Application of Artificial Intelligence to the Detection of Foreign Object Debris at Aerodromes' Movement Area

Source: From the author (2022).

Author: ASPAL/PILAV João Miguel Brito de Almeida
Master of Science Degree in Military and Aeronautical Sciences – Aviator Pilot
Force Academy, Sintra

Supervisor: CAP/ENGEL Gonçalo Charters Santos Cruz
Air Force Academy, Sintra

Co-Supervisor: MAJ/ENGEL Tiago Miguel Monteiro Oliveira
Air Force Academy, Sintra

ABSTRACT

The goal of the present dissertation is to develop a preliminary low-cost and passive system that detects Foreign Object Debris (FODs) at aerodromes based on computer vision with neural networks. FODs are a twofold problem, involving safety risks and high associated costs. Although some systems already exist to detect FODs, these are based on radars, making them expensive.

We build a dataset of images to test the viability of this solution, which was already attempted by other authors but the datasets are not publicly available. Moreover, we build a simplified architecture of the system to capture the images. In parallel, we develop a software pipeline which starts with image capturing scripts and ends in the evaluation of the models of neural networks we selected.

The datasets created result from three different electro-optical sensors: visible, near infrared and long-wave infrared. From the first, resulted a dataset of 9,260 images, from the second 5,672 and from the third 10,388.

Our approach to this problem is based on supervised learning with image classification and object detection and we train the models in subsets of the datasets. For image classification, we choose Xception as the neural network, achieving an 98.86% accuracy. In the case of object detection, we opt for a single-stage detector – YOLOv3 –, achieving an AP of 91.08%. Finally, we test the same models on new examples and verify a decrease in their performance to 77.92% accuracy for the classifier and 37.49% AP for the detector.

Keywords: Foreign Object Debris; Computer Vision; Dataset; Image Classification; Object Detection.

1 INTRODUCTION

In aviation, safety plays an important role, and prevention is the preferred method to assure it. Foreign Object Debris (FODs) are one of the biggest threats to aircrafts' ground operation (Kraus & Watson, 2001). In addition, the costs associated with FOD reach over $5 billion globally every year (McCreary, 2010). The most well-known FOD strike occurred with Concorde, which caused over 100 fatalities, and raised awareness for this problem.

Aerodromes perform regular visual inspections to the movement area every day to assure the safe circulation of aircraft. However, this traditional method of inspection is meagre because of their periodicity and capability of human detection. After the accident with Concorde, modern, radar and electro optical-based systems, started to be implemented at some major airports. These are capable of accurately detecting FODs in a wide range of weather conditions, yet, their implementation and cost are major downsides to aerodromes with medium and small number of movements.

In recent years, with the advent of machine learning, deep learning in the field of computer vision allowed for the implementation of solutions to problems that before would require visual inspection by humans. Two of the main advantages of these systems, compared to traditional ones are their lower cost and near-human accuracy.

2 LITERATURE REVIEW

2.1 FOD characterisation

The characterisation and definition of FOD are broad since anything that should not be at the movement area of an aerodrome is foreign to that place. In the case of the Portuguese Air Force (PoAF), FODs are divided into categories and types which classify FODs by their source and material. However, different organisations describe FODs differently from PoAF due to the very own broad nature of these objects.

The material of the most commonly found objects are made of metal (60%) and rubber (19%) and 50% are dark coloured. Tool pieces, ground equipment, pavement debris and metal from unknown sources are the objects with greater representation. In terms of size, FODs can be catalogued in two major groups: clusters of debris with individual size bellow 2 cm and FODs individually larger than 2 cm (90%). Although FODs are more prevalent on apron's areas, most of the strikes occur on runways and taxiways (McCreary, 2010), where the engine regimes and speed exponentiate safety risks.

The damage resulting from FODs is another concern involving this issue, since it is costly and may affect the readiness of aircraft. The Australian Transport Safety Bureau (ATSB) (2010), found that 11% of the FOD occurrences lead to wheel, engine and airframe damage. Moreover, McCreary (2010), concluded that FOD strikes occur 4.0 times per 10,000 movements, and 79% of those (3.2/10,000) inflicted damage to the aircraft. The most relevant damages inflicted by FODs are on tyres and gas turbines' blades. In terms of repairing and replacing, FODs inflict an average cost of $10,366 per strike.

Regarding military aviation, the risks associated with FODs are no different from civil aviation and the costs are higher. The aircraft which suffer more from FOD damage are the ones with turbofan or turbojet engines (NATO, 2004). Specifically, the F-16 and others are more susceptible to blade damage due to the position of the engine's air intake. In the PoAF, the FOD Prevention Plans define responsibilities, procedures and inspections, give a brief taxonomy of FOD and inform about the reporting methods. Inspections are carried out every day before, after and desirably during aerial activity. If requested, a sweeper truck must be available to go clean. The main objects found are parts of aircraft and ground vehicles, organic materials, safety wire, and debris from the pavement. Most FODs are found near parking spots, taxiways and runways. This means that they are not concentrated in a specific area of the aerodrome. In some cases, the operation of aircraft is limited at some airfields due to the conditions of the pavement, which is severely degraded to the point where aircraft cannot taxi.

