International Journal of Aviation Science and Engineering
e-ISSN: 2715-6958 p-ISSN: 2721-5342
AVIA

Original Article

Bird Detection System Design at The Airport Using Artificial Intelligence

Khairul Ummah[1], M Fadly Hidayat[1], Denni Kurniawan[2], Zulhanif[3], and Javensius Sembiring[1,*]

[1] Faculty of Mechanical and Aerospace Engineering, Institut Teknologi Bandung. Labtek II, Jl. Ganesa 10, Bandung, 40132 Indonesia
[2] Department of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Brunei, Gadong BE1410, Brunei Darussalam
[3] Department of Mechanical Engineering, Faculty of Engineering, Universitas Lampung, Jl. Prof. Soemantri Brojonegoro no. 1, Bandar Lampung 35144, Indonesia

* Correspondence: javen.sembiring@itb.ac.id

Received: 18 May 2022; Accepted: 12 September 2022; Published: 30 December 2022

Abstract. Bird strike is a process of crashing between bird and airplane which occurs in flight phase. Based on data, there are 40 times bird strike occurs every day (FAA, 2019). There are lot of research that already conducted to decrease number of birds at the airport. But it is not given significant changes. Hence, it is needed a model that can detect bird at the airport so that we can decrease the number of birds. Study already conducted by comparing motion detection with object detection and filter which can be used to improve detection quality. Model already developed using YOLOv4 object detection with 71.89% mean average precision. It is expected that object detection can be developed to become a bird repellent system in the future.

Keywords: artificial intelligent, bird detection, crashing, strike, motion

1. Introduction

Bird strike is a process of crashing between bird and airplane or other flying object when the aircraft is in the takeoff, cruise, roll, and landing [1]. The interaction between birds and flying object is something that has been happening for a long time in the world of aviation [2]. Bird strikes can be a significant threat to aircraft, especially in the aircraft structure. This has been proven by several fatal accidents that occurred due to bird strikes, such as Callibraith Rogers accident, Eastern Airlines flight 375, and US Airways flight 1549 [3]. Apart from occurring on aircraft, bird strikes can also occur on other aerial vehicle such as rockets and drones. Interaction between birds and drones can occur due to various reasons, such as interaction that occur when drones are widely used to observe bird behavior [4].

The FAA has issued regulations regarding bird strikes to explain the importance of reporting regarding bird strikes and wildlife strikes, known as The FAA National Wildlife Strike Database for Civil Aviation and how to report these problems [5]. Several studies have been developed to address bird strike problem such as auditory (or bioacoustics), visual, lethal, and even manual surveillance methods [5,6]. This paper was developed not only to improve flight safety but also to ensure the safety of these animals.

The objective of the paper is determined the visual AI model for detecting birds at airports, study the processes that can be added so that the visual AI model is suitable for use in several conditions at airport, then determine the complementary features for visual AI model for detecting birds at the airports area. The study will be developed using Python programming language with motion

e-ISSN: 2715-6958 p-ISSN: 2721-5342

detection and object detection codes and environment. The program will be developed to detect birds visually with trees, buildings, and open fields as background, and in sufficient light, low light, and foggy condition. The research focus on birds' detection system and camera placement in airside of the airport.

2. Materials and Methods

2.1 Data Collection and Pre-Processing

The dataset used is a bird dataset obtained from Google Imageset and Google Image with a total of 1400 data. The data consist of birds of various sizes and types. Based on research, the ratio of training data and testing data is 80% and 20%. After all the necessary data has been obtained, it is necessary to adjust the data. Labelling process is carried out using a web-based application called Roboflow. The output will be a .txt file which will conform to the format required for YOLO.

2.2 Data Filtering

In general, images have unwanted noise which affect the quality and resolution of the image. Therefore, it is necessary to apply data filtering to the images [7]. Grayscale is one of the filters. Grayscale eliminates some color parameters, so that the time needed to do iteration can be shorter [8]. However, due to a decrease in image quality, additional filter is needed. Edge enhancement is used to sharpening the lines in the image. There are several conditions that cannot be detected when the dataset used is only given a grayscale filter, such as foggy conditions. To detect in such condition, additional 200 images are given an image dehazing filter [9].

