Master of Science in Engineering

FOD Detection Model using Virtual Image and Drone

February, 2022

Department of Architecture & Architectural Engineering

The Graduate School

Seoul National University

Changhwa Hong

FOD Detection Model using Virtual Image and Drone

by

Changhwa Hong

A thesis submitted in partial fulfillment

of the requirements for the degree of

Master of Science in Engineering

Seoul National University

2022

FOD Detection Model using Virtual Image and Drone

December, 2021

Approved by Dissertation Committee:

Seokho Chi

Moonseo Park

Changbum Ryan Ahn

FOD Detection Model using Virtual Image and Drone

지도교수 박 문 서

이 논문을 공학석사 학위논문으로 제출함

2021년 12월

서울대학교 대학원

건 축 학 과

홍 창 화

홍창화의 공학석사 학위논문을 인준함

2021年 12月

위 원 장	지 석 호	(인)
부 위 원 장	박 문 서	(인)
위 원	안 창 범	(인)

Abstract

FOD Detection Model using Virtual Image and Drone

Changhwa Hong
Department of Architecture
The Graduate School
Seoul National University

FOD (Foreign Object Debris) is a concept that means an object that is located in an inappropriate place in the airport area and can cause damage to aircraft and personnel. FOD is a serious threat to flight safety and causes enormous direct and indirect economic damage every year. Although FOD detection and removal is an important task in airports around the world, most airports still use the traditional method using personnel or vehicles. This method not only consumes a lot of resources but also contains the risk of missing detection. And it has a limitation in that aircraft operations must be stopped during detection work. It also limits obtaining the details necessary to prevent FOD occurrences. In order to solve this limitation, recently, FOD detection studies using computer vision are being conducted.

However, previous studies used images with unsuitable scale for field application as training data. Images taken at a close range were used, which was not suitable for the field conditions where a small FOD was found in a large area such as a runway. Models trained from these data have limitations in being used in the actual field. Also, even

shooting data directly at a suitable scale would be a task that undermines the effectiveness of computer vision. Like other computer vision models, the process of preprocessing the training data also consumes a lot of time.

To solve these limitations, this study suggests a FOD detection model using virtual images as training data. By using the virtual image, it was possible to efficiently generate training data with a scale suitable for field application. Through the combination of FOD materials, backgrounds and solar conditions, the diversity of data was also secured. In addition, the time consumed in the data preprocessing process was greatly reduced. Images taken with a drone were used as test data. Three experiments were conducted. The first is a comparison of the performance of a model trained with a real image and a model trained with a virtual image. The second is the performance test of the model trained with virtual images taken at close range, similar to previous studies. The third is the performance test of the model trained with virtual images taken from a long distance in consideration of field application. As a result of the experiment, the effectiveness of the FOD detection model trained with virtual images was proven. And the detailed conditions for using this model in the field were analyzed.

The results of this study can contribute to the establishment of a FOD detection system using computer vision and drones. The results can solve the limitations of existing detection methods and previous studies. Furthermore, it can be used as a method of collecting basic data necessary to prevent and effectively remove FOD.

Keyword : FOD, Computer Vision, Object Detection, Virtual Image, Drone, Field Applicability
Student Number : 2020-25569

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

1.1. Research Background

Foreign Object Debris(FOD) is any object, live or not, located in an inappropriate location in the airport environment that has the capacity to injure airport or air carrier personnel and damage aircraft (FAA, 2020). In Korea, 『Airfield Facility Installation Standard』 defines it as an unintended foreign object or fragment that may be a temporary risk factor for aircraft operation within the airfield area (MoLIT, 2018).

Figure 1-1. Examples of FOD

FOD poses a major threat to navigation safety, such as inhalation into the engine or rupture of tires while moving the aircraft due to rapid airflow when taking off and landing an aircraft. A typical accident case is the crash of Air France 4590 flight at Paris de Gaulle Airport on July 25, 2020. The aircraft crashed in the vicinity of the airport 80 seconds after takeoff due to an explosion in a fuel tank. All 109 passengers on the plane were killed and four hotel staff members near the crash site were killed. The cause of the accident, which caused horrendous casualties, was a 43cm piece of metal that had fallen from another aircraft that had taken off just before. The metal ruptured the aircraft's landing tire and then hit the fuel tank at high speed, causing the explosion. (BEA, 2000). In addition, FOD causes enormous economic damage every year. In 2013, the FAA reported that direct and indirect economic damage from FOD was $4 billion annually.

Figure 1-2. Site of Air France 4590 crash

Figure 1-3. Cause FOD of accident(left), ruptured tire(right)

Due to these risks, each airport continues to conduct FOD detection and removal. However, while it is important to detect FOD that has already occurred, it is fundamentally necessary to reduce and prevent the occurrence of FOD through cause analysis. In order to analyze the cause of occurrence, detailed information such as FOD type, discovery location and discovery time is required, not simply information on the presence or absence of FOD. Each airport still performs FOD detection and removal and retains some of the information. However, currently, vehicles or airport personnel are mainly used to detect while walking directly on runways and taxiways. This existing method not only requires a lot of resources(manpower, equipment, time) but also always contains the risk of omission. And there is a limitation in that they have to make a separate schedule that limits the flight schedule. In terms of prevention, it is difficult to obtain detailed information about the occurrence of FOD. Recently, researches using computer vision technology, which have dramatically improved detection performance, are being conducted.

Figure 1-4. Current FOD detection and removal method

In order to detect and classify FOD using computer vision, the model training must be preceded. In this training process, training data is required. However, compared to the recent trend in which computer vision is widely used beyond the boundaries of the field, there are very few open image datasets that anyone can use. This limitation becomes more evident as the field of application and the required training data are specified. This was also true in the field of FOD detection, which requires a major FOD dataset.

If there is no open dataset, the training data(image) must be taken directly. However, there is a key challenge of FOD detection that should not be forgotten when acquiring training data by shooting directly. Unlike other object detection studies, the core of FOD detection is to detect small objects such as bolts and nuts in a large space such as a runway. The training data must also be secured at the appropriate scale. Images taken from too close a distance can increase the detection accuracy of the model, but will not be practical in the field. Also, images taken from too far away will be more practical, but

will be less accurate. Therefore, it is necessary to secure appropriate training data in consideration of this trade-off. However, most of the previous studies using computer vision used only images taken from a close distance as training data.

Figure 1-5. Training data sample(Hu et al., 2017), data collecting method(Cao et al., 2018)

In addition, directly shooting FOD training data at an appropriate scale is an inefficient method that consumes manpower and time. One of the biggest reasons for using computer vision is efficiency. Computer vision is used because it can perform tasks much more efficiently while showing similar or higher accuracy than that of humans. This efficiency must be ensured not only when using the model, but also in the process of training it. However, many studies easily overlook the effectiveness of the model training process. In particular, in fields where suitable data to be utilized, such as FOD detection, is remarkably

lacking, efficiency, a great advantage of computer vision, fades in the process of continuously securing training data.

To overcome these limitations in the training process, this study presents a FOD detection model trained with virtual images. A virtual image was used as a method to efficiently secure training data with a scale suitable for field application. Virtual images are widely used in computer vision to replace insufficient real images or to secure data diversity. In this study, various FOD virtual images were created using a 3D model and three experiments were performed with the trained model. The first is a performance comparison with a model trained with real images, the second is performance test at a close distance and the third is a performance test at a distance considering field applicability. For the test data to evaluate the performance, images taken directly from the FOD on the concrete pavement using a drone were used. If the detection model using virtual images and drones shows satisfactory performance, it will be possible to improve both the securing of appropriate training data and the efficiency of FOD detection. Furthermore, it will greatly contribute to FOD detection and prevention, and will be able to lay the foundation for the establishment of a FOD management system using computer vision.

1.2. Problem Statement

In this study, we try to solve the problems that occurred in previous studies that conducted FOD detection using computer vision. There are two problems.

The first is that training data with a scale that is difficult to apply to the actual field was used. A key challenge in FOD detection is finding small objects in large areas such as runways. In order to actually use the trained model, data with a scale that considers field applicability must be trained. In previous studies, most of the close-up images were used as training data. A model trained on data with an inappropriate scale is very impractical in the real world. As a result, the analysis of the necessary conditions when applying the model to the field is insufficient.

The second is the inefficiency that occurs in the process of securing appropriate data. In the field of FOD detection, due to the absence of a dataset, the data needed for training must be taken directly. However, it is very inefficient to obtain data while directly photographing various types of FOD at an appropriate scale. One of the most important advantages of computer vision is that it is far more efficient than humans can do it. However, securing a large number of training data through direct shooting greatly diminishes this advantage. In addition, depending on the training results and model performance, data acquisition often has to be done continuously, not just once.

1.3. Research Objective and Scope

The objective of this study is to suggest a FOD detection model using virtual images and drones and to analyze the conditions for using the model in the field. The problem of using training data with a scale unsuitable for field application and greatly reducing the efficiency of computer vision in the process of securing data can be solved by using virtual images as training data. Virtual images are recognized for their effectiveness as training data for computer vision models in various fields. If virtual images are used, it is possible to efficiently secure training data of various scales. We also chose drones rather than current personnel or vehicles as a means of detecting FOD. In the process of testing the performance of the model using the drone, the detection altitude suitable for the model and the minimum pixel required for detection by class were derived. These conditions presented in this study for practical use will be able to contribute to FOD detection research using computer vision in the future as well as using this model.

