Detecting and tracking the position of suspicious objects using vision system.

MUTHUKRISHNAN, R., KANNAN, S., PRABHU, R., ZHAO, Y. and BHOWMICK, P.

2023

Detecting and tracking the position of suspicious objects using vision system

Ramkumar Muthukrishnan[a], Somasundar Kannan[a], Radhakrishna Prabhu[a], Yafan Zhao[a], and Parijat Bhowmick[b]

[a]School of Engineering, Robert Gordon University, Aberdeen, UK
[b]Electronics and Electrical Engineering Department, Indian Institute of Technology, Guwahati, India

ABSTRACT

Vision-based object tracking is crucial for both civil and military applications. A range of hazards to cyber safety, vital infrastructure, and public privacy are posed by the rise of drones, or unmanned aerial vehicles (UAV). As a result, identifying suspicious drones/UAV is a serious issue that has attracted attention recently. The key focus of this research is to develop a unique virtual coloured marker based tracking algorithm to recognise and predict the pose of a detected object within the camera field-of-view. After detecting the object, proposed method begins by determining the area of detected object as reference-contour. Following that, a Virtual-Bounding Box (V-BB) is developed over the reference-contour by meeting the minimum area of contour criteria. In order to track and estimate the precise location of the detected object in two-dimensions during observations, a Virtual Dynamic Crossline with a Virtual Static Graph (VDC-VSG) was constructed to follow the motion of V-BB, which is considered as a virtual coloured marker. Additionally, the virtual coloured marker helps to avoid issues linked to ambient lighting and chromatic variation. To some extent, it can function efficiently during obstructions like rapid position fluctuations, low resolution and noises etc. The efficacy of the developed algorithm is evaluated by testing with significant number of aerial sequences, including benchmark footage and the outputs were outstanding, with better results. The suggested method will support future industry of computer vision-based intelligent systems. Potential applications of the proposed method includes object detection and analysis applied to the field of security and defence.

Keywords: Image processing, Cascade classifier, Object detection, Object tracking, Aerial images

1. INTRODUCTION

In defence applications, where optical sensors are used for tasks like detection,[1] monitoring and tracing, etc., optical systems have become a crucial component.[2] In particular, unmanned aerial vehicle (UAV) autonomy has advanced over the past few years and is now a standard practise in a wide range of commercial uses, including path planning, rescue and search, obstacle avoidance, vision-based target localization, motion tracing, target identification, etc. In a broad spectrum of both commercial and military applications, such as traffic observation, surveillance, defence, and security, object tracking plays a pivotal role.[3] Tracing an item across a series of frames is known as object tracking. The difficulties in doing this entail managing partial and overall obstructions of the item in certain frames, several objects flying in close range to one another, crossing one another, etc.[4] Further research is required in the field of tracing an item of interest by inferring its location from the movement of nearby items.[5] Beforehand, recently published research work was discussed below, which predominantly inspired the development of a solution in this research realm.[6]

To begin, a fundamental tracking architecture has been used in Ref. 7, in which regions of interest are traced in drone frames utilising an ameliorated approach for blob tracing. However, tracing based learning and identification techniques that integrate tracking and identification into a single framework have gained popularity in recent years. Thereafter, the Tracking Learning Detection (TLD) framework was proposed to trace multiple

Further author information: (Send correspondence to Ramkumar Muthukrishnan)
Ramkumar Muthukrishnan.: E-mail: r.muthukrishnan@rgu.ac.uk

targets from an aerial perspective.[3] In order to improve the identification of tiny objects, Micro-Doppler radars are commonly used in radar-based identification,[8] which has received little interest.[4] Moreover, these devices were used to detect the drone by (i) acoustic recognition, (ii) radio frequency recognition, and (iii) radar recognition. When a small UAV motor or related additional parts, like its fan blades, make noise, acoustic identification employs microphones to pick up on the noise. Particularly sensitive microphones are used to pick up the signal, which is then processed to pull out key elements. Machine learning (ML) is used to identify and recognise drones by feeding their characteristics into the model. Noise in the background is quite vulnerable to sound-based identification due to its small detection range.[6]

