International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Combination of Aggregated Channel Features (ACF) Detector and Faster R-CNN to Improve Object Detection Performance in Fetal Ultrasound Images

Fajar Astuti Hermawati[1,2]* **Handayani Tjandrasa**[1] **Nanik Suciati**[1]

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
[2]*Department of Informatics, Universitas 17 Agustus 1945, Surabaya, Indonesia*
* Corresponding author's Email: fajar.astuti15@mhs.if.its.ac.id

Abstract: The research proposed a method that combined non-deep learning detector that called Aggregated Channel Features (ACF) detector and Convolutional Neural Network (CNN) that named Faster R-CNN detector to extract a cross-sectional area of the fetal limb in an ultrasound image. This combination is appropriate to solve the problem of object detection where the object has no clear characteristic, it has shape variation, blurred, and no clear boundaries, which is difficult to solve using the common thresholding or the edge detection method. This method also deals with the ultrasound image analysis which the training set is small. The pre-trained CNN can establish the classification model from the small annotated training data. ACF detector provides the region proposals of the non-cross-sectional area as an input of pre-trained CNN. The proposed method could improve the average precision of detection result when it was compared with Faster R-CNN and ACF detector alone. Also, the combination method could reduce the elapsed time of the Faster R-CNN training phase significantly.

Keywords: Object detection, Faster R-CNN, Aggregated channel features (ACF), Ultrasound images.

1. Introduction

A cross-sectional area of the fetal limb is the sagittal view cross-section of the fetal arm or thigh in an ultrasound image. This area can be an indicator of the nutritional adequacy of the fetus during the pregnancy, by measuring the dispose part [1]. Additionally, the vast size of this area is used as a measurement to predict fetal weights on 3D ultrasound images [2-5]. Measuring fetal weight by using limb volume takes considerable time. An ultrasonographer must mark and calculate the area of each cross-section used in the calculation. Therefore we need a method to detect and segment the area automatically. Feng et al. [2] proposed a method for obtaining limb volume automatically by applying hierarchical marginal space learning method to detect the cross-sectional area of the fetal arm or fetal thigh.

A key challenge in detecting and segmenting the cross-sectional area of the fetal limb is the lack of clarity of boundaries and shapes of the region. The difficulty of determining the characteristics of the cross-sectional region and the nature of the ultrasound images containing the speckle noise and the artefacts led to the necessity of a convolutional neural network (CNN). CNN extracts the features directly from the annotated data at the training phase.

The development of CNN research is increasing rapidly in this last era. This revolution is supported by the rapid growth of parallel processing devices and by the higher the number of datasets. The use of CNN in the medical imaging field is also increasing. In [6], vessels were detected in ultrasound images using CNN. In [7], CNN was used to identify edges on musculoskeletal ultrasound images. A compute unified device architecture (CUDA) CNN was implemented to detect diabetic retinopathy in retinal images [8]. In [9], the lesion was recognized in the breast ultrasound image. In [10] and [11], the researches applied CNN to diagnose and segment the thyroid nodule. Also, classification of the abdominal ultrasound images was done with CNN [12]. In [13], CNN was used to detect papillary thyroid cancer in ultrasound images. All of these studies have

DOI: 10.22266/ijies2018.1231.07

successfully applied the CNN method to identify and perform other image processing such as edge detection and segmentation in medical images, especially on ultrasound images.

The problem solved in this study is how to extract or detect cross-sectional areas of fetal ultrasound images. The techniques used to identify objects in an image can be divided into two categories, i.e. the non-deep learning object detector and the deep learning method that implements CNN [14]. The Aggregate Channel Features (ACF) [15], Locally Decorrelated Channel Features (LDCF) [16], Spatial pooling+ [17] and also Viola Jones [18,19] are the non-deep learning methods. In a deep learning object detector, there is the Faster R-CNN [20] which is an updated version of Fast R-CNN [21] and R-CNN [22]. The problem in detecting the object in an image is to get the region proposal and test it according to the class of object. If the R-CNN method and faster R-CNN use an external region proposal method, such as selective search, the Faster R-CNN introduces a novel Region Proposal Network (RPN) that is scale-invariant in dividing the convolutional layer. The recognition phase of the Faster R-CNN method is 250 times faster than R-CNN and 100 times faster than Fast R-CNN. However, the time required for the RPN training phase in the Faster R-CNN becomes much longer than the previous two methods.

The layer model used in CNN dramatically determines the accuracy of the detection results. According to Tajbakhsh et al. [23], the preparation of layers in medical image analysis can use two techniques, i.e. the fine-tuning from the pre-trained network and the full-trained from scratch. Fine-tuning has advantages from the proven network reliability, such as the AlexNet and the googleNet. Moreover, the transfer learning of the pre-trained network is useful if the amount of training data is small. Tajbakhsh et al. [23] conclude that the use of pre-trained CNN such as AlexNet is superior in medical image analysis compared to full-trained CNN. The implementation of the pre-trained network in the medical image is also adopted by Ma et al. in [10] and Cheng & Malhi in [12].

In CNN training, we need annotated data input that distinguishes the object class and non-objects class. Determining of the non-object class requires high accuracy related to scale and size. To resolve the issue, Ribeiro et al. [24] implemented non-deep learning such as ACF and LDCF in the pedestrian detection problem to generate region proposals as the input of CNN methods. The question is whether the combination of ACF detector and Faster R-CNN able to provide excellent performance in recognising the cross-sectional area of fetal ultrasound images.