2.2 FOD detection systems

Although many airports still rely on the traditional methods of detecting FODs, larger ones started to implement radar and electro-optical solutions. Human/visual observation is the primary system to detect FOD and provides the best capability of judgement and assessment to assure safety (FAA, 2009).

One of the best ways to detect FOD is through FOD walks which, historically, is standard military practice (McCreary, 2010). However, it requires closing the runway for extended periods, which is not always possible, especially at aerodromes with considerable traffic. Currently, a number of automatic FOD detection systems are available on the market and these can be divided in two: fixed and mobile. Fixed systems provide continuous surveillance and are installed either on the light fixtures of runways and taxiways or on towers near them. Mobile systems, are installed on the back of vehicles solely dedicated that purpose, and must detect FODs while moving, at least, at 30 km/h.

A study conducted for 18 months on two runways at Boston-Logan Airport compared how many FODs were detected on both runways (FAA, 2019). On one runway, they would use FODetect® and on the other, only visual inspection. The detection system found 139 objects on the first runway, whereas the visual inspections found 21. A similar study at Seattle-Tacoma Airport, with the same equipment, detected 102 objects, more than the 30 objects detected by human inspection. Another two implementation evaluations took place at Dyess Air Force Base and Yuma Marine Corps Air Station – two air bases in the United States. This time, the mobile solution (FOD Finder™) was responsible for the detection task at both places. No FOD-related

incidents were registered During a six-month test period, and 15,750 FODs were detected and removed (NDCEE, 2011). The estimated annual cost of this system is US$128,305. The numbers presented in both implementation studies demonstrate the capability of these radar-based detection systems.

The accuracy of these systems is higher than 95% as regulated by the Federal Aviation Administration (FAA) while significantly reducing or not requiring runway closure time. The two major downsides of these systems are their acquisition and maintenance costs as well as the permissions required, making them difficult to install at medium and small airports.

2.3 Computer vision and machine learning background

According to Huang (1996) and Lakshmanan (2021), computer vision aims to develop computational models that imitate the human visual sensory and cognitive systems in order to develop autonomous systems that can complete tasks of image formation and machine perception. Computer vision is a subset of machine learning which, in its turn, is a subset of artificial intelligence (Alloghani et al., 2020; Sandhu, 2018).

The start of computer vision is linked to Roberts (1963) who contributed to the fundamentals of image understanding, such as edge finding, line fitting and model-based object recognition. Image understanding is one of the tasks of computer vision and comprises two relevant problems intrinsically connected – image classification and object detection. The goal of the first is to label the image with one class and the goal of the second to localise and classify the objects in an image. The concatenation of more powerful machine learning algorithms and computers with Graphics Processing Units as well as larger databases of images, boosted the development of image understanding in the 1990s (Shapiro, 2020).

Until the 2010s, the pipeline of image understanding comprised three stages: pre-processing, feature extraction and classification. Thereafter, with computer vision competitions fomenting the development of new architectures, deep learning models started to take over, delivering better results than conventional methods.

2.3.1. Learning methods

There are four main machine learning methods: supervised, unsupervised, semi-supervised and reinforcement learning. Supervised and unsupervised learning are two of the most relevant and the primary difference between them is that the former needs labels to be provided with the training data whereas the latter does not (Alloghani et al., 2020).

In supervised learning, the goal is for the model to generalise well on a training dataset in order to correctly predict unseen data. The parameters that constitute the model, are continually updated and optimised during training where a loss function is responsible for calculating the difference between the target and the predicted outputs. Often, the scarcity of labelled data as well as the laborious task of the labelling process are major barriers in the development of supervised machine learning solutions (Lakshmanan et al., 2021). Unsupervised learning algorithms look for uniformities in input data (Karazi et al., 2019) without assigning a target attribute, learning hidden patterns and acting on data without supervision. Thereby, this learning technique helps analysing large raw data so that users can better characterise the problem they are working with. When used in a stand-alone application, unsupervised learning tends to be less accurate than supervised. However, this is a largely unexplored and complex field of machine learning as explained by Yann LeCun (2016) at Neural Information Processing Systems 2016 (NIPS).

The two methods aforementioned have their own drawbacks, which semi-supervised and reinforcement learning can partially resolve. The idea behind most semi-supervised algorithms is

to combine the main tasks of supervised and unsupervised machine learning frameworks (Chapelle et al., 2006). This translates into frameworks' whose layouts are a composition of an unsupervised followed by a supervised algorithm. The most common technique involves feeding both labelled and unlabelled data to the model and the results are encouraging in the sense that unlabelled data can increase image classification tasks based on neural networks (NN) (van Engelen & Hoos, 2020). Reinforcement learning systems' goal consists of identifying by trial-and-error which actions yield the maximum numerical reward signal in the long term (Sutton & Barto, 2018). Here, the agent is in a state which is limited to a finite set of states and can take actions, which are also limited to a finite number – where both are determined by the environment. According to its decision, the agent can be rewarded or penalised. That action will affect the following states and, consequently, the following rewards (Géron, 2019).

