

# Face recognition with occluded face using improve intersection over union of region proposal network on Mask region convolutional neural network

Rahmat Budiarsa<sup>1</sup>, Retantyo Wardoyo<sup>2</sup>, Aina Musdholifah<sup>2</sup>

<sup>1</sup>Doctoral Program Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia

<sup>2</sup>Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia

## Article Info

### Article history:

Received Dec 6, 2023

Revised Jan 20, 2024

Accepted Jan 26, 2024

### Keywords:

Box regressor

Face recognition

Intersection over union

Mask region convolutional neural network

Occluded face

## ABSTRACT

Face recognition entails detecting and identifying facial attributes. Mask region convolutional neural network (R-CNN) method is a prominent approach, while prior research predominantly delved into refining loss functions and perfecting object and face detection, recognizing, and identifying faces using imperfect data remained relatively unexplored. This study focuses on an occluded dataset comprising Indonesian faces, wherein 'occluded' denotes facial data that lacks complete visibility-encompassing instances where objects obscure faces or are partially cropped. This investigation involves a deliberate experiment that tailors the intersection over union (IoU) of the region proposal network (RPN) to suit the nuances of occluded Indonesian faces, thereby augmenting accuracy in recognition and segmentation tasks. The innovation IoU in the strategic utilization of Anchors, which involves the exclusion of anchors falling beyond the image borders to optimize computational efficiency. The outcomes of this research are striking; it showcases a remarkable 14.75%, 10.9%, and 12.97% surge based on mean average precision (mAP), mean average recall (mAR), and F1-Scores compared to the conventional Mask R-CNN approach. Notably, our proposed model elevates the average accuracy by 10% to 15% and decreases running time by 21%, a noteworthy enhancement compared to the preceding model. This progress is substantiated by validation utilizing 300 instances dataset, reinforcing the robustness of our approach.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## Corresponding Author:

Retantyo Wardoyo

Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science,

Universitas Gadjah Mada

Building C, 4<sup>th</sup> Floor, Sekip Utara, Bulaksumur, Senolowo, Sinduadi, Mlati District, Sleman Regency,

Yogyakarta Special Region 55281, Indonesia

Email: rw@ugm.ac.id

## 1. INTRODUCTION

Face recognition is a technology that can be used to identify a person's face for various purposes, by comparing a facial image with a facial database and finding the most suitable facial database. Facial recognition technology can be utilized, one of which is in the identification process. Face recognition is a combination of the detection process and the identification process. Face detection plays a role in the face localization process, namely the process of finding the size and position of a face in an image, while the Face identification process determines the class of the object.

Face recognition is one of the computer vision domains. Face recognition is never apart from detection [1]–[3] and identification of facial objects [4]–[6]. Most of the researchers conduct research on face recognition based on multi-objects or in the input there are many faces, the problem with research usually lies in the detection which sometimes cannot detect many facial objects in one frame [7]–[9]. Objects that cannot be detected are mostly because information from these objects cannot be perfectly identified by a computer [10]–[12] image data that does not have full information is called an occluded image.

The object being lifted is a face image (occlusion of Indonesian face) that only displays facial information that does not have full information, for example, a face image taken at an angle that does not show all facial images or a facial image that is blocked by other objects such as masks. Objects with facial images (occlusion) that do not have all the facial image information can be due to acquisition from different angles or taking pictures of faces from different angles. This object was raised because it can affect detection and identification performance in facial images [2], [3], [12]. In addition, research in the field of face recognition can have a positive impact, such as in terms of helping search, security, and identification of a person's data through image input. This research can also help many parties in facial recognition with little facial image information.

The development of methods in computer vision cannot be separated from research conducted by researchers in the field of deep learning such as the region convolutional neural network (R-CNN) [13] method, Fast R-CNN [14], Faster R-CNN [15] and fully convolutional network (FCN) [10]. R-CNN [13] method is a development of convolutional neural network (CNN), where there is the addition of a region process which is useful for specializing object calculations in an image or frame. Fast R-CNN [14] develops R-CNN by only performing one CNN calculation. Faster R-CNN [15] develops the previous method by eliminating the process of external region proposals method and carrying out the region proposal network (RPN) process concurrently with CNN.

The newest deep learning method for object segmentation is Mask R-CNN. Mask R-CNN [11] can adapt human abilities in classifying object segmentation such as cars, animals, and humans. Mask R-CNN uses instance-aware semantic segmentation to label or classify objects, so it is able to distinguish objects in the same class. Mask R-CNN extends Faster R-CNN [15] by adding branches to predict mask branches in parallel with existing branches for bounding box regression and classification (recognition).

This research uses the Mask R-CNN method because this method is proven to have good performance in object detection and recognition [3], [11], [16]. In addition, this method can distinguish the same object in one image with instance-aware semantic segmentation or FCN. Mask R-CNN also uses the RPN to determine the location and shape of objects in the image.

