FOD-A: A Dataset for Foreign Object Debris in Airports

Travis Munyer
Dept. of Computer Science
Univ. of Nebraska Omaha
tmunyer@unomaha.edu

Pei-Chi Huang
Dept. of Computer Science
Univ. of Nebraska Omaha
phuang@unomaha.edu

Chenyu Huang
Aviation Institute
Univ. of Nebraska Omaha
chenyuhuang@unomaha.edu

Xin Zhong
Dept. of Computer Science
Univ. of Nebraska Omaha
xzhong@unomaha.edu

***Abstract*—Foreign Object Debris (FOD) detection has attracted increased attention in the area of machine learning and computer vision. However, a robust and publicly available image dataset for FOD has not been initialized. To this end, this paper introduces an image dataset of FOD, named FOD in Airports (FOD-A). FOD-A object categories have been selected based on guidance from prior documentation and related research by the Federal Aviation Administration (FAA). In addition to the primary annotations of bounding boxes for object detection, FOD-A provides labeled environmental conditions. As such, each annotation instance is further categorized into three light level categories (*bright*, *dim*, and *dark*) and two weather categories (*dry* and *wet*). Currently, FOD-A has released 31 object categories and over 30,000 annotation instances. This paper presents the creation methodology, discusses the publicly available dataset extension process, and demonstrates the practicality of FOD-A with widely used machine learning models for object detection.**

***Index Terms*—Image Dataset, Foreign Object Debris, Computer Vision, Machine Learning**

I. Introduction

Accidents caused by Foreign Object Debris (FOD) are responsible for severe injuries or death, and billions of dollars in damages to aircraft [1]. FOD is a critical safety hazard in airports, and machine learning and computer vision (MLCV) technology has been shown as a potential solution in exploratory research [2]–[5]. In order to facilitate this application of MLCV, a dataset of FOD images and annotations is required to be utilized and organized for more sophisticated and robust models and algorithms.

To foster future FOD-based MLCV work, we developed the novel dataset named Foreign Object Debris in Airports (FOD-A). FOD-A object categories are influenced by relevant FAA documents and previous research [1], [6]–[11]. These object categories are designed to cover several FOD types while including specific labels that are descriptive. Images are collected under varying light and weather conditions to ensure accurate simulation of typical airport environments. Varying conditions also ensure that the FOD-A dataset is challenging for modern MLCV algorithms. As a preview, FOD-A images with example bounding boxes are shown in Figure 1. Since FOD is a continually evolving datatype, it is important that a FOD dataset can be easily expanded. To enable extensibility, this dataset includes tools that allow the addition of new data with ease. Since extensibility and consistency may be contradictory traits, it will be important for several iterations of FOD-A to remain available. Once algorithms are compared using a consistent iteration of FOD-A, new algorithms can be implemented using the most current FOD-A data. This ensures all current FOD-A object categories are included in final detection algorithms.

Fig. 1. FOD-A dataset images with bounding box annotation examples.

Currently, there are several well-known general datasets available that contain diverse categories of common objects, (*e.g.*, bicycles, cars, desks, toasters) [12], [13]. Due to the location of the FOD datatype (*i.e.* airports), these datasets [12], [13] do not properly cover necessary categories of FOD (*e.g.*, luggage items, aircraft parts, tools). Because of this, comparisons to datasets of general objects will only be briefly made in Section III-C.

The rest of this paper is organized as follows. The related work is reviewed in Section II. A description of dataset creation methodology, FOD-A statistics, and the extension process is provided in Section III. Section IV presents the initial experimentation and algorithmic results. Finally, Section V concludes the paper and discusses future work. As a summary, the major contributions of this work are as follows: 1) the initialization and creation of the novel and publicly available

This work is partially funded by the NASA Nebraska Space Grant (Federal Award #NNX15AI09H). This work is also funded by Nebraska University (NU) Collaboration Initiative Grant.

dataset FOD-A; 2) the design and development of an efficient and abstractable method of image dataset creation; and 3) the implementation and evaluation of an initial algorithmic analysis of FOD-A.

