

Automatic Detection of Wrecked Airplanes from UAV Images

Anhar Risnumawan*, Muhammad Ilham Perdana^α,
Alif Habib Hidayatulloh[#], A. Khoirul Rizal^β, Indra Adji Sulistijono**,
Achmad Basuki[&], and Rokhmat Febrianto^π

Mechatronics Engineering Division
Politeknik Elektronika Negeri Surabaya (PENS)
Kampus PENS, Jalan Raya ITS Sukolilo, Surabaya, 60111 Indonesia
Email: {^αilhamperdana17, [#]alifhabib01, ^βherurizal,
^πrokhmat}@me.student.pens.ac.id, {*anhar, **indra,[&]basuki}@pens.ac.id

Received September 19, 2019; Revised November 16, 2019; Accepted December 1, 2019

Abstract

Searching the accident site of a missing airplane is the primary step taken by the search and rescue team before rescuing the victims. However, due to the vast exploration area, lack of technology, no access road, and rough terrain make the search process nontrivial and thus causing much delay in handling the victims. Therefore, this paper aims to develop an automatic wrecked airplane detection system using visual information taken from aerial images such as from a camera. A new deep network is proposed to distinguish robustly the wrecked airplane that has high pose, scale, color variation, and high deformable object. The network leverages the last layers to capture more abstract and semantics information for robust wrecked airplane detection. The network is intertwined by adding more extra layers connected at the end of the layers. To reduce missing detection which is crucial for wrecked airplane detection, an image is then composed into five patches going feed-forwarded to the net in a convolutional manner. Experiments show very well that the proposed method successfully reaches AP=91.87%, and we believe it could bring many benefits for the search and rescue team for accelerating the searching of wrecked airplanes and thus reducing the number of victims.

Keywords: Wrecked airplanes detection, UAV image, deep learning method, real-time detector, extra layers.

1. INTRODUCTION

First step for the search and rescue (SAR) team during an accident of a missing airplane usually is to locate the right accident site before performing a rescuing mission. Based on the latest contact and position of the missing airplane, the SAR team has to start searching for the accident site. For example,

an airplane missing on the mountainous area usually requires nearby



Figure 1. Examples of wrecked airplanes on several area. Highly appearance changing and deformable object can be observed. Thus, making the detection more challenging and interesting.

airplanes and helicopters to be deployed for the searching process. This process is tedious and much funding is required. Moreover, the SAR team has minimal time to locate the accident site before rescuing the victims.

The recent accident of a Malaysia airplane MH-370, [1], still becomes a mystery, and the location site is still unknown. Trillions of funds have been spent just to locate the accident. Technology starting from military airplanes, ships, helicopters with sophisticated sensors had been deployed to search the missing airplane. The vast ocean area of exploration makes the search and rescue nontrivial, and still an unsolved problem. An accident of an airplane also recently happen in Timika, Papua, [2]. The twin otter airplane disappeared after taking off from the Moses Kilangain Airport in Timika, Mimika District, en route to Ilaga, the capital city of Puncak District in Papua Province. Due to the vast exploration area, no access road, and lack of technology, the search process taking much time before rescuing the victims. Thus, the delay of rescuing the victims is primarily caused by taking excessively long for searching the accident site. New technology is crucial to be developed using the latest drones and visual detectors to quickly searching for the right accident site. However, using visual information for detecting the airplanes in the cluttered background, as shown in **Figure 1**, requires a sophisticated algorithm to distinguish the wrecked airplanes robustly or not.

Related works, [3]–[6], have been mainly developed for collision avoidance with other airplanes for safety flight system. Airplanes are detected using visual information and output bounding boxes to indicate the occurrence of airplanes. These works differ with the wrecked airplanes as the airplanes mostly not in the ideal shape of an airplane. Wrecked airplanes tend to have a highly deformable object. The airplane's part can be separated away from the main body, and change its color, which makes the detection nontrivial. The

work [7] has worked on the wrecked airplane also, but the approach differs much with our proposed. Using manually hand-crafted features could deteriorate the detection system, as the features are not optimized from the sample data, and it could hardly distinguish the desired wreckage in challenging terrain as the classification method might not deep enough to perform robust detection.