Research using Mask R-CNN [1], [3], [11] has an accuracy of 60% to 66%. We conducted research using the Mask R-CNN method for images occluded of Indonesian faces with 60% to 65% successfully detected, and around 35% to 40% not successfully detected. To overcome this problem, we conducted an intersection over union (IoU) of RPN modification experiment on Mask R-CNN to increase accuracy. Our research provides an accuracy of 14.75%, 10.9%, and 12.97% based on mean average precision (mAP), mean average recall (mAR), and F1-Scores higher than the original Mask R-CNN.

## 2. RESEARCH METHOD

This literature study was based on several articles retrieved from [www.scopus.com](http://www.scopus.com). These articles describe research that has been done on face detection and facial recognition. This section explains the articles used as references and the architectural models used as research methods.

### 2.1. Selection stage

Face recognition is one of the computer vision domains. Face recognition is never separated from detection [1]–[3] and identification of facial objects [4]–[6]. Most researchers conduct face recognition research on a multi-object basis, the problem with research usually lies in the detection which sometimes cannot detect many facial objects in one frame [7]–[9]. Objects that cannot be detected are mostly because the information from the object cannot be perfectly received by the computer, or in other words, the object is not all visible or only part of it [1], [10], [11].

The development of the first CNN method was named region convolutional neural networks (R-CNN) [13]. R-CNN is carried out by searching for regions or parts of images that can be objects, using the proposal region method, then each region will have a CNN for feature extraction. From this R-CNN research, other methods were obtained such as Fast R-CNN [14], Faster R-CNN [15], and the latest is Mask R-CNN [11] where this method combines Faster R-CNN by adding a new branch called Mask to perform segmentation and called FCN [10], [17].

Mask R-CNN [11] is a method that uses instance-aware semantic segmentation to label segmentation or classify objects so that this method can distinguish objects/instances within the same class.

mask R-CNN is usually used for detection [16], [18]–[22] and recognition [23]–[27] of objects. Mask R-CNN extends Faster R-CNN by adding a new branch for predicting mask objects in parallel with the existing branch for bounding box recognition. The segmentation on Mask R-CNN [11] is FCN [10] which uses a convolutional neural network to convert image pixels into pixel categories. The FCN changes the height and width of the middle layer features by remapping to the size of the input image via a transformed convolution layer so that the predictions have a one-to-one comparison correspondence with the input image in spatial dimensions (height and width).

Another method generalized Mask-R-CNN (G-Mask) [3] compared with several major methods including multi-scale CNN (MS-CNN) [28], contextual multi-scale region-based CNN (CMS-R-CNN) [29], scale-friendly deep convolutional network [30], multitask cascade CNN [31], and Faceness-Net [32]. Compared to the advanced MS-CNN method, the proposed method's average precision (AP) value was only 0.014 lower in the easy subset and 0.049 lower in the moderate subset. There are some gaps between the G-Mask method and MS-CNN on the hard subset. The reason may be that the MS-CNN methods employ a series of strategies for small-scale face detection, and thus they can handle more challenging cases.

Previous research has not provided much novelty on IoU RPN, even though RPN is a very important part. Previous research focused more on loss function and object and face detection but did not focus on identifying and recognizing faces with imperfect data. Thus, in this study, it is proposed to modify IoU in Mask R-CNN model for recognition occluded of Indonesian face. The modified RPN plays a role in increasing the accuracy of detection, coordinates, and segmentation.

## 2.2. Analysis stage

Mask R-CNN tries to improve the ability of object detection which currently uses bounding boxes, by using dense pixel-wise predictions to provide a more complex understanding of an image. CNN is widely used for feature extraction from an image in the form of feature maps. Mask R-CNN uses these feature maps as input for the FCN, which generates a matrix. The resulting matrix is 1 for all pixel locations that are part of the object and 0 for all other locations. This matrix is known as the binary mask. Mask R-CNN architecture is shown in Figure 1.