2.3.2. Conventional methods

Until the early 2010s, the paradigm of object recognition in images was mainly based on approaches 'hand-crafted' feature extractors such as Scale Invariant Feature Transform (SIFT) (Lowe, 1999) and Histograms of Oriented Gradients (HOGs) (Dalal & Triggs, 2005).The features extracted from SIFT and HOG are then applied to a trained classifier such as Support Vector Machine (Boser et al., 1992). These conventional methods were employed during several years and developed to achieve good performances. However, their results were surpassed when deep learning methods resurged in line with more powerful computers.

2.3.3. Deep learning methods

The idea behind NNs comes from the inspiration on neurons, but these are simplified mathematical and statistical models of how a visual cortex works. The image classification task assumes that an image can be represented by a set of features with different hierarchies: low, medium and high-level features (Sitaula et al., 2019). The main idea is that as we go deeper into the NN, the features become finer and carry more structural and semantic meaning.

The typical modern convolutional neural network comprises convolutional layers, pooling layers and fully connected layers (Alzubaidi et al., 2021). To control the output of the deep neural network, one must be able to measure its distance to the ground truth. The job of the loss function or cost function is to calculate that distance. The result works as a feedback signal for the neural network's training, whose objective is to lower the loss value via fine adjustments to the network's weights (Chollet, 2021).

The development of neural networks has been the driving force of many new applications in computer vision. Their robustness makes these networks work as good backbones or feature extractors for many different tasks such as object detection and image segmentation (Aloysius & Geetha, 2017; X. Li et al., 2016).

2.4. FOD detection with computer vision

The application of computer vision to FOD detection is not new to this dissertation. The first approach to the problem with this solution was made with conventional methods SIFT and HOG and had good results. However, these were trained with few examples, which potentiates overfitting. More recently, another set of authors employed object detection based on neural networks with more images and got better results. Despite this, their datasets are not publicly available.

3. SYSTEM ARCHITECTURE

The systems based on radar and electro-optical sensors mentioned above achieve good performances, yet they are expensive and require several permissions. On the other hand, a system based on electro-optical sensors embedded in vehicles which move around the aerodrome is a low-cost, effective and passive solution that satisfies most of our goals.

One objective of our work is to create an FOD dataset to test the viability of the implementation of the embedded system aforementioned. This involves having a mobile platform that simulates as close as possible the intended deployment of the system, hardware and connections, and creating a software pipeline. In Figure 1, we can observe both the training and deployment methodologies we want to employ in our system.

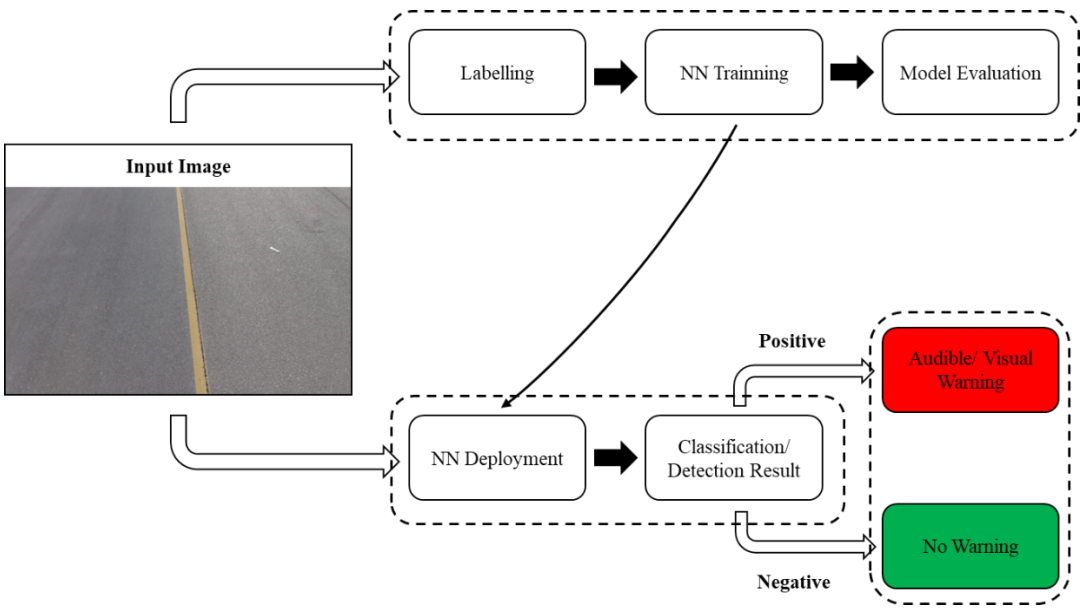

Figure 1 – System training (top) and deployment (bottom) methodologies. After labelling, training and evaluating a model, the latter should be uploaded to the system for its deployment. When the vehicle passes through an FOD in the field of view of the sensor, it should warn the driver of its presence recurring to a visual and/or audible warning.

3.1. FOD detection with computer vision

The image acquisition platform comprises three elements: vehicle, cameras or electro-optical sensors and setup. For the vehicle, we opt for a van whose height allows for a greater field-of-view of the cameras and for an easy installation of the system on its roof. As for the electro optical sensors, we choose cameras that work on different ranges of the electromagnetic spectrum.