In this study, we propose a method that combines the ACF detector and Faster R-CNN to detect a cross-sectional area of the fetal limb in an ultrasound image. Firstly, we apply a preprocessing step to reduce the speckle noise in the ultrasound image using the method proposed by Hermawati et al. [25]. Then, we implement ACF detector to process the training images with annotation bounding box. The ACF detector results that are not cross-sectional areas, become negative training images in the Faster R-CNN stage. We use the annotated positive training images and the negative images as the inputs of the CNN method. We apply the pre-trained AlexNet network model in CNN architecture. In this research, to show the increase of the performance in object recognition, we compare the proposed method with the Faster R-CNN and the ACF detector alone. This study also examines the elapsed time of the RPN network training phase in the proposed approach and the Faster R-CNN detector.

The paper is organized in four sections. After the first section, section 2 explains the proposed method that includes the preprocessing step in section 2.1, the ACF detector in section 2.2, the CNN architecture in section 2.3, the Faster R-CNN in section 2.4 and the performance measurement in section 2.5. Section 3 that consists of two subsections, presents the experimental results. In section 3.1, we show the time reduction in the CNN training phase. Section 3.2 illustrates the object detector experiments. The conclusion is presented in section 4.

2. Methods

The scheme of the proposed method is shown in Fig.1. Firstly, the ACF detector processes the annotated training images that contain the bounding box of object ground truth. The result of ACF object detector is separated into two types, i.e. the positive images and the negative images. The positive images are the bounding box regions that have a significant overlapping with the ground truth area, and on the other hand, the rests are negative images. The negative images from the ACF detector output and the cross-sectional ground truth images are used as input of the pre-trained AlexNet CNN. Furthermore, the re-trained CNN model was used to build the Region Proposal Network (RPN) on the R-CNN Faster and then to train it as many as four stages using the annotated training data.

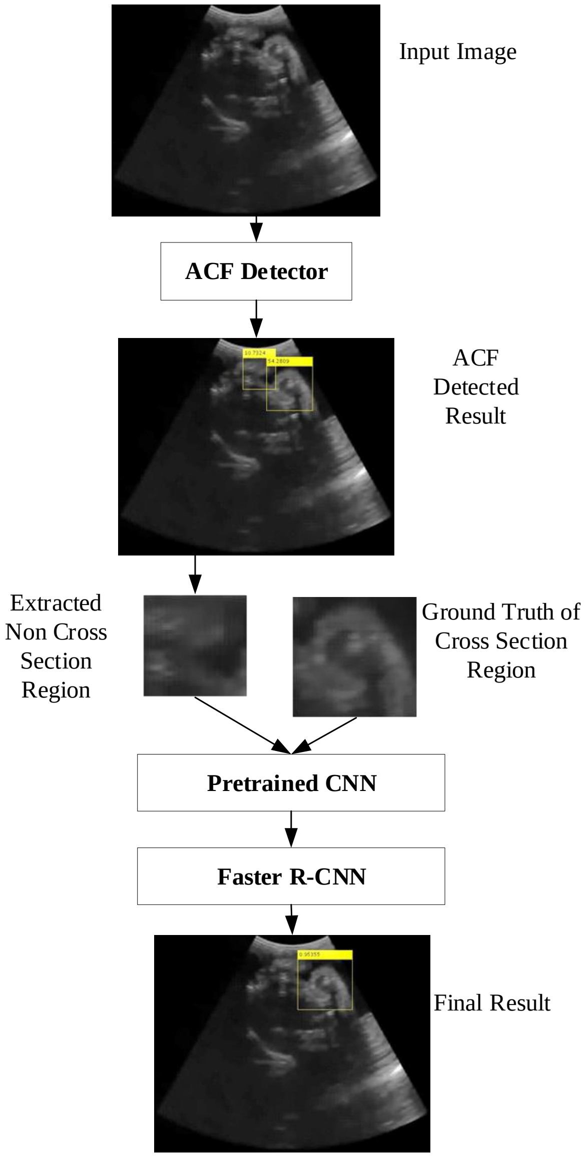
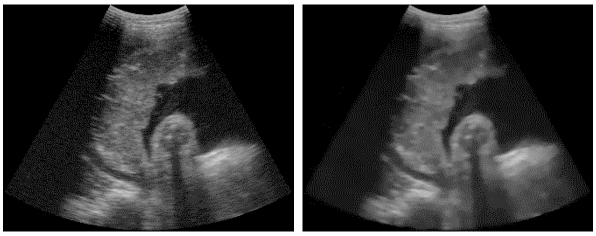


Figure.1 Proposed method scheme

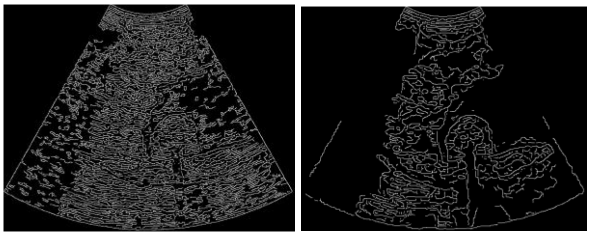
2.1 Preprocessing

Preprocessing step aims to remove the speckle noise on the ultrasound image. We implement the hybrid speckle reduction method that is proposed in [25]. The speckle reduction approach combines the spatial filtering, i.e. bilateral filtering and anisotropic diffusion with the multiresolution wavelet. The advantage of this method, it can eliminate the speckle noise while maintaining the edges. Fig. 2(a) and 2(b) show the image results before and after the preprocessing process. Fig. 3(a) and 3(b) present the edge detection results with Canny edge detector, which show the decrease in detail or noise in the non-edge area while there is no reduction in the edge area.



