DSP R-CNN: Direct Set Prediction Region with CNN features

Kwan-Yong Park
Department of Industrial and Management Engineering
Korea University
Seoul, South Korea
gamja331@korea.ac.kr

Jun-Geol Baek*
Department of Industrial and Management Engineering
Korea University
Seoul, South Korea
jungeol@korea.ac.kr

***Abstract*— Convolutional Neural Network (CNN) based object detection models struggle with differentiating objects from the background and with the separation and global interaction among multiple objects within an image. As a result, accurately capturing the location of objects in such images necessitates the use of the Region Proposal Network (RPN) structure. However, RPNs present several challenges in terms of performance and efficiency. This situation has led to an increasing focus on research in Transformer-based object detection models. While these Transformer-based models improve performance over their predecessors, they often compromise efficiency in terms of speed and training duration. The proposed method introduces a novel approach that interprets the channels of the feature map as compressed objects, fundamentally transforming the CNN paradigm by eliminating the need for Region Proposal in CNN-based object detection architectures. Utilizing a one-to-one matching function, it turns object detection into a direct prediction problem. Moreover, the DSP R-CNN model, developed from this method, streamlines the pipeline by dispensing with heuristic elements like Non-Maximum Suppression (NMS) and anchor box generation. The experiments on Circular pipe dataset show that this approach achieves higher accuracy and faster performance compared to the widely used CNN-based model Faster R-CNN and Transformer-based model DETR in the field of object detection.**

Keywords*— *Object Detection, Direct Set Prediction, Region Proposal Network, Convolutional Neural Network

I. INTRODUCTION

Object detection is an automated technique for distinguishing and identifying objects from the background in images. Object detection is increasingly being implemented in various fields including disease identification in the medical sector, fault detection in the manufacturing industry, military and robot vision [1]. With the miniaturization and diversification of objects, the importance of object detection is growing. Consequently, extensive research has been conducted in this area, with studies like Faster R-CNN, YOLO, and Mask R-CNN extracting features from objects based on Convolutional Neural Networks (CNN) specialized in image processing [2, 3, 4]. CNN-based object detection models require the utilization of a Region Proposal Network (RPN), which serves to indirectly identify areas that are likely to contain objects [5, 6, 7]. The reason is that in situations where multiple objects coexist within an image, it becomes challenging to simultaneously consider the separation and interaction among these objects. Therefore, without an RPN, CNNs might lack the necessary information for accurately predicting the location and size of each object.

However, the indirect capture of target areas by RPN has led to several issues. Firstly, there is a decrease in the flexibility of object detection models. RPN heavily relies on hyperparameters such as the size and ratio of candidate regions, and the size of Grid Cells. Secondly, there is a loss of information. Discrepancies exist between the target areas passed to the CNN and the actual areas. Furthermore, in the process of passing target areas to the CNN, roi pooling, which is responsible for size adjustment, results in information loss. Thirdly, there is a decrease in model efficiency. The training process of RPN involves encompassing multiple target areas, resulting in inefficiencies. Additionally, the post-processing manual task of Non-Maximum Suppression (NMS) is necessary to eliminate overlapping target areas. Moreover, since candidate regions generated heuristically often correspond to locations without objects, unnecessary elements are included during training when comparing these candidate regions with actual areas.

Recently, due to the limitations of RPN, There has been a need for models that compare actual areas with predicted areas directly, rather than indirectly generated target areas. However, it remains challenging to simultaneously consider the separation and interaction of different objects without the RPN structure [8]. In response, many studies have endeavored to address this issue by employing the Transformer architecture, which features an encoder-decoder structure [9, 10, 11, 12]. The reason Transformers can eliminate the RPN is due to their capability to explicitly model interactions between elements through self-attention, thereby enabling interactions among separated objects. That is, during the presence of multiple objects in images and videos, the encoder facilitates the separation among objects, while the decoder allows components within individual objects to interact, adjusting the size and position of the predicted bounding boxes.