The first camera has a sensor working on the visual spectrum (sensor01), the second on the visual and near infrared spectrums (sensor02) and the third on the long-wave infrared spectrum (sensor03). The first and the second sensors are connected to a NVIDIA® Jetson TX2 and a Raspberry Pi v3, respectively. The third sensor is a Gobi-384. The field-of-view (FoV) which the cameras provide when installed on the top of the van at 2.55 m of height and 38º of inclination in relation to the horizontal can be observed in Figure 2. In the case of sensor01, the height of the

trapezoid (blue) is 7.42 m and the width of the larger base 12.45 m. For sensor02, the height of the trapezoid (red) is 9.21 m and the width of the larger base 13.08 m. The sensor03 (green) has the smallest field-of-view with a height of 2.50 m and a width of the larger base of 2.59 m.

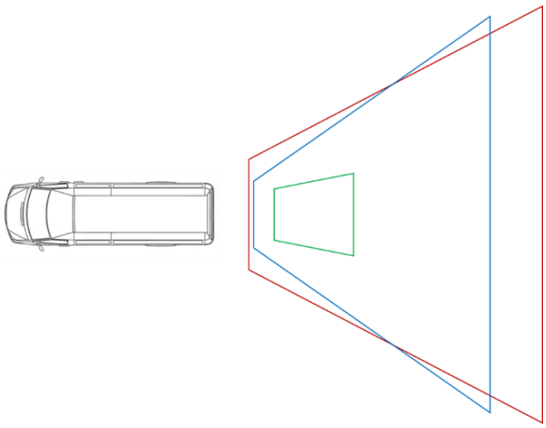

Figure 2 – Field-of-view of each sensor. 01 (in blue); 02 (in red) and 03 (in green). Image to scale.

3.2. Software pipeline

To create a dataset and test several machine learning algorithms, we needed tools that could help automatise and accelerate trivial processes. This necessity mainly came from the large quantity of data that composes a dataset.

Except for Gobi-384, the cameras' need a setup script to capture and save the images from the sequences. Raspberry's script consists of opening the camera, capturing and saving the image at a rate between, but not including, 0 and 2 fps. Similarly, the script for Jetson TX2 starts recording a video at 30 fps and then we choose the rate at which we save the frames, discarding the remaining. Then, we crop the images in tiles to feed the NNs, avoiding problems related with resizing, cropping and altering aspect ratio. We use an open-source image labelling tool – Label-Studio – to manually label all the original images we pre-selected, from each camera, containing FOD. In the case of the images for detection, the script works in a similar way, but it also creates annotation files. These are associated to each individual image and saves to an .xml file on PASCAL VOC format with the same name of the corresponding image. The fourth step of the pipeline is the neural network script. The last stage consists of a script to evaluate the neural networks and predict the classes of the test set.

In these scripts, we recurred to Keras, a high-level NN Application Programming Interface (API) written in Python. It is an open-source NN library which allows its users to readily specify network's configurations, tweak training parameters, evaluate and easily debug.

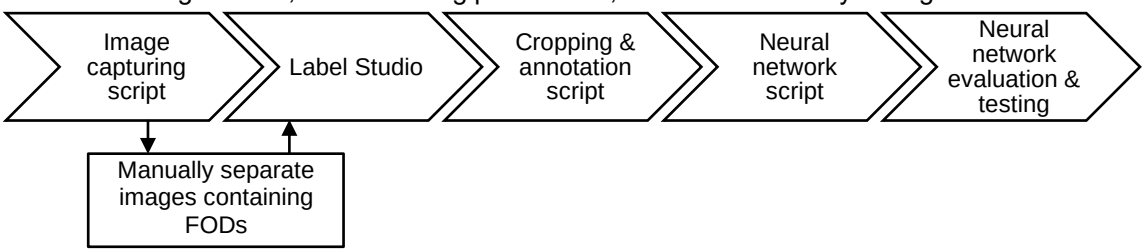

Figure 3 – Software pipeline representation.

3.3. Neural networks

Our proposed solution to detecting FODs is based on supervised machine learning with a classifier – Xception (Chollet, 2017) – and an object detector – YOLOv3 (Redmon & Farhadi, 2018). This decision has to do with several factors where speed in real-time classification and detection is necessary and computational board's processing capability is limited.

4. FOD DATASET

4.1. Existing datasets

Thanks to large datasets such as ImageNet (Deng *et al.*, 2009), more and more machine learning based approach to real-world problem solving is possible. Albeit some datasets provide millions of images and thousands of classes, some corresponding to FOD, the majority of the image is occupied by the target class(es). In our problem, the object of interest represents a small portion of the image due to the practicality of scanning large areas like aerodromes' surfaces without interrupting traffic.

Some authors had already addressed the FOD issue through machine learning when this project started, but most datasets are not publicly available. In the meantime, Munyer *et al*. (2022) released a large, publicly available, dataset compared to private ones, comprising 30,000 images and 31 object categories in three different lighting and two weather conditions. Although that dataset could be useful to our problem, the way the images were captured do not match the way how we expect to deploy our system.

We want to implement the cameras at the back of several vehicles, which limits the point of view of the camera in relation to the objects (FOD) in terms of angle and height. Combining the factors and the limitations of the aforementioned datasets, we decided to create our own dataset.

4.2. Objects selected for the dataset

In order to guide our selection of FOD, we relied on FAA's Advisory Circular 150/5220-24 (2009). Consequently, we chose objects that would resemble as much as possible the descriptions given on the Advisory Circular (AC), and other objects we found relevant based on other reports, papers and PoAF's prevention plans (ATSB, 2010; FAA, 2009; Herricks *et al*., 2015; McCreary, 2010; PoAF, 2018).