(a) (b)

Figure.2 Ultrasound image: (a) before noise reduction and (b) after noise reduction



(a) (b)

Figure.3 Binary edge image: (a) before noise reduction and (b) after noise reduction

2.2 Aggregated channel features (ACF) detector

Aggregated Channel Features (ACF) detector proposed in [15] used a combined feature which consisting of three channels of LUV colour space, a normalized gradient channel and a six-channel histogram of oriented gradient (HoG) and then arranged in a boosted tree. The block diagram of the ACF detector method to detect the cross-sectional areas in the ultrasound images is presented in Fig.4. ACF detector will extract the proposal region consisting of the positive area and negative region. Positive proposal regions are obtained from training data containing the bounding box of the ground truth area. Meanwhile, the negative proposal region is extracted automatically by the sliding window in all image area except the bounding box of ground truth area.

Let $I(x,y)$ be the RGB image of mxn size consisting of three channels. Firstly, the image is transformed into the LUV colour space, and then the gradient magnitude of the image I is calculated using the following formula.

$$M(i,j) = \sqrt{\left(\frac{\partial I(i,j)}{\partial x}\right)^2 + \left(\frac{\partial I(i,j)}{\partial y}\right)^2} \qquad (1)$$

Also, the gradient orientation of image I is expressed by the following equation.

$$O(i,j) = \tan^{-1}\left(\frac{\frac{\partial I(i,j)}{\partial y}}{\frac{\partial I(i,j)}{\partial x}}\right) \qquad (2)$$

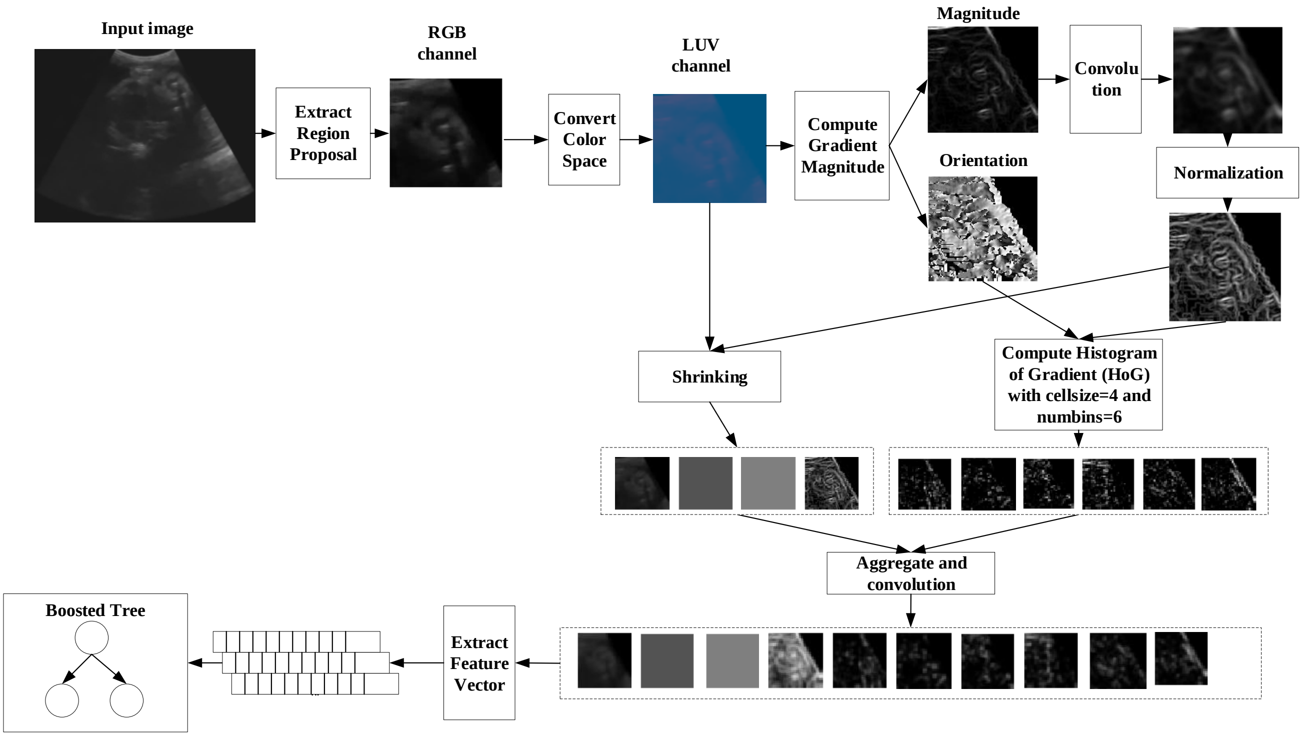


Figure.4 Block diagram of Aggregated Channel Features (ACF) detector

where $\frac{\partial I(i,j)}{\partial x}$ is the derivative *I* at the coordinates (*i*,*j*) in the *x*-direction and $\frac{\partial I(i,j)}{\partial y}$ is the derivative *I* at the coordinates (*i*,*j*) in the *y*-direction. The gradient image is smoothed using a convolution operation between the gradient image and a triangular filter [1 2 1] / 4. After that, the smoothed image is normalized to get the details of the gradient scale by the following equation.

$$\widetilde{M}(i,j) = \frac{M(i,j)}{S(i,j)+c} \qquad (3)$$

where *S*(*i*,*j*) is the smoothed image and *c* is a small normalization constant, e.g. *c* = 0.005.

