Foreign Object Detection in Aircraft Runways Using YOLO V3

Srikanth S, Dr. D. Sathya Srinivas
M.Sc. Computer Sciences and Engineering, Dr. MGR Educational and Research Institute, Chennai, India
Faculty of Centre of Excelle.64 nce in Digital Forensics, Dr. MGR Educational and Research Institute, Chennai, India

Abstract: The FOD-R is a collection of images that depict common types of foreign object debris (FOD) that can be found on runways or taxiways. The dataset has primarily been annotated using bounding boxes to facilitate object detection. However, the FOD-xR Dataset consists of an extra feature - color codes - that offer facts on the benefit of eliminating the debris. Lightweight objects are marked in green, while heavy objects are marked in red. This centralized system simplifies coordination efforts, organization's bottom line by minimizing delays and damage caused by FOD incidents. Overall, the FOD-R and FOD-xR Datasets are valuable re-assets for agencies seeking to decorate their FOD detection and removal processes.

Keywords - FOD-R, Bounding boxes, YOLO V3 Algorithm, Foreign Object Detection.

I. INTRODUCTION

Foreign Object Debris (FOD) incidents can result in serious harm, loss of life, and substantial financial damages fromairplane destruction. Since FOD is a major safety concern at airfields, scientists are investigating the potential use of machine learning and computer vision (MLCV) technology to address this issue. For the development of more advanced and reliable models and algorithms, a dataset of FOD images and observations is necessary to improve the precision and efficiency of MLCV technology. [1].

To support the development of machine learning and computer vision technology for detecting Foreign Object Debris (FOD) in airfields, a new dataset called FOD-A was created. FOD-A's object categories were chosen based on FAA documents and previous studies, and were designed to cover various types of FOD with specific descriptors. FOD-A's images werecaptured under different lighting and rainfall conditions to replicate real-world conditions and challenge modern MLCV algorithms. The dataset includes tools for easy addition of new data while maintaining consistency in object categories. Multiple versions of FOD-A will be available to ensure the dataset is comprehensive and up-to-date. Once algorithms are assessed using a consistent version of FOD-A, the latest FOD- A data can be used to develop new algorithms that include all FOD-A object categories. A sample FOD-A image with bounding boxes is displayed in Figure 1. Many widely-used datasets are available that include a range of common object categories such as bicycles, cars, desks, and toasters. However, these datasets are not suitable for covering the necessary categories of Foreign Object Debris (FOD) found specifically in airports, such as luggage items, aircraft parts, and tools. As a result, in Section III-C, only brief comparisons will be made to general object datasets as they do not properly cover the required FOD categories due to the unique location of FOD in airports.

This paper is organized as follows: in Section II, a review of previous research related to the topic is presented. Section III provides information on how the FOD-A dataset was created, including statistics and details on how the dataset was expanded. Section IV discusses the packages and algorithms used for analysis. The main findings and future research directions are summarized in Section V. The paper's main contributions are the creation of the unique FOD-A dataset, which is available to the public, the development of a scalable system for generating image datasets, and the design and testing of new algorithms for analyzing the FOD-A dataset.

Fig.1.Example of FOD-A Images annotated with bounding boxes

II. RELATED WORKS

The FAA has issued several documents offering guidance on detecting and managing Foreign Object Debris (FOD) [1], [7], [10], [11], [17]. Given that the Federal Aviation Administration (FAA) serves as the primary source of information on FOD, the categories of objects included in the FOD dataset are based on their classification [1], [7]. Re-chosen in accordance with FAA guidelines. Section III-A of the paper provides a detailed explanation of the process used to select these categories.

A. Existing FOD Datasets

The publicly available FOD dataset, [18] mainly focuses on recognizing metal, plastic, and concrete objects [19]. but these categories do not cover all common types of FOD according to the FAA. The publicly available FOD dataset focuses on recognizing metal, plastic, and concrete objects, but it does not cover all common types of FOD as defined by the FAA. This includes tools, colorful debris, natural debris, runway equipment such as makeup chips, and other types of FOD [1],[7],[14]. In comparison, FOD-A is a more comprehensive dataset for FOD detection and analysis, as it includes 31 different object categories, which are illustrated in Figure 3.