Augmentation and tilling are carried out as a preparation stage for small object detection. Augmentation is carried out by cropping, rotating, flip, zooming in and zooming out of the dataset used. While tiling is done by segmenting images into several parts to get additional datasets and to provide variation to the dataset.

2.3 Data Training

Data training requires Machine Learning Tools, large amount of dataset, and computer with adequate specification [10]. Training is conducted in Google Colab with hyper parameters as follows:

Table 1. Hyperparameters

Hyperparameter	Value
Batch Size	64
Subdivision	16
Width	416
Height	416
Channel	3
Class	1

Transfer learning is required in a model developed with YOLOv4. Transfer learning is a process for applying data that has been studied in previous research [11]. At this training stage, transfer learning will continue for up to 4000 iterations or when the loss is less than 2.0.

2.4 Model Evaluation

Model evaluation is carried out on the YOLO system every 1000 iterations. Main performance metrics is used to determined performance of the model [9]. Main performance metrics are divided into three, which are including: the true positive (TP) with correct detection of a ground-truth bounding box; a false positive (FP) which is an incorrect detection of a non-existent object or a misplaced detection of an existing object, and also the false negative (FN) which is an undetected ground-truth bounding box. Each metrics have average point. Mean Average Precision (mAP) is the mean of every average point. Model considered acceptable if the mAP value is larger than 50%.

2.5 Realtime Object Detection

Object detection developed with computer vision will be used to detect birds. Therefore, the object detection needs to be developed to be able to do real time detection. The system is developed by combining the weights with a webcam video. When an object is detected, features such as alerts on the main server can be implemented to improve the quality of detection. This is done with two means, printed warning and sound warning using the python display library.

2.6 X-Y Coordinates

Coordinates are obtained by comparing pixel size and actual size in meters with a static object as reference. The reference object is holding position marking, which according to ICAO, it is 0.8 meters long and 0.6 meters wide. In pixels, the holding position marking is 30 pixels long and 20 pixels wide. A point is determined as reference point (0,0). The distance between the reference point and center point of a bounding box is used to calculate the coordinate. The calculation uses these below equations.

$$x_{frame} = \left(\left(\frac{x_{min} + x_{max}}{2}\right) - \frac{width}{2}\right) * \frac{0.8}{30} \quad (1)$$

$$y_{frame} = \left(\frac{height}{2} - \left(\frac{y_{min} + y_{max}}{2}\right)\right) * \frac{0.6}{20} \quad (2)$$

2.7 Motion Detection

Motion detection is a detection method by comparing frame changes for each frame. Motion detection is used as a comparison of the video quality obtained by object detection with YOLOV4. There are five videos that are used as trial videos, namely in dark condition, birds at close distance and bright condition, birds at far distance and bright condition, birds at far distance and foggy condition, and birds at close distance and foggy condition. Each video is given a different threshold value so that the quality of motion detection can be improved. Determination of the threshold value is done by iterating several times until the best detection is obtained. The following are the threshold for each video, presented in table 2.

Table 2. Threshold for Motion Detection

Condition	Threshold
Birds at close distance and bright condition	0.4
Birds at far distance and bright condition	0.2
Birds in dark condition	0.1
Birds at far distance and foggy condition	0.1
Birds at close distance and foggy condition	0.4

3. Results and Discussions

The model developed is used to detect birds in bright, dark, and foggy conditions. The trained data is a combination of data from various conditions, starting from small, large, close-up, long-distance birds, have been given image filtering in the form of greyscale, edge enhancing, image dehazing, augmentation, and tiling. The following is a matrix evaluation table of the training, as in table 3.

Table 3. Training Results

Parameter	Value
TP	216
FP	133
FN	56
Precision	0.62
Recall	0.79
F1-Score	0.70
mAP	71.89

Training is targeted to be carried out until the loss or residual obtained is at a number less than 2 with a total iteration of 4100 iterations. The training is carried out for 16 hours for one training with a total of 5 training repetitions, this is because each training produces weights for different object readings. The final training is done using 1100 training datasets and 300 testing datasets, which produces weights that can detect birds in bright, dark, and foggy conditions, and have good performance in small object detection.