AC 150/5220-24 further describes the minimum performance for the FOD detection systems. Among others, these must be able to detect a metal cylinder, a sphere like a golf ball and 9 out of 10 of a group of objects placed within a 30 m by 30 m square in the coverage area. These 10 items must be smaller than 10 cm in any dimension unless otherwise specified. If the system is installed on a mobile platform, it must detect the FODs at a minimum speed of 30 km/h.

In total, we did three image acquisitions. For the first, there was no specific criteria to select the objects representing FODs except that they fit into the aeronautical context and were mentioned in most FOD related documents. Some of the objects were nuts, bolts, various metals, paint, and asphalt. The FODs selected for the second and third acquisitions followed more thorough selection process which was mainly based on the aforementioned AC. The FODs selected for the second and third acquisitions followed a more thorough selection process which was mainly based on the aforementioned AC. Most of the objects from the first acquisition were kept and we added a few more objects to partially correspond to AC 150/5220-24's list.

We did not divide the images per object class for two reasons. First, the number of images is small and that would force the framework to learn their features with fewer examples. Second, that approach would lead to the algorithm learning the features of each individual object failing at detecting different objects form the ones provided during training.

4.3. Data acquisition

During this project, we deployed our vehicle with the cameras three times to capture images to build a dataset. Along the three deployments we made several improvements to the sequence capturing. These improvements had to do with FOD placement and securing, image capturing scripts, camera positioning and overall workflow.

Figure 4 – Structure used to secure the cameras to the mobile platform with sensors 01 and 02 attached.

The first acquisition was the simplest, where the image capturing took place with the vehicle stopped at three locations in the aerodrome. For the following acquisitions, we built a structure that allowed for capturing images while moving. Due to some events on the second acquisition, we made a third one to have a better dataset. The final dataset only contains images from the third acquisition. All three acquisitions took place at the Portuguese Air Force Base no. 1 (LPST) in Sintra, Portugal, during different times of daylight with varying cloud coverage.

The third image acquisition took place at taxiway K of LPST during the morning and without any cloud coverage. During this acquisition, we made six passages along the taxiway – three in each direction to cover it in whole. We drove the vehicle at 15 km/h to reduce the distortion and blur of objects in the acquired images. In Figure 4, we can observe the number of occurrences of each object class by camera sensor.

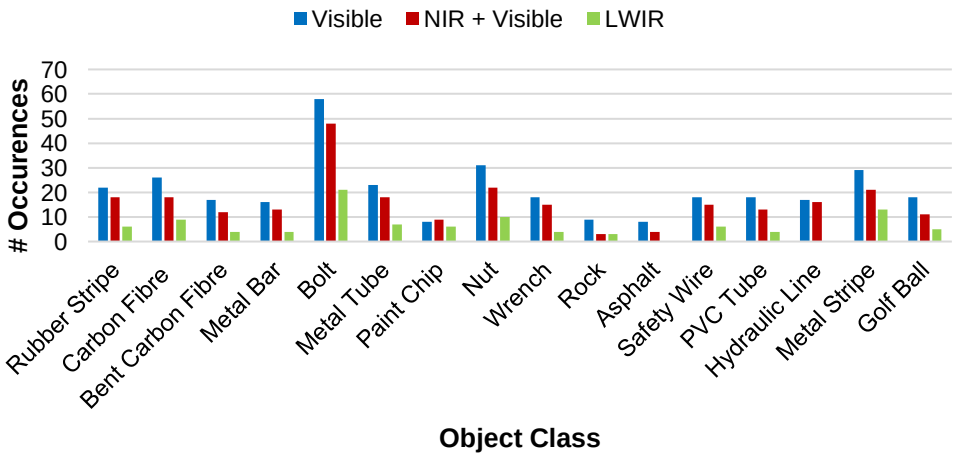

Figure 5 – Object occurrences by class and sensor for the third acquisition.

Table 1 provides a general perspective on the number of frames, labels and objects captured by each camera and the dimensions of the objects in pixels used in the dataset. From the table we can observe that although Gobi-384 captured the largest number of frames, it lacks two labels and two objects due to the problem aforementioned. On the other hand, the remaining two cameras captured less images but enclose all the labels and objects. The average size, size range and standard deviation of the width and height of sequences 01 and 02 are similar.

Table 1 – Capture sequence and objects characteristics of the third acquisition.

						Objects' width			*Objects' height*		
Name	*Spectrum*	*Resolution*	*Frames*	*Labels*	*Objects*	*Ave.*	*Range*	*Std. Dev.*	*Ave.*	*Range*	*Std. Dev.*
Sen01	Visible	1920× 1080	9,260	16	21	40	[4;258]	±38	27	[5;142]	±23
Sen02	NIR + Visible	1920 × 1080	5,672	16	21	37	[5;239]	±36	24	[5;137]	±22
Sen03	LWIR	384 × 288	10,388	14	19	22	[3;103]	±20	16	[3;49]	±10

There is no standardised definition of what a small object is in machine learning, leading some authors to take their own approaches. However, there seems to be a general consensus that objects smaller than 32×32 px are considered to be in the small object category. Moreover, on COCO detection evaluation metrics, small objects are considered to have less than 32^2 px^2. Another metric used to categorise the object's dimension is the median relative area. *(*Chen *et al*., 2017) merged images from Microsoft Common Objects in Context (MS COCO) (Lin *et al*., 2015) and Scene UNderstanding (SUN) (Xiao *et al*., 2010). We made our dataset publicly available at Harvard Metaverse website.

