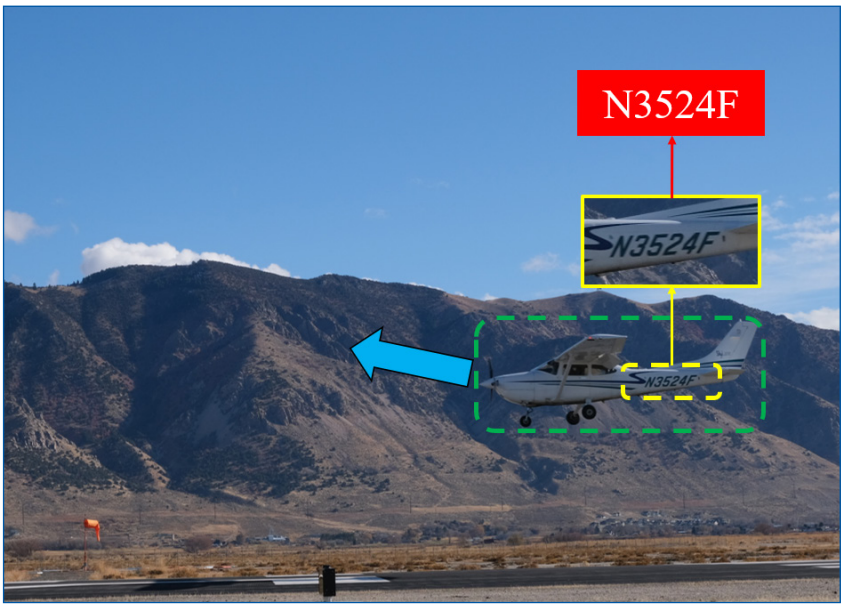
MOUNTAIN-PLAINS CONSORTIUM

MPC 22-476 | M. Farhadmanesh, A. Rashidi, and N. Marković

AUTOMATED IMAGE-BASED AIRCRAFT TRACKING AND RECORD-KEEPING FOR UTAH AIRPORTS



MOUNTAIN-PLAINS CONSORTIUM REGION 8

A University Transportation Center sponsored by the U.S. Department of Transportation serving the Mountain-Plains Region. Consortium members:

Colorado State University
North Dakota State University
South Dakota State University
University of Colorado Denver
University of Denver
University of Utah
Utah State University
University of Wyoming

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
MPC-639		

4. Title and Subtitle	5. Report Date
Automated Image-based Aircraft Tracking and Record-keeping for Utah Airports	August 2022
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Mohammad Farhadmanesh, Abbas Rashidi, Nikola Marković	MPC 22-476

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
University of Utah Department of Civil and Environmental Engineering 110 South Central Campus Drive, Suite 2000 Salt Lake City, UT 84112	11. Contract or Grant No.

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Mountain-Plains Consortium North Dakota State University PO Box 6050, Fargo, ND 58108	Final Report
	14. Sponsoring Agency Code

15. Supplementary Notes

Supported by a grant from the US DOT, University Transportation Centers Program

16. Abstract

Airport operational data is necessary for aviation decision-makers to quantitatively prepare airport master plans and fairly distribute national funds throughout the country. These data have other important applications, such as safety and security applications as well. Interestingly, most airports (i.e., typically general aviation airports) do not have control towers or staff to document the aircraft operations in their airport. That considered, many attempts have been made to automatically record the operations using radio, acoustics, and ADS-B technologies. While these methods come with some advantages, they cannot accurately monitor aircraft operations at non-towered airports due to technological and commercial shortcomings. Therefore, this project attempts to develop intelligent cameras for performing operation count and recognition by using machine learning techniques. The proposed method addresses the issues that are existed with the other methods. A camera system is a passive system that does not require cooperative aircraft. Moreover, our data analysis has shown that the proposed vision-based system holds the promise of an accurate system regarding both counting and recognizing the aircraft operations at non-towered airports during the busy traffic times in those airports.

17. Key Word	18. Distribution Statement
airports, air traffic control, automation, image analysis, traffic counting, vehicle detectors	Public distribution

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	50	n/a

Automated Image-Based Aircraft Tracking and Record-Keeping for Utah Airports

Mohammad Farhadmanesh
Dr. Abbas Rashidi
Dr. Nikola Marković

University of Utah
Salt Lake City, UT

August 2022

Acknowledgments

This project is funded by the Utah Department of Transportation (UDOT) and Mountain-Plains Consortium (MPC). The opinions and findings of the authors do not necessarily reflect the view and opinions of UDOT and MPC.

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented. This document is disseminated under the sponsorship of the Mountain-Plains Consortium, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

ABSTRACT

Airport operational data are necessary for aviation decision-makers to quantitatively prepare airport master plans and fairly distribute national funds throughout the country. These data have other important applications, such as safety and security applications. Interestingly, most airports (i.e., typically general aviation airports) do not have control towers or staff to document the aircraft operations in their airport. That considered, many attempts have been made to automatically record the operations using radio, acoustics, and ADS-B technologies. While these methods come with some advantages, they cannot accurately monitor aircraft operations at non-towered airports due to technological and commercial shortcomings. Therefore, this project attempts to develop intelligent cameras for performing operation count and recognition by using machine learning techniques. The proposed method addresses the issues that existed with the other methods. A camera system is a passive system that does not require cooperative aircraft. Moreover, our data analysis has shown that the proposed vision-based system holds the promise of an accurate system regarding both counting and recognizing the aircraft operations at non-towered airports during busy traffic times in those airports.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

There are many non-towered airports in the United States. These airports are important for the community since they provide vital services such as aerial firefighting, aeromedical flights, and law enforcement, besides corporate flights. In order to plan for the future development of these airports, aviation planners need the number of aircraft operations as well as the fleet mix information at these non-towered airports. Based on the Federal Aviation Administration, aircraft operations are classified into departure operations, landing operations, and touch-and-go operations. The touch-and-go operations are actually a more common type of operation at non-towered general aviation airports compared with major airports hosting airliners. As a result, having the total number of operations at non-towered airports is crucial for making managerial decisions at the airport and state levels. Continuous airport maintenance is the only way to keep these transportation corridors operable for the community. However, the associated national budgets should be fairly distributed, considering the airport operation volume throughout the year.

This paper reports the conducted research for developing an accurate approach for tallying airport-level aircraft operations using machine vision. The existing systems are comprised of automated acoustical counters, Automatic Dependent Surveillance–Broadcast (ADS-B)-based counters (a.k.a. transponder-based counters), radio click counters (a.k.a. general audio recording device), and security trail cameras. Nonetheless, the above-mentioned systems are not able to accurately collect the operational data at non-towered airports. ADS-B needs equipped aviation fleet with transponders. Acoustical counters cannot count landing operations since the aircraft sound is not noticeable during landings. The radio system also may lead to inaccurate counts. On the other hand, a passive camera-based counting system is proposed to overcome the challenges that exist with the use of these systems.

In order to develop a machine vision system for aircraft operation monitoring, we explored different camera layouts and various machine learning algorithms. Two camera layouts are proposed for full coverage of the aircraft operations at airports. The proposed camera layouts are determined with consideration of the possible airport configurations. The necessary algorithms to empower the camera footage with machine vision include aircraft detection, aircraft tracking, aircraft tail number detection, and aircraft tail number recognition. We used machine learning and deep learning methods to build the intelligent detection models to automatically perform the abovementioned tasks.

The feasibility of the developed system is tested on video data collected from general aviation airports within the state of Utah. Our team took the necessary safety measures while collecting the aircraft operational data using digital cameras. The collected data include different weather conditions (sunny, cloudy, and snowy) and different illumination conditions. The selected test locations allowed us to have a wide range of civil aviation fleets. Also, different airport configurations (i.e., runway and taxiway arrangement) are considered when the test locations are selected. The evaluation of the proposed system has shown that a vision-based system can provide accurate airport operational data. The proposed system is a good alternative to the existing systems and also provides aviation planners with a complementary tool for increasing the accuracy of the existing deployed counters in their studying airport.

1. INTRODUCTION

1.1 Introduction

Domestic airport operation volume and fleet mix information are essential for the preparation of airport master plans and appropriate allocation of state and federal budgets [1]. Furthermore, this information helps researchers in environmental studies regarding the noise emission and pollution related to aircraft operations [2], [3].

Awareness of airport operations data is a critical managerial tool used by planners in different sectors for a wide range of applications. The following briefly discusses how different entities use the information on airport-level operations counts. We divided applications of these counts into three groups of entities: public sector, private sector, and the airport itself, which could be both.

Public: State governments and the FAA can leverage accurate airport operations counts to make sound investment decisions [3]. In fact, any funding request for a capital improvement project needs to be justified by providing reliable information about the airport's operations counts. Moreover, these data are used as a basis for assessing airport capacity needs, determining staffing requirements, and allocating budgets for airport development projects [4]. The operations counts are also crucial to environmental studies, whose findings might advise against certain airport development projects. Actually, it is the number of aircraft activities that determine whether noise and air quality studies are needed for an airport development project (the FAA Order 5050.4) [5]. Additionally, flight safety could be compromised if FAA used underestimated operations counts for infrastructure planning, such as control tower justification as well as evaluation of dangerous obstructions to flying aircraft (Figure 1.1) [6]. Lastly, operations count data are essential elements of economic impact statements that eventually show the airport's financial gain considering its costs.

Figure 1.1 Examples of the obstructions to flying aircraft

Private: Private businesses located in the vicinity of an airport are often dependent on airport operations [3]. The private businesses located on an airport or located around an airport are related to airport operations. An increase in airport operations impacts these private businesses by causing demand for on-airport services, such as maintenance services, provided by fixed base operators (FBOs). In addition, it causes greater demand for off-airport services, such as hotels and fuel distribution systems [7]. Thus, banks and private investors can use operations counts when assessing loan requests for (non)towered airport businesses.

Airport: Airport managers are interested in having accurate operations counts for many reasons, such as determining the safety levels and knowing the demand for the service they provide [8]. Lau [9] states that the former solidifies the decisions made for procuring further insurance and justifying the addition of a parallel taxiway. The latter assists airport managers in determining fueling storage requirements as well as space requirements for terminal buildings (Figure 1.2) [10]. Furthermore, the volume of the aircraft operations is a factor in measuring airport performance. Lastly, the counts will directly determine whether noise mitigation procedures should be implemented in the airport [11].

Figure 1.2 A general aviation airport layout (Image from Google Earth)

1.2 Current Gap in Study

Detailed information about airport operations is typically documented by air traffic control (ATC) towers (Figure 1.3). However, more than 97% of U.S. airports do not have control towers [12]. That said, many attempts for aircraft operation detection have been proposed, including transponder-based methods, acoustical counters, and radio-based operation counters. Nevertheless, they have limited accuracy. Also, the current aviation fleet is not completely equipped with transponders to be counted by transponder-based counters.

Figure 1.3 Air traffic control tower

On the other hand, computer vision is an alternative approach, which is shown to be an effective solution for many related engineering tasks such as infrastructure modeling [13, 14] and inspecting [15]. The aviation sector needs a comprehensive system capable of counting all-type operations (departures and landings) for various airport layouts. We address this need by developing an automatic video-based air traffic surveillance system for counting and recognizing general aviation aircraft operations, which account for the vast majority of air traffic at non-towered airports. Moreover, an aircraft can be identified via its registration number, also known as the tail number, which must be printed on the body of the aircraft in accordance with the International Civil Aviation Organization (ICAO) regulations [16]. Therefore, a computer vision-based approach represents an economical alternative that can facilitate aircraft identification for record-keeping at airports.

1.3 Machine Vision

Developing a machine vision system for airport aircraft operation monitoring demands detailed considerations. It requires an appropriate camera layout that would avoid failures in capturing the operations or overcounting them. In addition, general aviation aircraft are typically small in size, and their recognition requires robust algorithms.

Furthermore, it is important to impose a limit on computation time to enable a real-time system for two reasons. First, it helps avoid data congestion and information delay that inevitably occurs with any non-real-time system. Second, it can be used to signalize flight clearance for pilots approaching the airport and thereby reduce runway incursion risk, which is a major safety concern (e.g., more than 1,511 runway incursions were reported in the last year alone [17]).

