Faraway Small Drone Detection based on Deep Learning

Jae Kyun Kim[1], Young Jin Shin[2], Jin Ho Kim[3], Jung Kn Kim[4], Jae-Hoon Kim[5], and Seong Dae Lee[6],*

[1]Department of Computer Engineering, Korea Maritime & Ocean University, Korea
[2,5,6]Department of Control and Automation Engineering, Korea Maritime & Ocean University, Korea
[3]Department of IT Engineering, Korea Maritime & Ocean University, Korea
[4]IST Corporation, Korea

Summary
The industry of small UAV (Unmanned Aerial Vehicle) such drones has recently been developing rapidly. The global UAV market is growing fast and is centered on the commercial and hobby drone markets, but it initially started with the military market. Due to the expansion of the drone market, related technologies including the miniaturization of drones and the increase in battery capacity have also been rapidly developing. As drone technology advances, the malicious use of drones including privacy violations and the transportation of small bombs have become serious problems. To prevent such problems, an anti-drone technology is also being developed. This paper carried out research on the anti-drone technology of drone detection. Studies on existing drone detection technologies have mainly been carried out through radars that converged with various sensors. The radar-based detection has problems such as the high cost and the need for an expert to operate it. Although studies using general web camera images without such radars have been conducted, they targeted only big and visually distinguishable drones. This study carried out exploration of the visually indistinguishable small drone images. The 2,085 images were used as a training set. 98.4% of the results were obtained through a mAP assessment of total 95-minutes video clip using the YOLOv3 model.

Key words:
Drone, Object Detection, Anti-drone, Deep Learning

1. Introduction

A small UAV is a flying object without a pilot. The industry of small UAV, known as “drones,” is developing very fast [1]. The global drone market has grown from when it was initially centered on the military market. Recently, the commercial drone market has been expanding rapidly, and the total market size is expected to grow from USD 11.4 billion in 2019 to USD 20.2 billion in 2025 [2]. Due to the growth of the drone market, drone technologies including the increase in the battery capacity and the miniaturization of drones have been rapidly developed. As drone technology develops, however, incidents of the abuse of the technology are gradually increasing. There are many serious problems related to drones over the world; privacy violation such as hidden camera and terrorism threats such as transportation of chemicals, radioactive materials, and small bombs with malicious objects [3]. To prevent malicious use of drones, many industries are also developing anti-drone technology which has combined an unmanned flying object detection technology and an unmanned flying object incapacitation technology. For the unmanned flying object detection technology, someone is developing the drone detection technology using various sensors like an ultra-high resolution radar, a microphone, a camera, and radio frequency [4]. In the technology, however, there are many drawbacks as follows: It is extremely costly, not portable, not easy to install the system for home and individual use, and necessary for an expert to operate the system. Previous studies on drone detection are mainly about detection through radars [5-6] and detection using specially shot images [7], but there is a bit lack of studies using general web camera images. As for the drone detection study using existing images, experiments were carried out on big drones and from the visually distinguishable drones. In order to ease the pains, we propose to detect visually indistinguishable small drones in images using a convolution neural network (CNN). CNN is one of the deep learning technologies and is widely used for processing images. This paper consists of the following sections. Section 2 describes related works such as CNN and the YOLO [8] algorithm. Section 3 describes experiments and Section 4 evaluates the performance of the proposed system. Finally, Section 5 draws conclusions and give a short outlook on future works.

2. Related Works

2.1 CNN

CNN is well known as deep learning algorithms and is widely used for processing images as well as has shown good performance in the image processing field. They are also known as shift invariant or space invariant artificial neural networks [9]. CNN is used in various fields including object classification [10], object detection [11], and image division [12]. In general, CNN composes two parts of feature extraction and classification as shown in Fig. 1. The former consists of a convolution layer and a pooling layer. The convolution layer is the layer that reflects the activation function after applying the filter to the input data. The pooling layer is the layer that consolidates the features of

Manuscript received January 5, 2020
Manuscript revised January 20, 2020

the extracted images from the convolution layer. The latter identities many objects or labels through general feed-forward layers.