Therefore, immediately locating the accident site using a dedicated technology is crucial to be developed. This paper aims to provide such a solution by developing an automatic wrecked airplanes system using visual information taken from aerial images. In practice, the wrecked airplane can have dramatic changing of visual appearance such as due to burning, and even its parts could be separated away far from the main body, with potentially leaving only the broken airplane body. Moreover, the complex background such as grass, soil, debris, trees, other objects which have shape and visual appearance similar to the wrecked airplane could deteriorate the detection. Thus, a widely-used deformable model using the composition of parts might not work, [3]–[6]. Therefore, in this system, a deep network approach is employed, mainly using a convolutional neural network (CNN). As suggested by the work [8] by proving theoretically, and empirically, [9], [10], more semantic information can be obtained from the deeper layers, while the earlier layers help to model geometrical details. Lower layers can help to locate the bounding box precisely but less invariant to appearance. The higher layers are more robust to appearance and deformable objects. With this in mind, we develop an architecture of the network that focuses more on semantics information by leveraging the higher layers features. Feature maps from the higher layers are processed progressively using convolutional networks to produce the last feature map that is a concatenation from the previous convoluted feature maps, as suggested by [11]. The lower net is a pre-trained VGG-16 model, [12]. In the proposed method, the conventional sliding-window approach is avoided as it could make the detection slower. A new dataset KAPUK3700 is also introduced containing specifically for wrecked airplane detection.

2. RELATED WORKS

Based on our knowledge, similar detection only for locating airplanes, not the wreckages. Detecting multiple airplanes has been proposed by the work [13], a deep learning algorithm has been proposed detecting airplanes below the horizon with the feature able to reject moving vehicles. Images below the horizon also can have a highly cluttered background that makes detection nontrivial. Another deep learning-based algorithm for detecting airplanes has been proposed by [14], [15]; the method boosted the SNR (signal to noise ratio) to detect airplanes in a long-distance and leverage the saliency of objects to detect the occurrence of airplanes. In the long-distance images usually have lower SNR value and not contrasted between objects and the background.

Hand-crafted features and classifiers methods also have been proposed by [16], [17]. The algorithm is deployed in the airplanes and must be robust to distinguish the airplanes or not in a limited power of computation. While another [17] proposed the HMM-based method to detect airplanes below the horizon using hand-crafted features.

In wrecked airplanes, the object highly appears differently from the airplane, such as color-changing, pose, orientation, and even deform its parts, such as the wing could be separated away from the plane's body. Thus, a sophisticated algorithm to be invariant to appearance and deformable model is crucial in the wrecked airplane detection. Fortunately, the invariant appearance and deformable model have been studied in the computer vision literature. The notable works [3]–[6] have proposed a deformable part-based model (DPM) and have shown excellent performance in the past years. DPM intuitively models an object using its parts information. The part is arranged accordingly, and part distance is used as a cost to optimize the model during learning. For example, to detect a human, DPM models each part, such as head, arms, and legs, and placed accordingly as a human model. In this way, DPM showed very well to detect humans and invariant to pose, color, and deformed objects. To deal with wrecked airplane detection, DPM might need to be modified since the airplane's parts can be separated away quite far from the airplane's body or even leaving only the body.

In order to deal with the complexity of the wrecked airplane object, a deep learning-based method can be devised. Deep learning [18] has shown very well performance in many visual perception tasks, such as text detection [19], [20], victims detection [8], [21], target tracking [22], [23], object detection [11], [24]. Accuracy can be boosted by a deeper network. Interestingly, the work [8] has been proven theoretically and empirically that the last layers can capture more semantics information or abstraction; thus, it is more robust to variation of pose, color, scale, and deformable object, that is suitable to detect the wrecked airplane robustly,

3. ORIGINALITY

In this paper, we are the first who study wrecked airplane detection from the UAVs image using a deep learning approach. Differing from the standard airplane detection where the object is in full shape, the wrecked airplane object is very complex; the airplane's parts can be missing or separated away from the main body, even leaving only the main body. Moreover, the airplane can have a high variation of scale, pose, and color for example, due to burning due to severe accidents. Our system is based on a deep network for a more robust distinguishing between a complex wrecked airplane and the background. Our system is an end-to-end network, and it can learn the network simultaneously from input to output. Thus, the learned model is obtained solely from the training samples and eliminating hand-crafted features design at the earlier stages.