Based on the results of the training data, a mAP value of 71.9% was obtained. This value can be categorized as sufficient and is considered valid for detecting objects. The low value is caused by the dataset that has many small birds. In testing, small objects that are not birds or noise are sometimes detected as birds. This can be seen from the high False Positive value which indicates that many non-bird objects are considered birds. The amount of training is limited to 4100 iterations with a loss of less than 2 to avoid overfitting, which will decrease the quality of the model.

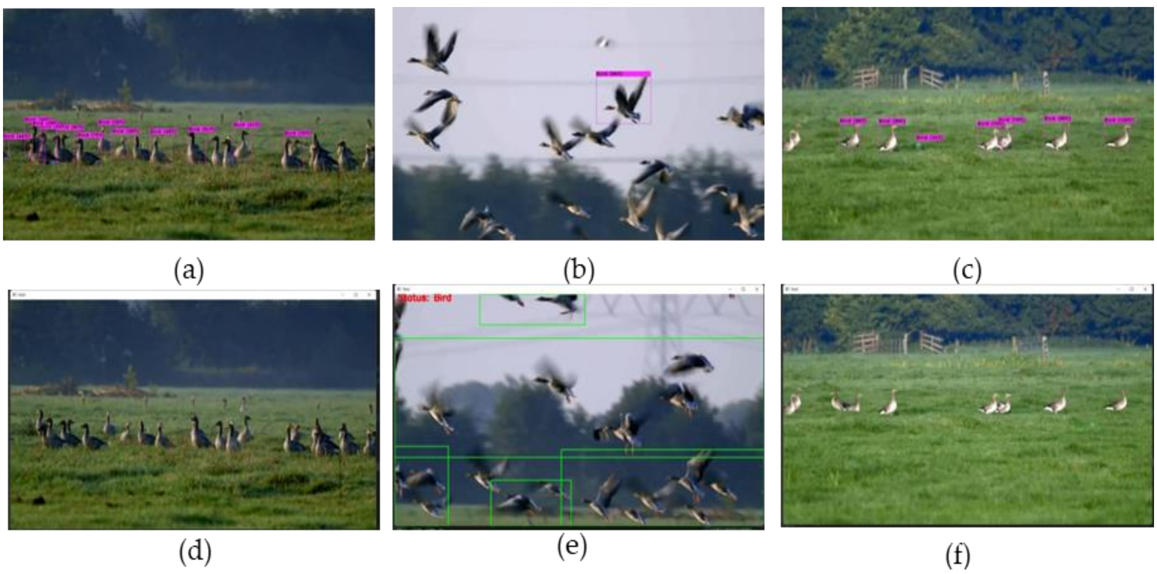

Figure 1. Big Birds in Close Range Analysis

Figure 1 (a), (b), and (c) shows the developed machine learning using YOLOv4 able to detect birds on grass or in flight with 30% to 90% accuracy. However, fig.1 (d), (e), and (f), which developed using motion detection unable to detect birds as good as YOLOv4 and have many false readings.