The material recognition dataset focuses mainly on zoomed- in images and only contains three categories of objects (metal, plastic, and concrete), which is insufficient for detecting all types of FOD as per FAA guidelines. On the other hand, FOD-A includes zoomed-out images with bounding boxes, as well as weather and light categorization data, and comprises 31 object categories. The material recognition dataset has only 500 object samples and is limited to three object categories (metal, plastic, and concrete). On the other hand, FOD-A has over 1376 object samples and 31 different object categories. As a result, FOD-A is more suitable for FOD detection tasks as it provides a more accurate representation of airfield terrain and a greater variety of object categories with more descriptive object orders.[15],[16]

B. FOD Detection Method Incorporated

In this research, data visualization is used to simplify complex data and make it easier to understand and gain insights. Specifically, the number of images in both the "high risk" and "low risk" categories is presented visually [8].

In order to accomplish this, the code employs the line "cat=os.listdir(data path)" to obtain a list of directories within the specified data path. The resulting variable, 'categories', is populated with the values ['high risk', 'low risk']. Once the categories are established, the number of labels is assigned to each category by utilizing the statement "la=[i for i in range(len(cat))]". This results in labels of [0, 1]. Each category is then paired with its corresponding label through the use of the line "la dict=dict(zip(cat, la))". This generates a dictionary that indicates the mapping of each category to its respective label, which in this case is 'high risk': 0, 'low risk': 1.

DATASET

The FOD category encompasses a broad range of objects, including Fastening nut, washers, and cords used in aircraft and engine fasteners, aircraft components like fuel caps, landing Gear debris, etc as well as tools used by mechan- ics, feeding supplies, flight line materials. Other items inthis category include runway and taxiway materials such as concrete and asphalt knobs, Rubber expansion joints, Flexible rubber fittings, Joint seals, Fibers, Filaments and debris from construction like turbulent ash, and pollutants resulting from weather conditions such as snow and ice [1],[2],[3].

Fig. 2. The provided images depict FOD objects under different light-level categories in FOD-A. Specifically, the left image represents an example ofa bright condition, the middle image depicts a dim condition, and the right image shows a dark condition.

To make the dataset useful for airport FOD management, images are collected in various conditions, as weather and light conditions can vary. The collection process incorporates wet and dry environments for weather variation and bright, dim, and dark light conditions for light variation. FOD-A includes categorization labels for weather and light level, and the images provided in Figure 2 show examples of light-level categorization. The dataset does not include a snowy category since snow is promptly cleared from the airport environment.[4],[12],[13].Any remaining moisture after snow clearance should still fit into the wet category. The dry and wet weather categories in FOD-A should cover the majority of weather types applicable to airports. The categorization annotations for weather and light level are in addition to the focus of the FOD-A, which is bounding box annotations for object detection [?]. The current method was tested using includes 1376 images [20], of which 690 images contain high-risk objects and the remaining 686 images contain low-risk objects. Figure 1 shows that most of the objects in the images are small. Images of common FOD were captured using portable and unmanned aerial vehicle (UAV) cameras in video (mp4) format. Using the UAV camera allowed for variation in recording distances, while the images collected by the portablecamera were closer to the object and had more drastic camera angle changes. The video format facilitated large-scale image collection, but presented some initial challenges, as not all frames in a video contained the target object(s), and empty frames could pollute the dataset. To address these issues, a small command-line tool was developed to efficiently trim videos to proper intervals and separate each frame to an imageformat, making the image collection process more efficient and enabling dataset expansion with simple instructions. The tool also normalized the images.

III. THE PROPOSED APPROACH

using the ’zip’ function. The resulting ’label dicta variable displays the categories alongside their respective labels [21].

2. Conversion of RGB image to Grey image: Converting color images to grayscale is typically not a primary focus of current image recognition systems, which generally operate on grayscale images. The reason for this is that the technique utilized for the conversion of color images to grayscale does not have a considerable impact on the image descriptors’ efficiency. Adding unnecessary information to the images can increase the amount of training data needed to obtain accurate results. By using grayscale images, the algorithm is simplified and computational requirements are reduced, which makes it easier to extract descriptors compared to working directly with-color-images-[9],[25],[26].

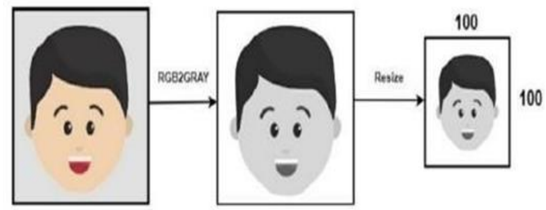

Fig. 3. The process of converting an RGB image to a grayscale image of 100x100 size.

To convert an image from color to grayscale, the cv2.cvtColor() function is used, where the input image and a flag indicating the type of conversion are passed as arguments [9]. In the case of the proposed method, the flag cv2.COLOR BGR2GRAY is used to convert the image to grayscale [27],[28].