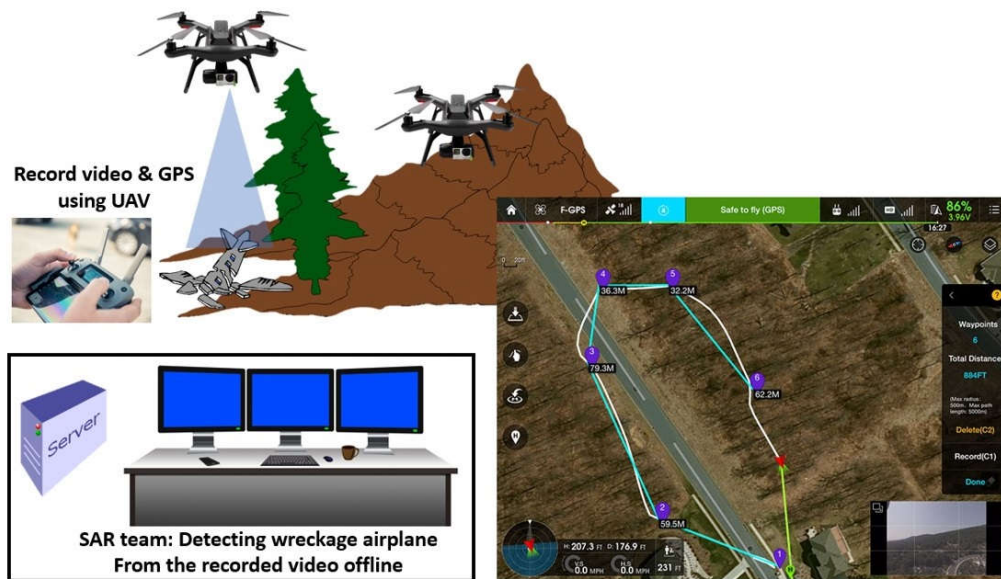


Figure 2. User / SAR team records a video and position, then SAR team analyze the recorded video to locate rapidly the occurrence of the wrecked airplane offline. Remote and autonomous UAVs can be selected from the operation.

4. SYSTEM DESIGN

Our overall system can be depicted in **Figure 2**. A drone is deployed to the suspected site either autonomously or remotely. Remote operation can be performed when the UAV is observable by the user, while the autonomous process is done by setting several waypoints, as shown in **Figure 2** bottom-right, for the UAV to generate paths on a fixed altitude. In this way, autonomous processes able to reach farther distances compared to remote operations. In practice, the autonomous process can reach 10 km radius distance from the user. Secondly, the recorded images or video together with the GPS position can then be sent to the search and rescue base station to inform the accident location rapidly.

In order to deal with the challenging wrecked airplane which can have a high variation of scale, pose, color, and deformable objects due to severe accident, the system is designed leveraging the benefit from the deep learning algorithm. The architecture of the proposed deep network is shown in **Figure 3**. During testing, each input image going through the proposed network (from the base model and the rest layers) is resized to 300x300x3 patch. Thus, for each image, there are five patches (single image as one patch, four divided

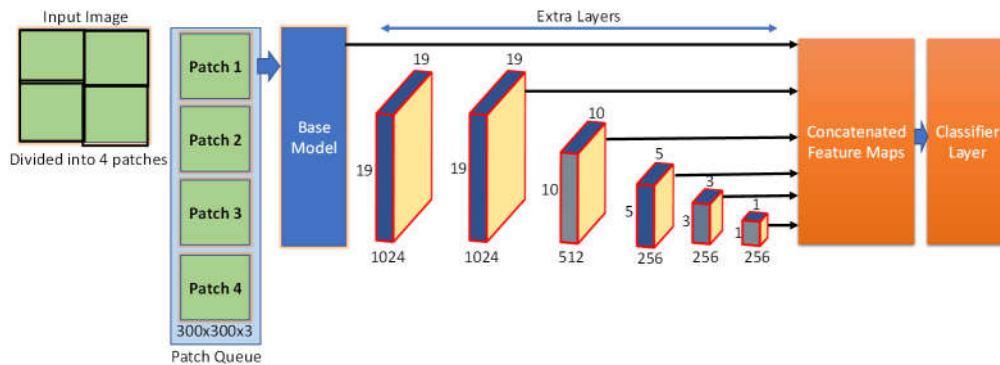


Figure 3. Architecture of the proposed deep network. Input image is divided into equal four patches. The patch is resized into 300x300x3 and arranged in a queue. Each patch is feed-forwarded to the network sequentially, from patch 1 to 5 (A single patch is from the full image itself). Several extra layers are added at the end of the base model to capture a more abstract or semantics information of the desired object.

regions as four patches) to be feed-forwarded to the network. This way is significantly faster compared to the standard deep learning that uses the common sliding-window method. In practice, five patches can help to increase recall scores. This is because the role of detecting wrecked airplane is not just about bounding box accuracy, but it is also the number of the estimated detection to the ground truth or actual wrecked airplane. We hope that we do not have a missing detection from the actual wrecked airplane. Reasonably accurate but still able to detect all the wrecked airplane is better.

Generally, an input 300x300x3 patch is feed-forwarded to our deep network resulting in 21 classes and produce a response map. From the map, a standard non-maximal suppression algorithm is employed to produce bounding boxes as an estimation of the detected objects. This procedure is repeated for the rest four patches.