II. Related Work

The FAA published several documents [1], [6]–[11] providing guidance on FOD detection and management. As the main resource for FOD information, the object categories in the FOD dataset are based off the FAA documentation [6], [7]. The details of the category selection process are described in Section III-A.

A. Existing FOD Datasets

A publicly accessible FOD dataset [4] does exist. However, this dataset primarily focused on material recognition, including the following three object categories: metal, plastic, and concrete [4]. Using only these three object categories does not cover all common types of FOD according to the FAA's [7] information. For example, these object categories cannot cover some types of tools, various common airport garbage (*e.g.,* paper, soda cans), animals and other natural debris, some runway materials such as paint chips, and other common FOD [1], [7]. For comparison, FOD-A provides 31 object categories (see Figure 3).

Furthermore, images contained in the material recognition dataset are in a zoomed-in format. It is likely that images collected during applied FOD detection tasks will not be zoomed into objects, so FOD-A provides images in a zoomed-out format with bounding boxes (see Figure 1). Also, the material recognition dataset contains about 3000 object instances, while the FOD-A dataset contains over 30,000 object instances. In summary, the FOD-A dataset is more appropriate for FOD detection tasks because it contains forms of annotation better suited to the airport environment (*i.e.* bounding box annotation plus weather and light categorization annotation), several more object instances, and descriptive object categories.

B. Related MLCV FOD Detection Methods

FOD-focused MLCV research is becoming increasingly common [2]–[5]. Several papers have been published that implement algorithms to detect FOD. These papers have created their own private datasets and have generally kept these datasets small. The effectiveness of major datasets in several research tasks [12], [13] can partially be attributed to focused dataset development and publicly accessible versions. It is a much larger task to create a robust dataset [14] when also presenting new detection methods. Thus, FOD-A should enable researchers to focus on the improvement of FOD detection algorithms.

P. Li and H. Li [3] did create their own small dataset for internal use. Their dataset consists of about 2000 images, with 100 images per object class [3]. Although their small and private dataset may be feasible for their proposed algorithms, it may not be suited to larger scale experiments.

Having considered the shortcomings and benefits of prior works discussed above, we have developed FOD-A with three main advantages: 1) FOD-A provides a wide range of descriptive object categories selected by FAA documentation, and includes a large number of instances for each category; 2) FOD-A is publicly available with documented expansion processes; and 3) FOD-A considers realistic and challenging data samples in varying weather and light conditions.

III. Dataset Construction

This section presents FOD-A in more details, including the creation methodology, FOD-A statistics, and the extension process.

A. Image Collection

According to the FAA, FOD commonly includes the following: "aircraft and engine fasteners (nuts, bolts, washers, safety wire, etc.); aircraft parts (fuel caps, landing gear fragments, oil sticks, metal sheets, trapdoors, and tire fragments); mechanics' tools; catering supplies; flight line items (nails, personnel badges, pens, pencils, luggage tags, soda cans, etc.); apron items (paper and plastic debris from catering and freight pallets, luggage parts, and debris from ramp equipment); runway and taxiway materials (concrete and asphalt chunks, rubber joint materials, and paint chips); construction debris (pieces of wood, stones, fasteners and miscellaneous metal objects); plastic and/or polyethylene materials; natural materials (plant fragments, wildlife and volcanic ash); and contaminants from winter conditions (snow, ice)" [6].

The FAA suggests that metal objects are the most common FOD (over 60 percent of the materials found in a one-year airport study) [6]. We generalize the FAA description into categories suitable for MLCV applications with priority given to objects that have the highest potential to harm aircraft (*i.e.* become ingested and damage jet engines, shred tires, etc.). The resultant categories for this implementation of the FOD-A dataset can be found in Figure 3.