To use deep convolutional neural networks, it is necessary to have images of a fixed size as input. To achieve this, the cv2.resize() function is utilized to alter the resolution of all images in the dataset to a common fixed size. In this case, the function is applied to the grayscale image and resizes to 100 x 100 pixels [29].

1) Image Scaling: To analyze an image, it is necessary to use a three-dimensional tensor in which each channel represents a unique pixel. It is important to note that all images must have corresponding 3D tensors and be of the same size, the images themselves may differ in content and have different characteristic tensors [8]. The requirement of properly formatted images poses a challenge When collecting data and preparing the model for most CNNs. Anyways, one way to address this limitation is by altering the input images before inputting them into the network [4].

The images are adjusted so that the pixel values fall between 0 and 1, and then they are transformed into four-dimensional arrays using the function np.reshape(). The shape of the array is (number of images, image size, image size, 1), where 1 represents grayscale. The neural network’s final layer has two outputs, one for “high risk” and one for “low risk,” which means it is categorical. Therefore, the data is changed into categorical labels.

B. Training the model

1) Building the model using CNN architecture: CNNs are a popular choice for a wide range of computer vision tasks due to their high performance [11]. In this study, the Sequential CNN approach is employed.

The proposed methodology comprises two primary com- ponents: a cascade classifier and a pre-trained CNN that comprises two 2D convolution layers and dense neuron layers. Below are the steps or procedures for implementing the Foreign Object Detection algorithm.

Algorithm 1: Foreign Object Detection

Input A dataset containing categories labeled as high risk and low risk.

Output An image that has been classified based on the presence of foreign objects on the runway.

for each image in the dataset do

1] Visualize the image in two categories and label them

2] Convert the RGB image to Gray-scale image

3] Resize the gray-scale image into 100 x 100

4] Normalize the image and convert it into 4-dimensional array end

for building the CNN model do

1] Add a Convolution layer of 200 filters

2] Add the second Convolution layer of 100 filters

3] Insert a Flatten layer to the network classifier

4] Add a Dense layer of 64 neurons

5] Add the final Dense layer with 2 outputs for 2 categories end Split the data and train the model

A. Data Processing

Data preprocessing is the technique of transforming raw data from its initial format to a more accessible and useful format, such as tables, images, videos, or graphs. The objective is to convert the data into structured information that conforms to a data model or structure, capturing the relationships between different elements of the data. This step is vital in data analysis, as it enables efficient data interpretation and extraction of insights [5],[22],[23].

In the proposed system, the data being processed is in the form of images and videos, and the task is carried out using Numpy and OpenCV. These tools aid in the extraction of meaningful information from images and videos. For instance, image segmentation techniques can be employed to identify specific objects within an image or video. Furthermore, OpenCV can be used to perform various image processing functions such as filtering, resizing, and thresholding. The combination of Numpy and OpenCV enhances the quality and accuracy of the structured data obtained from the images and videos, which can be used for In this specific case, an ideal number of 64 neurons is used, which is not too excessive, as too many neurons and filters can negatively impact the model's performance. Additionally, optimized stride values and pool size are employed to filter out the main components of the image, allowing for precise detection of the foreign object without causing over-fitting. detection of the foreign object without causing over-fitting. Figure 6 provides evidence that achieving high accuracy in object detection significantly reduces the cost of errors. This figure may show a comparison between the cost of errors for models with different levels of accuracy or highlightvarious applications such as machine learning and computer vision [24].

1) *Data Visualization:* Data visualization refers to the procedure of converting complex or abstract data into easy-to- understand and insightful representations through the use of encoding techniques. This approach helps in the identification and comprehension of patterns present within the dataset. [6]. To visualize the number of images in the 'high risk' and 'low risk' categories of the dataset, the code employs several steps. Firstly, the list of directories in the specified data path is assigned to the 'categories' variable using "cat=os.listdir(data
2) path)". Next, the 'labels' variable is created using the range of the number of categories. Finally, the 'label dicta variable is used to map each category to its corresponding label, which is accomplished by pairing the items in each iterator together image

The following step involves the initialization of a CNN model and the addition of a layer for convolution. After that, we perform pooling operations and add two more convolutional This layers. Then, we flatten the output of the convolutional Layer and add fully connected layers to create the final output layer.

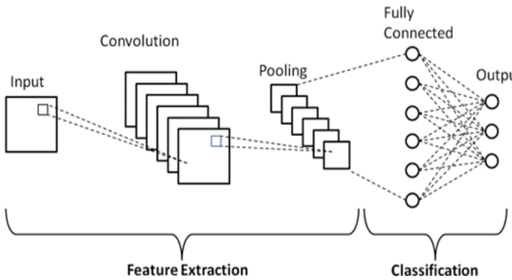

Fig. 4. The output shape can be calculated using the formula (inputsize - pooling window size + 1)/Leaps, where the default valueof Leaps is (1, 1) [7].