A pre-trained deep learning model is used as a base model in the proposed network connecting to the extra six convolutional layers that have a decreasing size of the layer. The work by [8] has shown theoretically and empirically the deeper layers of the network. *The work suggests that the last layers help to capture better semantic information or abstract of the desired object. Thus, it is more robust to the pose and color variation, and deformable object, which is suitable for detecting wrecked airplane.* Fortunately, in order to leverage this abstract objectness, six extra deep convolutional layers are intertwined to the end of the base network model, which is suggested by the work [11]. Convolutional features from the base model and extra layers are concatenated followed by a fully-connected layer that acts as a classifier detecting the wrecked airplane class or other objects classes. Twenty classes from the Pascal VOC dataset are used by replacing the airplane class to the wrecked airplane class, and additional background class. Thus, a total of 21 classes; 1 positive class of the wrecked airplane and 20 negative classes of other objects. A fully detailed network architecture is shown in **Table 1**.

Table 1. The network architecture employed in this work.

Layer	Type	Size	Kernel Size	Feature Map
Input	Image	300x300x3	-	-
1	Conv	300x300x64	3x3	64
	Relu	300x300x64	-	64
2	Conv	300x300x64	3x3	64
	Relu	300x300x64	-	64
	Max Pooling	300x300x64	3x3	64
3	Conv	150x150x128	3x3	128
	Relu	150x150x128	-	128
4	Conv	150x150x128	3x3	128
	Relu	150x150x128	-	128
	Max Pooling	75x75x128	3x3	128
5	Conv	75x75x256	3x3	256
	Relu	75x75x256	-	256
6	Conv	75x75x256	3x3	256
	Relu	75x75x256	-	256
	Max Pooling	38x38x256	3x3	256
7	Conv	38x38x512	3x3	512
	Relu	38x38x512	-	512
8	Conv	38x38x512	3x3	512
	Relu	38x38x512	-	512
9	Conv	38x38x512	3x3	512
	Relu	38x38x512	-	512
	Max Pooling	19x19x512	3x3	512
Extra-1	Conv	19x19x1024	3x3	1024
	Relu	19x19x1024	-	1024
Extra-2	Conv	19x19x1024	1x1	1024
	Relu	19x19x1024	-	1024
Extra-3	Conv	10x10x256	1x1	256
	Conv	10x10x512	3x3	512
Extra-4	Conv	5x5x128	1x1	128
	Conv	5x5x256	3x3	256
Extra-5	Conv	3x3x128	1x1	128
	Conv	3x3x256	3x3	256
Extra-6	Conv	1x1x128	1x1	128
	Conv	1x1x256	3x3	256
Clasification	Fully Connected	21	-	21

4.1 Dataset

We have collected our dataset, namely KAPUK3700, containing 3700 images for the training. Around 1200 positive images of the wrecked airplane, as shown in **Figure 4**, are taken from UAV aerial view, and 2500 negative images from various objects, as shown in **Figure 5**. The wrecked airplane images are very challenging; the airplane's parts can be separated away from the main body, and for some cases leaving only the body. All the airplane's parts are highly deformable; the parts are mostly not in the full and ideal shape. The color might significantly change such as due to burning and high collision impact. The negative images are taken from the ImageNet dataset [25]. For testing, 200 wrecked airplane images are taken from UAV aerial view. It is noted that testing images differ from the training images but still within the same distribution. We made our dataset publicly available and can be

downloaded from the following URL
<https://github.com/anh0001/KAPUK3700>.



Figure 4. Several positive samples of the wrecked airplane images from our dataset.

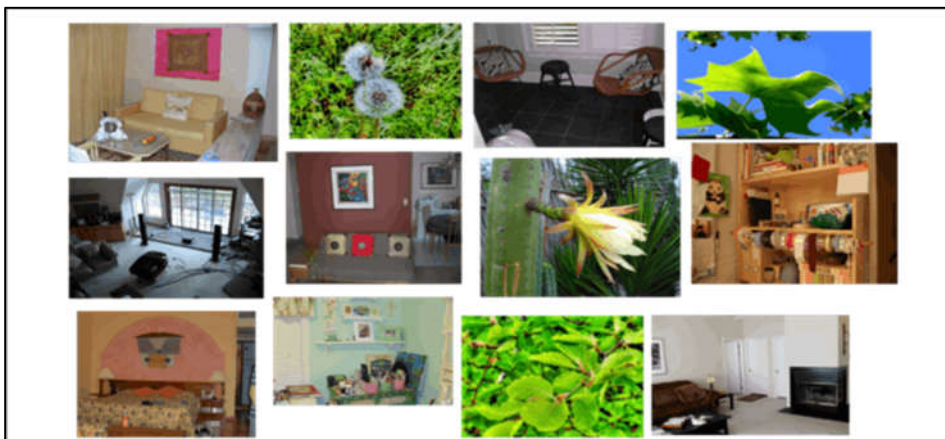


Figure 5. Several negative samples from our dataset.

