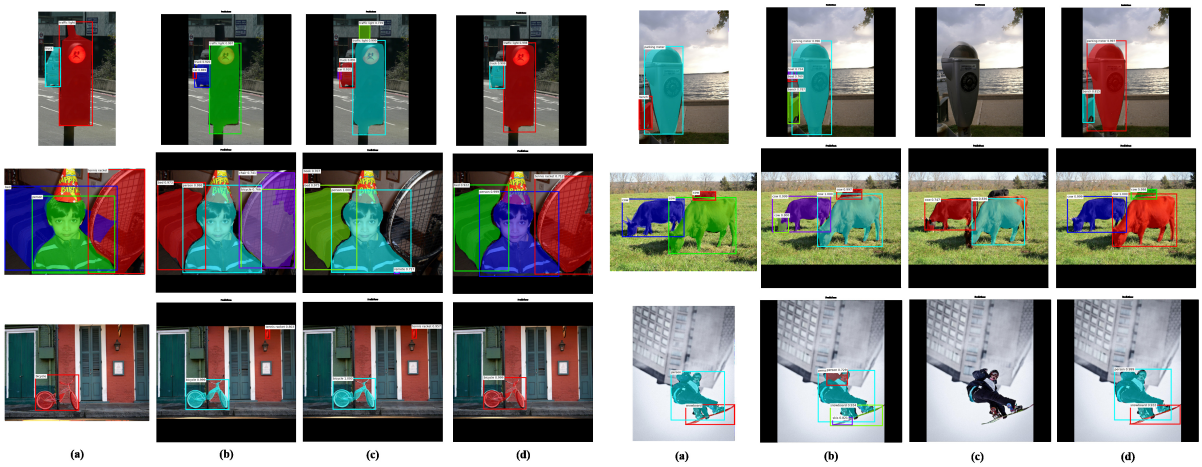


Fig. 4. Comparison of detection with PANet. The method without attention mechanism performs better as compared to PANet and proposed method + attention mechanism.

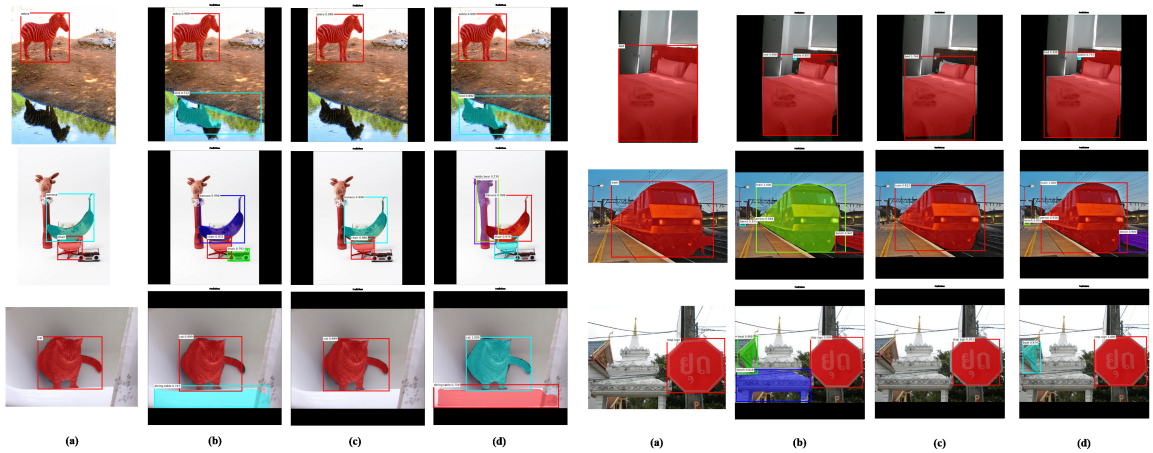


Fig. 5. Comparison of detection with PANet. The method + attention mechanism performs better as compared to PANet and proposed method without attention mechanism.

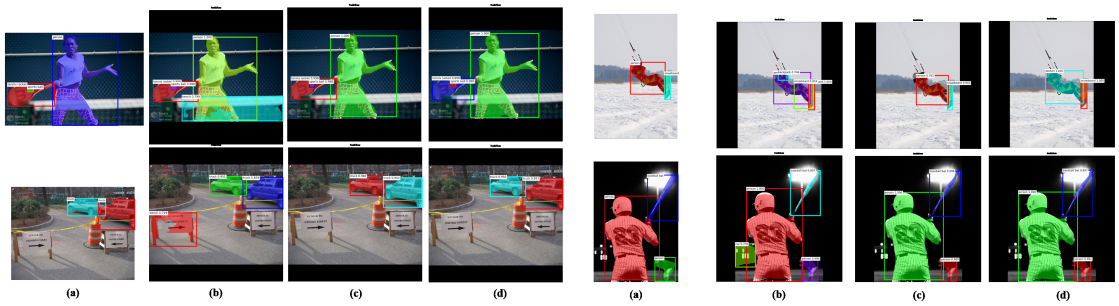


Fig. 6. Comparison of detection with PANet. Both proposed methods perform better as compared to PANet.