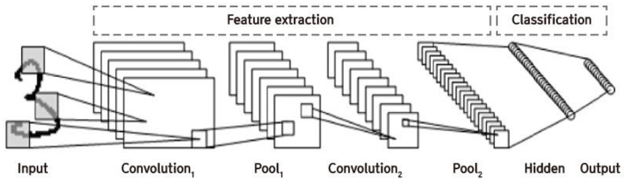

Fig. 1 Basic CNN architecture.

CNN has the following differences compared to the existing fully connected neural network: 1) maintaining each layer's input/output data shape, 2) maintaining the image's spatial information, 3) perceiving the features of the adjacent images effectively, 4) extracting and learning of images with plural filters, 5) consolidating the extracted images using the pooling layer, using very small parameters by sharing filters.

2.2 Object Detection

Drone detection is carried out using the object detection algorithm in this paper. Object detection involves solving the problem of a classification issue that classifies an object among a variety of objects from images, and a localization problem indicating location information through the bounding box where the object is located. Object detection using in-depth learning can be divided into two stages: Stage 1 detector simultaneously solving the localization and classification problems and Stage 2 detector solving the subsequent two problems. Object detection expresses an object to be detected with the bounding box in learning. The bounding box is used as a ground truth in learning by labeling the location, area, and width of an object in an image. The existing object detection algorithms, are the YOLO and the SSD families, which are a sort of the stage 1 detector, and the R-CNN family as the stage 2 detector. R-CNN needs several stages for the learning process, and so the object detection speed becomes slow, which is a shortcoming [13]. The Fast R-CNN, which is an improvement, receives and processes the input of the object candidate area in one go [14]. However, a lot of time is required for the process calculating the candidate area for object detection. The Faster R-CNN is a method to complement the Fast R-CNN, and is a method of reducing the calculation time by adding the neural network to calculate the candidate area to the network for object classification [15]. The stage 2 detector family is highly accurate, but more detection time is required, which can be a disadvantage. The YOLO, a stage 1 detector, classifies images and identifies the location information within the images in one go, and so it is suitable for this study that requires real time image classification. Consequently, this study performed drone detection using YOLO, a stage 1 detector.

2.3 YOLO

YOLO looks at images once by regarding the bounding box location within images and the class classification probability as one regression problem, and then YOLO estimates the types and locations of objects. In YOLO, the class probability of multi-hole anchor boxes belonging to the grid cell in which the center of the labeled object's bounding box is located. The object existence probability, the object center's location (x, y), and the width and height of the bounding box are estimated after dividing the input images with grid on the object. The YOLO network generates two output values, and has a confidence score information that calculates by using the bounding box information, the class probability and object existence probability as output. The confidence score is the score of whether an object exists within a bounding box and how much the class is reflected, and it can be calculated as equation (1). Here, IoU (Intersection over Union) is the abbreviation of the intersection over union, and it is the size of the intersection size/union size of the correct answer box and prediction box.

$$confidence\ score = P(class) \times IoU \qquad (1)$$

Compared to the Faster R-CNN which shows the speed of seven frames per second, YOLO was used in this study, as YOLO can process 155 frames when a fast version of 45 frames per second is considered as more effective in the drone detection field where real time detection is needed.