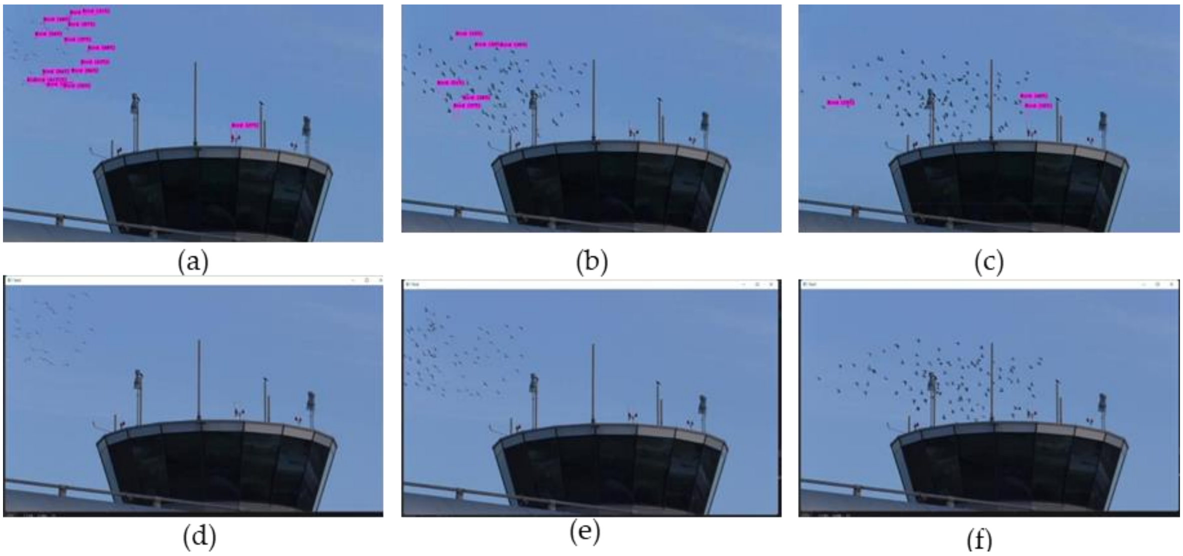

Figure 2. Small Birds in Long Range Analysis

Figures 2 (a), (b), and (c) shows that in long range, object detection still able to detect some birds, albeit in small percentage, while in fig.2 (d), (e), and (f), motion detection unable to detect any birds at all.

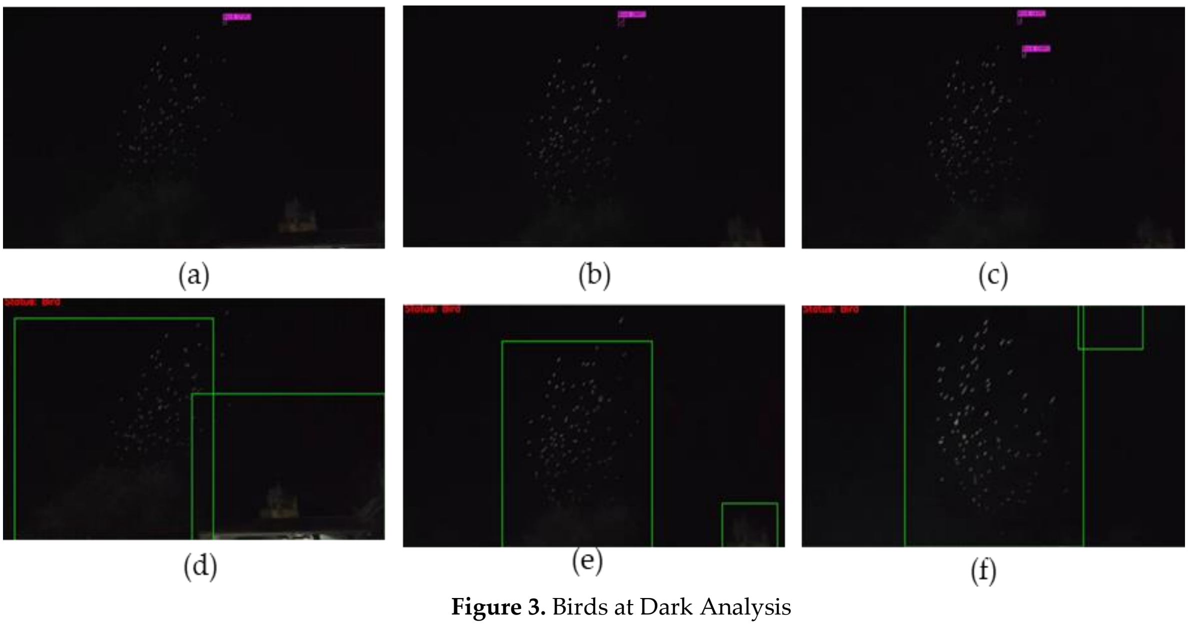

Figure 3. Birds at Dark Analysis

Figure 3 (a), (b), and (c) shows that in dark, object detection still able to detect birds with prediction value greater than 80%, while in fig.3 (d), (e), and (f) motion detection fail to detect birds and make incorrect detection.