3) Dividing the data into separate sets and training the CNN model: Partitioning the data into a training and testing set is a crucial step in developing reliable predictive models. To achieve this, a train-test split is performed where the dataset is split into two partitions, one for training the model and the other for evaluating its performance. In this case, the dataset is split into two parts, where 10% of the data is set aside for testing while the remaining 90% is used for training. The confirmation loss is monitored during the training process using Model Checkpoint. The Sequential model is then trained and tested on these sets, with 20% of the training data being used for validation. To avoid overfitting, the model undergoes 20 epochs during the training process, this value is chosen as a compromise between achieving high accuracy and avoiding the potential risk of overfitting. The proposed model is visuallyrepresented in Figure 5.

IV. RESULTS AND DISCUSSION

The dataset was utilized to train, validate, and test the model, which achieved an accuracy rate of 95.77%. (as shownin Figure 7). One of the primary reasons for the model's high accuracy down-sampling is the primary reason for achieving technique reduces the dimensional space of the input and grants translation in variance to the model's internal high accuracy in detecting foreign objects in images. This representation, resulting in a reduced number of that the model needs to learn.

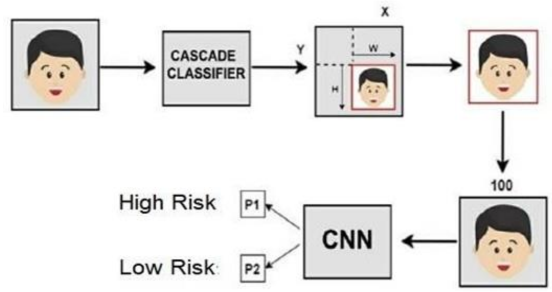

Fig. 5. Overview of the Model detection of the foreign object without causing over-fitting.

Figure 6 provides evidence that achieving high accuracyin object detection significantly reduces the cost of errors. This figure may show a comparison between the cost of errors for models with different levels of accuracy or highlightthe economic impact of misclassifications in this domain. By demonstrating the importance of high accuracy, the figure emphasizes the need for techniques like down-sampling, optimized stride values, and pool size, as mentioned in the paragraph, that can improve the model's performance and accuracy in detecting foreign objects. Ultimately, achieving high accuracy is vital for minimizing errors' cost in this context, where accuracy is crucial for ensuring safety and security.

The FOD system incorporates weather and light catego rization inputs, and experiments were performed to evaluate their efficiency using wet and dry weather inputs. The model with two inputs was effective in distinguishing between wet and dry backgrounds in FOD-A images, and its precision rapidly improved on confirmation data. However, some un- usual predictions were observed in the testing data, and it couldbe challenging to combine categorization and bounding box detection for advanced algorithms. Nevertheless, the weather and light inputs may have practical applications in the future.

V. CONCLUSION AND FUTURE WORKS

The paper starts by providing a clear explanation of the reason behind the work, and the learning and performance objectives of the model are presented in an easy-to-understand

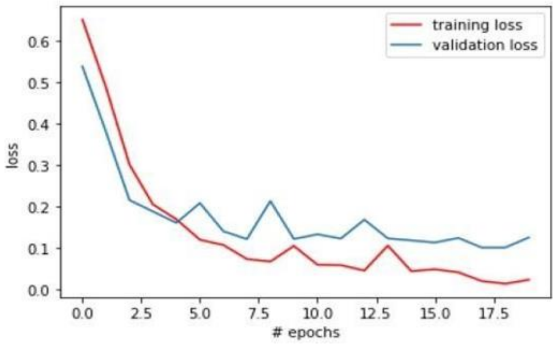

Fig. 6. This refers to a graph that shows the relationship between the number of training epochs and the loss observed in the dataset during training.

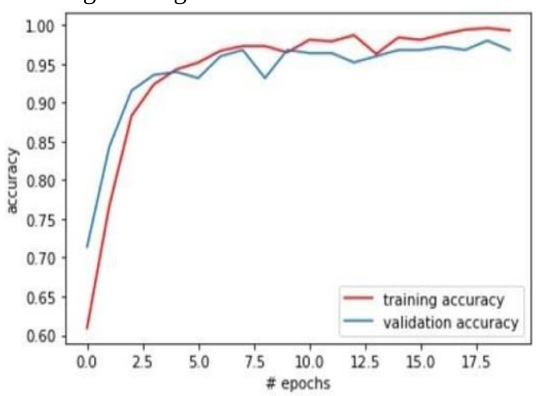

Fig. 7. This refers to a graph or chart that shows the relationship between the number of epochs used during the training process and the accuracy attained on the training dataset during the training process manner. The model has achieved high accuracy using basic machine learning tools and simplified approaches. The system has various applications, including preventing accidents and rapidly detecting and eliminating foreign objects.