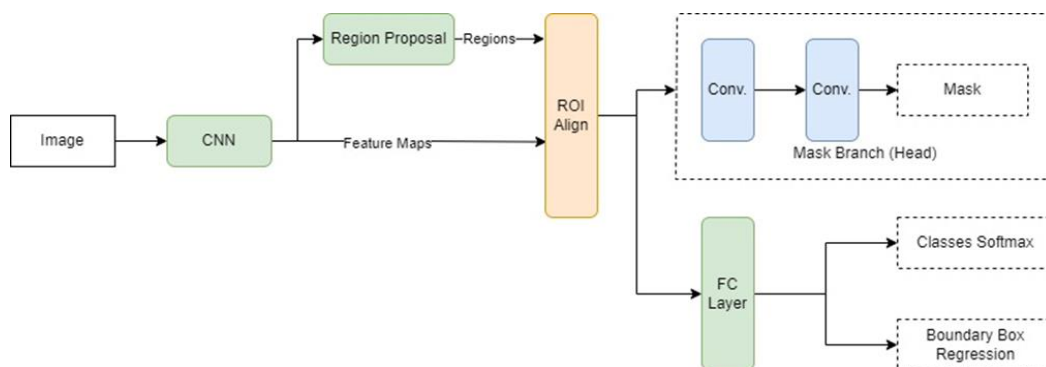


Figure 1. Mask R-CNN architecture

In this paper, we propose to use visual geometric group 16 (VGG16) as a CNN model. VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "very deep convolutional networks for large-scale image recognition" [33]. This model achieves a top-5 test accuracy of 92.7% on ImageNet, a dataset of more than 14 million images belonging to 1,000 classes. It is one of the well-known models submitted to ILSVRC-2014. It made improvements over AlexNet by replacing large kernel-sized filters (11 and 5 in the first and second convolutional layers, respectively) with multiple 3×3 kernel-sized filters one after another. CNN VGG-16 itself consists of 16 CNN layers. The VGG-16 itself only uses 3×3 convolutional (CONV) stride 1 and 2×2 MAX POOLING stride 2.

Region proposal network (RPN) is a fully convolutional network (FCN) [10] that image input for any size and output in the form of a box from the object proposal that has an objectivity score. RPN uses 9 types of anchors with 3 ratios and 3 scales. The ratio is 1: 1, 1: 2, and 2: 1. The scales are 128, 256, and 512. These scales and ratios are very important for overcoming the difference in ratios and scales. Since the 2,400-kernel window uses these 9 anchors, we have 21,600 anchors. Two methods are used to maximize the

number of anchors, namely ignoring the cross-boundary anchors or anchors outside the image. The second method is to use non-max suppression (NMS). The way it works is on the positive intersecting objects to carry out many operations on the same object. From these anchors, the anchor with the maximum value is selected. That way, the number will become 2,000 anchors, which is more efficient. IoU visualization is in Figure 2.

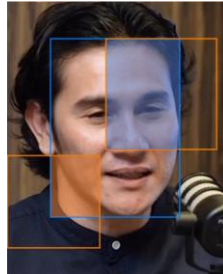


Figure 2. Illustration of IoU

Figure 2 shows that the blue box is the ground-truth box, and the orange box is the anchor. In Figure 2 there are two IoU which are blue and orange. The blue IoU indicates that  $\text{IoU} \geq 0.5$  (value 0.5 is used as an example of reference for using the minimum IoU value) which means this anchor is used as foreground (there are objects in the anchor), while the orange IoU shows that  $\text{IoU} < 0.5$  which means this anchor is used as background. The formula for the RPN is as (1):

$$\text{IoU} = \frac{\text{pixels}(A \cap Gt)}{\text{pixels}(A \cup Gt)} \quad (1)$$

where  $\text{IoU}$  is the ratio of the intersection of  $A$  with  $Gt$  to  $A$  union  $Gt$ ,  $A$  is Anchor or prediction box, and  $Gt$  is Ground truth boxes.

$\text{IoU}$  stands for intersection over union where if  $\text{IoU}$  is 0.7 then the image in the Anchor is considered an object, while if  $\text{IoU} < 0.7$  then the image in the Anchor is considered not an object. RPN will use the 13th Convolutional layer to generate map features. Its size is 512, where 256 kernel windows are obtained from positive anchors and an equal number from negative anchors. 512 kernel windows are obtained from  $40 \times 60 = 2,400$  kernel window/anchor locations, where later 2,400 anchors will justify the anchor location of the object and not. Object range values are 0 to 1, as well as non-objects. This value will be used for classification.

RPN uses 9 types of anchors with 3 ratios and 3 scales. The ratio is 1:1, 1:2, and 2:1. The scales are 128, 256, and 512. These scales and ratios are very important for overcoming the difference in ratios and scales. Since the 2,400-kernel window uses these 9 anchors, we have 21,600 anchors. Two methods are used to maximize the number of anchors, namely ignoring the cross-boundary anchors or anchors outside the image. The second method is to use NMS. The way it works is on the positive intersecting objects to carry out many operations on the same object. From these anchors, the anchor with the maximum value is selected. That way, the number will become 2,000 anchors, which is more efficient.

RPN loss function is a function to measure how big the error in the prediction is. The loss function is very useful for defining or giving positive labels to objects. In this case, 2 types of error measurement are carried out, namely the object/non-object classification and the loss function for box regression. The equation used in the RPN loss function is:

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \frac{1}{N_{reg}} \sum_i L_{reg}(t_i, t_i^*) \quad (2)$$

$$L_{cls}(p_i, p_i^*) = -p_i^* \log(p_i) - (1 - p_i^*) \log(1 - p_i) \quad (3)$$

$$L_{reg}(t_i, t_i^*) = \begin{cases} 0.5(t_i - t_i^*)^2, & \text{If } |t_i - t_i^*| < 1 \\ |t_i - t_i^*| - 0.5, & \text{otherwise} \end{cases} \quad (4)$$