In order to create a practical dataset that is applicable to airport FOD management, we collect images in diverse conditions. Weather and light conditions in airports vary, so a dataset of FOD objects must incorporate this fact into included data. Wet and dry environments provide weather variation for FOD-A image collection. For light variation, the image collection process incorporates bright, dim, and dark light conditions. Since each of these environmental variations could be easily abstracted to fit categorization tasks, FOD-A includes categorization labels for weather (dry and wet) and light-level (bright, dim, and dark). Example light-level categorization images are provided in Figure 2. Since snow is promptly cleared from the airport environment, it is unnecessary to include a snowy category. Any moisture that remains after snow is cleared should still fit into the wet category. FOD-A's dry and wet weather categories should cover the majority of weather types applicable to airports. Remarkably, the weather and light-level catgorization annotations are in addition to the

Fig. 2. Example images from various FOD-A light-level categories. Example *bright* image (left), an example *dim* image (middle), and an example *dark* image (right).

focus of the FOD-A, which is bounding box annotations for object detection.

Images of common FOD are collected in the video (mp4) format using both portable and unmanned aerial vehicle (UAV) cameras. UAV image collection allows variation in recording distances that could not be achieved with handheld portable cameras. The images gathered by the portable camera were closer to the object, and camera angles changed more drastically than the UAV camera. As a video is densely packed with images, the video format allows for large-scale image collection. However, utilizing the video format presented a few initial issues. Some videos do not have the target object(s) in each frame, and the empty frames could uselessly pollute the dataset. A video would have to be trimmed to proper intervals, and then each frame separated to an image format. Performing this task by hand is very time-consuming and prevents extensibility. We created a small command-line tool to solve a few of these issues. The benefits of this tool are as follows: 1) it allows for that dataset to be easily expanded using simple instructions; 2) it makes the image collection process more efficient; and 3) it normalizes the images.

This tool takes a video file as an input, and then creates a folder structure, a trimmed version of the video, and the frames for the video. The trimmed video can then be used as input into Computer Vision Annotation Tool (CVAT) [15], which makes the annotation processes efficient. Further details of the annotation process are described in section III-B. Each folder structure, once the annotations are added, corresponds to the set of annotations and frames for one video. The folder structures are stored together; this forms the original FOD-A format. We provide the tools that can convert the FOD-A format to the Pascal Visual Object Classes format (called Pascal VOC) [12] as commonly required by algorithmic processes.

The expansion tool is designed to further edit the video by automatically reducing the FPS of each video to 15. The tool then generates the frames after the FPS reduction. Because the original FPS of each video was about 60 FPS, this is an effective method to prevent duplicate images. The output location for the folders and frames is stored in a settings file to allow efficient expansion. Based on different settings, the expansion tool can automatically apply to either create a new dataset (if targeting a new folder) or to expand an existing dataset (if targeting an existing directory). The abstract design of the tools can enable the process and format to fit any image datatype.

Because each of the original videos include consistent light and weather conditions, the expansion tool automatically generates the categorization annotations once the weather and light conditions are specified. As the tool outputs each individual frame to the folder structure, it saves the correct categorization annotations along with a relative file path to the new image in a Comma-Separated Values (CSV) file.

The produced images and annotations for FOD-A have been uploaded to the the GitHub repository in the original format and in the Pascal VOC format [12]. This GitHub page also contains the detailed dataset expansion instructions. These instructions can also be used to create new image datasets. Additional images can be added to the dataset by inputting more videos into the expansion tool. Required changes to the dataset, such as format extension and data preparation, are made automatically by the expansion tool after additions. Inserting annotations to the created folder structure is the only additional requirement for extension. This enables the dataset expansion process to be efficient.

B. Image Annotation

To create a robust dataset, it is best to include as many instances of FOD objects as possible. Therefore, an efficient and quality annotation process was implemented. Video data can be quickly and accurately annotated using the existing tracking algorithms for videos. The efficient annotation process was provided by the open source tool, CVAT [15]. Since the FOD-A expansion tool outputs a matching trimmed video as well as the video's frames, the images are annotated as if they are still a video format. CVAT's video annotation algorithm only requires every ten frames (an interval that can be modified) to be annotated, with manual adjustments as necessary. The annotations for other frames are generated mathematically using the two manually created annotations. The in-between frames still need to be validated to ensure accuracy, but we found that this only requires minor adjustments. We then export the annotations in a standard XML format (*i.e.*, Pascal VOC) [12]. Once exported, we simply place annotations in the relevant folder created by the expansion tool.