Generally, RF dependent identification uses a RF device to monitor the response signal sent between a controller on the earth platform and a drone. Machine-learning algorithm was used to transform RF signals, which helps to categorise the Small UAV. The identification range of RF detection is very broad. However, it is unable to find unmanned aircraft.[9] A considerably more reliable and precise approach that has been successfully used to find aeroplanes is radar detection. Radars use high-frequency signals to send and collect returned signals. A moving drone may be located using the Doppler shift between the sent and returned signals. Radars, however, are unable to distinguish between stationary drones and flying birds, and they are also unable to detect hovering drones.[7] In Ref. 8 a revolutionary encryption-based drone identification method was presented to effectively identify an illegal drone. This method performs a dual-step validation of the obtained signal intensity indicator and the encryption key created from the UAV's location parameters. Moreover, a growing portion of the computer vision field is interested in visual detection.[10] Neural network-based strategies are recommended for this area's recent study for improved detection precision. Though the majority of research works to increase detection precision, the major goal of this project is to create a tracker that is reliable to deal with images threats and operate precisely in sky frame arrangement.

This article is organised as follows: Section 1 describes the introduction and related research work. Following that, Section 2 explains the developed and proposed models, and experimental analysis was carried out by comparing the developed models with each other, which is explained in Section 3. Section 4 concludes this article with a future plan.

2. PROPOSED METHOD

In this research, a dual classifier model was created to recognise unidentified aircraft and estimate their position in a continous flow of images that was taken by the vision system. Two unidentified aircraft identification models (Haar and LBP) were then subjected to experimental analysis, which helped to choose and propose the most efficient model. The proposed model is cutting-edge and very successful in identifying unknown aircraft. A simpler method for gradually developing HAAR-like classifiers was developed by Viola and Johns.[1] It sparked the creation of a model in the present research that is extremely effective at identifying moving, unrecognised objects (i.e., aircraft). Because of this, in order to advance, the main portion of the author's[1] work was used. The prominent contributions of this research work are;

- Preparation of data-set by manually extracting each frame from video.
- Training of robust models, which detect and track the unidentified aircraft.
- Development of a Virtual Dynamic Crossline with a Virtual Static Graph (VDC-VSG) through construction of the algorithm frame structure.

To begin with, the classifier model was first trained based on the item it was intended to identify (in this case, the goal was to track moving, unidentified aircraft in recorded footage).[5] As a result, the HAAR cascade classifier has to be properly trained before being used in this research. Using a laborious method, each frame from a sample video and a portion of a public database were prepared and collected to create the datasets. In comparison to positive pictures, there are a number of negative and positive images in this collection. Let 'α' indicate the positive pictures, and 'β' indicate the negative images being larger than the positive images. As a result, it defines 'p' as the number of positive frames and 'f' as the number of negative frames,[5]

$$\sum_{n=0}^{f} \beta \geq \sum_{n=0}^{p} \alpha. \tag{1}$$

To improve the detection ratio, the positive photos were in part manually trimmed. If the intended object area is not removed from the picture, the other item may be involved in the Haar characteristics, which might lower the classifier's effectiveness. In order to lower the noise component in positive photos, the images should be cropped in accordance with the needs of the training objective. Based on the importance of straightforward attributes, the object detection process categorises photos.[1] It is never easy to decide between pixels and features when describing an object detection characteristic. As compared to pixels, features offer greater benefits.[11] The use of features over pixels directly has a variety of justifications. First off, feature-based systems perform computations faster and with more efficiency than pixel-based systems.[1] Haar-based cascade classifiers were used, due to their improved display of fine-scale texture, which is appropriate for object recognition.

2.1 Haar-Like Features

Numerous researchers have often used the Haar characteristic for object identification. By modifying the concept of employing Haar-like wavelets, Viola-Jones utilised rectangular characteristics. Instances of the rectangular Haar-like features that may be used to detect objects are shown in Figure 1. It displays two, three, and a single four-rectangle feature. Since the area at the item's corner is thought to be darker than all other regions, while the area in the middle may be lighter, they are used to identify the common characteristics of the object. To calculate the variation in the contrast zone, take the total of the pixels inside the white rectangle and subtract it from the sum of the pixels inside the grey rectangle.

By swiping a static-size rectangle of the wavelets across the input picture at all dimensions, these Haar-like features may be used to identify the appearance of objects. Considering a detector with a 32 x 32 base resolution, the Haar-like feature that can be extracted from it is $180,000+$, which is a sizeable feature in terms of the amount of time needed to address it computationally. The extraction of features procedure is accelerated by using an integrated frame, an representation of the intermediate picture.