The steps to compute the histogram of the gradient (HoG) with the bin number = 6 and the cell size = 4 are as follows. For each subwindow called a cell with 4x4 size, the normalised gradient, $\widetilde{M}(i,j)$, is quantized into six bin histograms, $q_1, q_2, ... q_6$, with the orientation range 0-180, based on the value of *O*(*i*,*j*). The assignment of the gradient in the *q*-oriented bin uses the linear interpolation. The HoG feature size is $\frac{m}{cell\ size} \times \frac{n}{cell\ size} \times bin\_number$.

The aggregate features obtained are arranged in a decision tree and trained using bootstrapping and AdaBoost classifier alternately and repeatedly as many *N* stages [15]. At each step, the negative examples are extracted and accumulated with the previous ones.

2.3 CNN architecture

The pre-trained AlexNet CNN architecture consists of five convolutional layers, three max-pooling and three fully-connected layers. The complete CNN architecture can be seen in Table 1.

The first layer with the name 'data' is the input layer with the same size as the input image size. The size of the cross-sectional dataset image is changed to the image size used in AlexNet, i.e. 227x227x3. The middle layer consists of two repeating blocks including the convolutional layer, reLU, cross-channel normalization and max-pooling layer. The first block consists of the convolutional layer ('conv1') with 11x11 kernel size of 96 channels, followed by reLU and channel normalization of 5 channels per element, then ends max pooling ('pool1') with 3x3 layer size. The second block is the same as the first block consisting of the convolutional layer ('conv2') with kernel size 5x5, reLU, channel norm and max-pooling ('pool2') with 3x3 kernel size. Two repetitive blocks consist of the convolutional layer, i.e. 'conv3', 'conv4' which each kernel is 3x3 and reLU. One block consists of convolution layer, reLU and max pooling. The last layer includes a fully connected layer and softmax layer. In this AlexNet architecture, there are three fully connected layers and one softmax layer.

Table 1. CNN architecture

Name	channel	size	kernel
‘data’	3	227x227	
‘conv1’	3	227x227	11x11
‘pool1’	96	55x55	3x3
‘conv2’	96	27x27	5x5
‘pool2’	256	27x27	3x3
‘conv3’	256	13x13	3x3
‘conv4’	384	13x13	3x3
‘conv5’	384	13x13	3x3
‘pool5’	256	13x13	3x3
‘fc6’	256	6x6	6x6
‘fc7’	4094	1	1x1
‘fc8’	4096	1	1x1

For the transfer learning of the AlexNet, the last three layers of AlexNet are tuned to the new data classification that is cross-sectional thigh data sets. Fully connected layer 'fc8' is set in two classes: a cross-sectional region class and non-cross-sectional class. The weighting factor of learning rate is set to 20, and the bias factor of the learning rate is increased to 20 to speed up the training process.

2.4 Faster R-CNN

The Faster R-CNN method which proposed in [22] has two main steps: to extract and train approximately 200 region proposals using the Region Proposal Network (RPN) and to classify based on the features obtained. In the training process, there are four stages: training a Region Proposal Network (RPN), training a Fast R-CNN Network using the RPN from step 1, re-training RPN using weight sharing with Fast R-CNN and re-training Fast R-CNN using updated RPN.

To get the region proposals, in every position of the sliding windows, extracted several region proposals based on a reference box called anchors that have nine shapes with three different scales: 128^2, 256^2 and 512^2 as well as three aspect ratio, i.e. 1:1, 1:2 and 2:1. A region proposal is extracted if IoU (Eq. (8)) between ground truth and the anchor is greater than 0.7. The number of region proposals is reduced by eliminating areas located on the edges of the image.

The RPN consists of two networks. The first one is the CNN. The last five layers of the CNN are replaced with 3x3 convolution layer followed by ReLU layer, 1x1 layer followed by Reshape for RPN layer, softmax layer and RPN classification layer. The second network consists of two layers: 1x1 convolutional layer for regression box (RPN) Convolution and smooth-l1 Box Regression Output Layer.

RPN training process aims to minimize total loss function which is the aggregate of the data loss function at the classification stage and the regularization loss function. The formula of the aggregate loss function is as follows.

$$L = \frac{1}{N_c}\hat{L}_c + \lambda\,\frac{1}{N_{reg}}\,\hat{L}_{reg} \tag{4}$$

The data loss function is normalized by dividing it by N_c which is the size of a mini batch. Moreover, regularization function is standardized by dividing N_{reg} which is the number of anchor locations and multiplied by the balancing parameter λ.

Total data loss function is a log loss function of two prediction classes that are objects and not objects that are formulated as follows.

$$\hat{L}_c = \sum_i^N L_c(p_i, p_i^*) \tag{5}$$

where p_i is the probability of predicting an i-th anchor on a mini batch and p_i* is the ground truth of the i-th anchor which is set to 1 if positive and 0 if negative.

The regularization loss function that also called the regression function is a function that calculates the difference of a bounding box of ground truth object (t_i^*) with the bounding box of the predicted object (t_i), which is formulated as follows.

$$\hat{L}_{reg} = \sum_i^N p_i^* R(t_i - t_i^*) \tag{6}$$

The regression function is multiplied by p_i^* which means that the function will be activated if the anchor is positive.