The proposed monitoring system consists of four machine learning models in order to accurately perform the following tasks (as shown in Figure 1.4):

1. Aircraft Detection (green box in Figure 1.4)
2. Aircraft Tracking (blue arrow in Figure 1.4)
3. Aircraft Tail Number Detection (yellow box in Figure 1.4)
4. Aircraft Tail Number Recognition (red box in Figure 1.4)

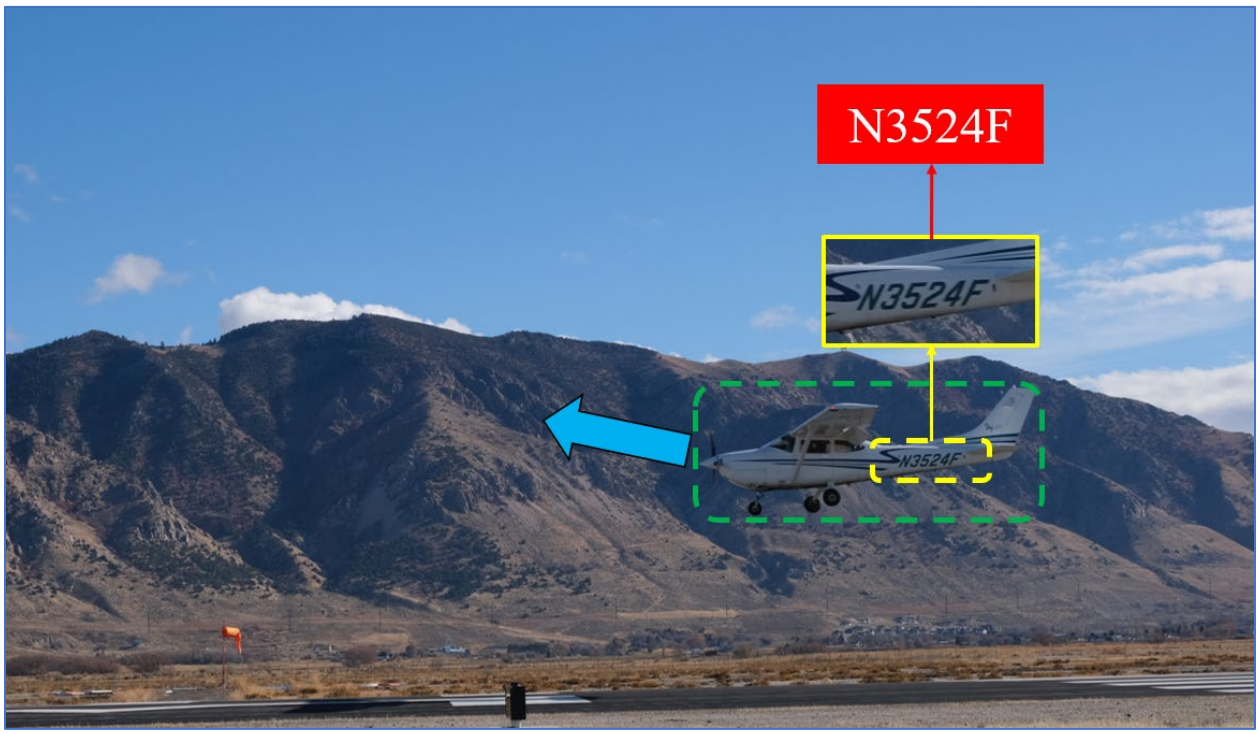


Figure 1.4 Main tasks performed by using the developed machine learning models

2. BACKGROUND

In this section, we review the literature regarding aircraft operation counting technologies and the vision-based methods for aircraft detection.

2.1 Aircraft Operation Counting Technologies

Acoustic, radio, and satellite-based methods are used to measure aircraft operations at non-towered airports. However, acoustic-based [18] and radio-based [19] (i.e., general audio recording device [GARD]) systems are incapable of recognizing the identity of the aircraft. Also, the acoustic technology can only detect take-off operations (i.e., departures) and misses the landing and touch-and-go operations. Additionally, as shown in Figure 2.1, airports with shared Unicom (radio) frequency are prone to inaccurate counts if the GARD technology is used.

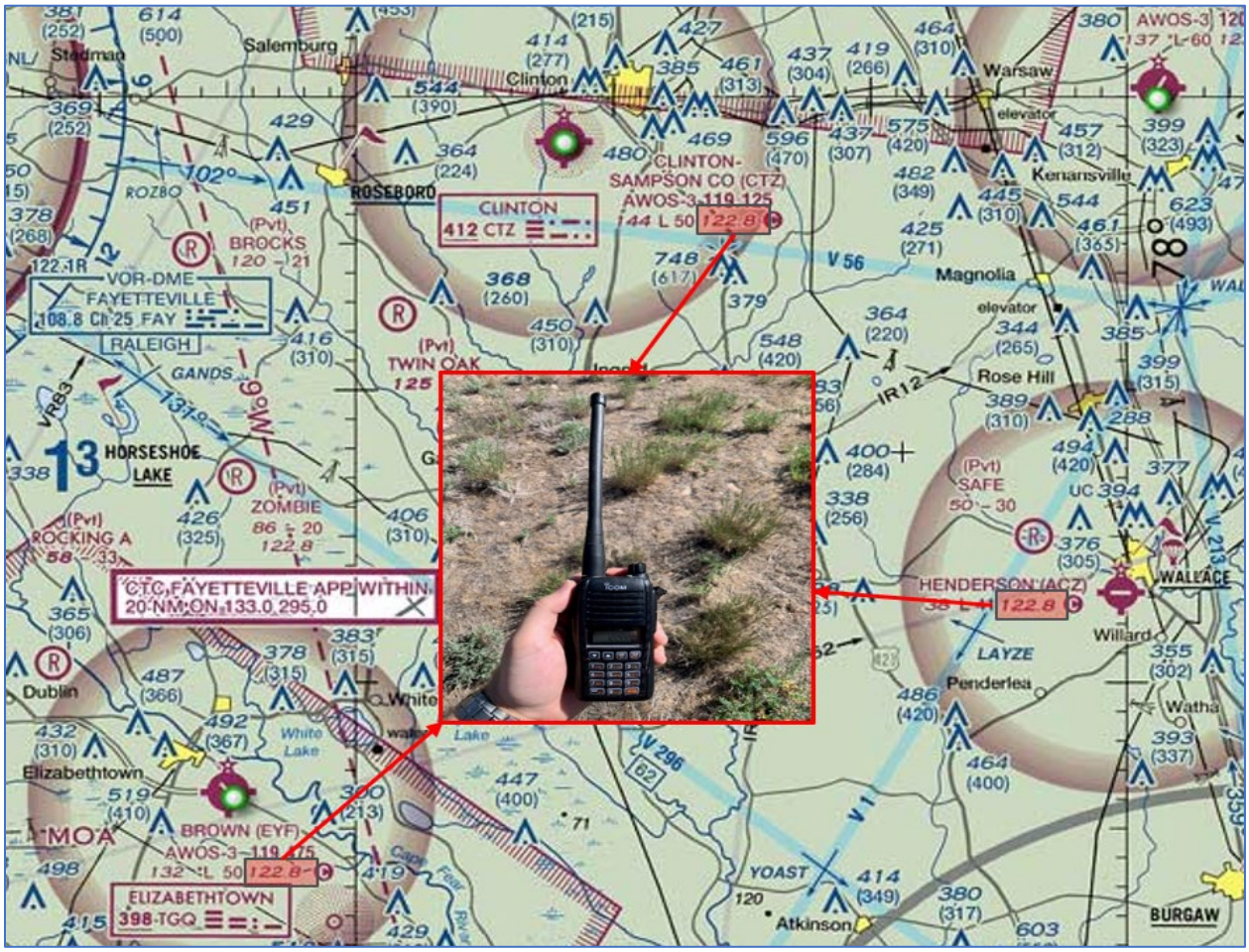


Figure 2.1 Airports with a shared Unicom frequency

Mott [20] proposed a satellite-based approach to recognize airport flight activities by encoding the signal transmitted by a transponder carried by aircraft. This method utilizes the Automatic Dependent Surveillance-Broadcast (ADS-B) system. There are several transponder types based on the signal they can transmit. Mode A/C and Mode S aircraft transponder signals are common for civilian use. Only in Mode S signals, each aircraft is assigned a fixed ICAO 24-bit address that can pair with aircraft registration in the FAA database (the identity of the aircraft) [21].

Nevertheless, this system is currently not sufficient because of the low equipage rate of the general aviation fleet with transponders (only about 75% [22]). In addition, the transponders of most of the equipped general aviation fleet (approximately 84%) cannot transmit Mode S signal that has aircraft identity information [23]. As a result, this system is not able to identify a large portion of the currently equipped aviation fleet. Besides, the dependency of this system's performance on aircraft cooperation reduces the reliability of using it. A vision-based system can properly solve this issue as it is not limited to cooperative aircraft.

Another important consideration that differentiates a vision-based aircraft detection system is that information about the airfield can be used to signalize flight clearance for pilots who approach the airport aerodrome. It would lessen runway incursion risk, which is a major safety issue (e.g., more than 1,511 runway incursions were reported in 2021 [17]). Runway incursion examples are shown in Figure 2.2.

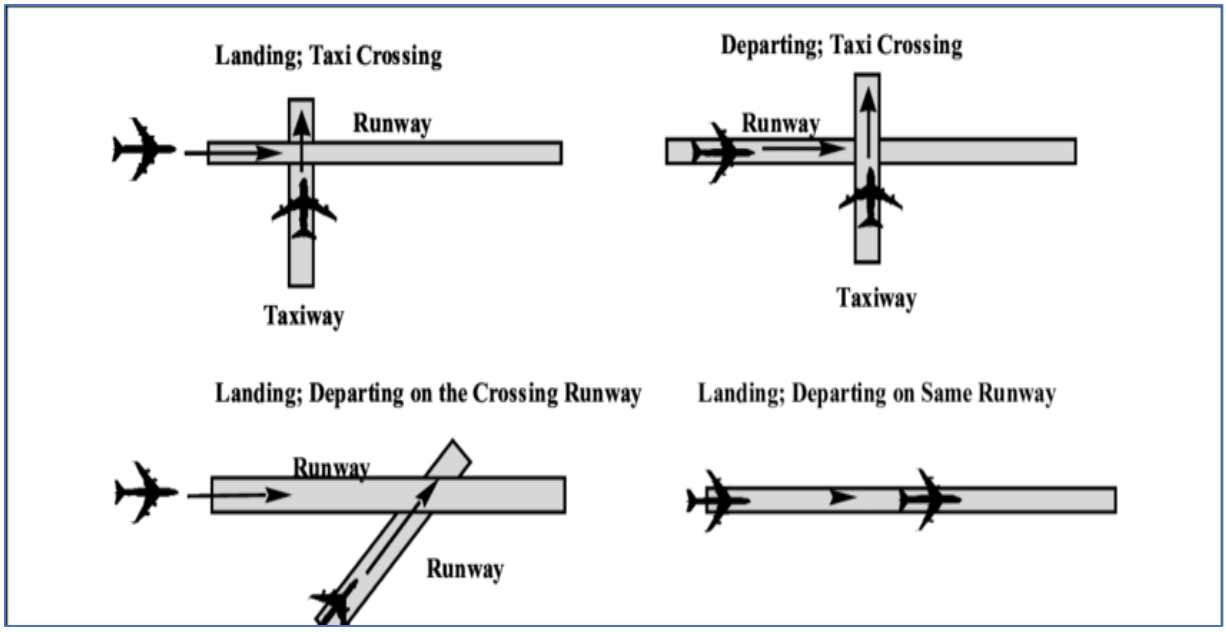


Figure 2.2 Runway incursion examples

2.2 Vision-based Aircraft Detection

The visual source of information for monitoring aprons has been typically two types of cameras: 1. pan-tilt-zoom (PTZ) and 2. closed-circuit television (CCTV) cameras that are installed in the airside area (Figure 2.3). Thirde et al. [24] developed a system with a multi-camera configuration to recognize the activities of the vehicles and personnel to monitor airplane servicing operations at the apron. Similarly, Koutsia et al. [25] use foreground subtraction and data fusion methods in order to build a network for activity tracking with PTZ and CCTV cameras using a central server. However, foreground subtraction alone is not sufficient to recognize and distinguish the activities of aircraft and service vehicles in general aviation airports because they are significantly small. A visual analysis of fueling and boarding activities can calculate the required time for the aircraft push-back process [26].

Figure 2.3 Pan-tilt-zoom camera for aircraft operation monitoring in apron area

Bloisi et al. [27] combined the theory of particle filter-based video-tracking (an algorithmic complex method) and data from radar to enhance the accuracy of the airplane positioning in the parking and apron zones. In the same way, a system named INTERVUSE [28] fuses the data retrieved from radar, cameras, and preplanned flights in a central server to cover the blind spots, thereby assisting the staff inside the air traffic control tower in spotting the operation and movements of the aircraft in airport. The sensors of these two technologies (camera and radar) work together and interchangeably to fill and cover the existing gaps of each individual technology.

Moreover, Advanced Surface Movement Guidance and Control Systems (A-SMGCS) that use cameras usually use CCTV cameras that are installed at high levels. By using these installed cameras in the airfield, Besada et al. [29] used blob analysis and foreground subtraction algorithm in order to localize and track the airplanes on the ground in the airport. Zhang et al. [30] used convolutional features calculated by deep learning methods for finding and tracking aircraft in the airport on the surface of the airfield. Thai et al. [31] improved these high (towered)-level camera systems by implementing deep neural network-based techniques for tracking the aircraft (Figure 2.4).