The formulation of the classification loss function  $L_{cls}$  is like (2)  $L(p_i, t_i)$ , except that the foreground and background classification loss  $L_{cls}(p_i, p_i^*)$  changes to multi-class classification loss. The loss function of RPN in (2) is the sum of classification ( $cls$ ) loss in (3) and regression ( $reg$ ) loss in (4). The classification loss is the entropy loss on whether it is a foreground or background. The regression loss is the difference between

the regression of the foreground box and that of the ground truth box. The equation used in the total loss function:

$$L_{total}(p_i, p_i^*) = L_{reg}(t_i, t_i^*) + L_{cls}(p_i, p_i^*) \quad (5)$$

where  $p_i$ : the predicted probability that the object contains,  $p_i^*$ : label with a value of 1 for a positive anchor and 0 for a negative anchor,  $\{p_i\}$ :  $\{p_1, p_2, \dots\}$  the set of predictive probabilities containing objects,  $N_{cls}$ : size in minibatch (512),  $N_{reg}$ : anchor numbers or numbers of anchors in minibatch (512),  $t_i$ : the four coordinates of the bounding box,  $t_i^*$ : the coordinates of the ground-truth window labeled positive,  $\{t_i\}$ :  $\{t_1, t_2, \dots\}$  set of four bounding-box coordinates,  $L_{cls}(p_i, p_i^*)$ : loss function foreground and background classification of each anchor,  $L_{reg}(t_i, t_i^*)$ : loss function regression,  $N_{cls}, \lambda$  (constant value),  $N_{reg}$ : hyperparameters for adjusting the weight between two losses, and  $i$ : index anchors in the minibatch.

From the total loss function in (5), it can be found that only anchors with positive values ( $p_i^* = 1$ ) will be subject to regression. All prediction and ground-truth coordinates are normalized using anchor location and anchor size (no subscript means prediction, the subscript "a" means anchor, \* means ground-truth). Based on (4), The regression parameter coordinates are defined in (6):

$$\begin{aligned} tx &= \frac{(x-x_a)}{w_a}, ty = \frac{(y-y_a)}{h_a} \\ tw &= \log\left(\frac{w}{w_a}\right), th = \log\left(\frac{h}{h_a}\right) \\ t_x^* &= \frac{(x^*-x_a)}{w_a}, t_y^* = \frac{(y^*-y_a)}{h_a} \\ t_w^* &= \log\left(\frac{w^*}{w_a}\right), t_h^* = \log\left(\frac{h^*}{h_a}\right) \end{aligned} \quad (6)$$

where  $t_i$ : prediction box,  $t_i^*$ : ground truth box,  $x, y$ :  $x$  and  $y$  coordinates,  $w, h$ : width and height,  $i$ :  $x/y/w/h$ .

FCN uses a convolutional neural network to convert image pixels into pixel categories [17]. FCN changes the height and width of the middle layer features by mapping back to the input image size via the transformed convolutional layer. The predictions have a one-to-one ratio correspondence with the input image in spatial dimensions (height and width). Given the spatial dimension position, the output from the channel dimension will be a prediction of the pixel category according to the location.

FCN uses dense prediction, namely pixel-wise class labeling, to label image pixels in determining segmentation classes. All pixels are predicted one by one. Because it uses dense prediction, FCN has a prediction sensor that is the same size as the original image. To predict the size of the FCN, the closer to the output, the smaller the size, but the deeper the prediction is. After the prediction, the segmentation will be carried out.

Bilinear interpolation is performed using linear interpolation. first in one direction and again in the other. Although each step is linear in sample values and position, the overall interpolation is not linear but quadratic across the sample locations. Bilinear interpolation is one of the basic resampling techniques in deep learning, computer vision, and image processing. Bilinear interpolation is also called bilinear filtering or bilinear texture mapping. Because the prediction layer class is not the same size as the prediction sensor, we generate up-sample back 32 times by inserting 31 paddings, which are initialized using bilinear interpolation. After getting a sensor of the same size as the prediction, a convolution is carried out. The result is the sensor layer that will be used for prediction.

In Mask R-CNN [11], we use FCN-8s. FCN-8s uses 3<sup>rd</sup>, 4<sup>th</sup>, and 5<sup>th</sup> pooling (conv7). Up-sample Conv-7 four times and up-sample again twice for pooling-4 so that it has the same size as pooling 3. After all, three poolings have a size of 28×28 like pooling 3, join the three-pooling using the 1×1 convolution. Which results in a deep depth of 21 with dimensions of 28×28. Up-sample back 8 times to get the insert padding, then convolution and up-sample back to get a more suitable segmentation to the image.

On Mask R-CNN, you must train the proposal object, object detector, and so on to get  $n$  to  $n$  segmentation sizes. For the results, FCN and Mask R-CNN have a difference. Mask R-CNN is better in segmentation because they change RoIPool to RoIAlign form with 10%-50% accuracy improvements and decouple masks and class predictions.