2.5 Performance measurement

The performance measures in this study is the average precision (AP). The following equation defines average precision (AP) which is a summary of the precision/recall curve.

$$AP = \frac{1}{11}\sum_{r=0}^{1} \max_{\hat{r}:\hat{r}\geq r} p(\hat{r}) \tag{7}$$

where r is a recall level with range between 0,0.1,0.2,0.3, ..., 1. $p(r)$ is precision level of the certain r value. Precision is the ratio of the number of detected cross-sectional areas and the number of total cross-sectional areas. The recall is the ratio of the number of detected area and total number of cases.

In object detection problem, Everingham et al. [26] determine the number of detected area using the

Intersection over Union (IoU). IoU is the ratio of the intersection between detected region (R_{d}) and ground truth region (R_{GT}), and the combination (union) of two areas which is formulated by the following equation.

$$IoU = \frac{R_d \cap R_{GT}}{R_d \cup R_{GT}} \tag{8}$$

3. Results and discussion

The experiments were done using three scenarios. The first scenario was to implement the ACF detector method. The second one was to use the Faster R-CNN and the third scenario was to combine both the ACF detector and Faster R-CNN, as our proposed method. The comparation was performed to see the influence of ACF detector on Faster R-CNN performance. We conducted two evaluations, i.e. the time performance of the training process on Faster R-CNN detector, and the detection performance of the fetal cross-sectional area. The tests were done using a computer with the following specifications: processor Intel Core i7 with 4GB memory and GPU 3.0 745M.

3.1 Evaluating of training process

The first experiment aims to determine the influence of the use of training data extracted from the ACF detector in the proposed method, to the decrease of training time on each phase in the Faster R-CNN detector.

The training process in the Faster R-CNN detector consists of the pre-trained CNN and the RPN training. The CNN training process requires two categories of input data, that is the cross-sectional area as positive images and non-cross-sectional area as negative images. The training data used in this research is divided into two kinds. The first training data is consist of 362 cross-sectional images and 362 non-cross sectional images that are created manually. The second one is comprising 77 cross-sectional objects and 31 non-cross-sectional objects that obtained from ACF detector.

Table 2 shows the comparison of the CNN training process using first training data and second training data. It can be seen that the training process of the first training data takes longer and the number of iterations is more significant with smaller epoch number. Although the average mini batch loss of the first training data is smaller than the second one, the classification accuracy of the testing data shows lower results than the second training data.

Table 2. CNN training comparation.

	1st Training Data	2nd Training Data
Epoch	10	20
Iteration	1300	60
Time (second)	944.45	78.97
Mini-batch loss average	0.125	0.289
Classification accuracy	0.9306	1

Table 3. Comparation of RPN training in Faster R-CNN

	1st Training Data	2nd Training Data
Stage 1		
Iteration	610	610
Time (second)	2113.97	1127.48
Mini-batch loss average	0.183	0.184
Stage 2		
Iteration	550	570
Time (second)	2490.28	1519.84
Mini-batch loss average	0.289	0.389
Stage 3		
Iteration	610	610
Time (second)	2062.55	1379.76
Mini-batch loss average	0.384	0.402
Stage 4		
Iteration	590	600
Time (second)	2332	1619.34
Mini-batch loss average	0.637	0.584

The pre-trained CNN is used to model the RPN network, as described in Section 2.4. The RPN network is trained using the Faster R-CNN object detector consisting of four stages. The object detector training uses the stochastic gradient descent momentum method with a maximum epoch 10, initial learning rate 0.00001 for phase 1 and 2 and 0.000001 for step 3 and 4. Comparison of training process between the trained CNN network which using the first training data and the second training data can be seen in Table 3. In stage 1 to stage 3, the mini batch loss average of the first RPN training is smaller than the second one. However, at the end of the stage, the second RPN training has an average loss lower than the first one. In all stages, the training time of the second RPN training is much smaller than the first one.

In the first experiment, we show the time required for the CNN and the RPN training phase in the Faster R-CNN method by using manually-created training images compared to the training data extracted from the ACF detector. From the experiment, there is a decrease in time during the training process both in the CNN training phase and in the RPN training phase. As presented in Table 2, with a smaller amount of training data, then the number of iterations in the CNN training phase is reduced. However, the accuracy of the classification increases. This means that the use of training data obtained from the ACF detector can improve the accuracy of the classification and reduce time required for CNN training. In Table 3, we can see that the time required for each phase of the RPN training step also decreased by an average of 37%.

3.2 Evaluating of detection performance

The second trial aims to show the performance of the combination of Faster R-CNN and ACF detector when identifying the cross-sectional object in the ultrasound image. We compare the proposed combination method with the ACF detector alone and the Faster R-CNN alone. To see the performance of the cross-sectional object detection in the ultrasound images, we present some examples of the detected results with the highest IoU scores (Fig. 5) and the lowest IoU scores (Fig. 6) and some undetected object by several methods (Fig. 7).