The scope of this study is three. First, we assumed the condition that FOD was visually detectable. Night or bad visibility conditions were not considered. Second, this study focused on the detection that should be made prior to the removal of FOD. Third, as FOD to be detected, four classes found frequently in the field were selected and experiments were conducted on them.

Chapter 2. Preliminary Study

2.1 FOD Detection Researches using Radar & EO/IR Camera

After FOD detection was recognized as an important task in airport operation, studies were conducted to detect it in various ways. Among them, studies using radar and EO(Electro-optic)/IR(Infra-red) cameras were actively conducted. Baoshuai & Wei (2017) and Wang et al. (2019) used a 77 GHz millimeter wave radar sensor and clutter map. Using the CFAR(Constant False Alarm Rate) algorithm that analyzes the relationship between the test cell and the surrounding cells, false alarms were classified and FOD was detected. The distance and angle of the FOD from the detection reference point were measured. An experiment was also performed to detect a moving object using a space-time joint estimation algorithm. Hong et al. (2018) performed an FOD automatic detection experiment using an EO/IR laser light camera and radar sensor. The detection method was specified by classifying it into a fixed type in which detection equipment is installed on facilities parallel to the runway, a mobile type in which detection equipment is installed in a vehicle, and a mixture of fixed and mobile types. Among them, they treated the fixed detection results and advantages and disadvantages as important. Song et al. (2018) used FMCW(Frequency Modulated Continuous Waveform) radar for FOD detection. Ground clutter was

applied to the FMCW to distinguish the grass area and the runway area, which were difficult to distinguish with the reflected signal. And the signal of the runway area without FOD was accumulated. Using this result, FOD located on the runway was detected.

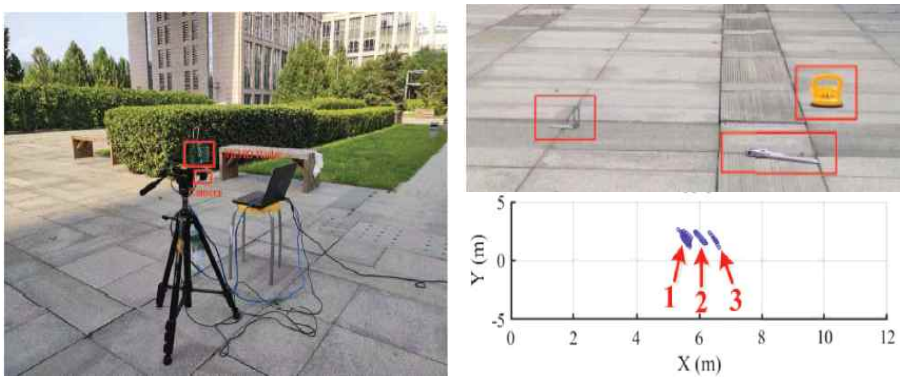


Figure 2-1. data measurement(left), signal processing result(right) (Wang et al., 2019)

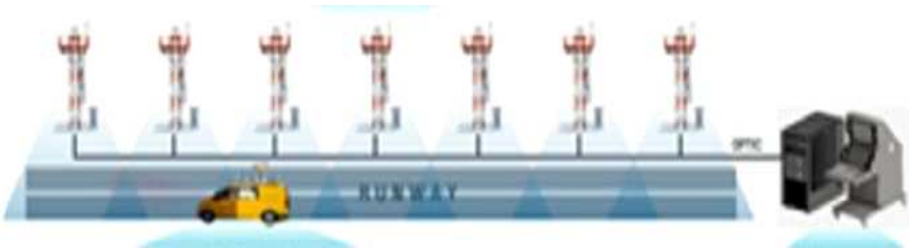


Figure 2-2. FOD automatic detection system (Hong et al., 2018)

However, the detection system using radar has several limitations. Since the equipment is expensive, a lot of money is required to build the system. In addition, there are problems such as frequent false alarms and false detections, limit of detection of living things and

frequent failures due to many management points. In particular, there is a limit to the detection of a small FOD with a small RCS (Radar Cross Section) even if the MMW (millimeter wave) radar technology is used. It is also difficult to distinguish whether an object suspected of FOD is a high-risk material or not (Lee et al., 2014). Due to these limitations, most airports except for some large airports did not apply the radar-based FOD detection system.

2.2 FOD Detection Researches using Computer Vision

As computer vision technology develops, studies using it are being conducted in the field of FOD detection. Xu et al. (2018) emphasized the importance of FOD material classification and appropriate training data. They presented a model for classifying metal, plastic and concrete materials. As learning data, images taken from an outdoor concrete road rather than an indoor environment were used. In order to increase the similarity with the field, the training data shooting time was divided into morning and afternoon. Cao et al. (2018) presented a model that distinguishes screws, stones, and backgrounds. The training data was obtained by installing a camera on the vehicle and photographing the runway surface. The detection process was divided into 2 stages. First, the location where the FOD is estimated was analyzed through RPN (Region Proposal Network) and then classification was performed using CNN (Convolutional Neural Network). Liu et al. (2018) obtained training data through cameras installed near the runway. They presented a model for detecting small steel balls, metal nuts, big screws and small screws of different sizes. Li & Li (2020) conducted an experiment to detect FOD consisting of 19 classes. Initially, there were 100 images for each class, but 40,000 samples were created through data augmentation.

What the previous studies have in common is that they used images taken at close range as training data. If an image taken at a close

range is used as training data, the model's performance can be derived excellently. However, this method overlooks an important issue in FOD detection. FOD detection is the search for small objects in large spaces such as runways. Therefore, it is difficult to apply the model trained with close range image to the actual field.

Another commonality of previous studies was that the data were obtained from photographs taken by hand. However, direct filming of FOD is an inefficient method. Even within the same class, each object has a different material, shape and size. Depending on the detection time, the information of an object may change. A lot of time is also consumed in the pre-processing of the training data. This inefficient data acquisition method greatly undermines the benefits of computer vision.

Figure 2-3. Training data from previous studies
(Xu et al., 2018; Liu et al., 2018)

2.3 Object Detection Researches using Virtual Images

In order to overcome the limitations of securing real data and to secure the diversity of data, many studies using virtual images are being conducted. Rajpura et al. (2017) noted that it is practically impossible to collect sufficient training data for applications that need to identify many subcategories. They generated a large amount of training data by synthesizing various 3D models and backgrounds. The scenario used in the experiment was to detect various objects in the refrigerator. The experiment was conducted by dividing the training dataset into three categories: synthetic only, real only, real(10%) + synthetic(90%). Real + synthetic showed the best performance. Sarkar et al. (2017) used 3D render images as training images to identify images drawn in catalogs. There are 5 classes to be identified. And training was done by changing the combination of background and texture. Real images taken indoors were used as test data. The classification process was performed by assigning only one object per image. Tian et al. (2017) compared the case of using only real data and the case of mixing real and virtual as autonomous driving training data. After creating a virtual city and road, virtual data was created in the same way as in real life, vehicles travel and capture data. In addition, three shooting directions were set. The performance of each direction was measured and compared.

Research using virtual images is being actively conducted in various fields. However, many studies are targeting relatively large objects or

small objects in an indoor environment. On the other hand, this study focused on finding small objects in outdoor environments such as runways.

Figure 2-4. Samples of training dataset (Rajpura et al., 2017)

Figure 2-5. Virtual camera(left) and diversity of data(left)
(Tian et al., 2017)

Chapter 3. Framework for Generating Training Data

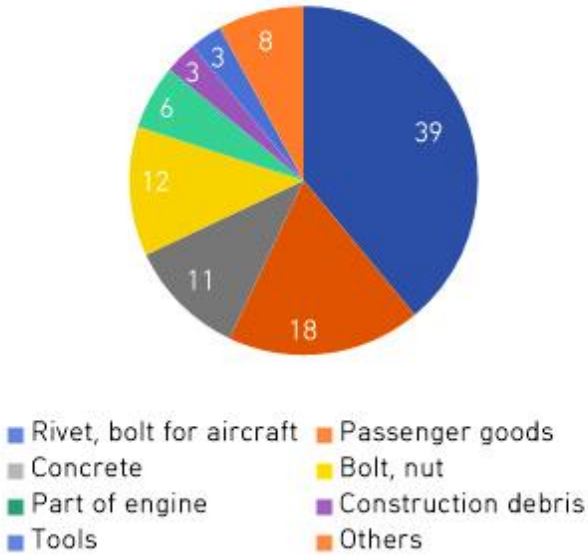
In Chapter 3, the framework for producing virtual images as training data was specified. Prior to establishing the framework, the target FOD to be detected in this study was selected. Four classes (blot, nut, fuel cap, tool) were determined as target FOD based on the FOD generation status announced by domestic and foreign organizations. Each class consists of 8 objects of different shapes and sizes. These 32 objects were obtained 3D models through open source sites. After adjusting the scale of the secured 3D model considering the size of the actual FOD, it was randomly placed on the virtual runway. In order to increase the diversity of data and the usefulness of the model, three materials(metal, rubber, plastic) were mapped to the virtual FOD. Several texture files with different color information were mapped within each material. The runway, which will be the background of the training data, also used 8 different concrete materials. In addition, considering that the site is outdoors, the solar conditions were also changed to create various combinations. Assuming that the virtual environment created in this way was taken with a drone, the videos were shot from an angle looking down from the vertical sky. Finally, the videos were divided into frames and labeled.