5. SYSTEM TRAINING AND TESTING

5.1. First steps

After the first data acquisition, the images were tested on Digits to understand the viability of this project. In our case, we used Alexnet and GoogLeNet, which are the pre-defined NNs ready for training and evaluating. The results were poor and preliminary. We believe that the small quantity of images in general, and the small number of images containing FODs in particular were two of the main factors contributing to poor and erratic prediction results. Nonetheless, the results obtained with the Digits models as well as the first dataset narrowed the path by helping to perceive what could work or not in terms of the network's hyperparameters and images.

5.2. Image cropping

The cameras' resolution is 1920×1080 px, which is large when compared to the input size of most neural networks. Feeding images this big to a NN comes at the cost of computing capacity that we do not have nor intend to implement on a low-cost embedded solution. On the other hand, the bigger the resolution of the original image the greater the accuracy performance of the network is because the detail also increases (Dong *et al.*, 2014).

We solve this problem with the tilling script aforementioned, which divides the images in tiles of adaptive size. It artificially reduces image size and required computational power while

making sure we keep the objects and their original features. One worrying aspect of this procedure is the loss of context. However, the background of our images is almost constant – grey asphalt and patches of white and yellow paint – meaning that there is practically no loss of context of the object.

We did not crop the LWIR images because their original size (384×288) is close to the input size of the neural networks we used. Furthermore, we did not use these images for training and testing. This decision was made based on the small number of images with FODs, the small FoV of the camera and our difficulty in finding the objects and labelling them.

Since one of the goals of the present dissertation is to determine which frameworks better suit our problem, we opted for a fixed tile size for image classification and object detection. The median area of the objects in relation to the original image area is between 0.0189% and 0.180%. By cropping the image into tiles of 256×256 px for classification, that ratio will now range between 0.180% and 0.763%. We followed the same principle for object detection, but with tiles of 416×416 px. The range of the median relative area changes from 0.0189% to 0.180% to 0.180% to 0.289%.

5.3. Implementation and results

5.3.1. Classifier

We trained the classifier on two types of subsets: balanced and imbalanced, and tested the models only on balanced test sets. The choice of training on imbalanced subsets was to have more images since, presumably, more images provide better results. The images were either labelled as 'fod' or 'no fod' and in tiles of 256×256 px. The train/validation/test split is in a proportion of 80/10/10, respectively. We applied transfer to the network with its weights pretrained on Imagenet (Deng et al., 2009).

We conduct for an unlimited number of epochs until convergence, where we would monitor the validation loss with a patience of 50 epochs for balanced subsets and 25 for imbalanced ones. To evaluate the performance of a model during training, we analyse their train and validation loss curves along the epochs. We tested different configurations of the hyperparameters and data augmentation. The dataset which contains all the images from the two balanced subsets delivers the best results in terms of accuracy (98.86%) and loss at 90.9 fps.

Analysing the plots, we can see how the train loss curve is approximately monotonically decreasing. On the other hand, the validation loss curve oscillates around a fixed value from start to finish, which might indicate an under representative validation set or overfitting. In our perspective, the problem has to do with overfitting, the limited number of images with FODs and few data augmentation. This allows the model to perform very well in the training examples ¬– high train accuracy –, losing generalisation capability.

5.3.2. Detector

The tile size employed for object detection purposes was 416×416 px. This image size allows YOLOv3 to infer in real-time while scoring good AP results. Just like we did for Xception, we opted for transfer learning with pre-trained weights on MS COCO (Lin et al., 2015). The train/validation/test split is in a proportion of 80/10/10, respectively. We tested the model in four different subsets based on each sensor and combinations of both like we did for Xception.

The subset with more images got the best results, scoring a 91.08% AP at 11.5 fps. Given the limitation established by the FAA of 30 km/h for mobile platforms in conjunction with capturing the FODs at least twice (>2.25 fps), we can say that the system can work in real-time.

Furthermore, we tested different IoUs to study its impact on the AP. As expected, the IoU grows, the AP decreases.

5.4. Evaluation with new objects

Given the results obtained in the previous sections, we wanted to test the robustness of the models by analysing their reaction to previously unseen objects. We did this because we believe that the images from the test set do not differ much from the images on the training set, leading to the results observed in the previous Sections.

Consequently, we collected new samples on a road-like surface to imitate the background of the previous acquisitions as close as possible. This time, we only captured images with sensors 01 and 02. The median relative areas of the objects in the resulting datasets are 0.0284% and 0.0615%. We employed these datasets only for testing the two best models obtained previously.

5.4.1. Results on the new dataset with image classification

The subset used for image classification is limited by the number of images with instances of FODs. Like we did before, we divided the images in tiles of 256×256 px with a horizontal and vertical overlap ratio of 0.5. From here, resulted 1,134 images for testing which contain FODs and 1,132 that do not, where 721 and 720, respectively, correspond to sensor01.