While Transformers have resolved the issues of the RPN structure, their self-attention operation renders them incapable of capturing the locality features of objects, making the detection of smaller objects unfeasible. Furthermore, the Transformer's structure, which has a low inductive bias, requires a large volume of image data during the model's convergence process and necessitates a lengthy training period, presenting a

*Corresponding author—Tel: +82-2-3290-3396; Fax: +82-2-3290-4550 limitation in its application for large-scale applications [13]. Therefore, to improve the time efficiency in the field of object detection, the application of CNN-based models is necessary.

The proposed method aims to enable object detection using CNN-based models without the RPN. The proposed method applies bipartite matching between the predicted and actual areas in the Transformer-based object detection model, Detection Transformer [10]. This study proposes the Direct Set Prediction Region with CNN features (DSP R-CNN), a method designed to eliminate the RPN in CNN-based object detection models by interpreting the channels of the feature map as objects.

Chapter 2 describes the theoretical background necessary for understanding the proposed model. Chapter 3 describes the detailed information about the proposed DSP R-CNN model. Chapter 4 describes the results of comparative experiments between Faster R-CNN and DETR. Finally, Chapter 5 describes the conclusions.

II. Related Works

A. *Region Proposal Network*

Fig 1 shows the mechanism of the Region Proposal Network (RPN). The RPN generates candidate regions, or anchor boxes, of various sizes and ratios for each grid cell, as depicted on the right side of Fig 1. It then compares these target boxes with the actual object areas (Ground-Truth, GT), thereby effectively pinpointing potential object locations to be forwarded to the CNN. CNN-based object detection models leverage this mechanism by separating objects at the RPN phase and facilitating interaction among these segregated objects during the CNN phase.

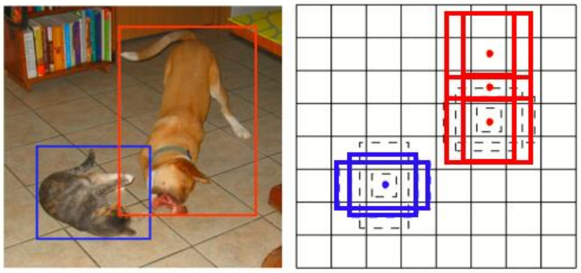


Fig 1. How RPN works

B. *Convolutional Block Attention Module*

Convolutional block attention module (CBAM) is a simple and effective attention module for feed forward CNNs [14]. Fig 1 shows the structure of CBAM. It sequentially infers attention maps along two separate dimensions, channel and spatial, from the feature map, and then multiplies each attention map by the input feature map. It allows for the establishment of a global reasoning framework across channels and supports the interaction of local information.

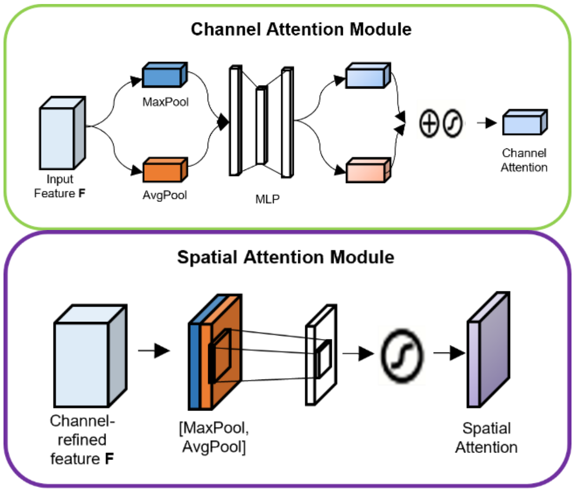


Fig 2. CBAM Structure

C. *Hungarian Matching Algorithm*

The Hungarian Algorithm is a methodology that solves assignment problems by finding all possible pairs in a bipartite graph that connects two independent groups, and then identifying the connection with the maximum weight [15]. When there are two sets of nodes, I and J, the cost incurred for I to process J is denoted as $c(i,j)$. Fig 3 shows a complete bipartite graph defined by two sets of nodes and the costs associated with their edges.