3.1 Target FOD Selection

As mentioned in the background, FOD is a concept that refers to all objects that may threaten navigation safety by being unplannedly located in the airport area including the runway. That is, the range of possible FODs is very vast. In this study, it was necessary to select the object to be detected first. The Federal Aviation Administration's FOD management guidelines stipulated that aircraft parts, repair tools, pavement peeling, wood, birds, and passenger goods were major FODs (FAA, 2020). The guidelines also stated that over 60% of the FOD was metals, followed by rubbers (about 18%). Similar FOD status could be confirmed in several data surveying domestic airports. Table 3-1 and Figure 3-1 show the main FOD status identified during the data survey.

Table 3-1. Major FOD types (top), essential detection objects in FOD automatic detection system (bottom)

Author	FOD type
FAA (2020)	Nut, bolt, washer
	Fuel cap, tire fragment, oil stick
	Maintenance, construction
	Concrete, asphalt, joint
	Plant, wildlife, ash
Author	FOD type
Incheon Univ. & Incheon International Airport (2014)	concrete, asphalt
	light sculpture
	Up to 20 cm metal pieces
	Aircraft or vehicle fuel caps
	big nut, hydraulic hose



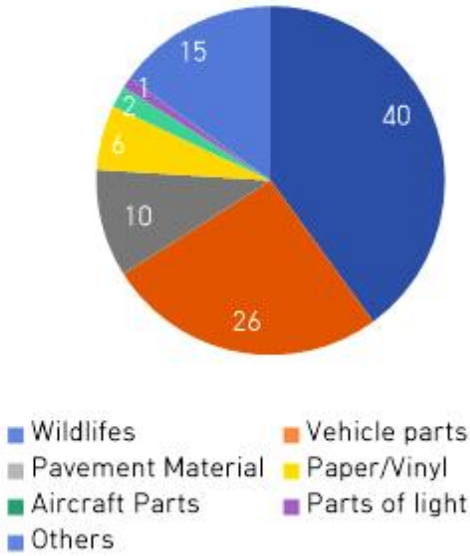


Figure 3-1. FOD Occurrence status
(Lee et al., 2014; Lim et al., 2017)

Based on the objects commonly mentioned in the occurrence status survey, the FOD class to be detected in this study was selected as bolt, nut, fuel cap, and tool (maintenance tool). Each class consists of 8 objects of different shapes and sizes to enhance the diversity of data and effectiveness in the field. Figure 3-2 summarizes the images of classes and objects used in this study. Each object utilizes an easily available open source 3D model. Even when training more classes or objects, the open source 3D model can be used sufficiently. For uncommon FOD, a method using a LiDAR or RGB-D camera is also possible.

Figure 3-2. Classes and objects used to create virtual images

3.2 Creating Virtual Environment

A virtual environment that will be the basis of the virtual image was created. Rhino, a commercial design software, and Grasshopper, a graphic algorithm editor, were used to create the virtual environment. First, a pavement surface with a length of 10m and a width of 2m was created to serve as a virtual runway. Then, the 32 3D models obtained earlier were scaled in consideration of the size of the runway and the actual FOD and placed on the starting point of the virtual runway. In order to secure the similarity to the actual field and the diversity of the learning data, the sizes of all 32 FOD models were set differently. Briefly, looking at the size of each class, the bolt is 15~72mm, the nut is 13~68mm, the fuel cap is 72~140mm, and the tool is 55~300mm. In order to increase the training data by taking advantage of the virtual image, a large number of FODs had to be placed on the virtual runway. To automate this, we used Grasshopper's algorithm. After inputting the length and width of the virtual runway, the type and quantity of objects to be placed were designated and the rotation angle and the number of placement cases were set at random. In the future, the amount of training data can be controlled by adjusting the number of objects to be placed according to the learning result of the FOD detection model. Figure 3-3 shows Grasshopper's algorithm for FOD placement, and Figure 3-4 shows the virtual runway where FOD is placed through the algorithm.

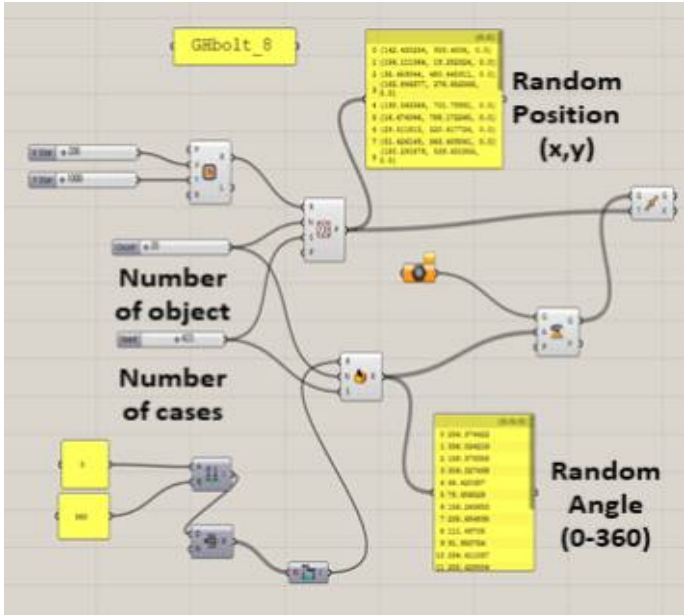


Figure 3-3. Algorithm for FOD Random Placement

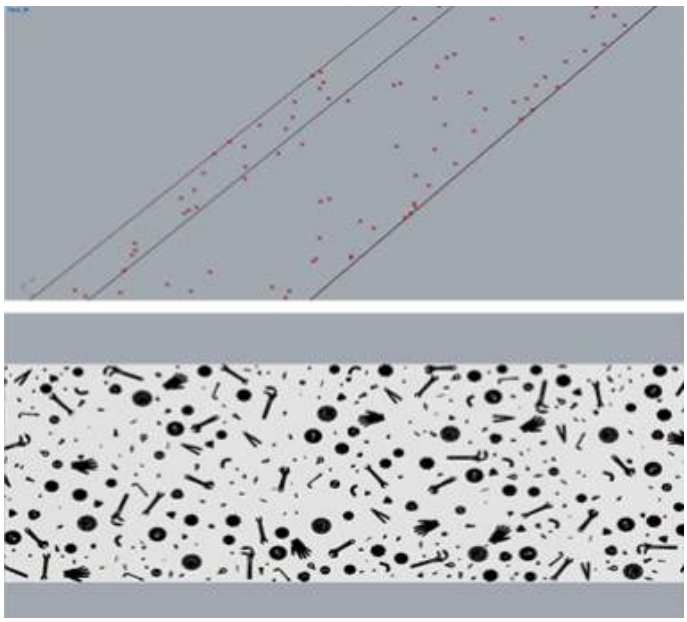


Figure 3-4. FOD placement on virtual runway

3.3 Generating Training Data and Preprocessing

The computer vision model extracts many features, including the shape, size and color of an object, from the color information of pixels in the input data. These extracted features enable classification and localization. In the FOD detection model, which has to find various objects on the runway placed in the external environment, the color information of the input data is a very important training area. For that reason, virtual images can generate more diverse and richer training data than real images.

Materials (metal, rubber, plastic) confirmed through preliminary investigation were mapped to the FOD of the virtual environment. Considering the actual material of the four classes, the ratio was about 81.2% for metal, about 9.4% for rubber and about 9.4% for plastic, giving the most weight to metal. Several types of material files are mapped to enable the creation of training data with various color information even for the same material. For the virtual runway that will affect the training results, eight concrete materials similar to the real one were mapped. Lastly, considering the external environment, data generation was carried out by changing the solar radiation that would change the color information. Data diversity was ensured through various combinations of FOD materials, backgrounds and solar conditions. After setting up the virtual environment, an image was created from an angle looking down from the vertical sky of the runway in the same way as a real drone shoots. Usually, 24, 30, and

60 are used for fps (frames per second) of a video, but in this study, 5 fps was applied to reduce the possibility of overfitting that may occur due to overlapping of the same objects. By dividing the generated videos into frames, a virtual image to be used as training data in this study is finally created. Figure 3-5 and Figure 3-6 are material samples used to create the virtual image and Figure 3-7 is a sample of the virtual image created through the above process.