Testing the images from the fourth acquisition on the trained model, we can see a drop in the performance of the classifier. This result was expected since most of the objects presented to the NN are novelties. However the obtained result is uplifting because we consider that a drop to 77.92% of accuracy is a good result taking into consideration the size of the training dataset and the shape training curves obtained for the model.

Objects that are similar to the ones used in the third acquisition such as the bolt, the plastic tube and the metals are correctly classified. It is clear that the model tends to classify novelties as FODs. Plants and cracks, as depicted tend to be detected over other, more dangerous, FODs such as a large tree branch. This particular result is not as bad as it would seem at first sight because plants may constitute an organic FOD, especially in large quantities.

5.4.2. Results on the new dataset with object detection

As in the previous Section, the subset used for object detection is limited by the number of images with instances of FODs. We divided the images in tiles of 416×416 px with a horizontal and vertical overlap ratio of 0.5. From here, resulted a subset of 423 images, where 248 contain FODs and 175 do not, and 237 belong to sensor01 and 186 to sensor02.

Similarly to the behaviour of the classifier, the detector had a performance drop. However, the performance decreased significantly more than that of the classifier, from 91.08% to 37.49%. In terms of inference time, it remained unchanged at 11.5 fps, the same value as in the third acquisition.

It is clear that the model tends to classify novelties as FODs, especially plants. This particular result is not as it would seem at first sight because plants constitute an organic FOD, especially in larger quantities.

6. CONCLUSION

FODs pose a double threat to aviation, where the first and most important has to do with safety and the second with the associated costs. Although airports try to mitigate this problem through several visual inspections every day, it is clear that this traditional method is not sufficient to satisfy the problem we are facing.

In recent years, radar and optical systems have been employed at several airports for early detection of FODs. These systems are either installed on the side of runways and taxiways – fixed – or coupled to a vehicle that moves around the airfield – mobile. Even though these systems are effective, they are also costly both in acquisition and maintenance-wise.

On the other hand, the development of computer vision along with machine learning brought several solutions to different fields of research where human inspection is necessary. Among others, these systems have the advantage of requiring low-cost solutions when compared to the traditional systems as is our case.

Our approach to this problem is twofold. Firstly, we build a dataset of images with three different sensors which operate in different wave lengths. This dataset resembles as much as possible the way a fully deployable system would work: with cameras mounted on the top of vehicles which regularly drive around the aerodrome. Secondly, we test classification and detection based supervised learning techniques to evaluate the pros and cons of each. The main goal of the present work is to build a preliminary system that resembles a full deployment. This system must be low-cost and non-intrusive to the normal operation of an aerodrome.

For the first part, we developed a system architecture with a mobile platform providing the base for the installation of electro-optical sensors. We opted for this system architecture because it does not require acquiring new platforms, it is considerably less costly than radar solutions and does not require as many permissions as radar-based technology does.

Our dataset resulted from the third acquisition. It contains 9,260 images from the visible sensor, 5,672 images from the visible plus near infrared sensor and 10,388 images from the long-wave infrared sensor. From the first sensor, 336 images contain FODs, from the second 256, and from the third 102. The median average area of the objects is between 0.0189% and 0.180%.

Finally, we evaluated the performance of the neural networks on the test subsets of images from the corresponding training subsets. The model with the best performance in classification achieved an accuracy of 98.86% at 90.9 fps, trained on a imbalanced dataset with images from both sensors. However, the training curve of the models, especially the validation loss in conjunction with the training accuracy, show signals of overfitting. Secondly, we trained the detection network. Again, the subset with all the images from both sensors provided the best results with an AP of 91.08% at 11.5 fps. Despite the good results, we suspected that the detector was overfitting because of the few number of images and the excellent results in the testing set.

This led to a fourth image acquisition with sensors 01 and 02 that intended to test the NNs on entirely new objects. As expected, the accuracy and average precision of the classffier and the detector dropped while retaining the fps. The classifier went from an accuracy of 98.86% to 77.92%, while the detector dropped from an AP of 91.08% to 37.49%.

In conclusion, the preliminary results obtained in the present dissertation constitute a good base to develop the system in future iterations.

BIBLIOGRAPHY

Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., & Aljaaf, A. J. (2020). A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. In M. W. Berry, A. Mohamed, & B. W. Yap (Eds.), *Supervised and Unsupervised Learning for Data Science* (pp. 3–21). Springer International Publishing. https://doi.org/10.1007/978-3-030-22475-2_1

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*(1), 53. https://doi.org/10.1186/s40537-021-00444-8

ATSB. (2010). *Ground operations occurrences at Australian airports 1998 to 2008* (Safety Report No. 42; ATSB Transport Safety Report, pp. 1–20). ATSB.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory - COLT '92*, 144–152. https://doi.org/10.1145/130385.130401

Chapelle, O., Schölkopf, B., & Zien, A. (Eds.). (2006). *Semi-supervised learning*. MIT Press.