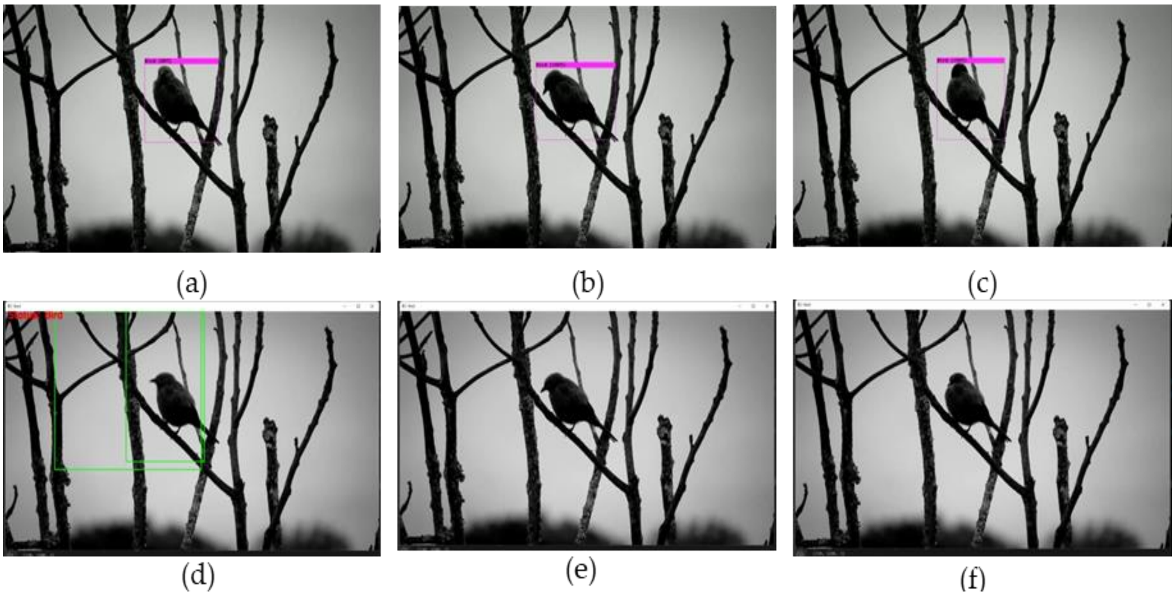

Figure 4. Close Range Birds at Foggy Condition Analysis

Figure 4 (a), (b), and (c) shows that in foggy condition, object detection able to detect 100% of birds in close range, while fig.4 (d), (e), and (f), motion detection fail to detect birds accurately.