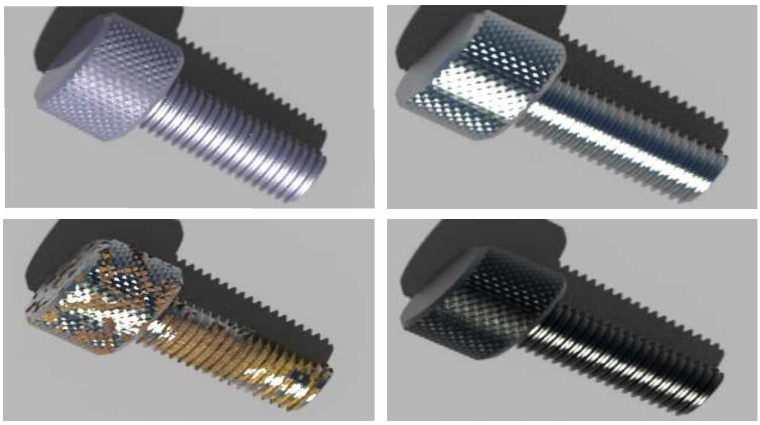


Figure 3-5. Metal material samples

Figure 3-6. Concrete material samples

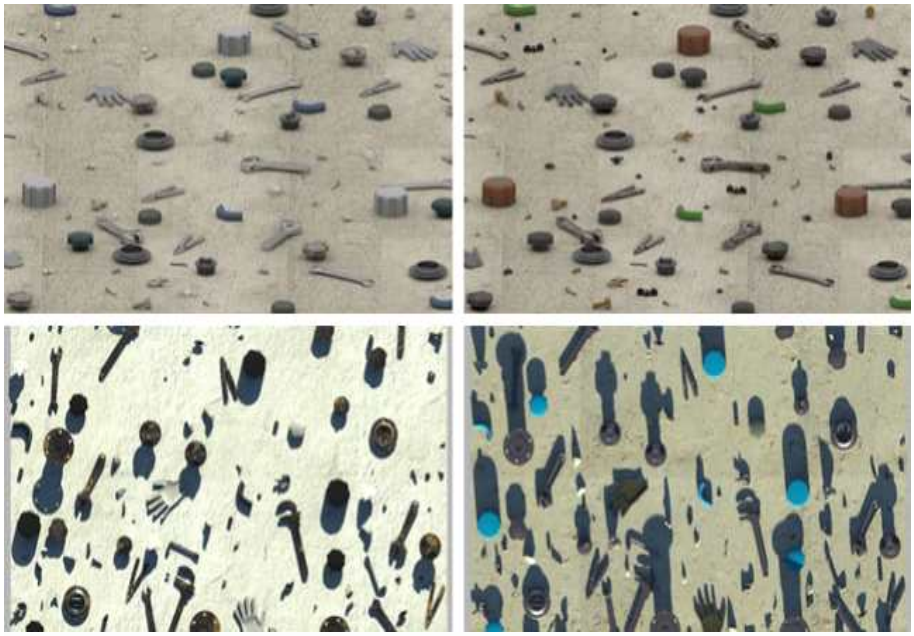


Figure 3-7. Virtual image samples

One of the tasks that consumes a lot of time and causes inefficiency in the training process of computer vision models is the labeling task. The performance of the model can be highly dependent on this labeling quality. For that reason, despite the rapid development of computer vision technology, labeling is still mostly done manually by humans. Although R&D on labeling automation is in progress, there are still many things that need to be supplemented to secure reliability and become common. In this respect, using virtual images as training data is a method that can greatly reduce the inefficiency of labeling. Each object is different data with different information, but since it has the same location and size in the virtual environment, it is possible to label a large number of objects in one operation. Figure 3-8 is a

labeling sample of frames taken at the same location.

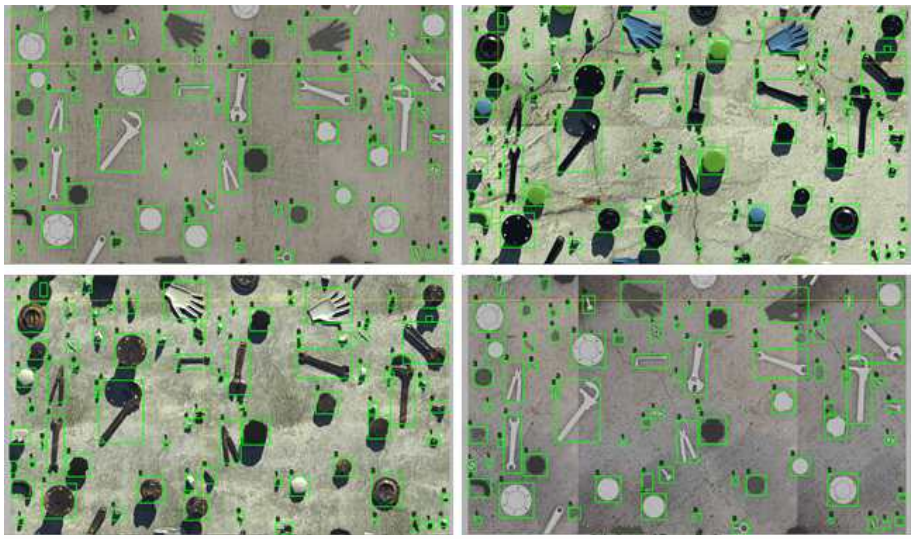


Figure 3-8. Labeling samples of virtual images

In order to train the computer vision model, one label file containing the coordinates and size information of the object is also required per frame. Using the framework presented in this study, this label file can also contain a lot of data in one operation. In the process of pre-processing the input data, a number was assigned to each class. The bolt was assigned as 0, the nut as 1, the fuel cap as 2, and the tool as 3. Figure 3-9 shows the preprocessed label file format. Information is displayed in the order of class number, normalize x center, normalize y center, normalize width and normalize height.

0	0.0863281	0.735417	0.0101562	0.0319444
0	0.524219	0.297222	0.0171875	0.0305556
0	0.694141	0.751389	0.0117188	0.0305556
0	0.822266	0.310417	0.0101562	0.0208333
0	0.289844	0.472222	0.0125	0.025
1	0.505469	0.883333	0.015625	0.025
1	0.257813	0.856944	0.015625	0.0305556
1	0.0785156	0.628472	0.0226563	0.0652778
1	0.226563	0.790972	0.0203125	0.0513889
1	0.889063	0.770833	0.028125	0.0333333
1	0.632422	0.681944	0.0335938	0.0472222
1	0.928906	0.656944	0.015625	0.0277778
1	0.849609	0.425	0.0132813	0.025

2	0.426953	0.25625	0.0539063	0.0986111
2	0.121875	0.23125	0.0765625	0.131944
2	0.326172	0.188889	0.0570313	0.1
2	0.654688	0.172222	0.04375	0.075
2	0.752734	0.1	0.0429688	0.0722222
2	0.166406	0.0979167	0.0515625	0.0875
3	0.802734	0.872917	0.0351563	0.0263889
3	0.546484	0.796528	0.0773437	0.176389
3	0.859375	0.807639	0.09375	0.184722
3	0.962109	0.7375	0.0492188	0.155556
3	0.329297	0.7875	0.0492188	0.227778
3	0.392969	0.741667	0.071875	0.263889

Figure 3-9. Label data format

Chapter 4. Experiments and Results

In Chapter 4, an experiment was performed to evaluate the performance of the FOD detection model trained with virtual images. Prior to the experiment, actual images to be used as test data were taken. The test image has a concrete pavement similar to the runway pavement as the background. Four classes were randomly distributed and the images were taken separately by detection altitude. Three experiments were conducted to test the model performance. The first is a performance comparison experiment between a model trained with a real image and a model trained with a virtual image. The second is a performance test experiment of a model trained with virtual images taken at close range, similar to previous studies. The third is a performance test experiment of a model trained with virtual images taken from a distance. As a result of the third experiment, it was confirmed that the performance of the model decreased as the detection altitude increased. Additional experiments were carried out to confirm the optimal detection height of this model and to derive the minimum pixel required for detection for each class.

4.1 Collect Test Data

To test the performance of the FOD detection model trained with virtual images, real images taken directly were used as test data. For the real image shooting, a concrete pavement site similar to the runway pavement was first hired. Objects of 4 classes (bolt, nut, fuel cap, tool) were randomly placed on the concrete pavement surface. Figure 4-1 is a sample of real objects used for shooting.

Figure 4-1. Real objects used for test data

The placed objects were filmed using a drone in the vertical sky above the pavement surface. To check the performance of the model by detection altitude, test data was also taken by altitude. From 1m to 6m,

the height was increased in 1m increments. DJI's MAVIC2 Enterprise was used as a drone for test data shooting.