Chen, C., Liu, M.-Y., Tuzel, O., & Xiao, J. (2017). R-CNN for Small Object Detection. In S.-H. Lai, V. Lepetit, K. Nishino, & Y. Sato (Eds.), *Computer Vision – ACCV 2016* (Vol. 10115, pp. 214–230). Springer International Publishing. https://doi.org/10.1007/978-3-319-54193-8_14

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1800–1807. https://doi.org/10.1109/CVPR.2017.195

Chollet, F. (2021). *Deep learning with Python* (Second Edition). Manning Publications.

Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, *1*, 886–893. https://doi.org/10.1109/CVPR.2005.177

Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, & Li Fei-Fei. (2009). ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. https://doi.org/10.1109/CVPR.2009.5206848

Dong, C., Loy, C. C., He, K., & Tang, X. (2014). Learning a Deep Convolutional Network for Image Super-Resolution. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision – ECCV 2014* (pp. 184–199). Springer International Publishing.

FAA. (2009). *AC 150/5220-24—Foreign Object Debris Detection Equipment – Document Information* (Advisory Circular No. 150). FAA. https://www.faa.gov/airports/resources/advisory_circulars/index.cfm/go/document.current/documentNumber/150_5220-24/

FAA. (2019). FAA Reauthorization Bill 2018 Foreign Object Debris (FOD) Detection Technology [[PowerPoint]]. https://www.faa.gov/sites/faa.gov/files/2021-11/Technology_Use_Airports_PL115-254_Section_142b.pdf

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (Second edition). O'Reilly Media, Inc.

Herricks, E., Mayer, D., & Majumdar, S. (2015). *Foreign Object Debris Characterization at a Large International Airport* (Technical Note DOT/FAA/TC-TN14/48; p. 89). FAA; https://www.airporttech.tc.faa.gov/Products/Airport-Safety-Papers-Publications/Airport-Safety-Detail/ArtMID/3682/ArticleID/36/Foreign-Object-Debris-Characterization-at-a-Large-International-Airport. https://www.airporttech.tc.faa.gov/Products/Airport-Safety-Papers-Publications/Airport-Safety-Detail/ArtMID/3682/ArticleID/36/Foreign-Object-Debris-Characterization-at-a-Large-International-Airport

Karazi, S. M., Moradi, M., & Benyounis, K. Y. (2019). Statistical and Numerical Approaches for Modeling and Optimizing Laser Micromachining Process-Review. In *Reference Module in Materials Science and Materials Engineering* (p. B9780128035818117000). Elsevier. https://doi.org/10.1016/B978-0-12-803581-8.11650-9

Kraus, D. C., & Watson, J. (2001). *Guidelines for the Prevention an Elimination of Foreign Object Damage/Debris (FOD) in the Aviation Maintenance Environment through Improved Human Performance*. FAA.

Lakshmanan, V., Görner, M., & Gillard, R. (2021). *Practical machine learning for computer vision: End-to-end machine learning for images* (First Edition). O'Reilly.

LeCun, Y. (2016, December 5). *Predictive Learning* [Keynote]. 2016 Conference on Neural Information Processing Systems, Barcelona, Spain. https://www.youtube.com/watch?v=Ount2Y4qxQo&t=1072s

Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., & Dollár, P. (2015). Microsoft COCO: Common Objects in Context. *ArXiv:1405.0312 [Cs]*. http://arxiv.org/abs/1405.0312

Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, *2*, 1150–1157. https://doi.org/10.1109/ICCV.1999.790410

McCreary, I. (2010). *FOD, Birds, and the Case for Automated Scanning* (1st ed.). Insight SRI Ltd.

Munyer, T., Huang, P.-C., Huang, C., & Zhong, X. (2022). FOD-A: A Dataset for Foreign Object Debris in Airports. *ArXiv:2110.03072 [Cs]*. http://arxiv.org/abs/2110.03072

NATO. (2004). *Best Practices for the Mitigation and Control of Foreign Object Damage-Induced High Cycle Fatigue in Gas Turbine Engine Compression System Airfoils*. NATO.

NDCEE. (2011). 2011 Annual Report (National Defense Center for Energy and Environment) (p. 13) [Anual Report]. National Defense Center for Energy and Environment. https://apps.dtic.mil/sti/pdfs/ADA574500.pdf

PoAF. (2018). *MBA5 330-3 (A)—Programa de Prevenção de Danos por Objetos Estranhos*.

Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *ArXiv:1804.02767 [Cs]*. http://arxiv.org/abs/1804.02767

Roberts, L. (1963). *Machine Perception of Three-Dimensional Solids* [PhD, Massachusetts Institute of Technology]. https://dspace.mit.edu/handle/1721.1/11589

Shapiro, L. G. (2020). Computer vision: The last 50 years. *International Journal of Parallel, Emergent and Distributed Systems*, *35*(2), 112–117. https://doi.org/10.1080/17445760.2018.1469018

Sitaula, C., Xiang, Y., Zhang, Y., Lu, X., & Aryal, S. (2019). Indoor Image Representation by High-Level Semantic Features. *IEEE Access*, *7*, 84967–84979. https://doi.org/10.1109/ACCESS.2019.2925002

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press.

van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, *109*(2), 373–440. https://doi.org/10.1007/s10994-019-05855-6

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., & Torralba, A. (2010). SUN database: Large-scale scene recognition from abbey to zoo. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3485–3492. https://doi.org/10.1109/CVPR.2010.5539970