TABLE I
PERFORMANCE OF OBJECT DETECTION BASED ON COCO DATASET *val-2017* SUBSET IN TERMS OF AP.

| Method | $AP$ | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ | $AR$ | $AR_{50}$ | F1-Score |
|---|---|---|---|---|---|---|---|---|---|
| Mask R-CNN + FPN [7] | 0.189 | 0.311 | 0.205 | 0.067 | 0.210 | 0.311 | 0.254 | 0.382 | 0.208 |
| Mask R-CNN + PANet [8] | 0.344 | 0.540 | 0.375 | 0.164 | 0.384 | **0.483** | 0.435 | 0.639 | 0.384 |
| Mask R-CNN + Our Method | **0.347** | **0.544** | **0.381** | 0.171 | **0.391** | **0.483** | 0.437 | **0.641** | 0.386 |
| Mask R-CNN + Our Method + attention | **0.347** | 0.542 | 0.378 | **0.172** | 0.390 | 0.479 | **0.438** | **0.641** | **0.387** |

TABLE II
INSTANCE SEGMENTATION PERFORMANCE COMPARISON ON COCO DATASET *val-2017* SUBSET IN TERMS OF AP.

| Method | $AP$ | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ | $AR$ | $AR_{50}$ | F1-Score |
|---|---|---|---|---|---|---|---|---|---|
| Mask R-CNN + FPN [7] | 0.151 | 0.256 | 0.160 | 0.052 | 0.173 | 0.242 | 0.168 | 0.326 | 0.201 |
| Mask R-CNN + PANet [8] | 0.302 | 0.504 | 0.320 | 0.132 | 0.337 | **0.444** | 0.388 | 0.604 | 0.340 |
| Mask R-CNN + Our Method | **0.307** | **0.511** | **0.324** | **0.142** | **0.344** | **0.444** | 0.392 | **0.607** | **0.344** |
| Mask R-CNN + Our Method + Attention | 0.306 | 0.508 | 0.323 | 0.140 | 0.342 | 0.443 | **0.393** | 0.606 | **0.344** |

perform better, and the last part of the visual comparison is shown in Fig. 6, shows that both of our proposed methods perform better. Each comparison uses 3 different images to compare the visual result comparison.

### *D. Object Detection Statistical Result*

The object detection performance between the baseline detection model and our method is shown in the Table. I, in this table, describes the statistical performance of $AP$, $AR$ and F1-score. Based on this result, if we compare it, we can see that the proposed method has better performance than the baseline models FPN and PANet in the Table. I, as we mentioned, we are using a different framework, and the results we get for the two baseline models are not similar to the original implementation results, especially for FPN, which has very low results in this object detection performance comparison. Overall, our proposed method is able to improve the performance of object detection by keeping accurate localization information aggregation consistent, especially for small and medium object sizes, however, If we compare the object detection result $AP$ for large objects, PANet has a better performance compared to our proposed method with attention mechanism, but other than that, both of our proposed methods perform better.

### *E. Instance Segmentation Statistical Result*

The instance segmentation of both of our proposed method and 2 baseline detection models are shown in the Table. II, according to the results, if we compare the instance segmentation results $AP$ of large objects, comparing with our proposed method with attention mechanism, PANet has better performance, but other than that, both of our proposed methods perform better.

## V. CONCLUSION

In this paper, a detection model is proposed to solve the inconsistency information issues during aggregation in multi-scale network to generate the feature maps. Our proposed method can improve the detection performance by keeping the consistency of the accurate localization information through the process of generating feature maps using concatenation. A modified spatial channel attention mechanism is also proposed to enhance the feature maps to recognize more objects after aggregating correct localization information to the feature maps. Both of our proposed method can improve the detection model. The first model is able to solve the inconsistency information issue that happens when generating feature maps and the second model is able to enhance the feature maps. Based on visualization results, both proposed models are able to detect the objects more accurately as compared to FPN and PANet.

## REFERENCES


[1] D. D. Onthoni, T.-W. Sheng, P. K. Sahoo, L.-J. Wang, and P. Gupta, "Deep learning assisted localization of polycystic kidney on contrast-enhanced ct images," *Diagnostics*, vol. 10, no. 12, p. 1113, 2020.

[2] Y.-C. Liu, D. D. Onthoni, S. Mohapatra, D. Irianti, and P. K. Sahoo, "Deep-learning-assisted multi-dish food recognition application for dietary intake reporting," *Electronics*, vol. 11, no. 10, p. 1626, 2022.

[3] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, vol. 1. Ieee, 2001, pp. I–I.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[7] 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.

[8] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[10] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*. PMLR, 2015, pp. 2048–2057.

[11] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[12] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," https://github.com/matterport/Mask_RCNN, 2017.