2.4 Kalman Filter

The Kalman Filter is a regression filter estimating the linear dynamic state based on the measured values containing noise [16]. The Kalman Filter estimates the combination distribution of the current state of values based on the past measured values. The Kalman Filter consists of two stages: prediction and update. In the prediction stage, the current stage variable values and accuracy are predicted. After the current stage values are actually measured, the current stage variables are updated by reflecting the difference between the predicted measured values and the actual measured values, based on the previously estimated stage variables in the update stage. The Kalman Filter is operated recursively. That is the current value is estimated based on the previously measured value, and the measured value or estimated value, except for the previous time-step stage value, is not used. Each estimated calculation consists of two stages. In the first stage, the measured value is predicted when the user input is given in the stage on the previous step-time measured state is calculated. This stage is called a prediction stage. In the next stage, the current state is estimated based on the above predicted measured value and the actual measured value. This stage is called an update stage. Depending on the system, the update stage occurs sometimes. While the predication stage is carried out several times, the update stage is sometimes only performed once.

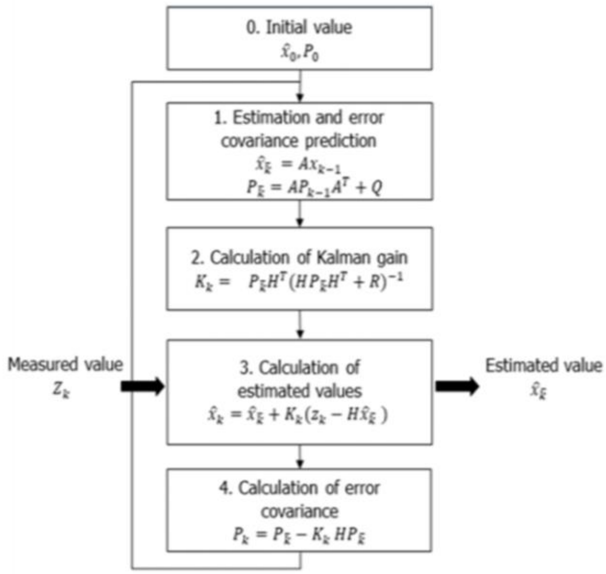

Fig. 2 Kalman Filter workflow.

3. UAV Detection

YOLO, a model used in this paper, detects an object in a single frame by splitting the image into frames. Therefore, there is a high possibility of misdetection. To solve this problem, we use the Kalman filter. The Kalman filter receives the previous frame as an input, detects recursively, and tracks the route. The workflow of this model is shown in Fig. 3:

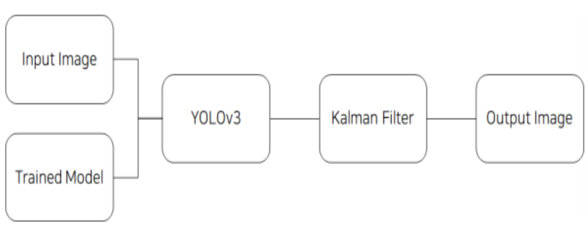

Fig. 3 Workflow of proposed system.

3.1 Generating Training Data

First of all, there are no training data for detecting visually indistinguishable drones. Therefore, it is necessary that we generate training data for our experiment. The input images used for training are shown in Fig. 4, which contain indistinguishable drones labeled using a box.

Fig. 4 Drone image as learning data.

Next, we should convert this training data into the input format which is suitable for YOLO model. The converted input format is expressed as real numbers as shown in Fig. 5:

0 0.506667 0.372222 0.066667 0.100000

Fig. 5 Converted result of Input data.

The results of conversion are saved as text files, and each column indicates class number, x and y coordinate of the object, and the width and height of the box.

3.2 Training the Detection Model

We additionally train the original detection model, YOLO, using the train data that we generate in 3.1. The output format of the model is same as the input.

3.3 Applying Kalman Filter

In this stage, we divide the output into time-steps and apply Kalman filter to each one. As stated in 2.4 above, the Kalman filter estimates the current state recursively by using the previous time-step state.

3.4 Path Tracking of Drone

In this stage, we get central coordinates using the x and y coordinate, and width and height values of a drone from the previous stage, and then store it in the memory. After that, we can track the path connecting central coordinates of each frame with a line.