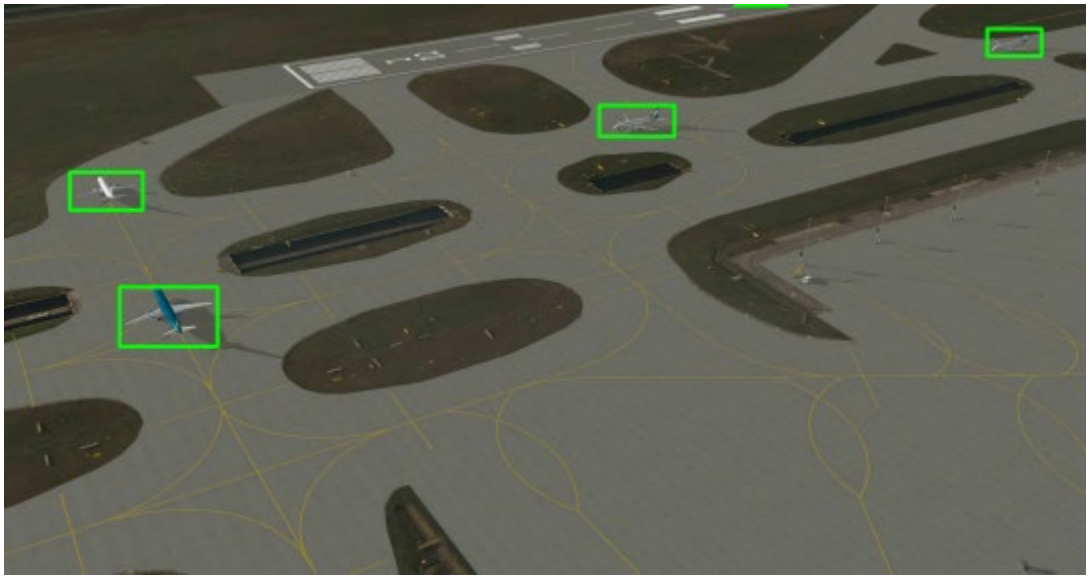


Figure 2.4 Aircraft detection from the towered level cameras [31]

Furthermore, some research was also conducted to identify tail numbers of the aircraft in the video footage from primary towered airports. Besada et. al [32] and Vidakis and Kosmopoulos [33] published technical papers to demonstrate the results of their research work developed for A-SMGCS to help staff inside the air traffic control towers detect and recognize the tail numbers of slow-moving grounded airliners in the apron or towers.

Since general aviation aircraft are smaller in size and have more challenging tail numbers compared with airliners, the aforementioned algorithms would have issues identifying the general aviation aircraft at non-towered airports. Therefore, in this project, we proposed vision-based methods for aircraft detection [34], aircraft tail number detection [35], aircraft identification [36], and aircraft operation recognition and count [37, 38].

3. METHODOLOGY

Four main tasks that are needed to be automated are:

1. Aircraft Detection
2. Aircraft Tracking
3. Aircraft Tail Number Detection
4. Aircraft Tail Number Recognition

Therefore, in this section, we explain how the necessary algorithms are constructed to perform the abovementioned tasks in an automatic fashion.

3.1 Aircraft Detection

For the task of aircraft detection, we tested both deep learning and machine learning methods. The former is tested for cases where there are several datasets available. The latter is tested for cases where there are few available datasets.

3.1.1 YOLO and SSD

YOLO and SSD are abbreviations for "You Only Look Once" and "Single Shot Detector." They are two modern deep neural network object recognition methods in images known for their raised speed of detection in comparison with their counterparts like region proposal-based networks.

In YOLO, which is a unified deep neural network for object detection, a regression solution is designed for the problem of localizing and classifying bounding boxes. Therefore, YOLO's increased detection speed results from applying only a single neural network to the input image to compute the spatial location of the object and its class probabilities. The latest version of YOLO, referred to as YOLOv4 [39], augmented new features to its previous network mainly to improve average precision while maintaining detection speed.

Despite the accuracy performance advantage over its previous versions, YOLOv4 still depends on powerful conventional GPU units for both training custom object detection models and real-time inference acceleration. To assist with YOLO's inference detection speed to be executed on embedded vision on single-board computers (SBC), Redmon et al. [40] proposed a variation of the YOLO architecture, tiny YOLO, which is much faster than the original YOLO at the cost of the lower recall.

SSD leverages multiple feature maps to enable accurate multi-scale object detection. To extract features in the deep network at multiple scales, SSD does not use the fully connected layers in its base network (i.e., VGG16) but replaces them with auxiliary networks named convolutional layers that have varying sizes. The absence of fully connected layers in the SSD network is one of the critical differences between YOLO and SSD architectures, making SSD faster than YOLO. Figure 3.1 shows the architecture of a typical object detection model that is based on deep neural networks.

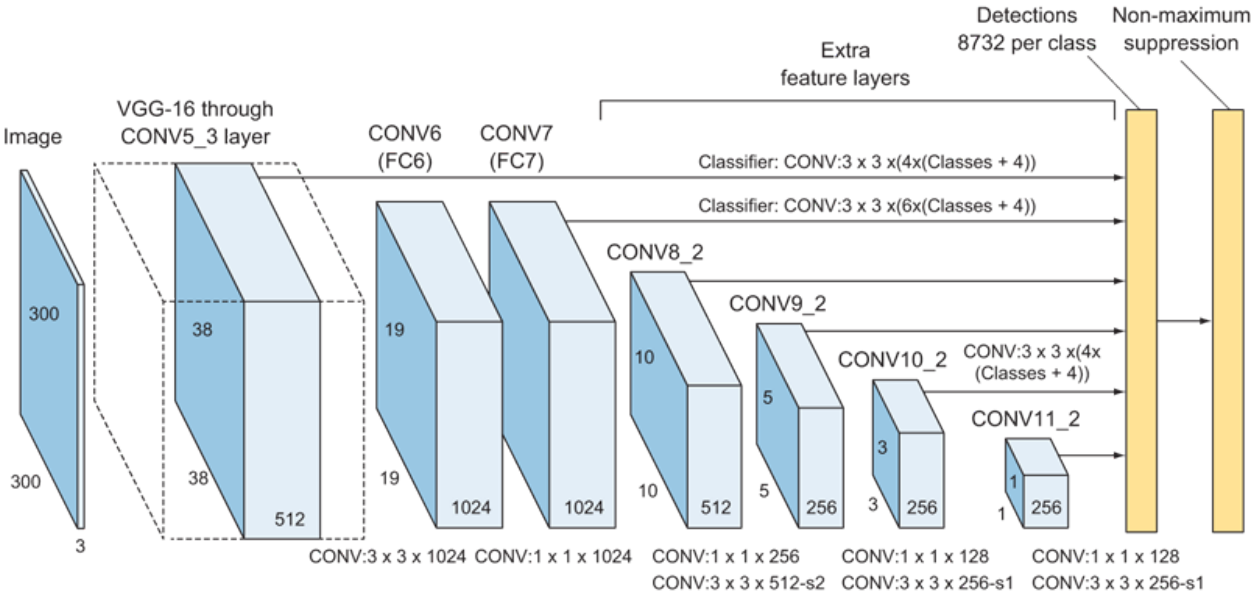


Figure 3.1 Architecture of a deep neural network for object detection

We used the three discussed deep-based object detection models to verify the Haar cascade classifier performance for the detection of the aircraft. YOLO-v4, YOLO-tiny-v4, and SSD are the three models, and their pre-trained models on the COCO dataset for the aircraft object are implemented in the Python platform using the OpenCV deep neural networks module. That said, only the airplane object, which is object category number five in the COCO dataset, is used for detection.

In order to decrease the high number of false-positive detections while maintaining a good portion of true-positive detections, a minimum of 30% was defined as their detection confidence threshold. The non-maximum suppression (NMS) function, available in OpenCV (which is an open source platform) for deep detectors, is activated with a threshold. This threshold was set at 0.5 to minimize multiple detections over a single aircraft (false positive detections) at each video frame in the program.

3.2 Aircraft Tracking

Once the aircraft is detected in our video footage, the expert system tracks the aircraft until it is out of the camera's field of view (FoV). This task is done using an object tracker. As Figure 3.2 illustrates, the detection bounding box is updated in every new video frame in order to contain the aircraft image in that video frame.

In order to lessen the computational intensity of the system, the aircraft detection task is preceded by a motion detection model and is succeeded by an object tracker. Both motion detection and object tracker models are much less computationally intensive, thereby decreasing the system processing time. During the aircraft tracking process, certain information such as aircraft tail number is extracted from the video footages. Also, the created trajectory is analyzed to determine the type of the operation (i.e., departure, landing, touch-and-go)

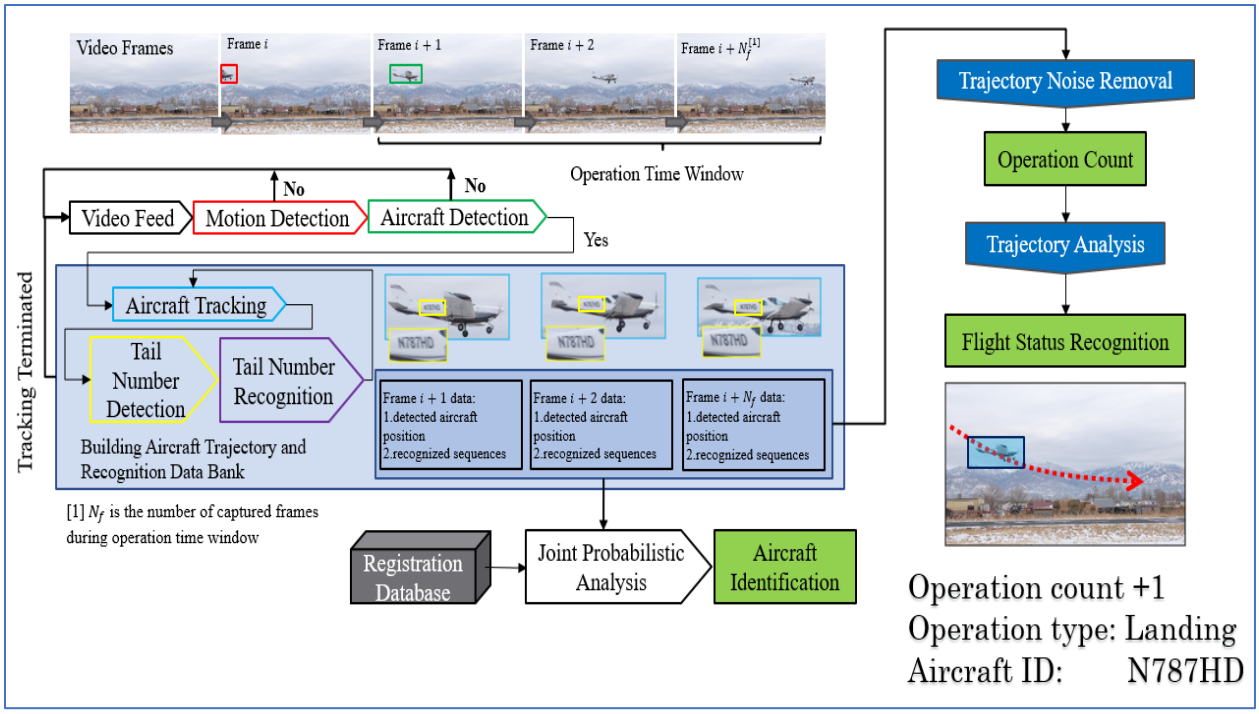


Figure 3.2 System flowchart

Correlation-based object trackers are used in this project. We did not use deep learning-based trackers because they are very computationally intense. As a result, the use of such trackers in any intelligent system would require significant computational power.

We tested several correlation-based object trackers as follows:

- __BOOSTING Tracker
- __MIL Tracker
- __KCF Tracker
- __TLD Tracker
- __MEDIANFLOW Tracker
- __MOSSE tracker
- __CSRT tracker

The performance of the first two object trackers for tracking general aviation aircraft in the recorded video footages was better than KCF and MEDIANFLOW. However, MOSSE and CSRT were the best in terms of accuracy and consistency in tracking the aircraft after it is detected via the object detection network. MOSSE stands for Minimum Output Sum of Squared Error, and CSRT stands for Channel and Spatial Reliability Tracking. MOSSE tracker is used in cases where the size of the aircraft bounding box does not change significantly during the tracking process. On the other hand, CSRT uses geometric invariant features for updating the location of the object, so it is a better candidate for cases where the object size changes significantly. Thus, we used the CSRT tracker when the bounding box is expected to change considerably.

3.3 Aircraft Tail Number Detection and Recognition

This section explains how we use the output of the general aviation aircraft detection algorithms and scene text detection algorithms to detect the aircraft tail number. We developed a camera layout for recording video footage from aircraft operations at general aviation non-towered airports. In this camera layout, two cameras are placed at each end of the runway considering the runway safety area. Each camera is pointed toward the runway ends. With this configuration and setup, the cameras record the landing aircraft off the runway surface level while getting close to the airport.

The same cameras will record the departing aircraft on the runway surface level since aircraft pilots usually go from the end of the runway to ensure a safety margin for a possible stop on the runway in case of engine failure or rejected take-off. Figure 3.3 shows a schematic display of the placement of the camera system and the cameras' FoV at one end of the runway.