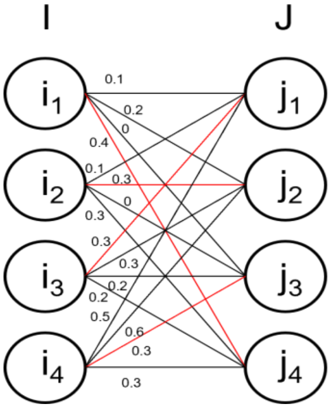


Fig 3. CBAM Structure

As shown in Equation (1), the technique seeks to find a perfect matching M in the complete bipartite graph that minimizes the cost.

$$L_{complte\ graph} = \sum_{(i,j)\in M} c(i,j) \tag{1}$$

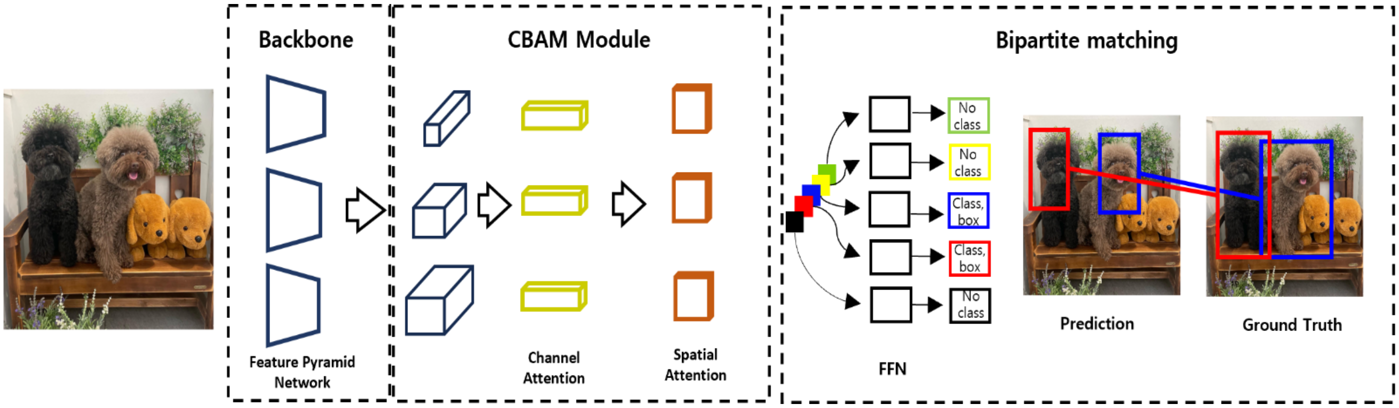


Fig 4. DSP R-CNN Structure

III. Methodology

In this chapter, we introduce the DSP R-CNN, an efficient training pipeline that achieves direct set construction between actual and predicted areas, thereby eliminating the RPN in CNN-based object detection methods. Fig 4 shows the overall structure of the proposed method. The DSP R-CNN consists of four main processes for training. (1) Due to the nature of CNN-based object detection models, it is not possible to separate objects or design global and local interactions within objects in a single stage. Therefore, features of objects are extracted using the Darknet-53 backbone structure, fine-tuned for multi-label classification, to segregate objects within the image [3]. (2) A CBAM module is configured to design global and local interactions within separated individual objects. (3) A Feed Forward Convolution Network (FFCN) comprised of 1x1 convolutions is established to summarize the compressed information of objects into class information and locations. (4) The predicted objects are matched one-to-one with the actual objects using the loss function represented in Equation (3). Subsequently, the model is trained to adjust the regions of predicted objects using the Hungarian loss function, as detailed in Equation (4).