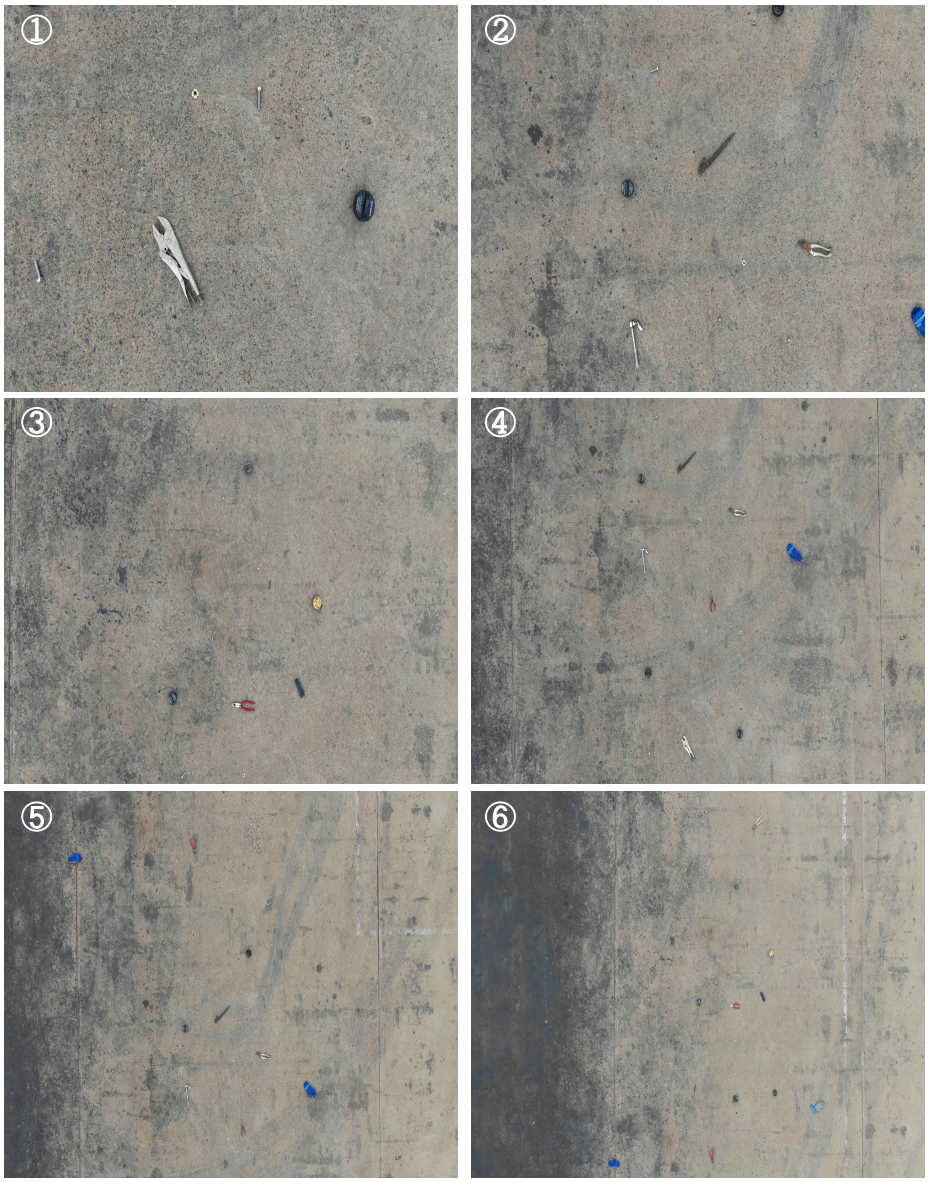


Figure 4-2. Test data samples
(1m to 6m altitude in numerical order)

4.2 Selecting Detection Algorithm

A typical deep learning algorithm used in computer vision is a CNN. CNN is useful algorithm for finding patterns to analyze images. CNN uses this pattern to classify images. The reason CNN is suitable for computer vision is that it trains while maintaining spatial information of images, unlike other algorithms. This algorithm derives global features by analyzing correlations with neighboring pixels rather than the entire pixel and performs classification and localization.

CNN model is divided into 1-stage detector and 2-stage detector according to the order of regional proposal and classification. The 1-stage detector performs both tasks simultaneously and the 2-stage detector performs sequentially (regional proposal → classification). The 1-stage detector has the advantage of operation speed and is evaluated to be more suitable for real-time detection. On the other hand, the 2-stage detector has a higher score in terms of accuracy. In this study, a 1-stage detector suitable for real-time detection was used in consideration of field usability. YOLO, a representative 1-stage detector model, was selected. Among them, the most recently released YOLOv5 was used. YOLO started in May 2016 and Joseph Redmon announced v1-v3 models, Alexey Bochkovskiy's v4 model and Glenn Jocher's v5 model were announced in 2020.

4.3 Comparison with model trained by real images

To verify the effectiveness of the virtual image as training data, the performance of the model trained with the real image and the model trained with the virtual image were compared. The real image used as the learning data was taken at a height of 1 m above the concrete pavement in a place different from the test data shooting site. Each model trained 120 real/virtual images and the number of objects included in the training data was set to an average of 11 to 12 per image. For the test data, a total of 42 images were used, including images taken by each class and images taken together in all classes.

Figure 4-3. Real(top)/virtual(bottom) images used as training data

The learning rate, IoU threshold, and confidence threshold values were fixed to 0.01, 0.6 and 0.001, which are the default values of yolov5 (these values were also fixed in subsequent experiments). 16 tests were conducted while changing the image size, epoch and mini batch size. In this experiment, the lightest model (YOLOv5s) among several YOLOv5 models was used and ubuntu and GeForce RTX 2080 (Ti) dual GPUs were used.

As a result of the experiment, it was confirmed that the performance of the model trained with the virtual image was not inferior to that of the model trained with the real image. In addition, the objects taken in the real image(training data) are the same objects as the objects used in the test image and the size of the image used for training was larger in real (1920*1080) than in virtual (1280*720). Considering these points, it can be judged that the possibility of the FOD detection model trained with the virtual image is sufficient.

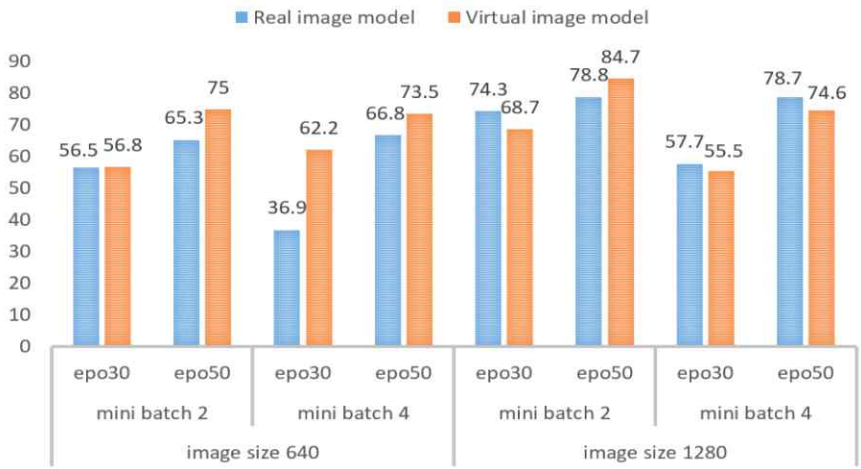


Figure 4-4. Comparison of real image model and virtual image model

4.4 Performance test at close range

The second experiment is to test the performance of the model by training images taken at close range, similar to previous studies. In previous studies, most of the data were collected by people taking pictures or installing cameras in vehicles. The shooting height was set about 1m to 1.5m above the ground. For comparison with previous studies, similar scales were applied to the training data and test data used in this experiment. A total of 4,800 training images were used by combining material, background and solar conditions. About 80,000 bolts and nuts, and about 40,000 fuel caps and tools are included. The size of the training image was 1280*720. Hyper parameters were set to image size 1280, epoch 20 and mini batch size 2. This was done using the YOLOv5m model and the same dual GPU as before. As a result of setting the mini batch size to 2, 2,400 iterations were performed in one epoch and a total of 48,000 iterations were performed. As the epoch exceeds 20, it can be confirmed that the performance of the model decreases due to overfittig. For the test image, the FOD on the concrete pavement was taken from a height of 1m using a drone. The size of the test image was 3840*2160.

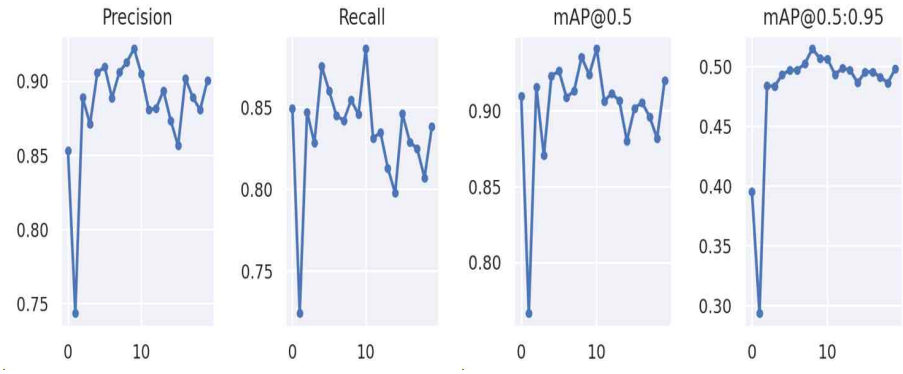


Figure 4-5. Training results for virtual images at close range

Figure 4-6. Close range test image samples

Figure 4-7. Close range test result image samples

Figure 4-5 is the training result of the second experiment and Figure 4-6 is a test image sample. Figure 4-7 and Figure 4-8 show the test result image and the performance of the model.

As a result of the test, AP for each class was 98.8% for bolt, 96.6% for nut, 92.5% for fuel cap and 98.6% for tool. And the mAP (mean average precision) of the model was confirmed to be 96.9%. Although it is difficult to make a simple comparison because the quantity, quality of the training data and class setting in previous studies are different, it was possible to estimate that the performance of the model presented in this study was similar or higher than that of previous studies using the mAP value. In addition, the studies with more than 90% performance specified in Table 4-1 were conducted by dividing the dataset taken at the same location into training data and test data. This model shows excellent performance despite the significant difference between the training data and the test data. Considering that point, it can be judged that the effectiveness of this model is higher.