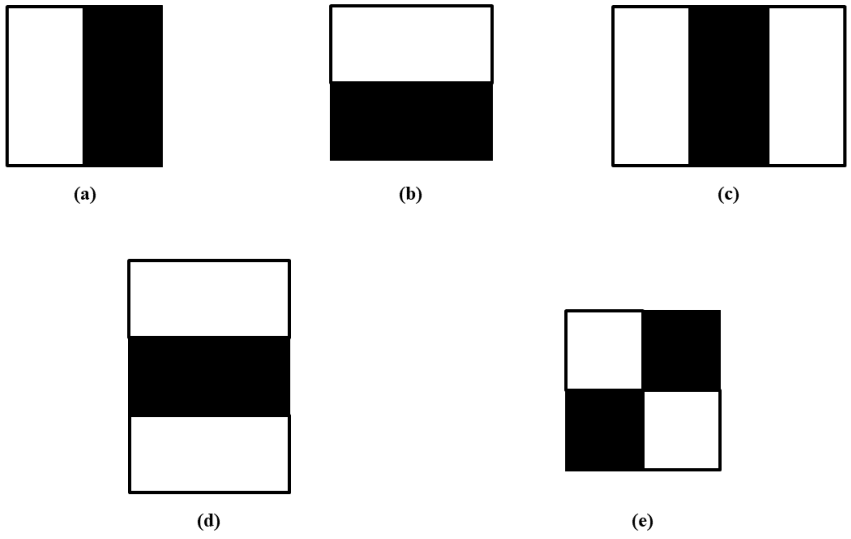

Figure 1. Haar wavelets Types. (a) Edge feature (Right-left), (b) Edge feature (Bottom-top), (c) Line feature (Vertical-middle), (d) Line feature (Horizontal-middle), (e) Cross feature (Diagonal)

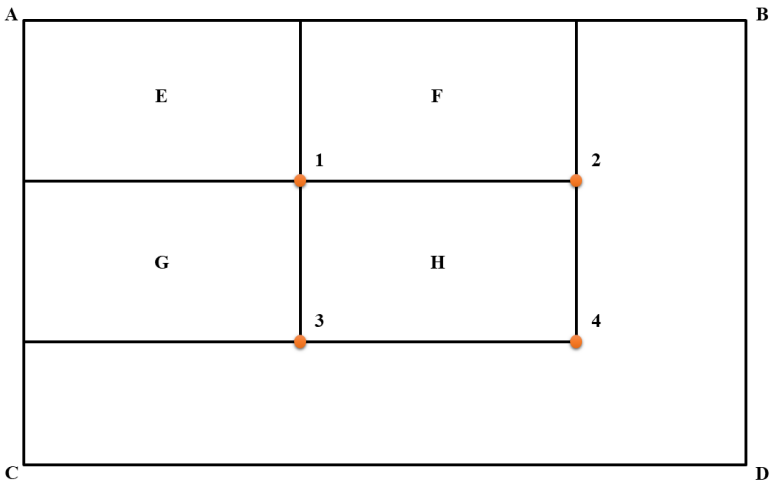

Figure 2. Quantifying pixel value using integral frame

2.2 Integral Picture

An intermediate version of the picture, which is defined as the integral image, may be used to quantify rectangle characteristics very instantly.[1] The total of the pixels to the right and on top of the coordinates x and y, inclusively, makes up the integral picture at those positions;[1]

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), \tag{2}$$

where integral picture is denoted by $ii(x, y)$, whereas the original picture is represented by $i(x, y)$. To locate and extract the features more rapidly, an intermediate version of the picture is required.[11] A new picture rendition known as an integrated image enables quick feature assessment.

Using Figure 2 as an illustration, it can be seen that an integral portion of the rectangle ABCD (EFGH) is utilised in order to quantify the pixel value and the total number of pixels in the rectangle. The total of the pixels on the rectangle E is represented by the integral value at position 1, E + F is represented by the integral value at position 2, E + F + G is represented by the integral value at position 3, and E + F + G + H is represented by the integral value at position 4. The total number of pixels in H is quantified to be $4 + 1 - (3 + 2)$.