A. *Backbone for the separation of objects*

In CNN-based object detection models, the model for the separation of objects is a crucial step for constructing direct set predictions, excluding the RPN. In this study, Darknet53 is utilized. When an initial image $3\times H0\times W0$ (with three channels) is inputted, it passes through Darknet53 and returns a feature map f with lower resolution $C\times H\times W$. For the extraction of object regions and features within the image, Cross Entropy Loss is used for Multi-Label Classification, as detailed in Equation (2). p_c represents the predicted probability for individual classes, and y_c indicates whether the corresponding class exists in the image, being either 0 or 1. Thus, the Cross Entropy Loss decreases as the predicted probability increases for the classes present in the image.

$$L_{cross-entropy} = -\frac{1}{N}\sum_{c=1}^{N} y_c \log(p_c) \qquad (2)$$

B. *Channel & spatial attention module for object interaction*

CNN-based object detection models have had the issue of being unable to consider interactions between objects without an RPN. Although fully connected layers enable these interactions, they can lead to the loss of positional information. Therefore, the RPN structure, which captures the location of prediction areas, is necessary. However, the RPN reduces the flexibility and efficiency of the model. This study aims to resolve this by recognizing feature maps as objects through the CBAM module. Images processed through Darknet53 result in feature maps of size N * H * W (where N is the number of channels). Each channel contains compressed information of specific objects, treating individual channels as objects. These N compressed objects facilitate more effective separation within the image through channel attention and allow smoother interactions between objects. Additionally, spatial attention within objects suggests their prediction areas. The compressed channels (objects), having completed both inter-object and intra-object interactions, are then input into a feed-forward 1x1 convolution network.

C. *Feed Forward 1x1 Convolution Network*

The final predictions are computed by three layers of 1x1 Convolution and ReLU activation functions. The Feed Forward Convolution Network (FFCN) predicts normalized central coordinates, height, and width of the prediction boxes from the input image and uses the softmax function to predict class labels. Typically, N predicts a fixed number of objects larger than the actual number of objects in the image. Therefore, an additional class label θ, representing 'background,' is used to indicate the absence of detected objects within a slot [10]. The information of the N compressed objects is independently decoded into box coordinates and class labels, resulting in N final predictions.

D. *Loss Function*

A primary challenge during the training of traditional CNN-based object detection models has been comparing target objects with all predicted objects to infer their location and class. As a solution, DSP R-CNN maximizes training efficiency by generating bipartite (one-to-one) matching between the actual objects and a fixed number of N predicted objects. The function for bipartite matching is as shown in Equation (3) [10].

$$\widehat{\sigma} = \sum_{i}^{N} L_{match}\left(y_i, \hat{y}_{\sigma(i)}\right) \tag{3}$$

In Equation (3), y represents the set of actual objects, and ŷ represents the set of N predicted objects. The process involves searching all possible permutations σ to find the pairing $\hat{y}_{\sigma(i)}$ that minimizes the match L_{match} loss function between the two sets. The matching cost(L_{match}) sequentially calculates the similarity between the N predicted classes and areas to the actual object y_i. In the actual object y_i = (c_i, x, z, w, h), c_i denotes the class label of each object. The class label can be an empty set, indicating the background. There are only N actual objects, similar to predicted objects, and those where c_i is an empty set do not get matched with predicted objects having an empty set for c_i and are excluded from training. In y_i x, z, w, h define the vector for the object's central coordinates, height, and width. In a specific pairing σ(i), when the predicted class is correct and the positional coordinates are $\hat{p}_{\sigma(i)}(ci)$, $\hat{b}_{\sigma(i)}(ci)$ the optimal pairing $\widehat{\sigma}$ and its matching cost are calculated as shown in Equation (4) [10].

$$L_{Match} = -1\{ci \ \neq 0\}\hat{p}_{\sigma(i)}(c_i) + 1\{ci \neq 0\}\ L_{box}\left(b_i, \hat{b}_{\delta(\ddot{i})}\right)) \tag{4}$$

In Equation (4), L_{box} is a formula that calculates the difference between two matched boxes. Once the optimal matching is found through the matching cost, it is optimized as per Equation (5) by a linear combination of the negative log-likelihood loss between matches and the loss of the prediction area [10].