The FCN [10], [11] loss function uses a formula or equation per pixel-softmax loss function. Matrix multiply+bias offset is a network with 3 classes. Input from the network is performed in an exponential operation. Then normalization is performed. In the learning process, the process of minimizing the value of cross-entropy loss (softmax) is carried out with the formula (7):

$$H(p, q) = -\sum_x p(x) \log q(x) \quad (7)$$

where  $p(x)$  is ground-truth probability (0/1) and  $\log q(x)$ ,  $L_i$  is predictive probability. Attempts are made to reduce the loss function by minimizing cross-entropy to get less value or loss function. This is a learning process in this method. The loss function is performed on all existing pixels.

Region of interest (RoI) is a sample in a "data set" that is identified for a specific purpose. The ROI concept is generally used in many application areas. For example, in medical imaging, a tumor's boundaries can be defined on an image or volume to measure its size. The endocardial border can be defined on the image, possibly during different cardiac cycle phases, for example, end-systole and end-diastole, to assess cardiac function. In "geographic information systems (GIS)", ROI can be taken literally as a selection of polygons from a 2D map. In "computer vision" and "optical character recognition", ROI defines the object's boundaries under consideration. ROI is also often used in face detection or object detection.

The size of the input image after CNN VGG16 will have the initial size divided by 32 and the object image. In RoIPool, the value of objects or images with size with the number of floats will be converted into an integer. Because the size of the object after the CNN VGG16 process and the size that softmax and regression boxes can accept is  $7 \times 7$  on the fully connected layer, resizing must be done using max pooling with the object size divided by 7 (for example,  $20/7=2.86$  rounded to 2). Because there is quantization that occurs twice, information is lost from input to output. This is fine for classifications such as the fast and Faster R-CNN.

There is a difference between RoIPool and RoIAlign. RoIAlign does not round up for results from CNN VGG16, so they are still using floats. For example, to get an image at coordinates 2.97 is done by bilinear interpolation. Bilinear interpolation will not change the coordinate value 2.97 obtained from CNN  $7 \times 7$ . This eliminates the quantization effects found in RoIPool.

### 3. RESULTS AND DISCUSSION

#### 3.1. Occluded of Indonesian faces dataset

The data in this study is facial section data (occluded of Indonesian faces) shown in Figure 3. Figure 3(a) is data that has all face information, while Figure 3(b) is occluded of Indonesian faces are face data that does not show 100% of the face, in other words, faces that are blocked by other objects or faces that are cut off. Occluded Indonesian faces usually occur due to taking pictures of faces with an angle that does not show the entire shape of the face, faces that are cropped, or are blocked by other objects such as eyeglass masks.

Dataset Occluded Indonesian faces are collected privately. The dataset used is 3,000+ Indonesian facial data. This dataset will be labeled manually to determine the class in the training and validation dataset, this manual labeling is also useful for determining the performance of the R-CNN Mask model. Meanwhile, data testing will use 100+ new data.



Figure 3. Dataset: (a) dataset that shows all face information, whereas and (b) dataset that does not show some facial information such as the use of masks and sunglasses as well as shooting at an angle that only shows half the face (occluded face)

#### 3.2. Experiment

From the previous explanation, Mask R-CNN can be a good method in terms of face detection and running time, which is almost the same as Faster R-CNN in detecting faces. In this paper, an architecture is designed to overhaul the FCN so that face detection can be even better because segmentation plays an important role in face detection. The face recognition system design proposed in this paper is shown in Figure 1. From Figure 1, we can see how the data flow is processed, starting from the incoming image data, then the CNN value will be calculated, then using the proposed region network, a feature map will be obtained. Then apply the RoI pooling layer obtained from the RoI Align calculation on the bounding boxes to bring all the RPN candidates on the feature map to the same size.

Meanwhile, segmentation is done using the FCN method, which combines several layers on the feature maps and CNN VGG-16 to perform segmentation. Proposals are forwarded to fully connected layers to classify and display bounding boxes for objects. The final step is to join forces between mask branches, box regression, and classification to get results in facial recognition.

The CNN backbone that we use is CNN VGG-16. RPN will use the 13<sup>th</sup> layer of CNN calculations to generate a feature map and use (1), where it is determined that if the input image contains a face object, it will be rated as 1. For images other than the face object, it will be considered the background, for this face image, which will be detected and predicted. There can be many face objects in one image, and using RPN and instance-aware semantic segmentation can label the same object in one image, for example, if there are two faces, labeling face 1 and face 2 will be done so that the 2 objects can be distinguished.

RoIAlign must obtain a RoI Pooling layer with a feature map size that can be processed by classification and box regression. RoIAlign calculates the face image's image/pixel by not rounding for the CNN VGG-16 results, so the numbers obtained are using floats. In other words, there is no quantization of the image coordinates.