Once the annotation process was completed for the initial data, the size of the dataset was too cumbersome for typical object detection methods and for ease of storage. At this point, the images included sizes that varied anywhere from $2k$ to $4k$ resolution, and a total dataset size of over 100 gigabytes. To overcome this issue while producing an extensible dataset, we created a second command-line tool. This resizing tool targets all applicable folders within the target folder, so it can be utilized on a single annotation/image combo or on the entire dataset at once. Once the images and annotations are resized, the dataset storage size is drastically reduced (to about 5 gigabytes in this case).

This resizing tool scales all properly formatted XML and image data to the specified size, whether smaller or larger than current size. We resize the images and annotations to 400×400 resolution to facilitate unified-size modeling, while the original images are also made available. The resize tool can optionally display all the images with their bounding boxes and labels.

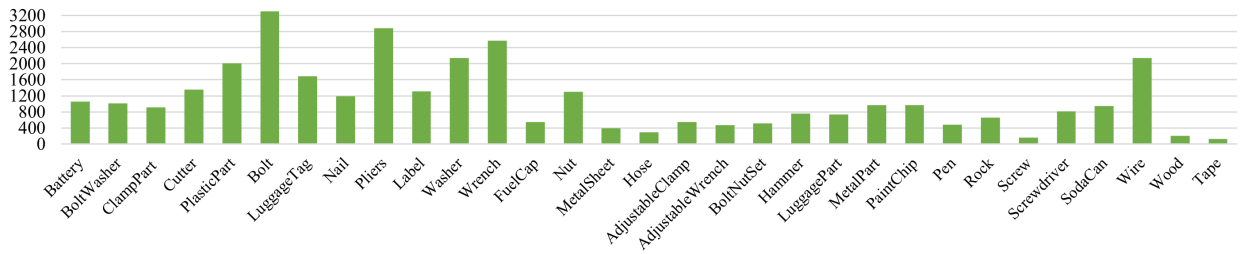


Fig. 3. Instances per category in the FOD-A dataset.

The images displayed in Figure 1 are an example output of this tool. This simply allows images to be visually inspected as the dataset is resized. Some major annotation errors in famous datasets been found [16]. This visual inspection process allows additional validation of annotations as FOD-A expands, which aids in the prevention of similar errors [16].

C. Dataset Statistics

After the initialization of the FOD-A dataset, there are a total of 31 object categories and over 30,000 annotation instances. Figure 3 shows instances per category for the bounding box annotations and Table I shows the statistics for the light-level and weather categories.

TABLE I
CATEGORIZATION STATISTICS

Weather		**Light-Level**		
Dry	Wet	Dark	Dim	Bright
26647	7216	4387	12464	17012

The material recognition dataset discussed in Section II [4], contained a total of 3 object categories and $3,440$ annotation instances. As such, the potential applications of FOD-A and the materials recognition dataset [4] may differ greatly. A few datasets may be considered when analysing FOD-A, such as the Pascal VOC [12] and the Microsoft COCO [13] datasets. However, the object categories contained in these datasets are of everyday objects and have a more general application scope than FOD-A. Image datasets of debris in airports should contain FOD specific object categories. For this reason, datasets of everyday objects are not directly comparable to FOD-A.