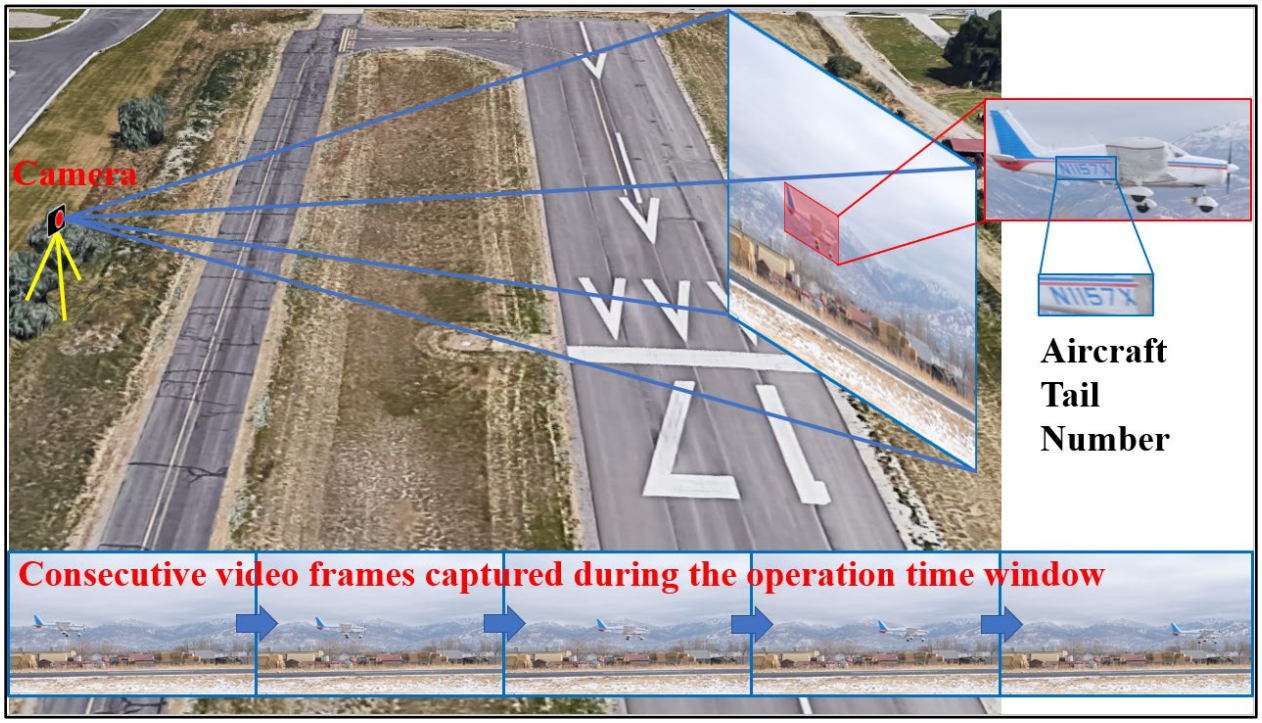


Figure 3.3 Camera system positioning at one end of the airport and its field of view

Figure 3.4 exhibits the flowchart of the system that organized the computer vision framework for aircraft operation count, aircraft activity recognition, and aircraft identification. In the first step, the system recognizes any motions by subtracting two consecutive video frames and analyzing the size of the resulted black and white blobs. These blobs could result from the motions of airport ground vehicles, airport personnel, nearby highway traffic, construction equipment, and animals (e.g., birds). Next, we utilize object detection models to detect the presence of aircraft among all possible moving objects in the video footage and very fast correlation-based object trackers, MOSSE and CSRT, to find the location of the aircraft in the rest of the video frames. The trajectory that is built by using the object tracker is further processed to count and recognize the type of aircraft activity (i.e., departure operation or landing operation).

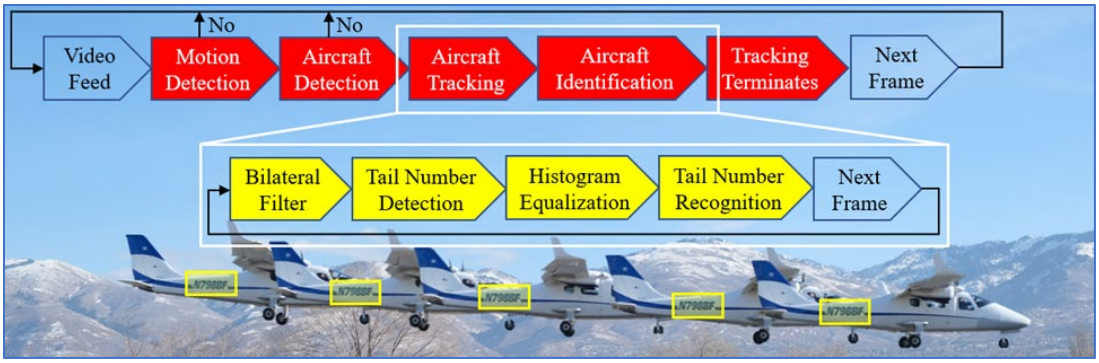


Figure 3.4 System flowchart during the flight operation time window

For detection and localization of the aircraft tail number (which is yellow-color-coded in Figure 3.4), it is best to search over the aircraft bounding box (which is red-color-coded) instead of the entire image (video frame) plane. In this way, we increment the accuracy of the tail number detection task by decreasing the size of the search space. Subsequently, the algorithms designed for scene text detection localize and extract the tail number image to be used as an input for the text recognition algorithm.

The aircraft's very high speed, especially during landing operations, can cause a blurry effect in video frames while recording the operation. That considered, in order to improve the quality of the images, we exert edge-preserving and contrast adjustment to the detected aircraft image box and the detected tail number image box.

A bilateral filter is applied to highlight the edges in the aircraft bounding box before tail number detection. A bilateral filter considers the weight of both the Euclidian distance of the image pixels and the color intensity differences by applying a Gaussian distribution while substituting the intensity value of an image pixel. The sharper edges are essential for the task of tail number detection.

Once the tail number is detected, we use histogram equalization to reinforce the contrast of the tail number characters compared with the background. Histogram equalization is an image processing technique to spread the intensity of image pixels by extending the histogram diagram and adjusting the contrast of the image. It is essential to apply the histogram equalization only to the tail number image box because it could potentially magnify the noise if devoted to the whole aircraft image box.

3.3.1 Tail Number Detection

Here we test two text detection techniques to evaluate their accuracy for applications with limited computation resources, which may be a problem in many general aviation airports, such as small local non-towered airports, general aviation airports, and backcountry airstrips. For that purpose, we propose a feature-based tail number detector and compare it with a text detector with a deep neural network-based backbone.

The feature-based detector operates based on an object detection technique named Haar cascade. As for the deep-based detector, we selected the TextBoxes algorithm developed by Liao et al. [41]. The TextBoxes model trained on the SynthText dataset is utilized for the purpose of comparison. The following illustrates how we construct the feature-based tail number detector.

In the United States, Federal Aviation Administration refers to tail numbers as the N-Number of the aircraft since registration numbers should begin with the letter "N" – which is a letter code specifically for the aviation fleet registered in the United States. Therefore, when the "N" character is found, the height of the tail number window is known to be the same size as the bounding box height of the detected "N." As Figure 3.5 indicates, the width of this window should be an extended width of the "N" bounding box.

Through experiments, it is demonstrated that extending the "N" bounding box width to four times the longer width accurately bounds the length of the actual tail number. Accordingly, we constructed a feature-based scene text detector to recognize the letter "N" in the images recorded from aircraft by means of the Haar cascade classifier technique [42].

Figure 3.5 "N" Detection

Haar Cascade Classifier: Haar cascade is known for its fast detection performance and comparable accuracy compared with deep neural networks. In this method, a rectangular window looks for the aircraft object by searching over the image plane using two mechanisms: 1) moving the window by a predefined number of pixels to fully cover the image; 2) resizing the original image for a few times (scale pyramid) to permit the multi-scale detection. In the next step, each candidate window will be classified into two classes: a positive class and a negative class, through several rejection/acceptance stages (Figure 3.6). This mechanism increases the speed of the process of detecting an object and also decreases the computational intensity because just a few primitive stages can remove the negative windows. A large number of weak classifiers build the final robust classifier at each stage, i.e.,

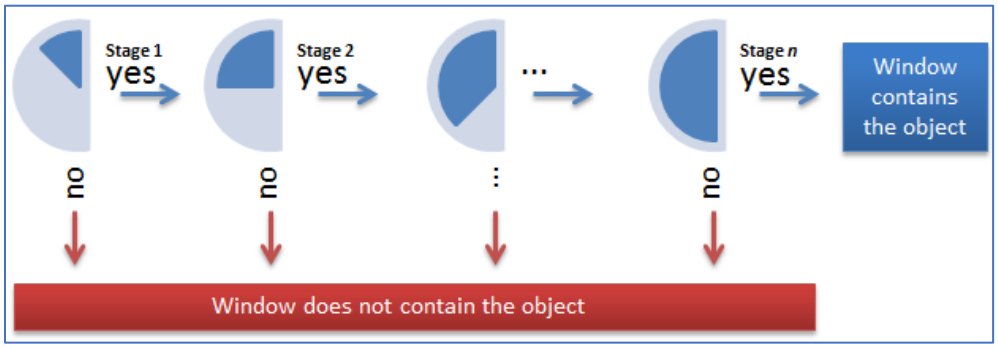


Figure 3.6 Haar cascade classifier stages

$$h(x) = \begin{cases} 1, & if \ \sum_{j=1}^{T} \alpha_j h_j(x) \geq \frac{1}{2} \sum_{j=1}^{T} \alpha_j \\ 0, & otherwise \end{cases}, \quad (1)$$

where x is a window of the original image, $h(x)$ is the final robust classifier, $h_j(x)$ is a Haar feature-based weak classifier, α_j is the weight of each classifier, and T is the total number of weak classifiers in the observed stage. Note that the final robust classifier of the resulting stage specifies if the window can encompass the object of interest or not. Windows that pass all stage criteria are known to encompass the object of interest (positive class). In this scene text detection problem, we define an image of the character of "N" as the positive class and any other image window as the negative class.

Haar Feature-based Weak Classifiers: A Haar feature and a threshold construct a Haar feature-based weak classifier. As Figure 3.7 demonstrates, Haar features are rectangular image filters with black and white small (sub-)rectangles adjacent to each other. The convolution operation of a Haar feature to the image window compared to a predetermined threshold indicates the value of $h_j(x)$ in the final robust classifier formula.

$$h_j(x) = \begin{cases} 1, & if \ p_j f_j(x) < p_j \theta_j \\ 0, & otherwise \end{cases}, \quad (2)$$

where p_j is the polarity coefficient, f_j a Haar-like feature, and θ_j is the predetermined threshold. The value of 1 means that the image subwindow contains the object based on the Haar feature-based weak classifier (i.e., $h_j(x)$). These classifiers are known as weak classifiers because a single classifier cannot accurately detect the associated subwindow.

Based on a theory, the abovementioned convolution operation is simply the subtraction of the value of the pixels that coincide with the Haar feature black sub-rectangles from the value of the image pixels overlapped with the Haar feature white sub-rectangles. Integral image representation formulated by Viola and Jones (31) enhances the computation efficiency of the Haar cascade approach even more by easing the Haar feature and image subwindow convolution operation.

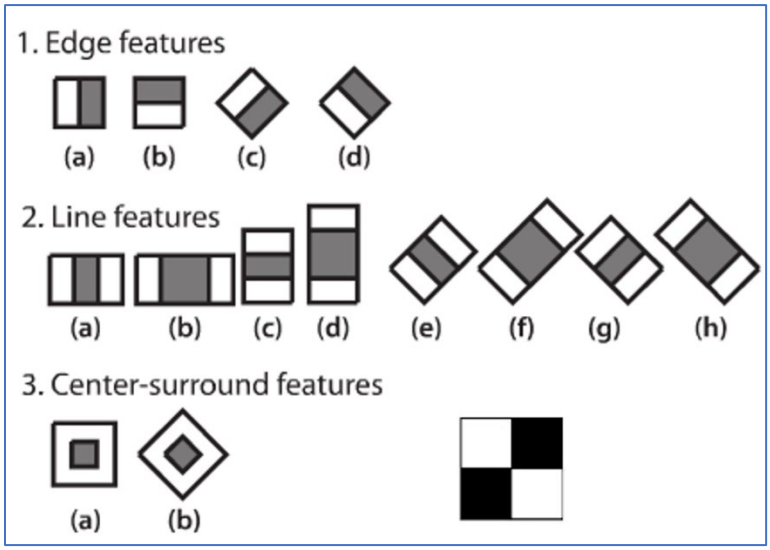


Figure 3.7 Haar-like features

The optimal distribution of the weak classifiers is important to building the ultimate robust classifiers at each stage. For instance, a 24x24 image size has 162,336 (upright) Haar-like features, more accurately 86,400 edge Haar-like features, 55,200 line Haar-like features, and 20,736 four-rectangle Haar-like features.

Training the Classifiers: An Adaboost algorithm is leveraged to find the best selection and value of the Haar cascade classifier parameters explained in the last two subsections. The aim is to find the best classification, considering precision and recall by raising the hit rate (i.e., true positive detection rate) and reducing the false detection rate at each stage. While a minimum hit rate of 0.999 is suggested by Lienhart et al. [43], we selected 0.997 as any higher value would significantly lower the speed of the training process. As for the maximum false alarm rate, a value of 0.4 is known to be efficient through experiment.

The training process is conducted in the OpenCV object detection module using Python. Regarding the hyperparameters related to the degenerate trees of Haar-like features (the weak classifiers), we utilized a fixed weight trim rate (0.95), maximal weak tree depth of 1, and maximal weak trees per stage of 100, which are the predefined values determined by the computer trainer application. We used a Gentle Adaboost to define the final form of the degenerate trees of Haar feature-based weak classifiers. Gentle Adaboost typically boosts the performance of the generalization in algorithms. Lastly, a grid search is performed over three maximum number of stages (i.e., 10, 15, and 20) and two sample window sizes (i.e., 20x20 and 24x24).