4.2 Training

In order to perform deep network training, a sophisticated hardware PC desktop with GPU support is setup. The hardware has specifications as follows: CPU Intel i7 with a memory of 16GB and the GTX 1080ti as GPU. The programming language is done primarily in MATLAB with Matconvnet deep learning library [26]. During learning, we set the learning rate to 0.0001 for the first 75 epochs and 0.00001 after.

During the experiment, we use the pre-trained model of VGG-16 [27] as a base model in our deep network. In practice, this pre-trained model helps to generalize the model for any object detection. More importantly, it helps to train and build a deep network when we have not enough data to perform training. The time for training could also be reduced significantly. During training, we set to fix the base model; we do not alter the base model net and

weights. We only train the extra layers, as shown in **Table 1**. All the fully-connected layers at the last layers from the VGG-16 are removed leaving only the convolutional layers which are used as the base model. The last convolutional layers of VGG-16 are used as the first and second extra layers.

Fortunately, our proposed extra network is based on [11] to leverage the semantics or abstract objectness at the end of layers. To perform training, we employ a similar loss function. It applies pre-defined boxes of each scale. Thus the loss function consists of three losses: confidence loss, localization loss, and multi-box loss [28]. Confidence loss is the score prediction class from the corresponding ground truth classes. Localization loss is computed as the offset of each pre-defined box to the ground truth boxes. Multi-box loss is calculated as the error to the corresponding ground truth box.

During training, if the accuracy showed a relatively lower score, it could be biased much to the negative images as the negative images usually have higher numbers as compared to the positives. Moreover, to train a deep network, our dataset is considerably not enough in the number of data. The network could easily overfit the training data resulting in a lower performance during testing. In order to deal with this, we combine our dataset with the Pascal visual object class VOC2007 [29] and VOC2012 [30]. The Pascal dataset contains 20 classes namely, **Person**: person; **Animal**: bird, cat, cow, dog, horse, sheep; **Vehicle**: airplane, bicycle, boat, bus, car, motorbike, train; **Indoor**: bottle, chair, dining table, potted plant, sofa, tv/monitor.

We use the training and validation images from the Pascal dataset. More specifically, we combine 5,011 of VOC2007, 11,540 of VOC2012, and 1200 KAPUK3700, thus totaling of 17,751 images for training. We found that data imbalance between the wrecked airplane and negative objects is relatively high. Fortunately, the Pascal has an airplane class, thus the 1200 KAPUK3700 images are placed in the airplane class. Also, the wrecked airplane data images are extended by flipping, distorting, and rotating. We roughly have 22,000 images, and 20% of those images are used for validation during learning. In this way, data imbalance between wrecked airplanes and negative images could be reduced.

During training, a failed optimized model can be observed when the loss is not getting lower, or the loss fluctuates. In order to solve this, the learning rate can be multiplied by 0.1. Thus, for example, if the current learning rate is 0.0001, then it can be reduced to 0.00001. It is also a best practice to multiply the learning rate by 0.1 after a fixed epoch or when the resulted loss is not decreasing significantly.

5. EXPERIMENT AND ANALYSIS

During learning, we set the number of epoch to around 110, with the number of batches 574, and each batch contains 32 images. An epoch is when the entire dataset is passed forward and backward through the deep network. For each epoch contains the number of batches, while each batch contains the

number of images. This process is to ensure we have enough memory for the whole dataset during learning.



Figure 6. Several result of the proposed method for detecting wrecked airplane.

To evaluate the proposed method performance, the evaluation has been performed by a comparison between the estimated bounding boxes with the ground truth. We calculated the intersection between the estimated and ground truth bounding box. For each test image i , an estimated bounding box \hat{b} is overlapped with the corresponding bounding box ground truth b^{gt} , where $\hat{b}, b^{gt} \in (x, y, width, height)$. For the whole test images, a set of estimated bounding boxes \hat{B} is compared with a set of ground truth bounding boxes B^{gt} . Precision and recall are then calculated and are defined as follows,

$$precision \stackrel{\text{def}}{=} \frac{\sum_i \mathcal{J}(B_i^{gt}, \hat{B}, th)}{|\hat{B}|}$$

$$recall \stackrel{\text{def}}{=} \frac{\sum_i \mathcal{J}(B_i^{gt}, \hat{B}, th)}{|B^{gt}|}$$

Where the scoring $\mathcal{J}: (B_i^{gt}, \hat{B}, th) \mapsto \{1: (B_i^{gt} \cap \hat{B}) \geq th, 0 \text{ otherwise}\}$ is a function that scores 1 if the estimated bounding box intersects with the corresponding ground truth bounding box greater and equal than $th = 0.5$ and scores 0 otherwise. Plots between precision, recall, and average-precision are