$$L_{Hungarian}(y, \hat{y}) = \sum_{i=1}^{N} [-\log\hat{p}_{\widehat{\sigma}(i)}(c_i) + 1_{\{c_i \neq \emptyset\}} L_{box}(b_i, \hat{b}_{\widehat{\sigma}}(i))] \tag{2}$$

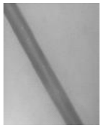
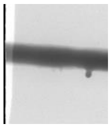
In the predicted pairings, if the class (Class) is 'no object,' meaning c_i is an empty set, it is not included in the loss function. Finally, L_{box} in Equation (4) and Equation (5)'s $L_{Hungarian}$, which includes L_{box}, is defined as per Equation (6) as a loss function that calculates the difference between the locations of the actual and predicted objects. L_{box} utilizes a normalized (between 0 and 1) L1 Loss and a scale-invariant generalized Intersection over Union (gIoU) loss.

IV. Experimental Results

A. Dataset

The experiments utilized circular pipe data from the manufacturing industry, specifically for the transportation of fluid materials (Caesar, et al., 2018; Yang, et al., 2021). Circular pipes refer to pipelines used for the transportation of fluid materials in various industries, including Oil/Gas and Chemicals/Petrochemicals. The circular pipe data, by its nature, may present class imbalances, and the data used in the experiments have class types and proportions as shown in <Table 1> [16].

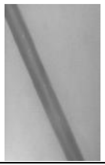
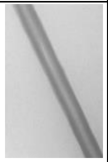
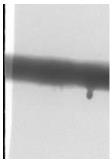
Table 1. Circular tube data

Class	Air-hole	Crack	Overlap	Unfused
Image				
개수	2051	119	219	408

B. Experimental setting

To address class imbalances in the circular pipe data, oversampling methods that do not distort the objects, such as Rotation, Gaussian Noise, Color Distortion, Resize, Random Flip, and Shearing, were used to augment the data. After data augmentation, the number of objects per class is as shown in <Table 2>.

Table 2. Number of objects per class by data augmentation

Class	Air-hole	Crack	Overlap	Unfused
Image				
Number	2051	1190	1095	1224

The data for training, validation, and testing are divided in a 6:2:2 ratio, as detailed in <Table 3>. The distribution for training, validation, and testing purposes follows the same 6:2:2 ratio, as indicated in <Table 3>.

Table3. Number of objects separated by Train, Val and Test

	Air-hole	Crack	Overlap	Unfused
Numbe r	2,051	1,190	1,095	1,224
Train (60%)	1,231	714	657	734
Val (20%)	410	238	219	195

C. Evaluation Metrics

To compare the performance of the models, metrics such as Average Precision (AP), Training Time, and Inference Time were used. AP is an evaluation metric used to assess the performance of object detection algorithms and corresponds to the area under the Precision-Recall curve. Precision and Recall are calculated based on the true and false classification results shown in <Table 5>. Precision is determined, as per Equation (7), by the ratio of correctly predicted objects among all predicted objects. Recall is calculated, as per Equation (8), by the ratio of correctly predicted objects among the actual objects. After calculating Precision and Recall, the area under the Precision-Recall Curve is computed to determine the AP value. However, since the ratio of Precision and Recall varies depending on the Intersection Over Union (IOU), which indicates the degree of overlap between the actual and predicted areas, object detection research commonly compares AP values based on IOU thresholds of 0.5 and 0.75.

Table 5. Confusion Matrix

	Actual Positive	Actual Negative
Predicted Positive	TP (True Positive)	FP (False Positive)
Predicted Negative	FN (False Negative)	TN (True Negative)

$$\text{Precision} = \frac{\text{TP}}{\text{TP+ FP}} \tag{7}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{8}$$

D. *Performance of The Proposed Method*

The experiments conducted various comparative tests based on the different sizes of objects and the set IOU values. For the circular pipe data, the Air-hole class was categorized as Small, the Crack class as Medium, and the Overlap and Unfused classes as Large Data. As shown in <Table 6> and <Table 7>, APS, APM, and APL represent the AP values calculated for small, medium, and large data, respectively.