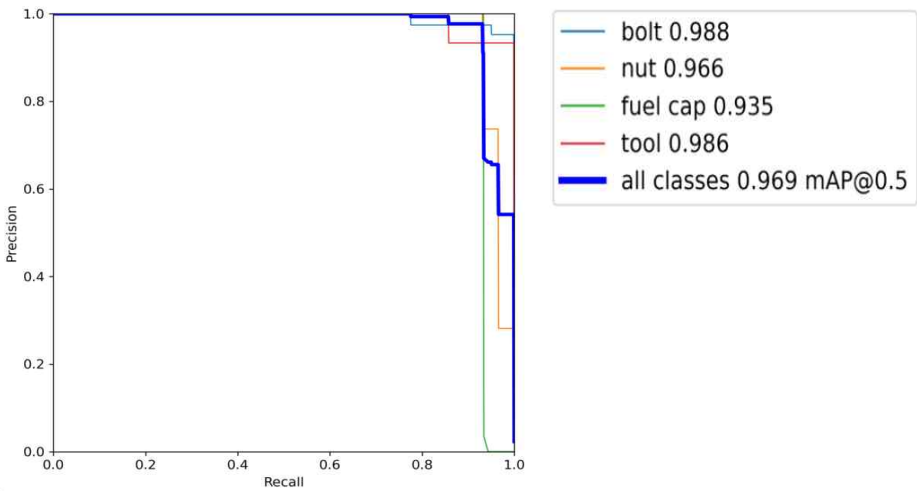


Figure 4-8. Performance of a model trained with close range

Table 4-1. Comparison of the performance of this study model with previous studies

Author	training data	Filming site	Number of classes	Performance
Xu et al. (2018)	3,000 images	different (training ≠ test)	3 classes (metal, plastic, concrete)	78% (accuracy)
Liu et al. (2018)	3,964 images	same (training = test)	4 classes (steel ball, nut, big/small screws)	95.66% (recall)
Cao et al. (2018)	9,785 images	same (training = test)	2 classes (screw, stone)	98.41% (mAP)
Li & Li (2020)	40,000 images (include test)	same (training = test)	19 classes (screw, plastic bottle screwdriver etc)	92.2% (mAP)
Ours (2021)	4,800 images	different (training ≠ test)	4 classes (bolt, nut, fuel cap, tool)	96.9% (mAP)

4.5 Performance test at a long distance

The third experiment is a long-distance performance test for actual field application. The detection height was divided into 2m, 4m and 6m. Figure 4-9 is a sample of training images for each height.

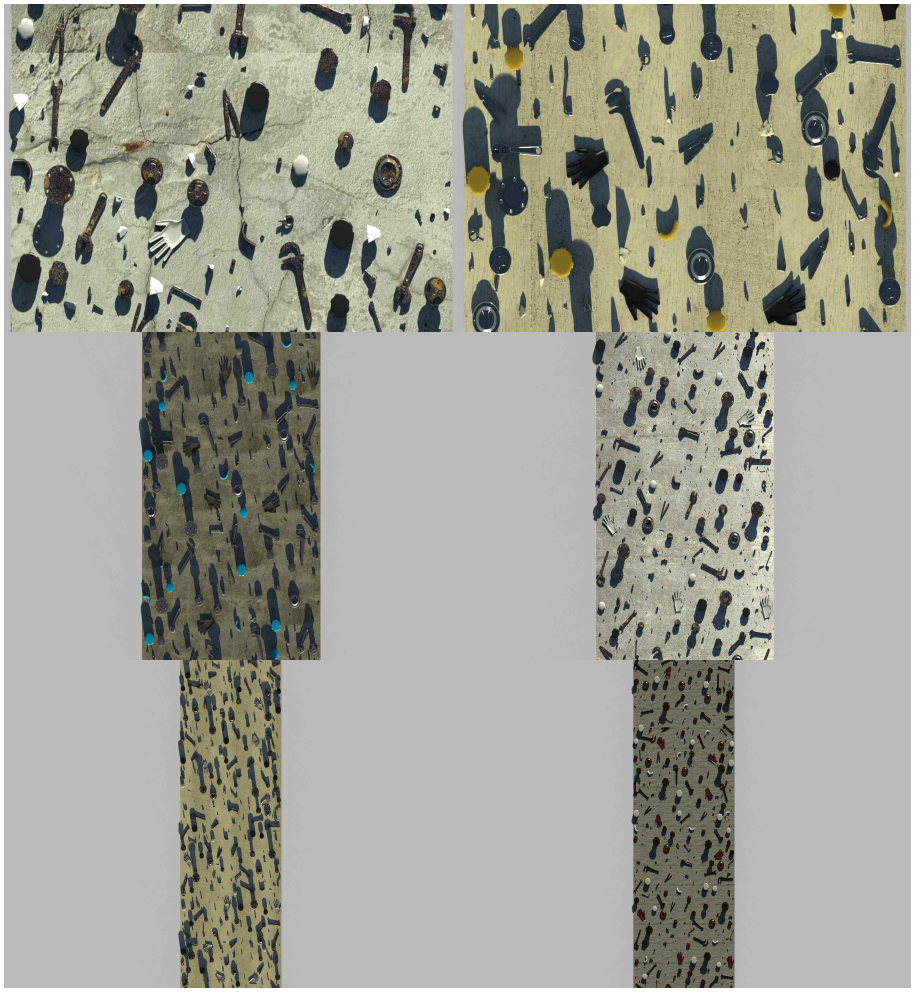


Figure 4-9. Training data taken from a distance (2m, 4m, 6m from the top)

It was found that the higher the shooting height, the more difficult it was to visually classify each class. In particular, it was not easy to accurately distinguish between bolts and nuts with the naked eye. Similarly, in the process of training the model, it was confirmed that the performance decreased as the height increased. In addition, in order to extract more features of each class, the size of the training image was set to 1280*720 at 2m and increased to 3840*2161 at 4m and 6m, but the performance could not be prevented from falling. In the 4m and 6m experiments, the number of training images was reduced by increasing the shooting speed to prevent overfitting of the same object.Although the size of the training image increased (1280*720 → 3840*2161), it is judged that setting the input image size to the same size had a negative effect on the model performance. Table 4-2 summarizes the test results by height conducted in the same way as before. The test image was 3840*2160 the same as before.

Table 4-2. Experimental conditions and test results by height

height /model	training images	image size	performance(%)				
			bolt AP	nut AP	fuel cap AP	tool AP	mAP
2m /YOLOv5m	4800	1280	99.5	97.5	95.5	93.0	96.4
4m /YOLOv5s	2448	1280	55.5	85.1	68.6	95.3	76.2
6m /YOLOv5s	2080	1280	00.0	00.0	83.9	60.0	36.0

Figure 4-10 is a sample of test image and result image by height.

Figure 4-10. Test image(left), result image(right) (2m, 4m, 6m from top)

It was found that even if the size of the original training image was large, the performance was lowered when the size of the input image was small. In other words, since large-sized images are compressed a lot, a lot of information that can analyze features is lost. In order to reduce image compression, the experiment was continued by changing two parameters.

First, we adjusted the original size of the training image and the test image. As shown in Figure 4-9 and Figure 4-10, as the height increased to 4m and 6m, spaces unnecessary for learning were being input together on the left and right sides of the image. To remove these unnecessary spaces, the horizontal size of the training and test images was cut from 3840 to 2160. By using the cropping method rather than compressing the image size, an experiment was conducted so as not to affect the object. Considering that the image input to the YOLOv5 model is square, the vertical size hardly changed (2161→2160). We cut out only the horizontal and proceeded to make square images.

The second is to adjust the size of the input image. Previously, the input was 1280*1280, but it was increased to 2560*2560 to reduce the compression rate of training and test images. When applied together with the adjusted training and test images (2160*2160), there were cases where the image was enlarged rather than compressed. The experiment conducted by increasing it to 2560 was conducted by Colab provided by Google and the GPU was also used by Colab's Tesla K80. Table 4-3 shows the results of experiments conducted by modifying two parameters.

Table 4-3. Results for experiments conducted by modifying two parameters.

height	train size	test size	input size	class	pixel	AP (%)	mAP (%)
4m	3840 * 2161	3840 * 2160	1280	B	98.71	55.5	76.2
				N	71.17	85.1	
				F	858.69	68.8	
				T	1395.94	95.3	
			2560	B	395.08	96.2	90.5
				N	284.33	84.2	
				F	3434.76	95.0	
				T	5584.85	86.5	
	2160 * 2160	2160 * 2160	1280	B	149.14	90.4	87.5
				N	134.81	99.0	
				F	1180.24	83.8	
				T	2535.75	76.6	
			2560	B	597.04	97.1	96.3
				N	539.22	99.5	
				F	4719.57	94.6	
				T	10140.98	93.8	
6m	3840 * 2161	3840 * 2160	1280	B	51.36	00.0	36.0
				N	47.03	00.0	
				F	356.72	83.9	
				T	691.47	60.0	
			2560	B	205.28	7.7	44.9
				N	188	1.7	
				F	1426.88	81.5	
				T	2765.06	88.8	

Through the experiment, the model performance (mAP) at 4m and 6m was confirmed as shown in Table 4-3. With this experiment, it was found that the performance of the model varies greatly depending on the original size of the training and test images and the size of the input image. When the original image of 2160*2160 is input as 2560*2560, the optimal detection height of this model is judged to be 4m.

In addition, it is judged that the performance of the model will vary depending on the performance of the camera mounted on the drone taking pictures. The value that changes depending on the size of the original image, the size of the input image and camera performance is the number of pixels representing one object. Table 4-3 shows the pixel representing the smallest object for each experiment. It was found that the more pixels representing the object, the better the detection performance of the model.

In this study, we tried to present a reference point for detection and a realistic model. For this purpose, two additional experiments were conducted. The first is an experiment to confirm the change in detection performance according to the change in the number of pixels representing the object. The target class is bolt and the target performance is set at 95% mAP. Table 4-4 and Figure 4-11 show the experimental results. When the number of pixels representing one bolt exceeds about 206, it was confirmed that the detection performance was more than 95%.