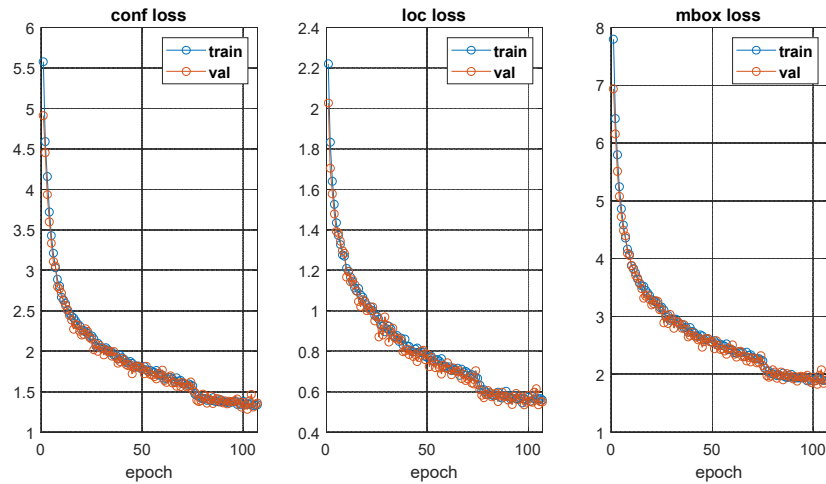


Figure 7. Confidence (left), localization (mid), and box (right) losses during learning model. It is noted that our model progressively moving towards an optimum model.

usually employed to show the performance of detection. *Basically, Precision measures how accurate the prediction is, recall measures how good the method finds all the positives, and average-precision (AP) is the area under the precision-recall curve.* Average-precision is usually employed to indicate the overall performance of detection.

Training loss can be shown in **Figure 7**. Three losses that are confidence loss, localization loss, and multi-box loss are shown in the figure with the increasing epoch. The output of the method can be shown in **Figure 6**, the wrecked airplanes can be detected successfully by the method. Unfortunately, few bounding boxes could not be fitted perfectly, this could be attributed to the heavily damaged parts of the airplane which likely have no effective visual features and appear as debris that visually similar with the background such as burned wood and soil. In practice, we consider this still acceptable as long as our method able to detect the desired object. High precision but missing the actual wrecked airplane is very critical as the user / SAR team can have a difficult time searching the accident sites resulting in the delay of performing help to the victims.

We also test the performance for different input sizes using the pre-trained model VGG16, tested using KAPUK3700 test images. We tried to increase the input net size to 512x512x3. The result is shown in **Figure 8**. The result indicates not a significant improvement can be achieved but increasing the complexity of the model; thus, more memory is occupied. Therefore, we still stick to 300x300x3 input net size.

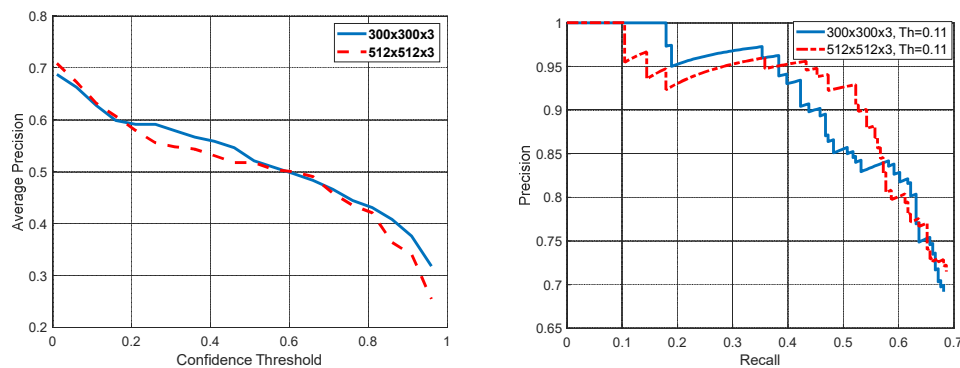


Figure 8. Comparison of average precision for the input network of 300x300x3 and 512x512x3 using pre-trained deep model VGG16 (a), and precision-recall (b) using confidence threshold 0.11. This result indicates that increasing the input net size does not improve much in the precision, with the trade-off between memory occupancies; then, the complexity of the model is also increase.

The quantitative performance of the proposed system is shown in **Figure 9**, and the comparison of AP with other methods is shown in **Table 2**. We evaluate the method using the KAPUK3700 testing dataset which contains 200 images. In the evaluation, precision and recall are shown based on the bounding boxes detection results. In the comparison, DeepVGG16 is the standard deep learning of VGG16, DeepExtraLayers is the deep network by [11] with extra layers as in our proposed. Our proposed shows very well as compared to other methods. It is noted that lower precision and high recall would likely be produced by predicting with many bounding boxes. On the other hand, high precision and lower recall would likely be produced by predicting with less number of bounding boxes. A right balance between high precision and high recall is the best performance.