Table 6. Circular tube data AP Comparison

model	AP	AP50	APs	APm	APL
F-RCNN	63.0	69.5	**47.7**	61.2	77.5
DETR	32.8	44.3	22.3	48.4	50.4
Ours	**67.3**	**74.2**	40.2	**69.9**	**81.7**

AP represent the results when the IOU is set to 0.75, and AP50 denotes the results with an IOU of 0.5. In both datasets, DSP R-CNN demonstrates superior performance in AP across different IOUs. However, as indicated in <Table 6>, there is a performance decline in Small data, attributed to Darknet-53's inability to effectively detect small objects. It is also noteworthy that DETR exhibited significantly lower performance with circular pipe data, due to the Transformer structure requiring a substantial amount of data and iterations for convergence. <Table 8> and <Table 9> present the experimental results from a temporal perspective.

Table 8. Circular tube data Training, Inference Time

Model	Training Time	Inference Time (FPS)
F-RCNN	**1.17 Hours**	12
DETR	2.0 Hours	15
Ours	1.33 Hours	**20**

The training time was longer than Faster R-CNN due to the additional convolution layers required to design Channel and Spatial Attention. Yet, the DSP R-CNN model, which eliminates the RPN, shows remarkably higher performance in Inference Time.

<Figure 8> displays the results of inferring circular pipe data using the DSP R-CNN model.

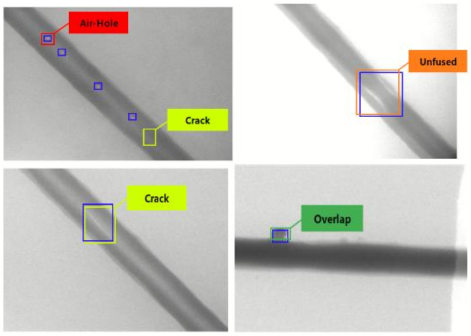


Figure 8. Circular tube data inference result – the blue box represents the actual area.

Furthermore, the performance is compared based on the CBAM module. The dataset used was circular pipe data. <Figure 9> and <Figure 10> display the performance comparison of AP50 and APS according to the number of CBAM modules.

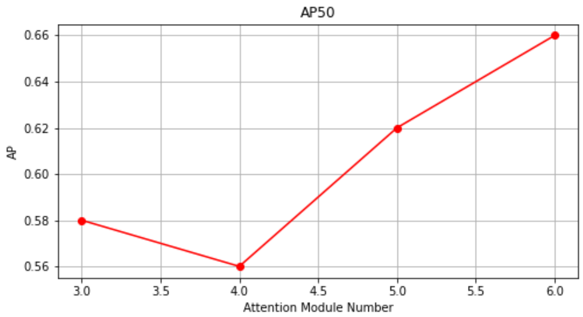


Figure 9. AP50 performance change according to the number of CBAM modules

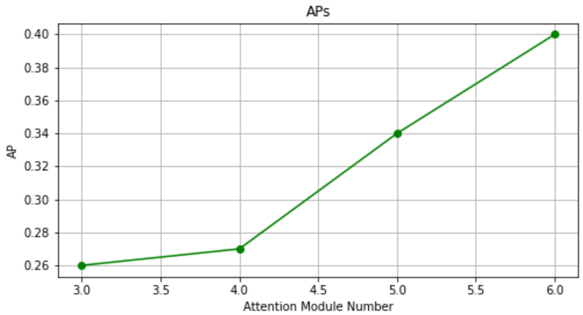


Figure 10. APs performance change according to the number of CBAM modules

The experimental results indicate that the performance increases with the addition of CBAM modules. This suggests that the channel attention and spatial attention in CBAM facilitate smoother interactions between objects. Notably, an increase in the number of CBAM modules showed an improved performance range, especially in smaller-sized data. This implies that spatial attention, being a global operation, has addressed the limitation of traditional vanilla attention that was unable to detect smaller objects.