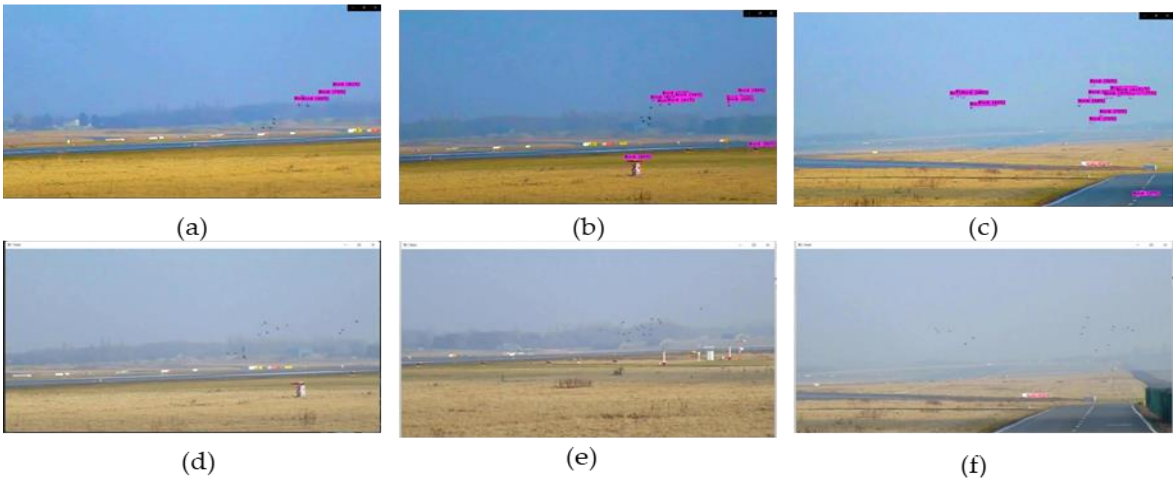

Figure 5. Long Range Birds at Foggy Condition Analysis

Figure 5 (a), (b), and (c) shows that object detection able to detect birds in long distance and foggy condition with accuracy of 30%-90%. Meanwhile, fig.5 (d), (e), and (f) shows that motion detection unable to detect any birds.

Object detection is suitable for detecting birds in bright, dark, foggy, long-range, and close-range condition. The accuracy of the detector is varied ay 30 to 100 percent. Meanwhile, motion detection is unable to detect most of the birds and have many incorrect detection even with customized threshold. Bounding box produced by object detection is more accurate than motion detection's bounding box.

The model is design to detect objects in images. This feature is the basic of object detection which has been determined and trained. The model is also capable to detect objects in video. Grayscale, edge enhancement, and image dehazing filters are applied to give the model the capability to detect object in various environment conditions. Augmentation and tiling also enables the model to detect object in long distance or small object detection.

The model has a feature to detect object in real time. It requires additional python library and other improvement to detect birds in real time. It also has capability to produce audio and writing output should a bird was detected in real time.

Figure 6. Real Time Detection with Audio and Writing Warning

The developed object detection is also able to map birds' position in two-dimensional space. It is able to produce coordinate in x and y with respect to the center of the frame. Not only in two-dimensional space, but also in three-dimensional space by adding bird eye mapping. This feature enables operator to find out the position of birds in airport as bird surveillance.

Figure 7. Object Detection with x,y,z coordinate with respect to camera

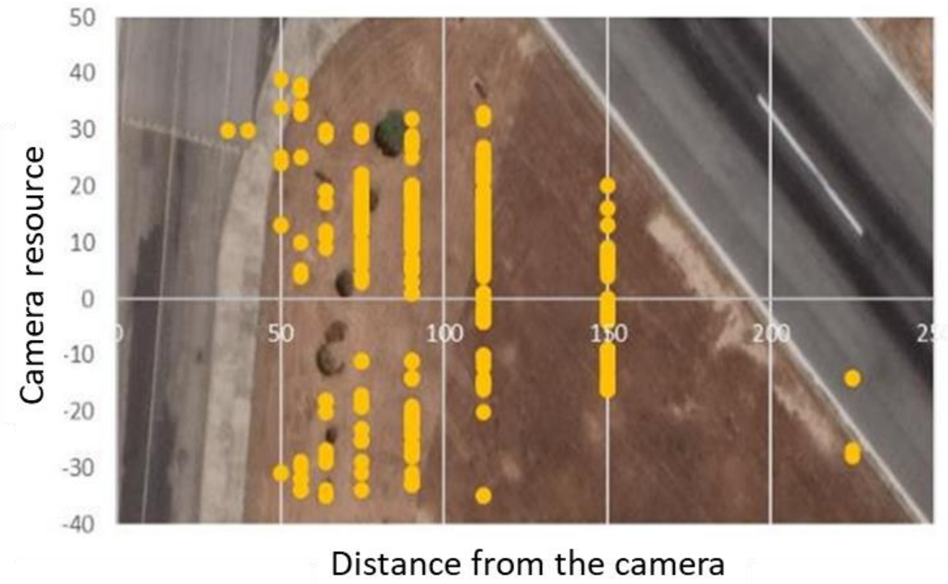

Figure 8. Bird Eye Mapping

The model has also a dashboard feature to provide information of the number of the birds in certain areas. This feature is used to determine whether the area is at risk of bird strike. The ratio between area occupied by birds and the total area of the frame is calculated as follows.

$$Ratio = \left(\sum area(bbox)/area(frame)\right) \quad (3)$$

If the ratio is < 0.3 then the area will be declared safe from birds, but if the ratio is > 0.3 then the area will be declared unsafe from birds, so it is necessary to take preventive measures to drive away birds. Here's an example dashboard based on the video used.