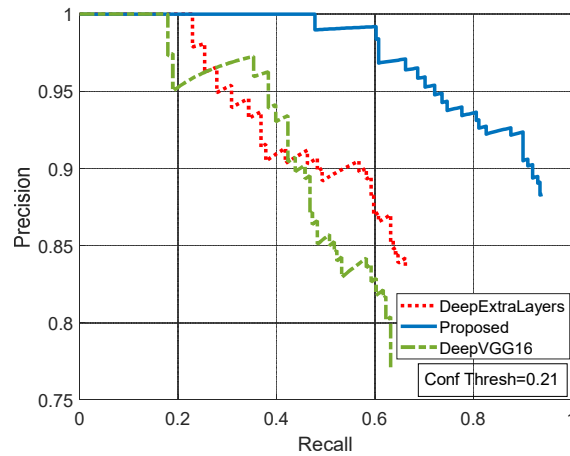


Figure 9. Our method (proposed) shows relatively well as compared with the other deep networks by achieving higher precision and recall (higher AP).

In **Table 2**, our method shows very well and successfully reaches AP=91.87% as compared to other methods. The confidence threshold of 0.21 is selected based on **Figure 10**.

Table 2. Average-Precision (AP) comparison

No	Deep Net	AP (%)
1.	DeepVGG16	59.11
2.	DeepExtraLayers	62.39
3.	Proposed Method	91.87

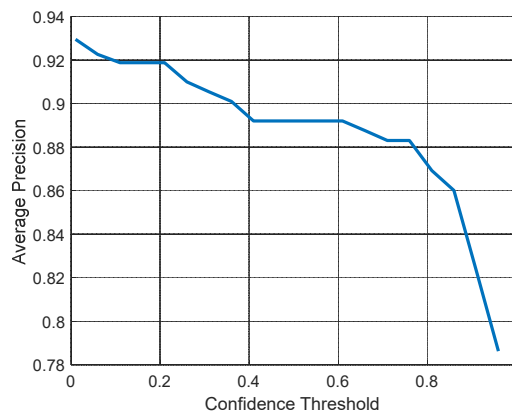


Figure 10. Confidence threshold selection using average-precision. Based on the curve, we selected the threshold to be 0.21. Lower than 0.21 value, the proposed might include quite a number of detection boxes around the desired object.

6. CONCLUSION

Quickly searching the accident site is a nontrivial task and requiring sophisticated technology in order to help the search and rescue team. Therefore, In this paper, an automatic system to detect wrecked airplanes has been elaborated. A new deep network has been proposed to distinguish robustly the wrecked airplane that has high pose, scale, and color variation, more importantly, high deformable object. The network leverages the end of the layers to capture more abstract and semantics information for robust wrecked airplane detection. The network is intertwined by adding more extra layers connected at the end of the layers. To reduce missing detection which is crucial for wrecked airplane detection, an image is then composed into five patches going feed-forwarded to the net in a convolutional manner. Experiments show very well and successfully reach AP=91.87%. For future work, it is interesting to improve the method to be able to detect the full wreckage with a sophisticated user interface for the SAR team. Moreover, the method can support multi drones for exploration.

Acknowledgments

This research has been supported by the Ministry of Research, Technology, and Higher Education of Indonesia with the grant *Penelitian Terapan* scheme number 7/E/KPT/2019.

REFERENCES

- [1] J. Y. CNN, **Search for missing MH370 plane ends but mystery remains**, *CNN*. [Online]. Available: <https://www.cnn.com/2018/05/29/asia/mh370-search-ends-intl/index.html>. [Accessed: 17-Jun-2019].
- [2] **Search efforts for missing aircraft in Papua continued on Thursday - ANTARA News**. [Online]. Available: <https://en.antaranews.com/news/133044/search-efforts-for-missing-aircraft-in-papua-continued-on-thursday>. [Accessed: 06-Jan-2020].
- [3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, **Object detection with discriminatively trained part-based models**, *TPAMI 2010*, 2010.
- [4] H. Azizpour and I. Laptev, **Object detection using strongly-supervised deformable part models**, in *European Conference on Computer Vision*, 2012, pp. 836–849.
- [5] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, **Cascade object detection with deformable part models**, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2241–2248.
- [6] J. Yan, Z. Lei, L. Wen, and S. Z. Li, **The fastest deformable part model for object detection**, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2497–2504.