There are 248 positive image samples and 415 negative image samples leveraged as the data for classifier training, which is downloaded from random images extracted from the web. A validation dataset of 10 aircraft operations video footage with 100 digital frames verified the use of the model with a 24x24 window sizes for higher accuracy of “N” (or aircraft registration/tail number) detection rate. The optimum model continued to be trained for 15 stages and terminated the training process after getting close to the overall false alarm rate criteria, which is specified by max_false_alarm_rate and number_of_stages. This model is also examined with its deep neural network counterpart (named TextBoxes), which is further tested in the next section.

Implementation: Haar cascade classifiers, although efficient, often result in moderate to high false-positive object detection rates. Because of that, we extracted those false detections that possess fewer overlapping detected bounding boxes (i.e., neighbor bounding boxes) than an associated threshold (i.e., minimum neighbors number). Nevertheless, the experiments demonstrated that there would still occasionally be several detections over a single object of interest (i.e., aircraft registration/tail number). Because of that matter, the detections are further repressed by an algorithm named refinement filter, illustrated in Figure 3.8.

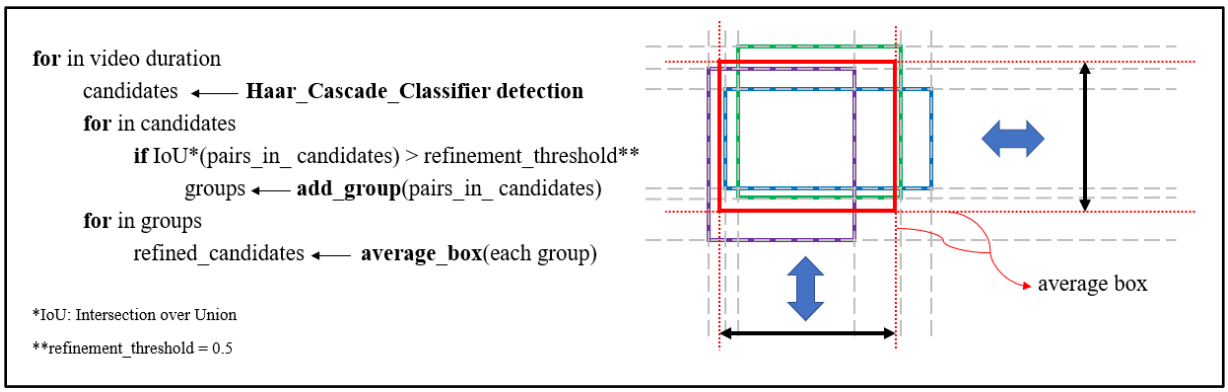


Figure 3.8 The refinement filter algorithm

Consequently, the best detections are kept. The CRNN algorithm will have the detected image of the aircraft tail number as the network input and digitalize it to an editable format. The product can extract aircraft information, such as the aircraft type, engine model, manufacturing year, and weight, from the published FAA registration databases.

3.3.2 Tail Number Recognition

We employ a convolutional recurrent neural network (CRNN) [44], a modern deep learning algorithm known for its accuracy and speed. This model combines a deep convolutional neural network and a recurrent neural network into an end-to-end character recognition neural network model. There is no constraint for the length of the tail-number-like object that the CRNN model can recognize. This feature is beneficial for the task of tail number recognition because the length of the tail number can change from three letters to six letters (according to the ICAO grammar for tail numbers).

The feature maps in the convolutional layer of the CRNN network relate each receptive field (i.e., sliding bounding box in the input image demonstrated in Figure 3.9) to a feature sequence, which will be later classified into one of the predefined labels (i.e., uppercase English alphabets, numbers, and a ‘blank’). The used overlapping sliding window (Figure 3.9), instead of connected component-based techniques (e.g., Tesseract OCR library), is primarily helpful for our problem with tail numbers that are haphazardly arranged (tail numbers that have overlapping characters). Lastly, a transcription layer implements the connectionist temporal classification (CTC) prediction module explained in [45] to eliminate the repeated labels over single characters and the outputs like the ‘blank’ windows’ outputs. Consequently, the output of the CRNN optical character recognition model would be sensitive and need the accurate localization of the target sequence (i.e., aircraft registration number/tail number) in the input image.

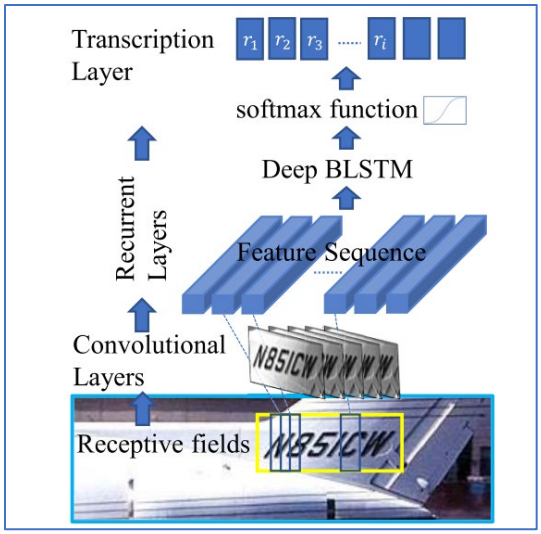


Figure 3.9 Each vector in the feature sequence of the CRNN network is associated with a receptive field. A BLSTM (bidirectional long short-term memory) network predicts a label distribution for each vector (receptive field).

In the previous section, we explained how we detected the input for the aircraft tail number recognition module from the video footage data from operating aircraft in an airport environment. It is noteworthy that, in this project, we used a CRNN model constructed by leveraging two popular text recognition datasets named MJSynth and SynthText. These datasets have synthetic images that together enjoy 14.4 M word box samples. MJSynth and SynthText are both suitable text recognition datasets for tail number recognition since they have the needed diversity in their fonts, borders, shadows, background colorings, and projective distortions processed when building these datasets [46].

4. DATA COLLECTIONS

Several data collections have been conducted to find the optimal setting for video capturing configuration as well as the localization of the cameras in the two configured layouts in the previous section. Furthermore, the collected data are utilized to evaluate the accuracy and feasibility of the system. The test locations that we visited are five public-use general aviation airports in Utah:

1. Bountiful Airport
2. Brigham City Municipal Airport
3. Spanish Fork Airport
4. Heber City Airport
5. Logan-Cache Airport

The following figures demonstrate the general aviation airports' test locations and layouts. The order of these follow the above list.

Bountiful Airport

Brigham City Municipal Airport

Spanish Fork Airport

Heber City Airport Logan-Cache Airport

Figure 4.1 Test location airports in Utah (Images from Google Earth)

Spanish Fork Airport and Brigham City Municipal Airport have a more mixed aviation fleet than the other airports. A wide range of aircraft types, from general aviation aircraft to large airplanes, are observed during data collection. This fact produces another challenge for detection of the aircraft operation because the broader range of aircraft leads to a more comprehensive range of aircraft operation speed, acceleration, and landing approach. This dataset is useful for us to customize the software for such general aviation airports.

The selection of the Logan-Cache Airport was originally intended to assess the system performance for more complex taxiway-runway arrangement cases. As the plots demonstrate, the operations at this airport include three runway lanes with the associated taxiway and taxiway-runway passages. The strategic passages that cover the majority of the activities in these airports can be detected and examined for the best placement of the cameras.

As Figure 4.1 demonstrates, a fine combination of runway sizes can be observed in the test locations. The size of the airports influences the mix of the operating aircraft in the airport as well. That considered, a very good mix regarding aircraft types has been recorded in the conducted data collection sessions. For instance, Bountiful Airport enjoys high general aviation aircraft traffic. The challenges that existed with general aviation aircraft detection and recognition relate to their small sizes and irregular tail number shapes in comparison with large airplanes.

The most jet traffic among our test locations can be seen at Heber Valley Airport. This type of aircraft tends to fly (depart or land) at a much higher speed and acceleration in comparison with the other types of aircraft. The higher operation speed raises the possibility of a blurry effect in the captured video frames. As a result, a higher rate of video data capturing might be crucial for accurate aircraft operation recognition at the abovementioned airports.

4.1 Camera Layout Plans

After carefully observing the various airport layout plans and interacting with local airport authorities and operators, two possible camera layouts for small general aviation airports are proposed. Each of these layouts has a different data recording configuration, setup, and requirements. A video-based air traffic surveillance system aims to capture all flight operations in an airport. Aircraft pilots operate from the end of the runway to have a safety margin for a stop on the runway in case of an engine failure/rejected take-off.

Therefore, two ends of the airport runway are determined as strategic locations for camera placements in layout 1 to record aircraft operations (Figure 4.2, top). As explained and as shown in Figure 5 (bottom right), departure operations, which begin from either end A or end B, are recorded on the level of the ground in the demonstrated field of view (FoV). Accordingly, the arrival operations are captured while the aircraft lands and is off the ground level (Figure 4.2, bottom left).

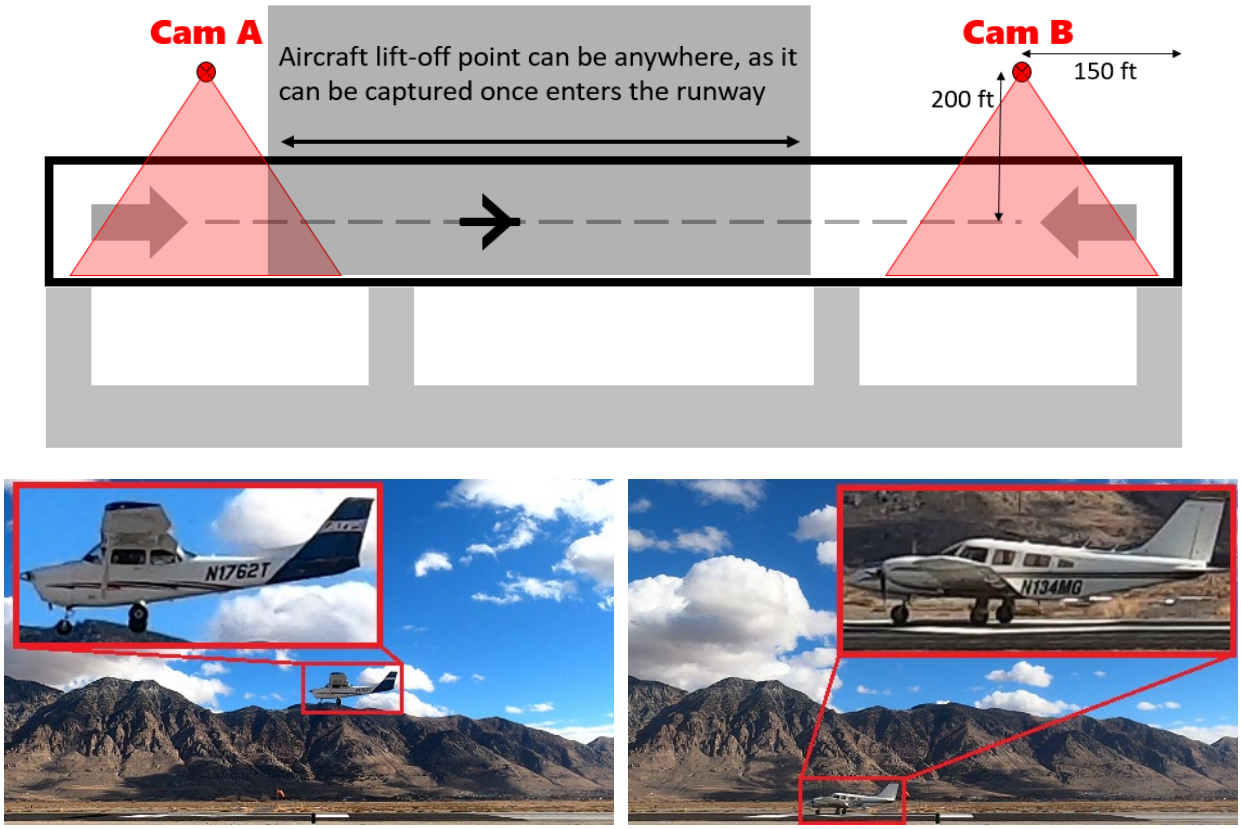


Figure 4.2 Top: Camera deployment in layout 1; Bottom: Field of view in Camera A for an arrival operation (left) and a departure operation (right) on the runway area

While camera layout 1 can have all aircraft operations, consisting of touch-and-go operations, some aircraft operations might not be visually identifiable in the cameras' FoV (Figure 4.3: Difficult to read aircraft tail number for identification purposes). That considered, layout 2 is devised for cases where higher recognition accuracy is required. Figure 4.4 demonstrates the camera placements and orientations in the airport for layout 2.

Figure 4.3 View of a landing aircraft with a difficult-to-read tail number in layout 1 field of view

Since all operations require a passage over connectors, camera layout 2 FoVs have a view of flight operations (either departure or arrival) on the ground level and can distinguish them based on the aircraft motion direction in the respective connector.