Boundary Box Total Area	Average Box Area	Frame Area	ratio	Condition
290708	669.834101	921600	0.315438	Affected by bird

Figure 9. Birds Detection Dashboard

Based on the experimental results and analysis that has been carried out, the developed object detection has succeeded in detecting birds in pre-determined conditions, but there are several things that can be improved. The dataset used can be given more variations to increase the mAP. This can be done by providing a bird dataset with more diverse conditions and adding more non-bird datasets. In addition, the system can be developed to detect various types of birds, because there are many types of birds at airports and require different monitoring.

The object detection can be further developed to design a bird repellent laser model. It can be achieved by integrating the program into a simple laser shooter system that can beam a laser when sighted birds are detected. The object detection can be developed further to determine the location of birds in 3 dimensions at airports and make it easier for airports to reduce the number of birds found at airports, by implementing a 3D surveillance system using several cameras that highlight certain locations, as well as combining the frames of each image into a three-dimensional shape.

The system can be developed into bird surveillance at airports by installing cameras from several points that have a high potential for birds, such as areas with tall grass or puddles. Then with the help of the coordinate system that has been made, a bird distribution plot at the airport can be mapped from the top view. So that the bird repellent car with bioacoustics can directly focus on areas that have a large number of birds.

The system can be developed to be a bird repellent by using a laser to replace manual shooting of birds. The laser can be directed at birds and will disturb the birds but not injure the birds, so that the birds can be driven out of the airport. The model can be developed with a camera with a minimum resolution of 1080p which can be used to detect objects with additional filters specifically adjusted to dark and bright conditions which can be turned on and off through the control system of the airport.

4. Conclusions

Based on the research and system development that has been carried out, the following conclusion are obtained that the visual AI has been developed with the YOLOV4 system, resulting in a mAP of 71.89%. To make object detection able to detect birds in various conditions, image filtering process is needed which will be given to each environmental condition at the airport, namely: Greyscale and edge enhancement allow YOLOv4 to detect birds in both dark and bright conditions. The image dehazing allows YOLOv4 to detect birds in foggy conditions, while the augmentation and tiling allow YOLOv4 to detect birds at long distances or in small size. The object detection model with YOLOv4 has higher accuracy and better consistency compared to motion detection in detecting birds in bright, dark, and foggy conditions. The model can also be added real time features and x-y-z coordinates to improve the quality of the model.

References

[1] E. Regulation, *Bird Strike History of Birds Strike* (2014).

[2] I. C. Metz, J. Ellerbroek, T. Muehlhausen, D. Kugler, and J. M. Hoekstra, Aerospace **7**, 1-20 (2020).

[3] Z. Naukowe, P. Slaskiej, and S. Transport, Sci. Jour. of Sliesian Univ. of Tech. **108**, 3-8 (2020).

[4] M. Lyons *et al.*, Aust. F. Ornithol. **35**, 51-56 (2018).

[5] A. K. Jha, S. Sathyamoorthy, and V. Prakash, Procedia Struct. Integr. **14**, 416-428 (2019).

[6] A. F. El-Sayed, *Bird strike in aviation: Statistics, analysis, and management* (2019).

[7] M. Mirmozaffari, Eng. Trans. **1**, 1-5 (2021).

[8] V. M. Mohan, R. K. Durga, S. Devathi, and K.S. Raju, Adv. Intell. Syst. Comput. **381**, 353-361 (2016).

[9] R. Padilla, S. L. Netto, and E. A. B. Da Silva, Int. Conf. Syst. Signals Image Process. **7**, 237-242 (2020).

[10] C. Janiesch, P. Zschech, and K. Heinrich, Electron. Mark. **31**, 685-695 (2021).

[11] C. C. Aggarwal, Data Classif. Algorithms Appl., 657-665 (2014).