- [7] A. Risnumawan, M. I. Perdana, A. H. Hidayatulloh, A. K. Rizal, and I. A. Sulistijono, **Towards an Automatic Aircraft Wreckage Detection Using A Monocular Camera of UAV**, in *2019 International Electronics Symposium (IES)*, 2019, pp. 501–504, doi: 10.1109/ELECSYM.2019.8901632.
- [8] I. A. Sulistijono and A. Risnumawan, **From concrete to abstract: Multilayer neural networks for disaster victims detection**, in *2016 International Electronics Symposium (IES)*, 2016, pp. 93–98.
- [9] M. D. Zeiler and R. Fergus, **Visualizing and understanding convolutional networks**, in *European conference on computer vision*, 2014, pp. 818–833.
- [10] P. Agrawal, R. Girshick, and J. Malik, **Analyzing the performance of multilayer neural networks for object recognition**, in *European conference on computer vision*, 2014, pp. 329–344.
- [11] W. Liu *et al.*, **Ssd: Single shot multibox detector**, in *European conference on computer vision*, 2016, pp. 21–37.
- [12] A. G. Howard *et al.*, **Mobilenets: Efficient convolutional neural networks for mobile vision applications**, *ArXiv Prepr. ArXiv170404861*, 2017.
- [13] J. James, J. J. Ford, and T. L. Molloy, **Below Horizon Aircraft Detection Using Deep Learning for Vision-Based Sense and Avoid**, *ArXiv Prepr. ArXiv190303275*, 2019.
- [14] J. James, J. J. Ford, and T. L. Molloy, **Learning to Detect Aircraft for Long-Range Vision-Based Sense-and-Avoid Systems**, *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 4383–4390, 2018.
- [15] G. Hu, Z. Yang, J. Han, L. Huang, J. Gong, and N. Xiong, **Aircraft detection in remote sensing images based on saliency and convolution neural network**, *EURASIP J. Wirel. Commun. Netw.*, vol. 2018, no. 1, p. 26, 2018.
- [16] S. Hwang, J. Lee, H. Shin, S. Cho, and D. H. Shim, **Aircraft detection using deep convolutional neural network in small unmanned aircraft systems**, in *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, 2018, p. 2137.
- [17] T. L. Molloy, J. J. Ford, and L. Mejias, **Detection of aircraft below the horizon for vision-based detect and avoid in unmanned aircraft systems**, *J. Field Robot.*, vol. 34, no. 7, pp. 1378–1391, 2017.
- [18] Y. LeCun, Y. Bengio, and G. Hinton, **Deep learning**, *nature*, vol. 521, no. 7553, p. 436, 2015.
- [19] **Aksara jawa text detection in scene images using convolutional neural network**, 2017.
- [20] A. Risnumawan, I. A. Sulistijono, and J. Abawajy, **Text detection in low resolution scene images using convolutional neural network**, in *International Conference on Soft Computing and Data Mining*, 2016, pp. 366–375.
- [21] I. A. Sulistijono *et al.*, **Implementation of Victims Detection Framework on Post Disaster Scenario**, in *2018 International*

- Electronics Symposium on Engineering Technology and Applications (IES-ETA)*, 2018, pp. 253–259.
- [22] M. K. Anwar *et al.*, **Deep Features Representation for Automatic Targeting System of Gun Turret**, in *2018 International Electronics Symposium on Engineering Technology and Applications (IES-ETA)*, 2018, pp. 107–112.
- [23] M. K. Anwar, A. Risnumawan, A. Darmawan, M. N. Tamara, and D. S. Purnomo, **Deep multilayer network for automatic targeting system of gun turret**, in *2017 International Electronics Symposium on Engineering Technology and Applications (IES-ETA)*, 2017, pp. 134–139.
- [24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, **Region-based convolutional networks for accurate object detection and segmentation**, *TPAMI 2016*, 2016.
- [25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, **Imagenet: A large-scale hierarchical image database**, in *2009 IEEE conference on computer vision and pattern recognition*, 2009, pp. 248–255.
- [26] A. Vedaldi and K. Lenc, **Matconvnet: Convolutional neural networks for matlab**, in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 689–692.
- [27] K. Simonyan and A. Zisserman, **Very deep convolutional networks for large-scale image recognition**, *ArXiv Prepr. ArXiv14091556*, 2014.
- [28] C. Szegedy, S. Reed, D. Erhan, D. Anguelov, and S. Ioffe, **Scalable, high-quality object detection**, *ArXiv Prepr. ArXiv14121441*, 2014.
- [29] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, **The PASCAL visual object classes challenge 2007 (VOC2007) results**, 2007.
- [30] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, **The pascal visual object classes challenge 2012 (voc2012) results (2012)**, in URL <http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html>, 2011.