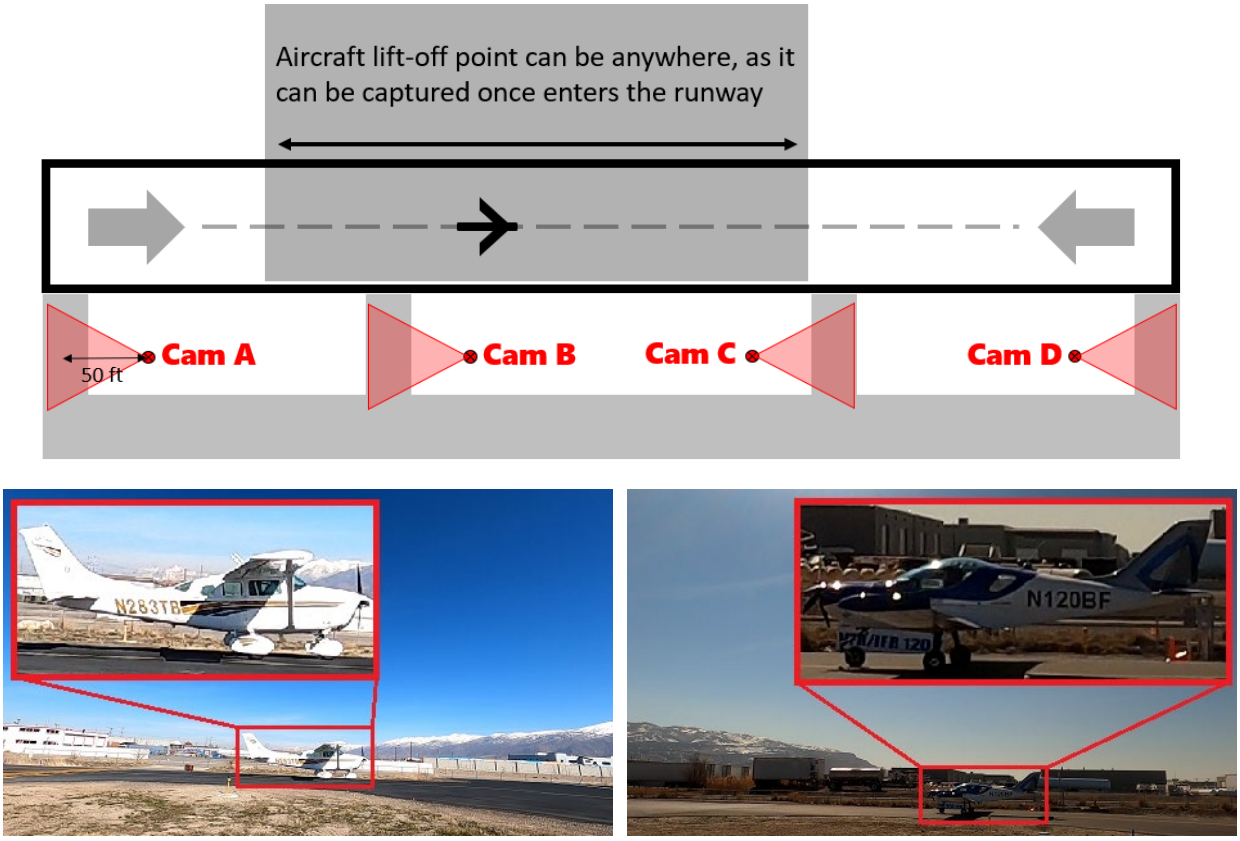


Figure 4.4 Top: Camera deployment in layout 2; Bottom: Field of view in Camera A for a departure operation (left) and an arrival operation (right) on the taxiway-runway connector area

Camera layout 2 should be used for finding touch-and-go activities' occurrences as a spinoff use. To do so, the tallied arrival operations at connector passages are subtracted from all landing aircraft operations that are seen at the end connectors (Cam A and D in Figure 4.4). Figure 8 demonstrates the Cam A FoV while observing a landing operation. Table 4.1 shows air traffic visual data details provided by each camera layout separately.

Table 4.1 Visual data field provided by camera layout 1 and 2

Visual data set field	Layout 1	Layout 2
Able to count all flight operations	✓	✓
Able to distinguish departure operations from landing operations	✓	✓
Able to count touch-and-go activities	✓	✓
Able to capture aircraft tail number for departure and arrival operations	✓	✓
Able to capture aircraft tail number for touch-and-go activities	✓	✗
Able to distinguish touch-and-go activities from non-touch-and-go arrival operations	✗	✓

4.2 Experimental Setup in Test Locations

In this project, the dataset was collected by using low-priced commercial off-the-shelf cameras (GoPro Hero 8, GoPro Hero 3, and Fuji Film XT-30). These cameras were mounted on steel tripods (Figure 4.5). Several video capturing resolutions were tested: 4K linear, 4K wide, 2.7K (1.4x), 1080 (HD) (2x), XP+, and 1,760x3,120 (2x). According to the results, 1,080 video resolution with 30 frames per second and 2x zoom selected is optimum because it ensures enough image pixels and number of frames in an aircraft operation time window for aircraft operation detection and aircraft operation count.

Figure 4.5 Data collection setup

Nonetheless, the rise in video resolution would assist us in the task of aircraft identification through the recognition of the aircraft tail number. The system evaluation section demonstrates the level that the increased video resolution influences the results of the aircraft identification in both camera layouts separately.

We performed several data collection sessions at the discussed airport test locations to include a wide variety of video data in our datasets, including all possible variations. These variations are not limited to different weather/illumination conditions, other moving objects in the scene, different airport layouts, or different aircraft types.

During the project time, the project lead visited all five test locations and conducted multiple data collection sessions at each general aviation airport. With the assistance of the student team and the airport operators, the project lead conducted 29 data collection sessions in all four seasons of the year. The following pictures show a few video frames of the camera FoVs as well as the equipment configuration and camera setup at some of the data collection sessions.

Skypark 09-25-2020

Skypark 10-14-2020

Brigham City Municipal Airport 10-28-2020

Brigham City Municipal Airport 11-18-2020

Brigham City Municipal Airport 11-19-2020

Skypark 12-10-2020


Skypark 12-15-2020


Spanish Fork Airport 12-23-2020


Spanish Fork Airport 02-19-2021

Heber Valley Airport 06-03-2021

Heber Valley Airport 06-04-2021

Logan-Cache Airport 08-10-2021

Figure 4.6 Sample screenshots from cameras' fields of view during data collection sessions

In some cases, the small sizes of the aircraft tail numbers are even difficult for human eyes to recognize. Figure 4.6 demonstrates the various aircraft types and the different shapes of the associated tail numbers printed on the aircraft fuselage. The direct sunlight reflected from the aircraft body is another instance of a difficult-to-read aircraft tail number.

The reflected sunlight from the aircraft body reduces the tail number reading possibility. Therefore, a polarizer filter can reduce the effect of solar radiation on the camera lens. In cases where the sun shines directly into the camera lens, an IR cut lens can lessen the effect of the input light into the camera lens.

All data collection sessions are spread throughout different seasons of the year for a better generalization of the research product. From the beginning of the project, the project leader gathered the required data with a crew from Dr. Rashidi and Dr. Markovic's lab. After performing each data collection, the video data are meticulously evaluated and processed with the developed algorithms (software).

We performed several trials and errors in order to build the most current software platform and camera layout configuration. Among them are the video configurations and camera specifications, camera positions at the airports, and software debugging. The next section examines the system using the collected data from the assigned test locations (airports).

5. RESULTS

In this section, we evaluate the designed system by using standard metrics. First, the accuracy of the camera layouts for capturing the aircraft operations is assessed. Next, we report the accuracy of the computer vision algorithms, including aircraft operation detection and tail number identification. This section also evaluates the accuracy of the camera layouts in capturing different aircraft operations.

5.1 System Performance

5.1.1 Aircraft Detection

As illustrated in Table 5.1, the performance of the algorithm is tabulated for general aviation aircraft detection for each camera layout and flight operation status separately. Despite its rapid processing time with 42 FPS, tiny YOLO was not able to provide sufficient true positive detections in departure operations at both camera layouts and was removed from Table 5.1 for the rest of the algorithm analysis. The recall observed when using tiny YOLO is zero and 0.2 for departure operations at layout 1 and layout 2, respectively. The lower value of the recalls (Table 5.1) for detecting aircraft in departure operations versus arrival operations is because of the complex background of the scene and occasionally partial occlusions of aircraft with long bushes in the runway shoulder area.

Table 5.1 Aircraft detection accuracy

Operation	**Method**	**Camera Layout 1**				**Camera Layout 2**			
		#frames	**FPS**	**precision**	**recall**	**#frames**	**FPS**	**precision**	**recall**
Arrival Flights	Haar	1352	33	0.88	0.71	493	29	0.79	0.58
	ref Haar	1352	32	0.98	0.71	493	29	0.82	0.57
	SSD	1352	26	0.84	0.60	493	26	0.98	0.88
	YOLOv4	1352	4	0.97	0.89	493	4	1.00	0.96
Departure Flights	Haar	2451	31	0.88	0.38	1948	30	0.69	0.55
	ref Haar	2451	31	0.89	0.38	1948	29	0.85	0.45
	SSD	2451	27	0.57	0.16	1948	25	0.99	0.97
	YOLOv4	2451	4	0.99	0.73	1948	3	1.00	0.96

In both camera layout 1 and camera layout 2, YOLOv4 had a better performance compared with Haar cascade and SSD (Table 5.1) regarding the precision and recall metrics. Figure 5.1 illustrates higher robustness of YOLOv4 in detection at various quartiles as well. Nevertheless, this algorithm falls behind its counterparts because the YOLOv4 processing rate is 3-4 FPS.

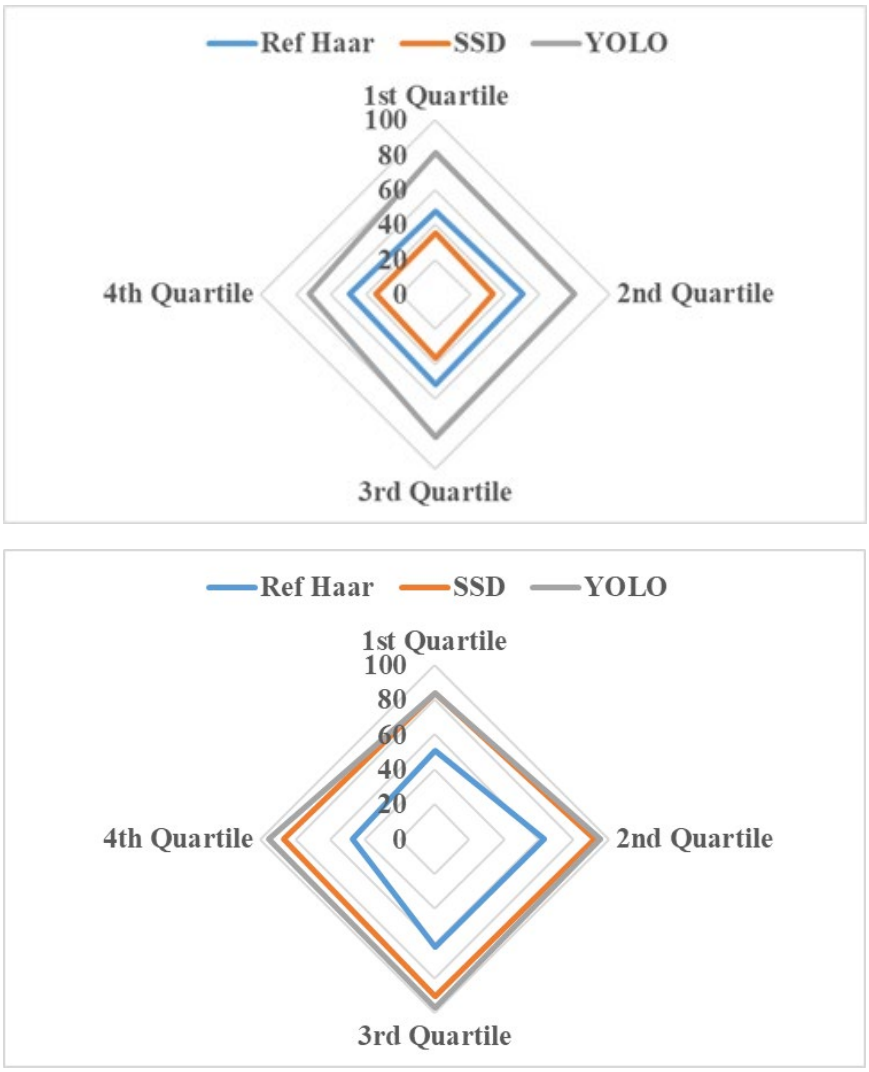


Figure 5.1 Average true positive detection rate per quartiles in layout 1 (left) and layout 2 (right)

Among those detection algorithms that maintained a real-time processing time, Haar cascade outperformed the SSD in detection results for the first camera layout. As for 3,803 frames (extracted from 59 flight operation time windows) examined for camera layout 1 and tabulated in Table 5.1, the Haar cascade algorithm attained 4-7 more FPS and higher precision and recall values compared with SSD. This increase in performance can be seen especially for departure operations, which are more difficult to detect. The developed refinement algorithm enhanced the Haar cascade classifier's precision from 0.88 to 0.98 for detections in arrival operations while maintaining the recall value. Figure 5.1 illustrates the average true positive detection values for all methods for each operation separately. Figure 5.1 shows the considerably higher true positive detection performance attained by the refined Haar cascade classifier compared with SSD for the entire flight operation's time window in the first camera layout, with an almost 50% true positive detection performance at all of the operation time window quartiles.