Table 4-4. mAP change according to the change in the number of pixels representing bolt

Pixel	51.36	98.71	149.14	180.46	193.86
mAP(%)	0	55.5	90.4	96.7	89.4
Pixel	205.91	232.66	271.81	292.31	335.57
mAP(%)	98.3	98.7	97.3	98.7	97.2
Pixel	369.96	380.16	395.08	403.34	
mAP(%)	98.9	99.5	96.2	99.5	

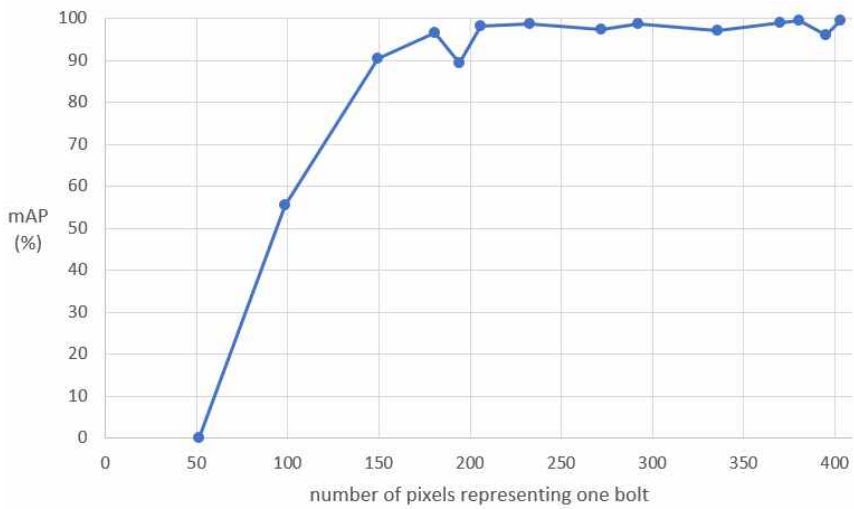


Figure 4-11. Chart of mAP change by number of pixels representing bolt

In the case of bolt, nut and fuel cap that are relatively similar in shape and size, the detection performance and minimum pixel derivation method of this model are expected to make a significant contribution. However, in the case of tools, there is a big difference in shape and size even between objects in the same class, so it is necessary to

classify the class in more detail.

The second is an experiment in which test data is set similar to reality and then detected. Although the test data used earlier is real data, many FOD exist simultaneously in one image. These data were not suitable for representing the actual situation where FOD occurred on the runway. Accordingly, similar to the actual situation, the number of objects included in one test image was reduced.

Figure 4-12. Original test image(left), modified test image(right)

Figure 4-12 is a sample 4m test image modified to include one object for each class in one image. Figure 4-13 and Figure 4-14 are the results of FOD detection with the modified test images. The mAP was 94% and the detection performance for fuel cap and tools was inferior compared to before the test image was modified. As can be seen from Figure 4-14, this model detected FOD well even in a test similar to the real situation.

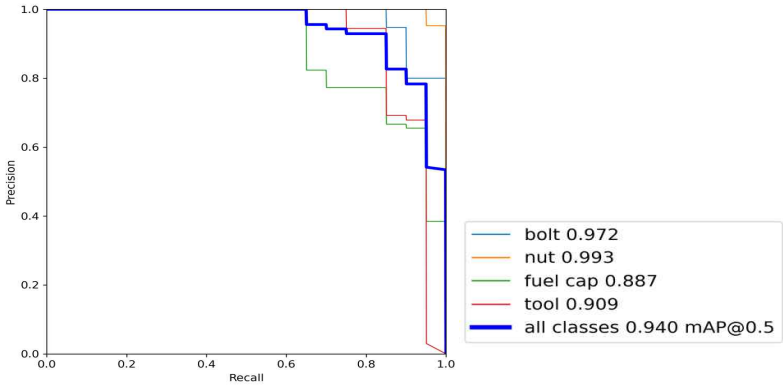


Figure 4-13. 4m test result with modified test images

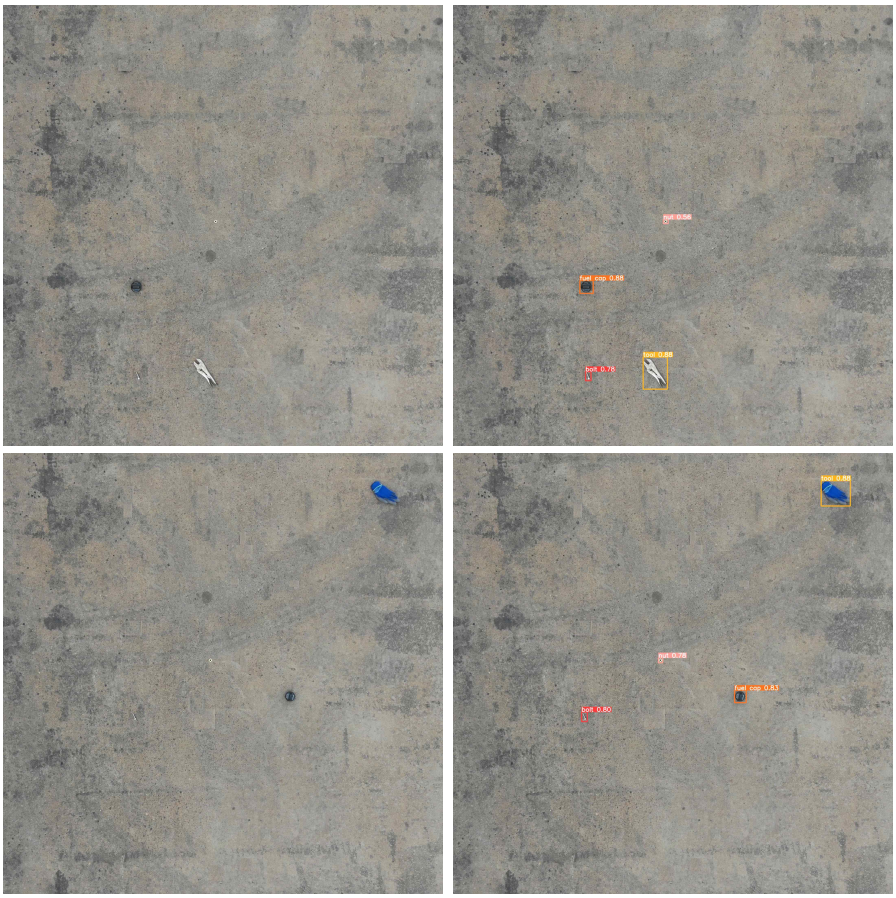


Figure 4-14. Modified test image(left), result image(right)

Chapter 5. Discussion

In this study, in order to confirm the effectiveness in the field, real images of the FOD on the concrete pavement were used as test data. As can be seen from the previous experimental results, this model showed satisfactory performance for the target FOD to be detected. However, in some experimental results, the pavement pattern was recognized as FOD. If the object to be found is correctly found, it is called a true positive, and if an object other than the target is found, it is called a false positive. Figure 5-1 is a sample of false positive.

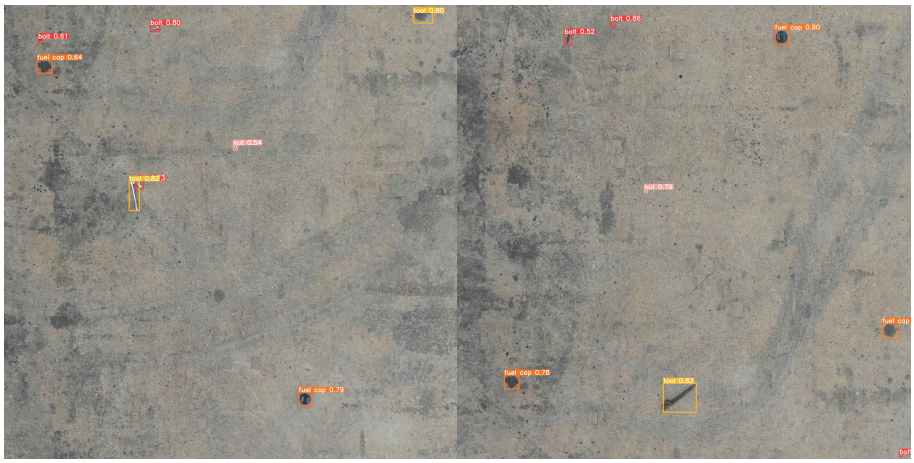


Figure 5-1. Sample of false positive

If FOD false positives continue to occur in the actual field, a lot of resources will be wasted in the process of checking and eliminating them. As a way to reduce false positives, it is necessary to consider training the model on a runway with no FOD at all. By learning a

clean runway, it is expected that it will be possible to reduce the number of cases in which the pattern of concrete pavement is recognized as FOD. This method will help to recognize the existence of FOD even if there is a case where the model cannot accurately identify the class of FOD detected.

Another area that needs to be discussed is the review of adding radar and EO/IR sensors to the drone camera used to detect FOD in this study. In this study, detection was performed using a general camera, so detection was limited at night and in bad visibility. It is judged that this limitation can be solved by using radar and EO/IR sensors. In addition, the limitations of existing studies using radar and optical/infrared sensors (many installation points, expensive system construction cost, low detection rate for FOD with small vertical component) are also expected to be partially resolved.