FCN here functions in the form of segmentation by labeling images using dense prediction, namely pixel-wise class labeling the results of FCN in the form of branch masks. Decouple mask and class prediction are carried out by generating masks for all existing classes, and all masks are binary masks. For example, if we have 21 classes then we generate binary masks for each class. The difference occurs because FCN only generates masks once for all classes, which means FCN only has 1 class. The mask branch can predict K (K=number of classes) masks per RoI, using only the K<sup>th</sup> mask, where k is the class predicted by the classification branch. Because the prediction layer class is not the same size as the prediction sensor, a simple backup is generated by padding the insert 31 (32-1=31) times, which is initialized using bilinear interpolation. After getting a sensor of the same size as the prediction, a convolution is carried out. The result is the sensor layer that will be used for prediction.

In this research, experiments will be carried out using the Mask R-CNN method by modifying the RPN section to improve detection accuracy, determine accurate coordinates, and increase segmentation, to increase the accuracy of the recognition process. The calculation of the loss function is performed on the RPN and FCN sections. The modified part is the IoU of the RPN. In this section, an overhaul will be carried out by removing anchors that are outside the size or area of the image and only using anchors that are inside the image. The difference can be seen in Figure 4.

The blue line in Figure 4 is the ground truth box while the orange is the anchor. Figure 4(a) shows the original Mask R-CNN where anchors that are outside the image are also used. Figure 4(b) shows a modified Mask R-CNN that will be used by eliminating the use of anchors that are outside the image to save running time. This research model uses Figure 4(b).

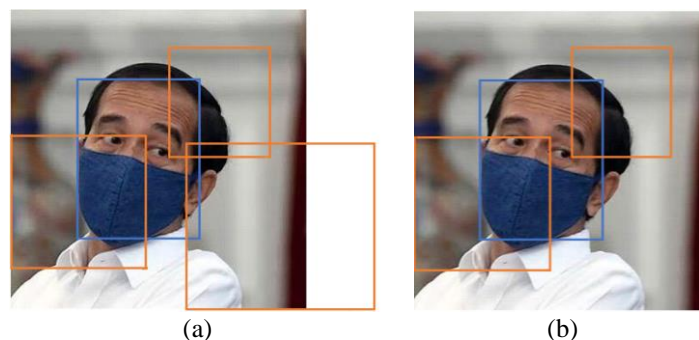


Figure 4. Comparison of anchor usage: (a) with anchor outside the image and (b) without anchor outside the image

This paper provides experimental recommendations for IoU of RPN such as size and image quality as inputs to provide accuracy and running time. Experiments in this study used the original Mask R-CNN method and Mask R-CNN with modifications IoU of the RPN section. This research compares the running time and accuracy of using the models, to determine the comparison as shown in Table 1.

In this experiment, each model and method in Table 1 uses 20 epochs and 1,000 steps per epochs with 3,000+ dataset occluded of Indonesian face. Mask R-CNN with modifications to IoU of RPN obtained an increase in the percentage accuracy of mAP, mAR, and F1-Score each by 14.75%, 10.9%, and 12.97% higher than the original Mask R-CNN. The running time of our method is 4,781(s) faster than Mask R-CNN. This increase in accuracy and acceleration of running time cannot be separated from reducing the inefficient IoU of the RPN process, namely removing anchors that are outside the size of the image and focusing on the anchors that are in the image.

We compared the segmentation and recognition results of the Mask R-CNN with modifications to IoU of the RPN with the original Mask R-CNN model. We did a comparison with 300 validation data. Comparisons were made with manual annotations, the Mask R-CNN model, and the Mask R-CNN with modifications to IoU of the RPN as shown in Figure 5. Figure 5(a) is the result of the annotation done manually, Figure 5(b) is the result of the model we built using GIoU, while Figure 5(c) is the result of the Mask R-CNN model.

We can see the difference between Mask R-CNN and the method we developed in Figure 5. The method we use provides increased segmentation results and accuracy. The segmentation generated by our model is better at retrieving the occluded Indonesian face. The average accuracy of facial recognition with our model is increased by 10% to 15% with 300 datasets validation on our architecture model. Apart from accuracy and segmentation, our loss function model is better than the original Mask R-CNN model. The loss function of the original Mask R-CNN and our model are in Figures 6 and 7.

Table 1. Running time, mAP, mAR, and F1-Score

Method	Occluded of Indonesian face object			
	Running time (s)	mAP	mAR	F1-score
Mask R-CNN	22715	0.6761	0.7465	0.7095
Mask R-CNN + modification IoU of RPN	17934	0.8235	0.8555	0.8392

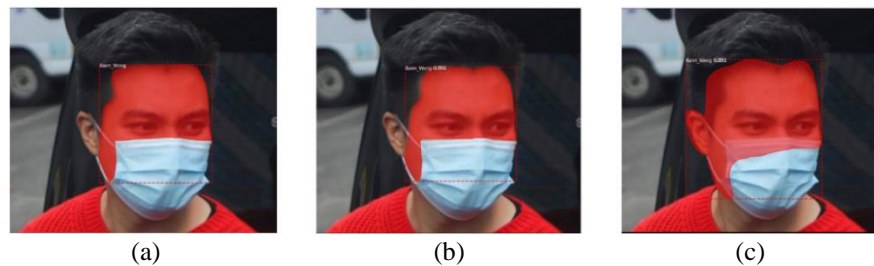


Figure 5. The experimental results of occluded Indonesian face: (a) original image, (b) prediction of our model, and (c) prediction of Mask R-CNN