In the analysis of the camera layout 2, SSD is shown to be a more robust algorithm while considering the real-time processing factor. As tabulated in Table 5.1 for camera layout 2, it is true that the refined version of the Haar cascade classifier has an acceptable performance in detection considering both accuracy matrixes, recall, and precision. Nonetheless, SSD offers higher values of recall and precision for both of the departure operations and landing operations with 25 FPS processing time.

As illustrated in Figure 5.1 right, the refined version of the Haar cascade maintained a moderate true positive detection rate and detected the general aviation aircraft for more than 50% of the digital video frames during each operation; however, SSD excelled at this camera layout for better true positive detection rates. The SSD's better performance resulted from a closer FoV in layout 2 for departure operations compared with the other layout.

5.1.2 Operation Count and Classification

Each camera layout was able to independently perform an acceptable operation recording because of their sufficient coverage of the scene from the determined location at the airfield. Each camera layout alone covered both departure and arrival operations during the video surveillance. The collected data have flight operations from different types of general aviation aircraft and various weather conditions (sunny, overcast, rainy, and snowy). Table 5.2 illustrates the mix of the flight operation captured by the two camera layouts during the data collection sessions. During the observation time (in-field data collection), the observed operations by the assigned student's crew totaled 288 and 91 for camera layout 1 and camera layout 2, respectively (Table 5.2).

Table 5.2 Camera layout operation mix during observation in data collection time

	Camera Layout 1			**Camera Layout 2**		
	# Observed Operations	**# Captured by Cameras**	**# Detected by Software**	**# Observed Operations**	**# Captured by Cameras**	**# Detected by Software**
Departure	79	79	76	52	52	52
Landing*	209	207	201+1**	39	39	38
Total	288	286	277	91	91	90

It is shown in **Table 5.3** that camera layout 1 offers camera FoV that captured 99.3% of operations (i.e., accuracy regarding FoV selection) in the airport environment. This accuracy is for capturing flight operations (i.e., departure operations and landing operations, including arrivals and touch-and-goes). Furthermore, the software detected 96.9% of the recorded flight operations in the video footage in layout 1. In a similar fashion, the camera layout 2 system had 100% accuracy for camera FoV selection and 95.8% accuracy for operation detection via our vision-based software. **Figure 5.2** and **Figure 5.3** display the AIVATS software output displayed on the video footage screen (layouts 1 and 2) for further illustration.

Table 5.3 Accuracy of the operation count task during observation

System	**Accuracy for Layout 1**	**Accuracy for Layout 2**
Camera FoV selection	99.3%	100%
Software (automatic vision based)	96.9%	95.8%
Camera + Software	96.2%	95.8%

Table 5.3 illustrates that the AIVATS software performance is very high for the two operations' count and operation-status distinguishing tasks. From 377 operations captured by cameras in airports, only 10 false-negative detections and one misclassification (i.e., false-positive) have resulted from using the software application.

Figure 5.2 AIVATS software performance during (top image) and right after (bottom image) the aircraft operation in layout 1

Figure 5.3 AIVATS software performance during (left image) and right after (right image) the aircraft operation in layout 2

5.1.3 Tail Number Detection and Recognition

Both of the aircraft tail number detection algorithms showed comparable accuracies for finding the bounding box of the aircraft tail number in the video footage. It should be noted that the Haar Cascade classifier could detect the bounding boxes at a much faster speed by using its rejection/approval stages for finding the object of interest. On the other hand, TextBoxes had a minor superiority regarding the accuracy of the bounding box localization compared with the ground truths. As **Figure 5.4** shows, the accuracy of both algorithms is high considering their average precision and recall value for video frames captured per operation time window.

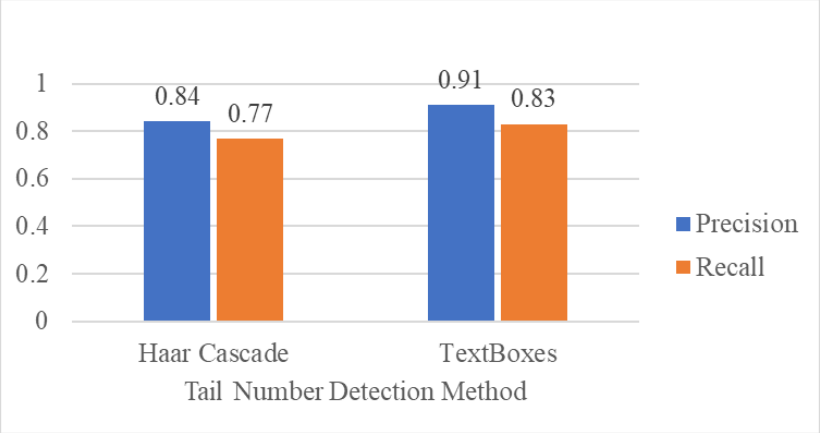


Figure 5.4 Tail number detection (localization) performance metrics. The blue and orange bars show the average precision and recall value of the tail number detection methods for video frames captured per operation time window.

Figure 5.5 shows the average of the mean, minimum, and maximum OCR IoU score for the recognized characters of the aircraft tail numbers per operation using the found bounding boxes by means of Haar cascade and TextBoxe algorithms.

The following figure demonstrates that the TextBoxes algorithm results in slightly more aligned bounding boxes to the spatial location of the aircraft tail numbers in the digital video frames. It is shown there is an 8% difference between the mean OCR IoU scores of the two tail number detection methods. Also, the error bar of the Haar cascade model is extended more than the error bar of the TextBoxes. Nevertheless, the considerably larger dataset (i.e., SynthText) used for training purposes for the TextBoxes network is one of the main reasons for this difference in OCR IoU score.

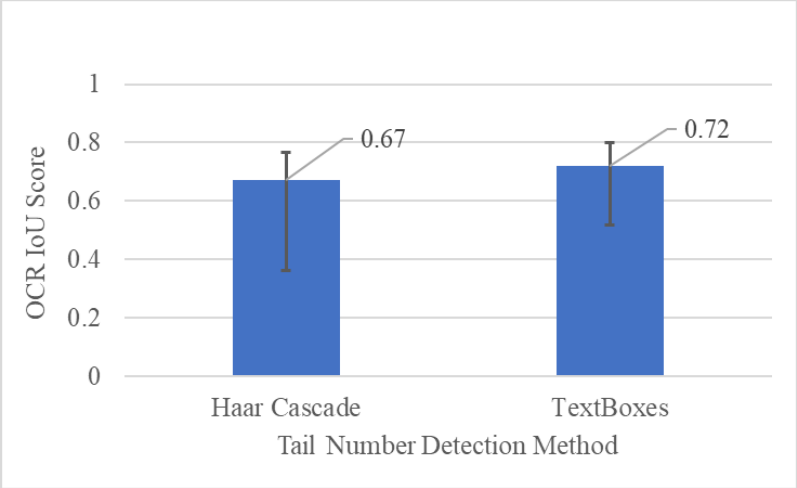


Figure 5.5 OCR performance of the CRNN over the detected bounding boxes detected by the two tail number detection methods. The blue bars indicate the average of the mean OCR IoU score of the recognized tail numbers per aircraft operation. The lower end and upper end of the black vertical error bars show the average of the minimum and maximum OCR IoU score per aircraft operation, respectively.

Table 5.4 summarizes the two layout accuracies for the aircraft identification task. While the configured software identified 64% of the total number of aircraft operations collected from layout 1 FoV, only about 14% of the identification errors resulted from the software. Despite having a moderate accuracy difference for the task of operation count and operation status recognition compared with layout 1, camera layout 2 showed a better performance in aircraft tail number identification.

The camera layout 2 results determine better identification accuracy. Only 5% of unidentified aircraft are identified because of software errors. Closer range of view and lower aircraft speed in the operation time window of the FoV can be considered the primary reasons for the enhanced identification accuracy in camera layout 2. The other factors are the visibility of the tail numbers (e.g., the imprinted tail number sizes). Moreover, they are sometimes cluttered and unclear. Furthermore, in about 6% of the cases, there was no imprinted tail number on the body of the operating aircraft (Figure 5.3).

Table 5.4 Aircraft operation identification (fleet mix) accuracy of the system

		Layout 1	**Layout 2**
% (correctly) identified		64%	80%
% unidentified	**Aircraft with no printed tail number**	7%	6%
	Not visible (small, cluttered, unclear tail numbers)	15%	9%
	Software error	14%	5%

Figure 5.6 Aircraft with small and not imprinted tail numbers

6. CONCLUSIONS

This paper presented an adaptive computer vision-based system for aircraft operation detection and tail number identification by processing the visual information of the airport cameras. The proposed system covers all airport operations through the camera system deployed at each end of the runway or the cameras deployed adjacent and oriented toward the entrance taxiways at airports. The field experiments demonstrated promising results regarding the detection of the aircraft, operation count, and identification of the tail numbers. The automatic transformation of the detected tail numbers into an editable format highly depends on the implemented OCR method.

The proposed intelligent system uses a framework that leverages several video frames to detect the aircraft and its tail number during the flight operation time window. This approach avoided issues such as reading the difficult-to-read tail numbers because of the occlusion of the tail numbers by the aircraft wing or due to the spatial transformation of the tail number caused by aircraft maneuvers. With this approach, a high value for the average maximum tail number character recognition score was achieved when the system was tested for the recorded aircraft operations at five general aviation airports within the state of Utah. These airports are:

1. Bountiful Airport
2. Brigham City Municipal Airport
3. Spanish Fork Airport
4. Heber City Airport
5. Logan-Cache Airport

We also developed a multi-frame-based probabilistic OCR system to connect the per frame aircraft tail number recognitions to the FAA registration database for registered general aviation aircraft in the United States. This was done with two objectives. First, the developed system was needed in order to handle the possibly different number of recognized characters obtained from tail number recognition algorithms in consecutive digital video frames. The second objective was to enhance the accuracy of the system by completing the partially recognized tail numbers due to the errors raised by the recognition networks.

The future work path for this research would be the fusion of the developed system with a secondary data source such as ADS-B. This data fusion would increase the accuracy of both systems. In this way, we can also have a better understanding of the percentage of the operations conducted by those aircraft not equipped with a transponder. Hybrid systems increase the accuracy of the individual systems and cover the limitations of each individual system. Also, this hybrid system can provide the airport authorities with more information if only one of the information sources is used. For example, some aircraft tail numbers are not readable by using vision-based systems, and an ADS-B system would solve the problem. Likewise, a vision-based system can recognize the identity of aircraft that do not have transponders.

6.1 Challenges

The proposed system uses digital cameras to record the required video frames for detection purposes. Many general aviation airports do not have the electricity infrastructure to provide the necessary power for executing the camera system. In order to overcome this challenge at these airports, electrical power can be provided by using solar-powered digital cameras. These cameras have recently become more easily accessible and can be installed at low costs.

Another challenge the proposed system might face is that it uses the visual information captured from the airfield under different weather and aircraft conditions. This could lead to misidentification of some of the aircraft operations at the airport. This matter is thoroughly discussed in the previous section. The discussed factors relate to the size of the tail numbers (if they are too small), the shape of the tail number (if it is abnormal or cluttered), the visibility of the tail number (if the used color is noticeable or if there is any imprinted tail number on the fuselage), and illumination and weather conditions (if it is too sunny or too dark).

In order to solve the abovementioned challenges, we used preprocessing modules to improve the quality of the aircraft tail number image. The employed preprocessing modules are designed to help the system identify the difficult-to-read aircraft tail numbers. In fact, these modules improve the cluttered images so that the new image looks more readable compared with the original image captured by the cameras. It should be noted that these techniques are only useful when the original image has a minimum quality to retrieve the lost information. Otherwise, other approaches should be taken. That is why a multi-frame-based approach is beneficial. It is almost impossible possible to lose all visual information of an aircraft tail number in several consecutive video frames (around 100) compared with the single-frame-based detection approach used in some intelligent systems.

6.2 Limitations

It is important to note that the developed computer vision-based aircraft operations detection/identification has the drawbacks of other vision-based systems. Therefore, any extreme weather condition that decreases the visibility in the camera field of view would negatively affect all vision-based systems. Regarding nighttime operation, we suggest the use of night vision cameras for those airports with limited lighting resources in their airfield. The considerably lower rate of nighttime aircraft activities at non-towered general aviation airports could also help the system perform at a high performance at small general aviation airports with no runway lighting.

By testing it for the collected data at different airports, the developed system has shown to be robust as it extracted the tail number image in all aircraft operations for those aircraft that had a tail number readable by human eyes. It should be considered that the inaccuracy regarding tail number recognition is related to the performance of the optical character recognition algorithm. The developed system can be used at all types of airports, from small backcountry airstrips to airports that have multiple runways and complex taxiways. In the latter case, the user should take careful consideration in order to avoid extra counting by the system. Double counting the aircraft operations is possible if in a multi-runway airport where two runways begin in close proximity to each other.