4. Experimental Results

4.1 Experimental Environment

In the data set used for this experiment, the small drone image, which is visually indistinguishable, does not exist in open data set, and so this study made data set using the labeling tool. Among a total of 5,734 images, this study labeled 2,085 images but not in the other 3,649 images where a drone did not appear. The labeling was used as a labeling tool such as labelimg. As for the hyper parameter used for learning, it had a batch size of 64, and the subdivision value is 8, which is the size for learning, 2,000 epochs. Table 1 shows the computing resources used for learning.

Table 1: Computing Environment

OS	Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-66generic x86_64)
CPU	Intel (R) Xeon(R) CPU E5-1660 v3 @ 3.00GHz
RAM	64GB
GPU	TITAN RTX 24GB
CUDA Version	10.1

The model used for learning was the YOLOv3 darknet53.conv.74, and Fig. 6 shows its structure.
An experimentation is evaluated with the mAP (mean Average Precision) used for the object detection area in the computer vision. As a result, Fig. 7 shows the graph indicating the loss of learning process and a 98.4% mAP result was obtained.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
	Convolutional	32	1 × 1	
1×	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
2×	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
8×	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
	Convolutional	256	1 × 1	
8×	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
	Convolutional	512	1 × 1	
4×	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Fig. 6 Model architecture of learning process.

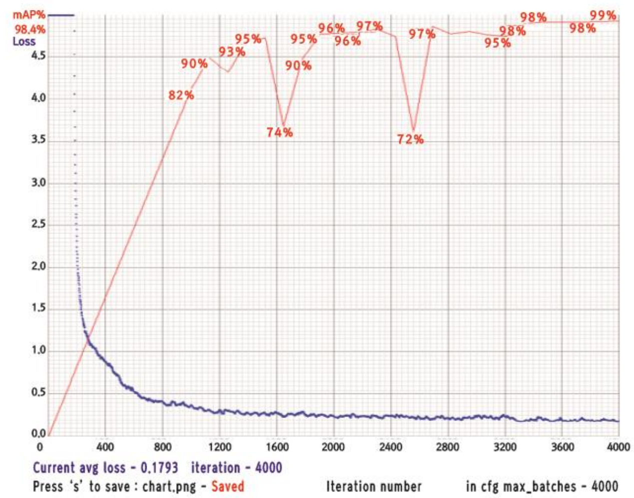

Fig. 7 Loss and mAP Graph.

The results of detection and tracking of drone on the actual test data are shown in Fig. 8.

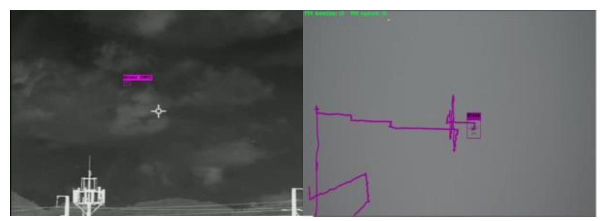

Fig. 8 Results of detection and tracking of drone.

5. Conclusions

This study carried out real time drone detection research using general web camera images without specialist equipment or technology used in existing drone detection technology. As a result of performing a mAP assessment using a total of 95 min-images, a 98.4% result was obtained. According to the experiment result, real time detection with more than 30 frames per second was possible, but overfitting to the training data is suspicious, due to similarity of the learning images. In the training data used for learning, only one drone was shown in one image, and so multi-detection was impossible. For further study, additional learning is planned by collecting multi drone images.

Acknowledgments

Following are results of a study on the "Leaders in INdustry-university Cooperation +" Project, supported by the Ministry of Education and National Research Foundation of Korea.