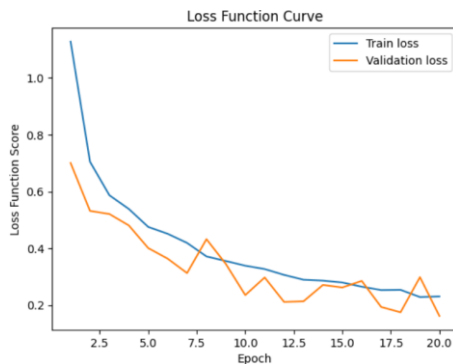


Figure 6. Loss function Mask R-CNN

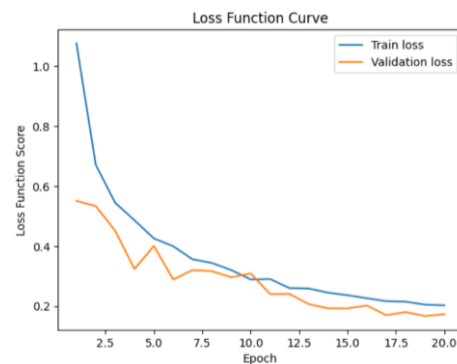


Figure 7. Loss function Mask R-CNN with modification IoU of RPN



#### 4. CONCLUSION

In this paper, a method for occluded Indonesian face image recognition based on an improved structure of Mask R-CNN was proposed. We use 3,000 datasets consisting of 2,700 training data and 300 validation data with 20 epochs of 1,000 batches. Improvements to the use of Anchors by removing the use of anchors outside of the image to save running time. Segmentation and loss function in our model is also slightly better than the R-CNN mask. The results of the research, our model increase in the percentage accuracy of mAP, mAR, and F1-Score each by 14.75%, 10.9%, and 12.97% higher than the Mask R-CNN method. meanwhile, the running time with our model is reduced by 21% compared to Mask R-CNN. The average accuracy of occluded Indonesian face recognition with our model increased by 10%-15% from the previous model by using 300 validation datasets in our model.

#### ACKNOWLEDGEMENTS

The publication is made as part of a final college assignment and an assignment for the *Pendidikan Magister menuju Doktor untuk Sarjana Unggulan (PMDSU)* grant funded by the Ministry of Research and Technology of the Republic of Indonesia.




#### REFERENCES

- [1] W. Wu, Y. Yin, X. Wang, and D. Xu, "Face detection with different scales based on Faster R-CNN," *IEEE Transactions on Cybernetics*, vol. 49, no. 11, pp. 4017–4028, Nov. 2019, doi: 10.1109/TCYB.2018.2859482.
- [2] O. Cakiroglu, C. Ozer, and B. Günsel, "Design of a deep face detector by Mask R-CNN," in *2019 27th Signal Processing and Communications Applications Conference (SIU)*, Apr. 2019, pp. 1–4, doi: 10.1109/SIU.2019.8806447.
- [3] K. Lin *et al.*, "Face detection and segmentation based on improved Mask R-CNN," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–11, May 2020, doi: 10.1155/2020/9242917.
- [4] J. Deng, Y. Zhou, and S. Zafeiriou, "Marginal loss for deep face recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jul. 2017, pp. 2006–2014, doi: 10.1109/CVPRW.2017.251.
- [5] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: deep hypersphere embedding for face recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 6738–6746, doi: 10.1109/CVPR.2017.713.
- [6] X. Zhang, Z. Fang, Y. Wen, Z. Li, and Y. Qiao, "Range loss for deep face recognition with long-tailed training data," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 5419–5428, doi: 10.1109/ICCV.2017.578.
- [7] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.
- [8] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proceedings of the British Machine Vision Conference 2015*, 2015, pp. 41.1--41.12, doi: 10.5244/C.29.41.
- [9] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: a unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.
- [10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3431–3440, doi: 10.1109/CVPR.2015.7298965.
- [11] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.
- [12] Y. Du and Q. Wang, "Multi-angle face detection based on improved RFCN algorithm using multi-scale training," in *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, Apr. 2021, pp. 319–322, doi: 10.1109/ICSP51882.2021.9408676.
- [13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.
- [14] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.
- [15] 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, 2017, doi: 10.1109/TPAMI.2016.2577031.
- [16] J. Yu, M. Wu, C. Li, and S. Zhu, "A street view image privacy detection and protection method based on Mask-RCNN," in *2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, Dec. 2020, pp. 2184–2188, doi: 10.1109/ITAIC49862.2020.9338847.
- [17] J. Ji, X. Lu, M. Luo, M. Yin, Q. Miao, and X. Liu, "Parallel fully convolutional network for semantic segmentation," *IEEE Access*, vol. 9, pp. 673–682, 2021, doi: 10.1109/ACCESS.2020.3042254.
- [18] H. Jiang and E. Learned-Miller, "Face detection with the Faster R-CNN," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, May 2017, pp. 650–657, doi: 10.1109/FG.2017.82.
- [19] Z. Zhou, M. Zhang, J. Chen, and X. Wu, "Detection and classification of multi-magnetic targets using Mask-RCNN," *IEEE Access*, vol. 8, pp. 187202–187207, 2020, doi: 10.1109/ACCESS.2020.3030676.
- [20] X. Sun, P. Wu, and S. C. H. Hoi, "Face detection using deep learning: An improved faster RCNN approach," *Neurocomputing*, vol. 299, pp. 42–50, Jul. 2018, doi: 10.1016/j.neucom.2018.03.030.
- [21] Y. Bai, W. Ma, Y. Li, L. Cao, W. Guo, and L. Yang, "Multi-scale fully convolutional network for fast face detection," in *Proceedings of the British Machine Vision Conference 2016*, 2016, pp. 51.1--51.12, doi: 10.5244/C.30.51.
- [22] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *Advances in neural information processing systems*, vol. 29, pp. 379–387, 2016.
- [23] L. Yu, Y. Hu, X. Xie, Y. Lin, and W. Hong, "Complex-valued full convolutional neural network for SAR target classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 10, pp. 1752–1756, Oct. 2020, doi: 10.1109/LGRS.2019.2953892.
- [24] J. Liu, C. Fang, and C. Wu, "A fusion face recognition approach based on 7-layer deep learning neural network," *Journal of*