6.3 Recommendations

For airports with very long runways, the operator can use an extra camera near the connector where intersection departure is predicted. It is noteworthy that most general aviation airports would not need such cameras because the abovementioned mid-entrance taxiways (connectors as shown in **Figure 6.1**) are typically used for the purpose of taxiing arrival aircraft operations. Those arrival operations are already captured by the camera at the end of the runway.

Moreover, there is a command named “No intersection departure,” which makes the operations in an airport more organized. Hence, it would be easier to control the airport operations for radio stations as well. During our data collection sessions, we did not observe any intersection departure at the test location (general aviation) airports.

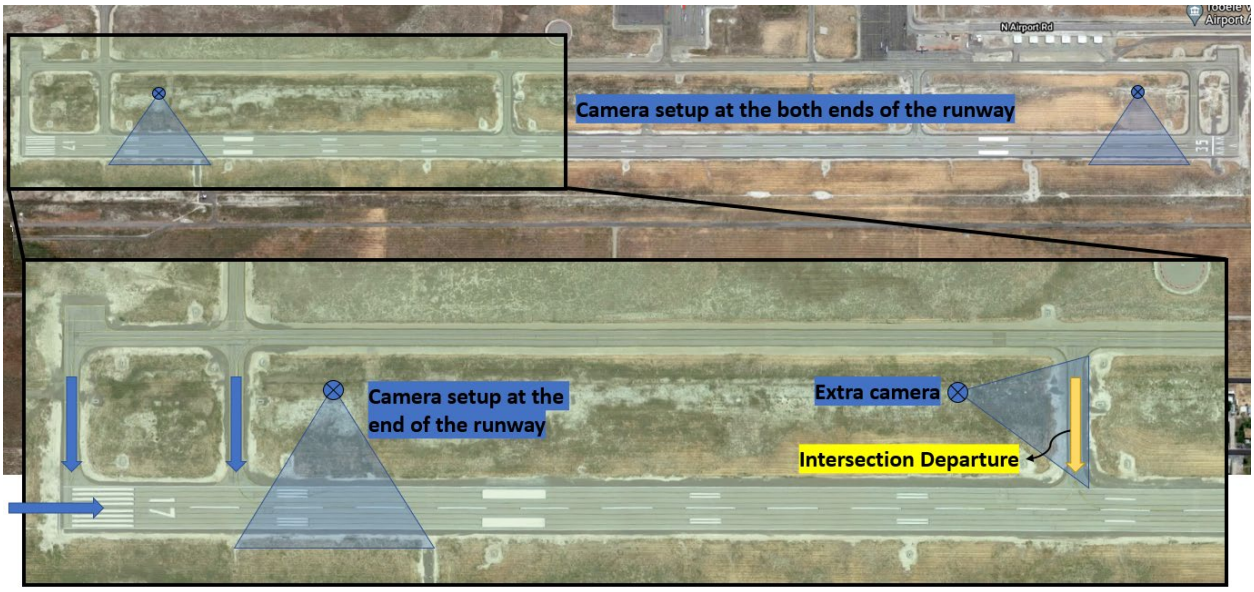


Figure 6.1 A 2D aerial view of the camera layout at an airport (Tooele Valley Airport)

The proposed system could be advantageous for several applications. Those applications are not limited to counting the number and volume of the operations for planning purposes only. One of the important applications is the system's use for automating the billing association process for fees related to landing operations. This application was not possible by the previous vision-based aircraft identification methods because there is no aircraft operation classification module (classifying the operation into departure operations and landing operations) devised in those methods. The proposed rapid tail number detection system can also be used for larger aircraft, while the previously proposed approaches had shortcomings for recognizing small general aviation aircraft. This increases the generalizability of the proposed system.

Regarding the tail number detection algorithm, it is essential to mention that the rapid detector can be trained to detect all prefixes for any country. For the current study, we only trained the model for detecting the letter "N," which is the prefix for aircraft registered in the United States, because our collected data only included U.S. general aviation aircraft. However, for international airports, we suggest using the deep tail number detector since it is possible to observe airplanes from various nationalities and, as a result, with various prefixes.

Finally, in this report, two camera layouts are proposed to record video footage from the airport. Camera layout 1 records the two ends of the runways. This camera layout is beneficial for cases where there is a cost limitation, and counting numbers is more important than the recognition of the tail numbers. In cases where there is a more available budget for purchasing cameras, camera layout 2 would be helpful. In this layout, cameras are placed adjacent to the entrance taxiways and have a closer range of view, thereby resulting in higher accuracy for detecting and recognizing the aircraft tail number.

REFERENCES

[1] M. J. Muia and M. E. Johnson, *Evaluating methods for counting aircraft operations at non-towered airports*, no. Project 03-27. 2015.

[2] J. H. Mott, M. L. McNamara, and D. M. Bullock, "Estimation of aircraft operations at airports using nontraditional statistical approaches," in *2016 IEEE Aerospace Conference*, 2016, pp. 1–11.

[3] M. J. Muia, "An analysis of the methods used to calculate customer operations at non-towered airports and of the associated managerial uses of operations information," 2000, [Online]. Available: https://www.proquest.com/pagepdf/304672330?accountid=14677.

[4] F. Sanchez, "ACCEPTABLE METHOD OF COUNTING AIRCRAFT OPERATIONS AT NON-TOWER AIRPORTS. FINAL REPORT," 1996.

[5] Federal Aviation Administration, "National Environmental Policy Act," 2021. https://www.faa.gov/regulations_policies/orders_notices/index.cfm/go/document.current/documentnumber/5050.4 (accessed Oct. 17, 2021).

[6] S. B. Young and A. T. Wells, *Airport planning and management*. McGraw-Hill Education, 2011.

[7] P. A. Jolicoeur and A. J. Khattak, "How airport context and service are related to general aviation aircraft operations," *Transp. Res. Rec.*, vol. 1788, no. 1, pp. 116–123, 2002.

[8] T. Li and A. A. Trani, "A model to forecast airport-level General Aviation demand," *J. Air Transp. Manag.*, vol. 40, pp. 192–206, 2014.

[9] S. K. Lau, "General Aviation Flight Data Monitoring, Fly with Intelligence–Best Practices to Improve the Safety and Efficiency of Flight Operations," *CAPACG White Pap.*, 2007.

[10] B. Chèze, P. Gastineau, and J. Chevallier, "Forecasting world and regional aviation jet fuel demands to the mid-term (2025)," *Energy Policy*, vol. 39, no. 9, pp. 5147–5158, 2011.

[11] Q. Zhang, J. H. Mott, M. E. Johnson, and J. A. Springer, "Development of a Reliable Method for General Aviation Flight Phase Identification," *IEEE Trans. Intell. Transp. Syst.*, 2021.

[12] Air Safety Institute, "Operations at nontowered airports," 2020. [Online]. Available: https://www.aopa.org/-/media/files/aopa/home/pilot-resources/asi/safety-advisors/sa08.pdf?la=en.

[13] I. Brilakis, H. Fathi, and A. Rashidi, "Progressive 3D reconstruction of infrastructure with videogrammetry," *Autom. Constr.*, vol. 20, no. 7, pp. 884–895, 2011.

[14] F. Dai, A. Rashidi, I. Brilakis, and P. Vela, "Comparison of image-based and time-of-flight-based technologies for 3D reconstruction of infrastructure," in *Construction Research Congress 2012: Construction Challenges in a Flat World*, 2012, pp. 929–939.

[15] C. Koch, S. G. Paal, A. Rashidi, Z. Zhu, M. König, and I. Brilakis, "Achievements and challenges in machine vision-based inspection of large concrete structures," *Adv. Struct. Eng.*, vol. 17, no. 3, pp. 303–318, 2014.

[16] International Civil Aviation Organization, "Convention on Civil Aviation," 1981. [Online]. Available: www.icao.int.

[17] FAA, "Runway Safety Statistics," 2020. https://www.faa.gov/airports/runway_safety/statistics/ (accessed Jan. 10, 2021).

[18] J. H. Mott and N. A. Sambado, “Evaluation of Acoustic Devices for Measuring Airport Operations Counts,” *Transp. Res. Rec.*, vol. 2673, no. 1, pp. 17–25, 2019.

[19] Florida Department of Transportation, “Operations counting at non-towered airports assessment,” 2018. [Online]. Available: http://www.invisibleintelligencellc.com/uploads/1/8/4/9/18495640/2018_ops_count_project_final_report_09102018.pdf.

[20] J. H. Mott, “Measurement of Airport Operations Using a Low-Cost Transponder Data System,” *J. Air Transp.*, vol. 26, no. 4, pp. 147–156, 2018.

[21] C. Yang, J. Mott, and D. M. Bullock, “Leveraging Aircraft Transponder Signals for Measuring Aircraft Fleet Mix at Non-Towered Airports,” *Int. J. Aviat. Aeronaut. Aerosp.*, vol. 8, no. 2, p. 1, 2021.

[22] Federal Aviation Administration, “Ads-b in and out, installation.,” 2022. https://www.faa.gov/air_traffic/technology/equipadsb/installation/.

[23] J. H. Mott, C. Yang, B. Hardin, S. Zehr, D. M. Bullock, and J. R. Bell, “Technology Assessment to Improve Operations Counts at Non-Towered Airports,” 2019.

[24] D. Thirde *et al.*, “A Real-Time Scene Understanding System for Airport Apron Monitoring.,” in *ICVS*, 2006, vol. 6, p. 26.

[25] A. Koutsia, T. Semertzidis, K. Dimitropoulos, N. Grammalidis, and K. Georgouleas, “Automated visual traffic monitoring and surveillance through a network of distributed units,” in *ISPRS*, 2008, pp. 599–604.

[26] T. Van Phat, S. Alam, N. Lilith, P. N. Tran, and B. T. Nguyen, “Aircraft Push-back Prediction and Turnaround Monitoring by Vision-based Object Detection and Activity Identification,” 2020.

[27] D. Bloisi, L. Iocchi, D. Nardi, M. Fiorini, and G. Graziano, “Ground traffic surveillance system for air traffic control,” in *2012 12th international conference on ITS telecommunications*, 2012, pp. 135–139.

[28] N. Pavlidou *et al.*, “Using intelligent digital cameras to monitor aerodrome surface traffic,” *IEEE Intell. Syst.*, vol. 20, no. 3, pp. 76–81, 2005.

[29] J. A. Besada, J. Portillo, J. García, J. M. Molina, Á. Varona, and G. González, “Image-based automatic surveillance for airport surface,” in *4th International Conference on Information Fusion, Fusion*, 2001, pp. 11–18.

[30] X. Zhang, M. Ding, and W. Wang, “Visual Tracking Algorithm for Aircrafts in Airport,” in *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, 2018, vol. 1, pp. 311–314.

[31] P. Thai, S. Alam, N. Lilith, P. N. Tran, and B. N. Thanh, “Deep4Air: A Novel Deep Learning Framework for Airport Airside Surveillance,” *arXiv Prepr. arXiv2010.00806*, 2020.

[32] J. A. Besada, J. M. Molina, J. García, A. Berlanga, and J. Portillo, “Aircraft identification integrated into an airport surface surveillance video system,” *Mach. Vis. Appl.*, vol. 15, no. 3, pp. 164–171, 2004.

[33] D. G. Vidakis and D. I. Kosmopoulos, “Facilitation of air traffic control via optical character recognition-based aircraft registration number extraction,” *IET Intell. Transp. Syst.*, vol. 12, no. 8, pp. 965–975, 2018.

[34] M. Farhadmanesh, A. Rashidi, and N. Markovic, “Implementing Haar Cascade Classifiers for Automated Rapid Detection of Light Aircraft at Local Airports,” in *Computing in Civil Engineering*, 2021.

[35] M. Farhadmanesh, A. Rashidi, and N. Markovic, “An Image Processing Method for Light Aircraft Tail Number Detection in General Aviation Airports,” *Transp. Res. Board 101th Annu. Meet. Transp. Res. Board*, no. TRBAM-22-02213, 2022.

[36] M. Farhadmanesh, A. Rashidi, and N. Marković, “General Aviation Aircraft Identification at Non-Towered Airports Using A Two-Step Computer Vision-Based Approach,” *IEEE Access*, 2022.

[37] M. Farhadmanesh, A. Rashidi, and N. Markovic, “Image Processing and Machine Learning Techniques for Automated Detection of Planes at Utah Airports,” Utah. Dept. of Transportation. Research Division, 2021.

[38] M. Farhadmanesh, N. Markovic, and A. Rashidi, “An Automated Video-based Air Traffic Surveillance System for Counting General Aviation Aircraft Operations at Non-Towered Airports,” *Transp. Res. Rec.*, 2022.

[39] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” *arXiv Prepr. arXiv2004.10934*, 2020.

[40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[41] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, “Textboxes: A fast text detector with a single deep neural network,” in *Proceedings of the AAAI conference on artificial intelligence*, 2017, vol. 31, no. 1.

[42] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, 2001, vol. 1, pp. I–I.

[43] R. Lienhart, A. Kuranov, and V. Pisarevsky, “Empirical analysis of detection cascades of boosted classifiers for rapid object detection,” in *joint pattern recognition symposium*, 2003, pp. 297–304.

[44] B. Shi, X. Bai, and C. Yao, “An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, 2016.

[45] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[46] J. Baek *et al.*, “What is wrong with scene text recognition model comparisons? dataset and model analysis,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4715–4723.