IV. ALGORITHMIC ANALYSIS

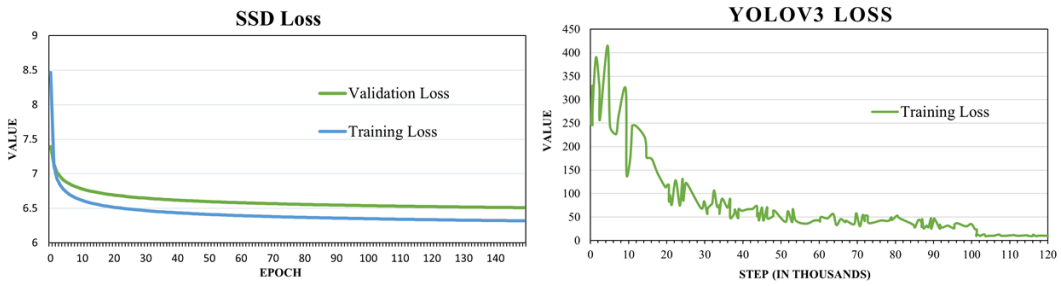
To validate dataset functionality in abstract scenarios, it is necessary to implement FOD-A in several common MLCV algorithms. As mentioned previously, this dataset focuses on bounding-box based object detection functionality, but the potential of FOD-A is expanded by including categorization annotations. The categorization annotations are tested using a simple binary classification model described later in this section. The viability of the bounding box annotations are tested using the two famous algorithms: You Only Look Once Version 3 (YOLOv3) [17] and Single-Shot Multi-box detector (SSD) [18]. Existing implementations of SSD [19] and YOLOv3 [20] are used. The categorization accuracy and mean intersection over union [21] (IOU) metrics are used to compare results between the algorithms. Both metrics are calculated using predictions on the validation dataset. Accuracy is computed as the number of correct categorization predictions over the total number of predictions. To calculate the mean IOU [21], the IOU value of predictions in true positive categorization cases are averaged. IOU values from only true positive categorization cases facilitates separation from accuracy results.

The initial experimentation began using YOLOv3. As shown in Figure 4(b), the loss reaches a value of about 7.05. For this implementation [20], the loss is calculated using the method presented in the original YOLOv3 paper [17]. In this experimentation, YOLOv3 produces categorization accuracy of 12.42% and a mean IOU of 47.58% on FOD-A validation data. As the accuracy and mean IOU metrics suggest, the YOLOv3 algorithm commonly predicts an incorrect categorization label, but regularly produces correct bounding boxes.

As shown in Figure 4(a), SSD loss reaches a value of about 6.51; the loss result approaches convergence after about 140 epochs. The loss in this implementation [19] is calculated using the method defined in the original SSD paper [18]. SSD provides categorization accuracy of 71.81% and a mean IOU of 68.05%.

In this experimentation, SSD produced better results than YOLOv3. However, the scope of this paper is mostly restricted to the presentation of the FOD-A dataset, as algorithm optimization for FOD-A is future work. As intended, FOD-A proved to be difficult for the modern YOLOv3 and SSD algorithms. This provides room for future algorithmic enhancement, in both efficiency and accuracy.

Since FOD-A also includes categorization annotations, we examine this functionality with a binary categorization model using transfer learning. With the output layer removed and substituted for both a max pooling and a fully-connected layer with two output neurons, this binary classification model was built using the MobileNetv2 [22] architecture with weights pretrained on ImageNet [23]. FOD-A includes weather and light categorization annotations. To test the functionality of the categorization annotations, we conducted experiments using the two weather annotations, *wet* and *dry*. The binary categorization model quickly became skillful, and was able to effectively distinguish between a *wet* and *dry* background in FOD-A images. The accuracy quickly improved to the maximum percentage on validation data. The model is capable of correctly categorizing most images, but there were still some outlier predictions on the testing data. Although the weather classification alone can be solved quickly, the combination of the categorization and bounding box detection prove difficult



(a) (b)

Fig. 4. Loss curve from SSD and YOLO experiments.

for modern algorithms. Additionally, the weather and light annotations could prove useful in future practical work.