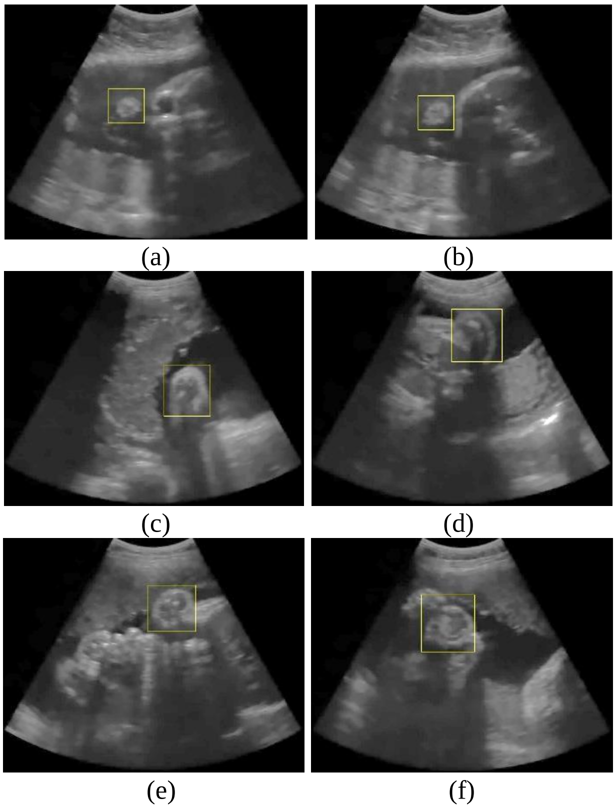

(a) (b)
(c) (d)
(e) (f)

Figure.5 Detection results with highest IoU: (a) and (b) ACF detector, (c) and (d) Faster-RCNN, and (e) and (f) Faster RCNN+ACF detector

ACF detector is capable of providing the highest IoU for a small cross-sectional area, as shown in Fig. 5 (a) and Fig. 5 (b). However, the ACF detector does not recognize cross-sectional regions that have a low contrast to the background as shown in Fig. 6 (a), and only identify the partial areas as in Fig. 6 (b). The ACF detector cannot detect the object located on the edge of the scanning area with the incomplete shape as in Fig. 7 (a).

The Faster R-CNN method can recognize the sizeable cross-sectional object and the high contrast area as shown in Fig.5 (c) as well as the low contrast area as in Fig.5(d). But the Faster R-CNN method cannot identify the small object that surrounded by the other object with similar brightness, as in Fig. 7 (b). The Faster R-CNN can identify the objects as in Fig. 6 (c) and Fig.6 (d) even though not precisely. The combination method, Faster R-CNN+ACF, can recognize all of the cross-sectional objects from the given images including the object that cannot be identified by the two previous methods (Fig. 7 (a) and Fig. 7 (b).

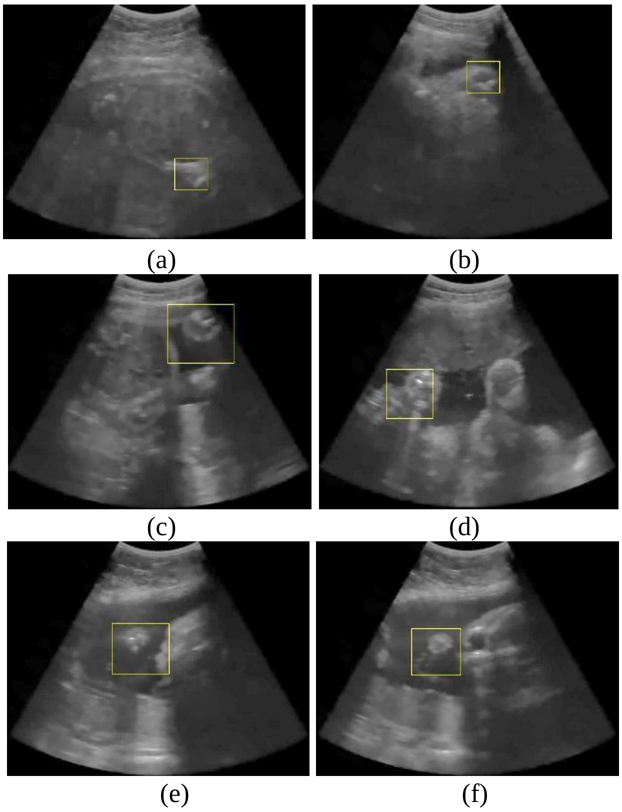
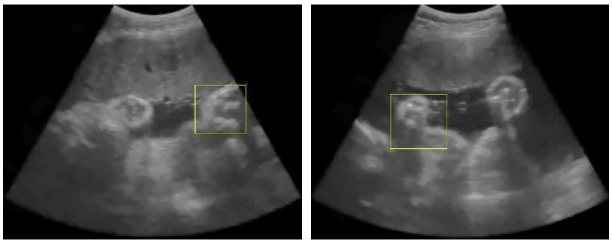

(a) (b)
(c) (d)
(e) (f)

Figure.6 Detection results with lowest IoU: (a) and (b) ACF detector, (c) and (d) Faster-RCNN, and (e) and (f) Faster RCNN+ACF detector



(a) (b)

Figure.7 Detected objects by the proposed method that is undetected by: (a) ACF detector and (b) Faster R-CNN

To evaluate the method quantitatively, Fig. 8 presents the average precision curve for each threshold value of IoU. The average precision of each approach begins to fall when the threshold value is 0.5. Even the ACF detector starts dropping at threshold value equals to 0.4. Thus, the best threshold value of this cross-sectional object detection is 0.5.