REFERENCE

[1] Munyer, T., Huang, P. C., Huang, C., & Zhong, X. (2021, October 6). FOD-A: A Dataset for Foreign Object Debris in Airports.

[2] S. Srinivasan, R. Rujula Singh, R. R. Biradar and S. Revathi, "COVID- 19 Monitoring System using Social Distancing and Face Mask Detec- tion on Surveillance video datasets," 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2021.

[3] N. Saurabh *et al.*, "Efficient Face Mask Detection For Different Surveil- lance Applications Using Python",

[4] A. Das, M. Wasif Ansari and R. Basak, "Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV".

[5] Dey, S.K., Howlader, A., Deb, C. (2021). MobileNet Mask: A Multi- phase Face Mask Detection Model to Prevent Person-To-Person Trans- mission of SARS-CoV-2.

[6] H. Adusumalli, D. Kalyani, R. K. Sri, M. Pratapteja and P. V. R. D. P. Rao, "Face Mask Detection Using OpenCV",

[7] N. Yonemoto, S. Futatsumori, A. Kohmura, K. Morioka and N. Miyazaki, "Multi-static Millimeter Wave Radar Connected by Radio over Fiber for Foreign Object Detection on Runways".

[8] Annie Silviya S H, Pathmanaban J, Kingsley S, Sriman B, Suryaa Narayanan K and Santhosh Kumar E, "Prediction and Prevention Analysis Using Machine Learning Algorithms for Detecting the Crime Data".

[9] G. Harriat Christa, J. J, A. K and K. M. Sagayam, "CNN- based Mask Detection System Using OpenCV and MobileNetV2,".

[10] Y. Wang, Q. Song, J. Wang and H. Yu, "Airport Runway Foreign Object Debris Detection System Based on Arc-Scanning SAR Technology,".

[11] based Mask Detection System Using OpenCV and MobileNetV2,".

[12] Y. Wang, Q. Song, J. Wang and H. Yu, "Airport Runway Foreign Object Debris Detection System Based on Arc-Scanning SAR Technology,".

[13] Yadav, Shashi. (2020). Deep Learning based Safe Social Distancing and Face Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence. International Journal for Research in Applied Science and Engineering Technology.

[14] M. Ote, S. Jeong and M. M. Tentzeris, "Foreign Object Detection for Wireless Power Transfer Based on Machine Learning,".

[15] Sneha Sen and Khushboo Sawant 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1070 012061. DOI 10.1088/1757-899X/1070/1/012061

[16] K. Suresh, M. Palangappa and S. Bhuvan, "Face Mask Detection by using Optimistic Convolutional Neural Network,".

[17] I. B. Venkateswarlu, J. Kakarla and S. Prakash, "Face mask detection using MobileNet and Global Pooling Block,".

[18] W. Vijitkunsawat and P. Chantngarm, "Study of the Performance of Machine Learning Algorithms for Face Mask Detection,".

[19] N. Yonemoto, S. Futatsumori, A. Kohmura, K. Morioka and N. Miyazaki, "Multi-static Millimeter Wave Radar Connected by Radio over Fiber for Foreign Object Detection on Runways,".

[20] Federal Aviation Administration,"Airport foreign object debris manage- ment,".

[21] Federal Aviation Administration, "Airport foreign object debris detection equipment," Federal Aviation Administration, Tech. Rep., 09 2009, advisory circular: 150/5220-24

[22] C. M'Sila, R. Ayad and N. Ait-Oufroukh, "Automated Foreign Object Debris Detection System based on UAV,".

[23] J. Wang, S. Yan, L. Zhou, R. Fan, X. Han and X. Wang, "Research on Application of New Technology in Foreign Object Debris Detection Field,".

[24] W. Wang, B. Wu, S. Yang and Z. Wang, "Road Damage Detection and Classification with Faster R-CNN,".

[25] D. Arya et al., "Crowdsensing-based Road Damage Detection Challenge (CRDDC'2022),".

[26] X. Wang, Z. Lan, Z. Li, Y. Lin and Q. Xiao, "A Practical Application of Runway Foreign Object Debris Detection System at the Airport,".