V. Conclusion

MLCV has produced promising results for various tasks in FOD. However, to the best of our knowledge, a proper publicly available dataset of FOD has not been initialized prior to the proposed work. This paper introduces FOD-A and proposes an abstractable method of image dataset creation. As discussed previously, FOD-A object categories have been selected based on prior FAA documentation and research. This enables comprehensive coverage of common FOD. Moreover, we have developed an efficient and publicly documented expansion process and intend to make several extensions of FOD-A available on the GitHub repository. An efficient expansion process is important since FOD is a continually evolving datatype. To simulate airport environments, images in FOD-A contain varying light and weather conditions. In addition to the bounding box annotation, we provide these weather and light conditions as categorization labels. We also validate these approaches for both the object detection and the categorization functionalities of FOD-A on different algorithms. The experimental results demonstrate FOD-A's practicality and difficulty.

There are several research paths that could be followed to build on this work. One direction for future work is to develop more efficient and accurate object detection techniques for the FOD datatype. Once detection algorithms are improved, further works could explore practical experimentation in airports.

VI. Resources

GitHub: https://github.com/FOD-UNOmaha/FOD-data

References

[1] M. Alexander-Adams, "Fact sheet – foreign object debris (fod)," https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=15394, Nov 2013.

[2] Z.-D. Yuan, J.-Q. Li, Z.-N. Qiu, and Y. Zhang, "Research on FOD detection system of airport runway based on artificial intelligence," *Journal of Physics: Conference Series*, vol. 1635, p. 012065, nov 2020. [Online]. Available: https://doi.org/10.1088/1742-6596/1635/1/012065

[3] P. Li and H. Li, "Research on fod detection for airport runway based on yolov3," in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 7096–7099.

[4] H. Xu, Z. Han, S. Feng, H. Zhou, and Y. Fang, "Foreign object debris material recognition based on convolutional neural networks," *EURASIP Journal on Image and Video Processing*, vol. 2018, p. 21, 04 2018.

[5] X. Cao, P. Wang, C. Meng, X. Bai, G. Gong, M. Liu, and J. Qi, "Region based cnn for foreign object debris detection on airfield pavement," *Sensors (Basel, Switzerland)*, vol. 18, p. 737, 3 2018.

[6] Federal Aviation Administration, "Airport foreign object debris management," Federal Aviation Administration, Tech. Rep., 09 2010, advisory circular: 150/5210-24.

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

[8] E. Herricks, P. Lazar, E. Woodworth, and J. Patterson, "Performance assessment of a mobile, radar-based foreign object debris detection system," University of Illinois and Federal Aviation Administration, Tech. Rep., 09 2011.

[9] E. Herricks, E. Woodworth, S. Majumdar, and J. Patterson, "Performance assessment of a radar-based foreign object debris detection system," University of Illinois and Federal Aviation Administration, Tech. Rep., 02 2011.

[10] E. Herricks, P. Lazar, E. Woodworth, and J. Patterson, "Performance assessment of an electro-optical-based foreign object debris detection system," University of Illinois and Federal Aviation Administration, Tech. Rep., 03 2012.

[11] E. Herricks, E. Woodworth, and J. Patterson, "Performance assessment of a hybrid radar and electro-optical foreign object debris detection system," University of Illinois and Federal Aviation Administration, Tech. Rep., 05 2012.

[12] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, Jan. 2015.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[14] J. Zhang, F.-W. Sun, J. Song, A. Von Ancken, and R. Zhai, "Fine-grained image classification via spatial saliency extraction," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018, pp. 249–255.

[15] "Computer Vision Annotation Tool," https://github.com/openvinotoolkit/cvat, 2021, last accessed on July 1.

[16] C. G. Northcutt, A. Athalye, and J. Mueller, "Pervasive label errors in test sets destabilize machine learning benchmarks," 2021.

[17] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. [Online]. Available: http://arxiv.org/abs/1512.02325

[19] "object-detection-in-keras," https://github.com/Socret360/object-detection-in-keras, 2021, last accessed on July 6.

[20] "keras-yolo3," https://github.com/experiencor/keras-yolo3, 2021, last accessed on July 6.

[21] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2014.

[22] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," 2019.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.