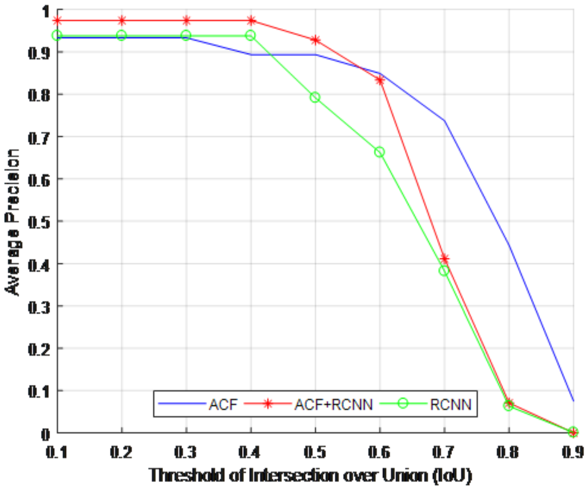


Figure. 8 Average precision per threshold of IoU

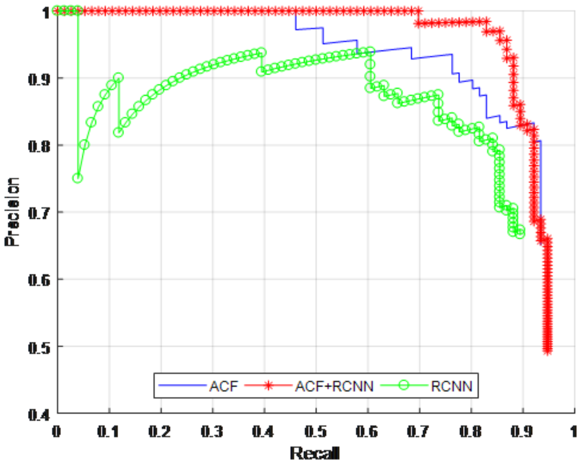


Figure.9 Precision recall curve

Table 4. Average precision comparation

Method for Each Scenario	Average Precision
ACF Detector	0.894
Faster R-CNN	0.791
ACF+Faster R-CNN	**0.928**

Fig. 9 shows a precision / recall (PR) curve for the threshold value 0.5. The precision level of the combination of Faster R-CNN+ACF detector has the precision of 1 when the recall is 0.7, whereas the precision of ACF detector starts to decrease when recall is larger than 0.4. The lowest precision is the Faster R-CNN method alone which begins to decrease its precision when recall is less than 0.1. Table 4 summarises the comparison of average precision (AP) performance of each experiment scenario for the same threshold value of 0.5. The first scenario using ACF detector gives the AP of 0.894. The second scenario implementing Faster R-CNN has the AP of 0.791. The third one applying the combination of ACF detector and Faster R-CNN results in the best average precision of 0.928.

In the second experiment, we can see that the cross-sectional object has various shapes such as small circle (Fig. 5 (a) and Fig. 5 (b)) and a rather big one (Fig. 6 (c)), has no clear boundaries (Fig. 5 (d)), has low contrast to its background (Fig. 6 (a)), attached to another similar object (Fig. 6 (d)) and the incomplete object in the edge of image (Fig. 7 (a)). It is difficult to recognize the cross-sectional objects using common thresholding methods or edge-based segmentation methods. The object detector methods both non-deep learning such as the ACF detector and deep-learning such as the Faster R-CNN can identify the cross-sectional objects as shown in Fig. 5 and Fig. 6. But both the ACF detector and the Faster R-CNN cannot recognize the cross-sectional area in Fig.7. The combination method, ACF+Faster R-CNN, can identify all cross-sectional objects in ultrasound image even those that are unsuccessfully recognized by both approaches alone, as shown in Fig.7. The combination method also can increase the average precision when compared with ACF detector or Faster R-CNN alone. So, the combination of Faster R-CNN and ACF detector can improve the ability to recognize the cross-sectional area in fetal ultrasound images. This capability will facilitate further cross-sectional object analysis, such as to calculate the size of the region.

4. Conclusion

In this research, a method that combines the non-deep learning method of ACF Detector and the Faster R-CNN deep-learning method can recognize the cross-sectional objects that have shape variation, have no clear boundaries, have incomplete shape, attached to the similar object, and have a low contrast to its background. The non-deep learning method is used to extract non-cross-sectional regions as the input of pre-trained CNN processes. The combination of Faster R-CNN and ACF detector methods can increase the average precision of cross-sectional objects detection in ultrasound images and can decrease the training time of Faster R-CNN object detector. This method can be developed to recognize other objects in fetal ultrasound images such as abdominal or head circumference.

References

[1] S. Rueda, C. L. Knight, A. T. Papageorghiou, and J. Alison Noble, "Feature-based Fuzzy Connectedness Segmentation of Ultrasound Images with an Object Completion Step", *Medical Image Analysis*, Vol. 26, No. 1, pp. 30–46, 2015.

[2] S. Feng, K. S. Zhou, and W. Lee, "Automatic Fetal Weight Estimation Using 3D Ultrasonography", In: *Proc. of SPIE, Medical Imaging 2012: Computer-Aided Diagnosis*, Vol. 8315, pp. 1–7, 2012.

[3] L. M. M. Nardozza, M. F. Vieira, E. Araujo Júnior, L. C. Rolo, and A. F. Moron, "Prediction of Birth Weight Using Fetal Thigh and Upper-Arm Volumes by Three-dimensional Ultrasonography in a Brazilian Population", *The Journal of Maternal-Fetal & Neonatal Medicine*, Vol. 23, No. 5, pp. 393–398, 2010.