Small, medium, and large data, respectively. The AP values represent the results when the IOU is set to 0.75, and AP50 denotes the results with an IOU of 0.5. In both datasets, DSP R-CNN demonstrates superior performance in AP across different IOUs. However, as indicated in <Table 6>, there is a performance decline in small data, attributed to Darknet-53's inability to effectively detect small objects. It is also noteworthy that DETR exhibited significantly lower performance with circular pipe data, due to the Transformer structure requiring a substantial amount of data and iterations for convergence. <Table 8> and <Table 9> present the experimental results from a temporal perspective. The training time was longer than Faster R-CNN due to the additional convolution layers required to design Channel and Spatial Attention. Yet, the DSP R-CNN model, which eliminates the RPN, shows remarkably higher performance in Inference Time. <Figure 8> displays the results of inferring circular pipe data using the DSP R-CNN model.

V. Conclusion

This study proposes the first direct object detection model based on the CNN model structure and bipartite matching loss. This approach has shown similar or improved performance and faster training and inference speeds compared to Faster R-CNN and DETR in real-world circular pipe data. Particularly, the traditional DETR has been known to suffer from reduced performance in detecting small objects due to its global attention operation. The proposed method in this research addresses this by employing spatial attention operations, thus enhancing the detection performance for small objects. The backbone model, Darknet-53, plays a crucial role in separating objects. The performance of object detection models can be significantly influenced by the classification capabilities of the backbone, and there is a notable variation in performance based on the number of CBAM modules. Therefore, future research will focus on performance comparisons contingent on changes in the backbone and the number of modules.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2022R1A2C2004457). This work was also supported by Samsung Electronics Co., Ltd. (IO201210-07929-01) and Brain Korea 21 FOUR.

References

[1] P. H.L, “A Survey on Moving Object Detection and Tracking Techniques,” International Journal Of Engineering And Computer Science, Apr. 2016.

[2] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, Jun. 2017

[3] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement.,” arXiv (Cornell University), Apr. 2018.[4]K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1, 2018.

[4] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1, 2018.

[5] Y. Kim, “Robust Selective Search,” ACM SIGIR Forum, vol. 52, no. 1, pp. 170–171, Jan. 2019.

[6] T. Kong, A. Yao, Y. Chen, and F. Sun, “HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection,” IEEE Xplore, Jun. 01, 2016. https://ieeexplore.ieee.org/document/7780467 (accessed Jan. 26, 2022).

[7] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, “Selective Search for Object Recognition,” International Journal of Computer Vision, vol. 104, no. 2, pp. 154–171, Apr. 2013.

[8] G. Brazil and X. Liu, “M3D-RPN: Monocular 3D Region Proposal Network for Object Detection,” Oct. 2019.

[9] K. He, X. Chen, S. Xie, Y. Li, Piotr Dollár, and R. Girshick, “Masked Autoencoders Are Scalable Vision Learners,” Nov. 2021.

[10] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and Sergey Zagoruyko, “End-to-End Object Detection with Transformers,” arXiv (Cornell University), May 2020.

[11] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, “Learning RoI Transformer for Oriented Object Detection in Aerial Images,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2019.

[12] Dai Zhigang, B. Cai, Y. Lin, and J. Chen, “UP-DETR: Unsupervised Pre-training for Object Detection with Transformers,” Computer Vision and Pattern Recognition, Jun. 2021.

[13] Alexey Dosovitskiy et al., “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” arXiv (Cornell University), Oct. 2020.

[14] S. Woo, J. Park, J.-Y. Lee, and In So Kweon, “CBAM: Convolutional Block Attention Module,” Jul. 2018.

[15] A. Frank, “On Kuhn’s Hungarian Method?A tribute from Hungary,” Naval Research Logistics, vol. 52, no. 1, pp. 2–5, Feb. 2005.

[16] D. Yang, Y. Cui, Z. Yu, and H. Yuan, “Deep Learning Based Steel Pipe Weld Defect Detection,” Applied Artificial Intelligence, pp. 1–13, Sep. 2021.