References

[1] H. G. Kim, "Analysis on Patent Trends for Industry of Unmanned Aerial Vehicle", Proc. of The Korean Entertainment Industry Association, pp.90-93, 2017.
[2] Commercializations Promotion Agency for R&D Outcomes,S&T Market Report, Vol. 67, pp.10, 2019.
[3] S. H. Choi, J. S. Chae, J. H. Cha, and J. Y. Ahn, "Recent R&D Trends of Anti-Drone Technologies", Electronics and Telecommunications Trends, Vol. 33, No.3, 2018.
[4] D. H. Lee, "Convolutional Neural Network-based Real-Time Drone Detection Algorithm", Journal of Korea Robotics Society, vol. 12, no.4, pp.425-431, 2017.
[5] C. H. Seol, and Y. C. Chung, "X-band Phased Array Antenna Radar Design For Drone Detection", Proc. of The Institute of Electronics and Information Engineers, pp.363-365, 2019.
[6] B. T. Koo, S. H. Han, J. H. Choi, G. S. Kim, Y. S. Beak, D. W. Kim, L. S. Eo, J. Y. Kee, and M. Park, "Implementation of AESA based Intelligent Radar System for Small Drone Detection", Proc. of The Institute of Electronics and Information Engineers, pp.1466-1467, 2019.
[7] K. W. Lee, K. M. Song, J. H. Song, C. H. Jung, W. K. Lee, M. J. Lee, and Y. K. Song, "Implementation of Radar Drone Detection Based on ISAR Technique", The Journal of Korean Institute of Electromagnetic Engineering and Science, pp.159-162, 2017.
[8] REDMON, Joseph, et al, You only look once: Unified, real-time object detection, In: Proceedings of the IEEE conference on computer vision and pattern recognition, p. 779-788, 2016
[9] Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", Proc. of International Conference on Neural Information Processing Systems, pp. 1097-1105, 2012.
[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, 2016.
[11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single Shot Multibox Detector", European Conference on Computer Vision, pp. 21-37, 2016.
[12] Kolesnikov and C. H. Lampert, "Seed, expand and constrain: three principles for weakly-supervised image segmentation", European Conference on Computer Vision, pp. 695-711, 2016.
[13] R. Girshic, "Fast R-CNN", Proc. of the IEEE International Conference on Computer Vision, pp.1440-1448, 2015.
[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks", Advances in Neural Information Processing Systems, pp.91-99, 2015.
[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, and A. Rabinovich, "A Going Deeper with Convolutions", Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp.1-9, 2015.
[16] G. C. Kim, S. H. Son, M. S. Kim, J. W. Jeon, I. J. Lee, J. H. Lee, J. H. Cha, H. C. Choi, Object Tracking Method using Deep Learning and Kalman Filter, Jounal of Broadcast Engineering, Vol. 24, No.3, pp. 495-505, 2019

Jae Kyun Kim received the B.S. and now in M.S course. in Computer Engineering from Korea Maritime & Ocean University in 2018. His research interests include Natural language processing and Machine learning.

Young Jin Shin is an undergraduate student in Control and Automation Engineering from Korea Maritime & Ocean University. His research interests include Network and Artificial Intelligence.

Jin Ho Kim is an undergraduate student in Information Technology Engineering from Korea Maritime & Ocean University. His research interests include Health care, Senior Industry and IT.

Jung Kn Kim received the M.S. and Ph.D. in Control and Instrumentation Engineering from Korea Maritime & Ocean University in 2001 and 2008, respectively. He is the CEO in IST(Intelligent System Tchnology) since 2012. His research interests include high performance surveillance system design and integration.

Jae Hoon Kim received the M.S. and Ph.D. in Computer Engineering from Korea Institute of Science & Technology in 1988 and 1996, respectively. He work at Electronics and Telecommunications Research Institute 1988-1997, He has been a Professor for Korea Maritime & Ocean University since 1997. His research interests include Natural language processing, Information retrieval, Corpus linguistics, Sentiment analysis.

Seong Dae Lee received the M.S. and Ph.D. in Computer Engineering from Korea Maritime & Ocean University in 2001 and 2007, respectively. He has been a Professor for Industry-University Cooperation in Korea Maritime & Ocean University since 2009. His research interests include Database, Big Data Analysis, Data Mining and Machine Learning.