[4] W. Lee, M. Balasubramaniam, R. L. Deter, L. Yeo, S. S. Hassan, F. Gotsch, J. P. Kusanovic, L. F. Gonçalves, and R. Romero, "New Fetal Weight Estimation Models Using Fractional Limb Volume", *Ultrasound in Obstetrics and Gynecology*, Vol. 34, No. 5, pp. 556–65, 2009.

[5] W. Lee, M. Balasubramaniam, R. L. Deter, S. S. Hassan, F. Gotsch, J. P. Kusanovic, L. F. Gonçalves, and R. Romero, "Fractional Limb Volume - A Soft Tissue Parameter of Fetal Body Composition: Validation, Technical Considerations and Normal Ranges During Pregnancy", *Ultrasound in Obstetrics and Gynecology*, Vol. 33, No. 4, pp. 427–440, 2009.

[6] E. Smistad and L. Løvstakken, "Vessel Detection in Ultrasound Images Using Deep Convolutional Neural Networks", in *Carneiro G. et al. (eds) Deep Learning and Data Labeling for Medical Applications. DLMIA 2016, LABELS 2016. Lecture Notes in Computer Science*, Vol. 10008, pp. 30–38, 2016.

[7] S. I. Jabbar, C. R. Day, N. Heinz, and E. K. Chadwick, "Using Convolutional Neural Network for Edge Detection in Musculoskeletal Ultrasound Images", In: *Proc. of the 2016 International Joint Conference on Neural Networks*, pp. 4619–4626, 2016.

[8] R. Parmar, "Detecting Diabetic Retinopathy from Retinal Images Using CUDA Deep Neural Network", *International Journal of Intelligent Engineering and Systems*, Vol. 10, No. 4, pp. 284–292, 2017.

[9] M. H. Yap, G. Pons, J. Marti, S. Ganau, M. Sentis, R. Zwiggelaar, A. K. Davison, and R. Marti, "Automated Breast Ultrasound Lesions Detection using Convolutional Neural Networks", *IEEE Journal of Biomedical and Health Informatics*, Vol. 22, No. 4, pp. 1218–1226, 2017.

[10] J. Ma, F. Wu, J. Zhu, D. Xu, and D. Kong, "A Pre-trained Convolutional Neural Network based Method for Thyroid Nodule Diagnosis", *Ultrasonics*, Vol. 73, pp. 221–230, 2017.

[11] J. Ma, F. Wu, T. Jiang, Q. Zhao, and D. Kong, "Ultrasound Image-based Thyroid Nodule Automatic Segmentation using Convolutional Neural Networks", *International Journal of Computer Assisted Radiology and Surgery*, Vol. 12, No. 11, pp. 1895–1910, 2017.

[12] P. M. Cheng and H. S. Malhi, "Transfer Learning with Convolutional Neural Networks for Classification of Abdominal Ultrasound Images", *Journal of Digital Imaging*, Vol. 30, No. 2, pp. 234–243, 2017.

[13]H. Li, J. Weng, Y. Shi, W. Gu, Y. Mao, Y. Wang, and W. Liu, "An Improved Deep Learning Approach for Detection of Thyroid Papillary Cancer in Ultrasound Images", *Scientific Reports*, Vol. 8, No. April, pp. 1–12, 2018.

[14] D. Ribeiro, J. C. Nascimento, A. Bernardino, and G. Carneiro, "Improving the Performance of Pedestrian Detectors using Convolutional Learning", *Pattern Recognition*, Vol. 61, pp. 641–649, 2017.

[15] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 8, pp. 1–14, 2014.

[16] W. Nam, P. Dollár, and J. H. Han, "Local Decorrelation For Improved Detection", In: *Proc. of the 27th International Conference on Neural Information Processing Systems*, pp. 424–432, 2014.

[17] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Strengthening the Effectiveness of

Pedestrian Detection with Spatially Pooled Features", In: *Proc. of the 13th European Conference on Computer Vision, ECCV 2014*, pp. 562–577, 2014.

[18] P. Viola and M. Jones, "Robust Real-time Face Detection", In: *Proc. of the Eighth IEEE International Conference on Computer Vision*, Vol. 20, p. 7695, 2001.

[19] F. A. Hermawati and H. Budianto, "A Video Based License Plate Detection System Using Viola-Jones Method", In: *Proc. of the 2nd SciTech Internasional Seminar*, pp. 63–69, 2013.

[20] 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.

[21] R. Girshick, "Fast R-CNN", In: *Proc. of the IEEE International Conference on Computer Vision*, Vol. 2015 Inter, pp. 1440–1448, 2015.

[22]R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", In: *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[23] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?", *IEEE Transactions on Medical Imaging*, Vol. 35, No. 5, pp. 1299–1312, 2016.

[24] D. Ribeiro, A. Mateus, P. Miraldo, and J. C. Nascimento, "A Real-time Deep Learning Pedestrian Detector for Robot Navigation", In: *Proc. of 2017 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2017*, pp. 165–171, 2017.

[25] F. A. Hermawati, H. Tjandrasa, and N. Suciati, "Hybrid Speckle Noise Reduction Method for Abdominal Circumference Segmentation of Fetal Ultrasound Images", *International Journal of Electrical and Computer Engineering*, Vol. 8, No. 3, pp. 1747–1757, 2018.

[26]M. Everingham, L. Van~Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge", *International Journal of Computer Vision*, Vol. 111, No. 1, pp. 98–136, 2015.