- Electrical and Computer Engineering*, vol. 2016, pp. 1–7, 2016, doi: 10.1155/2016/8637260.
- [25] H. El Khiyari and H. Wechsler, "Face recognition across time lapse using convolutional neural networks," *Journal of Information Security*, vol. 07, no. 03, pp. 141–151, 2016, doi: 10.4236/jis.2016.73010.
- [26] K B Pranav and J Manikandan, "Design and evaluation of a real-time face recognition system using convolutional neural networks," *Procedia Computer Science*, vol. 171, pp. 1651–1659, 2020, doi: 10.1016/j.procs.2020.04.177.
- [27] M. Nimbarte and K. K. Bhojar, "Biased face patching approach for age invariant face recognition using convolutional neural network," *International Journal of Intelligent Systems Technologies and Applications*, vol. 19, no. 2, 2020, doi: 10.1504/IJISTA.2020.107216.
- [28] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *ECCV 2016*, 2016, pp. 354–370.
- [29] C. Zhu, Y. Zheng, K. Luu, and M. Savvides, "CMS-RCNN: contextual multi-scale region-based CNN for unconstrained face detection," *Deep learning for biometric*, pp. 57–79, 2017, doi: 10.1007/978-3-319-61657-5\_3.
- [30] S. Yang, Y. Xiong, C. C. Loy, and X. Tang, "Face detection through scale-friendly deep convolutional networks," *arXiv preprint arXiv:1706.02863*, 2017.
- [31] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.
- [32] S. Yang, P. Luo, C. C. Loy, and X. Tang, "Faceness-Net: face detection through deep facial part responses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 1845–1859, Aug. 2018, doi: 10.1109/TPAMI.2017.2738644.
- [33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *The 3rd International Conference on Learning Representations (ICLR 2015)*, 2015, pp. 1–14.

## BIOGRAPHIES OF AUTHORS






**Rahmat Budiarsa**    is currently pursuing his doctoral program in the Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia. He took a bachelor's degree in informatic engineering program, from the Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia, in 2019 and a master's degree in Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia in 2020 (PMDSU Master and PhD Program). His research areas of interest include machine learning, artificial intelligence, and deep learning. He can be contacted at email: rahmat.budiarsa09@mail.ugm.ac.id.



**Retantyo Wardoyo**    is a lecturer and a researcher at the Department of Computer Science, Universitas Gadjah Mada. He obtained his bachelor's degree mathematics in at Universitas Gadjah Mada, Indonesia. He obtained his master's degree in computer science at the University of Manchester, UK, and His doctoral degree in computation at the University of Manchester Institute of Sciences and Technology, UK. His research interests include intelligent systems, reasoning systems, expert systems, fuzzy systems, vision systems, group DSS and clinical DSS, medical computing, and computational intelligence. He can be contacted at email: rw@ugm.ac.id.



**Aina Musdholifah**    is a lecturer and a researcher at the Department of Computer Science, Universitas Gadjah Mada. She obtained a bachelor's degree in computer science, from Universitas Gadjah Mada, Indonesia. She obtained a master's degree in computer science at the Universitas Gadjah Mada, Indonesia, and Her doctoral degree from Computer Science Universiti Teknologi Malaysia, Malaysia. Her research interests include genetics algorithms, fuzzy logic, bioinformatics, soft computing, and machine learning. She can be contacted at email: aina\_m@ugm.ac.id.