Chapter 6. Conclusion

Previous studies that studied FOD detection models using computer vision used training data with a scale that was not suitable for actual field application. In addition, in the process of securing training data, the efficiency, which is the advantage of computer vision, was greatly faded. In this study, the training data was created as a virtual image. This is a way to solve the scale problem in the close range image used in previous studies and it was able to significantly improve the efficiency that fell in the process of securing and preprocessing the training data. The amount of data can be adjusted according to the training results and the diversity of data can also be secured.

In this study, three experiments were conducted. The first was a comparison of the performance of a model trained with a real image and a model trained with a virtual image, the second was a performance comparison with previous studies trained with close range images and the third was an experiment by detection height to confirm the long-distance performance. In the first experiment, it was confirmed that the virtual image with diversity did not deteriorate as useful as training data compared to the real image. In the second experiment, the model trained with close range(1m) virtual images showed 96.9% performance. In previous studies that showed excellent performance, training data and test data were taken at the same location. However, the 96.9% performance of this model, which is significantly different

from the training data and the test data, showed meaning and contribution. In the third experiment, the detection height was increased to 2m, 4m, and 6m to prepare for actual field use. It showed excellent performance up to 2m, but the performance was significantly lowered in the 4m and 6m experiments. The experiment was continued by adjusting the size of the training and test images and the size of the input image. It was confirmed that the performance of the model was improved. The optimum detection height of this model for actual field use was confirmed to be 4m(96.3%). And we derived the minimum pixel value required to detect bolt among the classes of this study. Finally, to verify the effectiveness, the performance was verified using test data similar to the actual situation.

This study tried to suggest a FOD detection model that can be used in airport areas. Since it was trained with virtual data with an appropriate scale, it is judged that it can be used for actual FOD detection using drones. This can be the basis for building a future FOD management system using computer vision. In the future, more detailed research (performance comparison by algorithm, speed of detection drones, removal method after detection, etc.) for actual field application is needed. And research to improve performance at higher altitudes or at night and in severe weather is needed.

Bibliography

Airfield Facility Installation Standard. Notice of Ministry of Land, Infrastructure and Transport, No 2018-751 (2018).

Baoshuai, W. & Wei, Z.(2017). FOD Detection Based on Millimeter Wave Radar Using Higher Order Statistics. 2017 IEEE International Conference on Signal Processing, Communications and Computing, Xiamen, Chnia

Bureau d'Enquêtes et d'Analyses pour la Sécurité de l'Aviation Civile (2020). Accident on 25 July 2000 at La Patted'Oiein Gonesse(95) to the Concorde registered F-BTSC operated by Air France. Preliminary report translation.

Cao, X., Wang, P., Meng, C., Bai, X., Gong, G., Liu, M. & Qi, J.(2018). Region based CNN for foreign object debris detection on airfield pavement. Sensors, 18(3).

Federal Aviation Administration.(2020). Advisory Circulars 150/5210 – 24 - Airport Foreign Object Debris(FOD) Management. Retrieved from https://www.faa.gov/documentLibrary/media/Advisory_Circular/draft-150-5210-24a.pdf

Gao, Q., Hong, R., Chen, Y. & Lei, J.(2021). Research on Detection Algorithm of Foreign Object Debris and Small Targets in Airport Runway Based on SSD. The 2nd International Conference on Computing and Data Science, CA, USA

Hong, J., Kang, M., Kim, Y., Kim, M. & Hong, G.(2018). Experiment on Automatic Detection of Airport Debris (FOD) using EO/IR Cameras and Radar. Journal of Advanced Navigation Technology, 22(6), 522-529.

Hu, K., Cui, D., Zhang, Y., Cao, C., Xiao, F. & Huang, G.(2017). Classification of Foreign object debris using integrated visual features and extreme learning machine. Communications in Computer and Information Science, 773, 3-13.

Lee, K., Lee, J. & Kim, D.(2014). A Study for Efficient Foreign Object Debris Detection on Runways. Journal of the Korean Society for Aviation and Aeronautics, 22(1), 130-135.

Li, P. and Li, H.(2020). Research on FOD Detection for Airport Runway based on YOLOv3. Proceedings of the 39th Chinese Control Conference, Shenyang, China

Liu, Y., Li, Y., Liu, J., Peng, X., Zhou, Y. & Mruphey, Y.(2018). FOD Detection using DenseNetwith Focal Loss of Object Samples for Airport Runway. 2018 IEEE Symposium Series on Computational Intelligence,

547-554.

Papadopoulos, E. & Gonzales F.(2021). UAV and AI Application for Runway Foreign Object Debris(FOD) Detection. 2021 IEEE Aerospace Conference, MT, USA

Rajpura, P. S., Bojinov, H. & Hegde, R. S.(2017). Object detection using deep CNNs trained on synthetic images. arXiv:1706.06782v2

Rozantsev, A., Lepetit, V. & Fua, P.(2015). On rendering synthetic images for training an object detector. Computer vision and image Understanding, 137, 24-37.

Sarkar, K., Varanasi, K. & Stricker, D.(2017). Trained 3D models for CNN based object recognition. 12th International Conference on Computer Vision Theory and Applications, Porto, Portugal.

Song, S., Kim, B., Kim, S., Kim, M., Kim, Y., & Lee, J.(2018). Ground Clutter Modelling and Its Effect of Detection Performance in FOD FMCW Radar. Journal of the Korea Society for Simulation, 27(4), 61-68.

University of Incheon.(2014). Final report on the development of an automatic detection system for foreign object debris(FOD) on the runway. Ministry of Land, Infrastructure, and Transport, research

identification number : 1615006845.

Wang, J., Xueyin, G. & Wei, S.(2019). Airport Runway FOD Detection System Based on 77GHz Millimeter Wave Radar Sensor. 2019 IEEE International Conference on Integrated Circuits, Technologies and Applications, 140-143.

Xu, H., Han, Z., Feng, S., Zhou, H. & Fang, Y.(2018). Foreign object debris material recognition based on convolutional neural networks. EURASIP Journal on Image and Video Processing. https://doi.org.10.1186/s13640-018-0261-2.

국 문 초 록

FOD(Foreign Object Debris)는 공항지역 내 부적절한 장소에 위치하여 항공기 및 항공인력에게 피해를 줄 수 있는 물체를 통칭하는 개념이다. 이 FOD는 운항안전을 심각하게 위협하는 요소이며 매년 엄청난 규모의 직·간접 경제적 피해를 유발하고 있다. 전 세계 공항에서는 FOD 탐지 및 제거를 중요한 업무로 수행하고 있지만, 아직도 대부분의 공항에서는 인력이나 차량을 이용하는 전통적인 방법을 사용하고 있다. 이러한 방법은 많은 자원이 투입될 뿐만 아니라 탐지누락의 위험성을 내포하고 있으며 작업시간 동안 항공기 운항을 중단해야 하는 한계점이 있다. 또한 FOD 발생을 근본적으로 예방하기 위해 필요한 세부정보를 확보하는 것도 제한된다. 이러한 한계점을 해소하기 위해 최근에는 컴퓨터 비전을 활용한 FOD탐지 연구들이 진행되고 있다.

그러나 기존의 연구들은 현장 적용에 부적합한 스케일의 이미지들을 학습데이터로 사용했다. 활주로처럼 넓은 지역에서 작은 FOD를 찾아야 하는 현장여건에 맞지 않게 근거리에서 촬영된 이미지들을 사용했다. 이러한 데이터로 학습된 모델은 실제 현장에서 쓰이기에는 한계가 있다. 또한 적합한 스케일로 데이터를 직접 촬영하더라도 그것은 컴퓨터 비전의 큰 장점 중 하나인 효율성을 크게 퇴색시키는 작업이 될 것이다. 다른 컴퓨터 비전 모델들과 같이 확보한 데이터를 전처리하는 과정도 많은 시간이 소모된다.

이러한 한계점들을 해소하기 위해 본 연구에서는 가상이미지를

학습데이터로 활용한 FOD 탐지모델을 제시한다. 가상이미지를 활용함으로써 현장적용에 적합한 스케일의 학습데이터를 효율적으로 생성할 수 있었다. FOD 재질과 배경, 일사조건의 조합을 통해 데이터의 다양성도 확보할 수 있었다. 또한 데이터 전처리과정에서 소모되는 시간을 대폭 줄일 수 있었다. 테스트데이터는 드론으로 실제 촬영한 이미지를 활용했다. 실험은 크게 3개로 나누어 진행했다. 첫 번째는 실제이미지로 학습한 모델과 가상이미지로 학습한 모델의 성능 비교실험이다. 두 번째는 이전연구들과 유사하게 근거리에서 촬영된 이미지로 학습한 모델의 성능평가이다. 세 번째는 현장적용을 고려하여 원거리에서 촬영된 이미지로 학습한 모델의 성능평가이다. 실험의 결과로 가상이미지로 학습한 FOD 탐지모델의 효용성을 입증하였으며 본 모델을 현장에서 사용하기 위한 세부조건들을 분석했다.

본 연구의 결과는 컴퓨터 비전과 드론을 활용한 FOD 탐지체계 구축에 기여할 수 있다. 많은 자원이 소모되는 기존의 탐지방식과 현장적용이 제한되는 이전 연구들의 한계점을 개선시킬 수 있다. 더 나아가 FOD 발생을 예방하고 효율적으로 제거하기 위해 필요한 기초자료를 수집하는 방법으로도 활용할 수 있을 것이다.

주요어: FOD, 컴퓨터 비전, object detection, 가상이미지, 드론, 현장실효성

학 번: 2020-25569