2.3 AdaBoost Algorithm for Selecting Features

A series of cascade classifiers grouped in steps make up a cascade object detector. Weak learners who must make decisions make up each stage. Using a method known as boosting (also known as AdaBoost), each step chooses fewer characteristics. Adaboost combines the mean mass of the judgments made by the dull classifiers to produce a highly intricate classifier.[11]

Let's use the Adaboost classifier as an example and imagine that it is learning based on the advice of a team of experts. $E_n(x_i)$ represents the categorization outcome for each expert E_n for the given input x_i. $E_n(x_i)$ can only take couple of outputs that are expressed as $+1$ or -1, i.e., $E_n(x_i)$ ϵ $(-1, +1)$, in order to distinguish between the practicing vector of dual outputs. K (x_i) stands for the experts' aggregate view. It stands for the weighted total of expert recommendations combined linearly, which is represented as follows:[11]

$$K(x_i) = w_1 E_1(x_i) + w_2 E_2(x_i) + \cdots + w_n E_n(x_n), \tag{3}$$

where w_1, w_2, w_3, w_4, w_5, ..., w_n are the mass assigned to every proficient recommendation, and $E_1(x_i)$, $E_2(x_i)$, $E_3(x_i)$, $E_4(x_i)$, $E_5(x_i)$, ..., $E_n(x_i)$ reflect the judgements made by n proficient.

The dull learner chooses the best threshold classification function for every feature, aiming to misclassify the fewest possible cases.[1] Thus, the components of a weak classifier, $h_j(x)$, are a feature (f_j), a threshold (θ_j), and a parity (p_j) that denotes the sign of the inequality:[1]

$$h_j(x) = \begin{cases} 1, & \text{if } p_j f_j(x) < p_j \theta_j \\ \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

A dull classifier's stages could provide either a required or non-required results.[11] The presence of an object on the frame is indicated by a positive value, and the absence of an object is shown by a negative value.

2.4 Haar cascade classifier training setup

The effectiveness of practicing the cascade classifier is determined by the training settings. One Ada-boost classifier is used in the training process, and it goes through several phases. The cascade classifier receives a minimum hit rate of 0.995, which also serves as a measure of the effectiveness of the training process. The false alarm rate determines the highest possible number of incorrect detections that can be produced during training. The Max False Alarm Rate (MFA_{Rate}) option is used to provide this number.[12]

During stage 1 training, MFA_{Rate} improved from 1 to 0.042 in 6 seconds of training time. Similarly, during the next stage of training, MFA_{Rate} improved from 1 to 0.142 for 12 seconds of training time. In the last stage of training, MFA_{Rate} improved from 1 to 0.824 in 33 minutes, and 32 seconds of training time. The real positive rate rises significantly between MFA_{Rate} values of 0.2 and 0.3 and stays high until MFA_{Rate} values of 0.3 and 0.9. Although practicing time is essentially stable, it gradually changes until MFA_{Rate} is between 0.4 and 0.9. However, practicing time drops unexpectedly after stage 18 (false alarm achieved) has been completed before it reaches stage 20, while detection time rises (false alarm reached).

2.5 Cascade classifer

This method is defined as a linked classifier that becomes increasingly complicated as it moves through the stages. Due to its minimal false-positive rates, it rejects the majority of the $-^{ve}$ windows while permitting the $+^{ve}$ examples seen in Figure 3. It is a cascading model in which the decision of the 1^{st} classifier dictates the action of the 2^{nd} classifier, and every step classifier initiates the following classifier if the outcome is affirmative. Higher detection rates are the goal of the classifiers. The sub-window is automatically rejected if any step in the cascade has a negative result. On the other hand, if the desired outcome (the target item) is obtained, the classifier keeps going through more phases until all the components are inspected. In most cases, the cascade steps are produced using learning algorithms (AdaBoost) by changing the threshold value to produce the fewest false negatives feasible. Higher detection rates and false-positive rates are nonetheless produced by lower AdaBoost threshold values. Furthermore, every part of the numerously divided windows has the ability to add data to this attentional cascade.[11]

2.6 Local Binary Patterns

This technique uses a different pattern, which is not parameterised for representing images that encode global and local targeted object properties into a small feature histogram.[11] Although LBP was first developed to examine pattern characteristics, it has proven to be a highly effective tool for describing the global and local structures of a picture. It has been tested in a variety of applications, including remote sensing, image and retrieval video, biomedical analysis, aerial image analysis, motion analysis, and many more fields.

The LBP format, which assigns decimal numbers to each pixel, is used to organise an image's pixels. LBP codes are the abbreviation for the Local Binary Patterns format. By deducting the value of the central pixel from its neighbours, LBP takes each pixel's 3 × 3 neighbour into account (8). A positive number is symbolised by the number 1 if the outcome is a positive number. If not, it equals 0. Beginning at the top left and moving clockwise, this technique will produce binary numbers (zeroes and ones) that may be merged. Local binary patterns are the terms used to describe the generated integers. The LBP value is converted, and the resultant decimal value is used to mark the pixels. A 3 x 3 neighbourhood pixel's local binary transformation is shown in Figure 4.

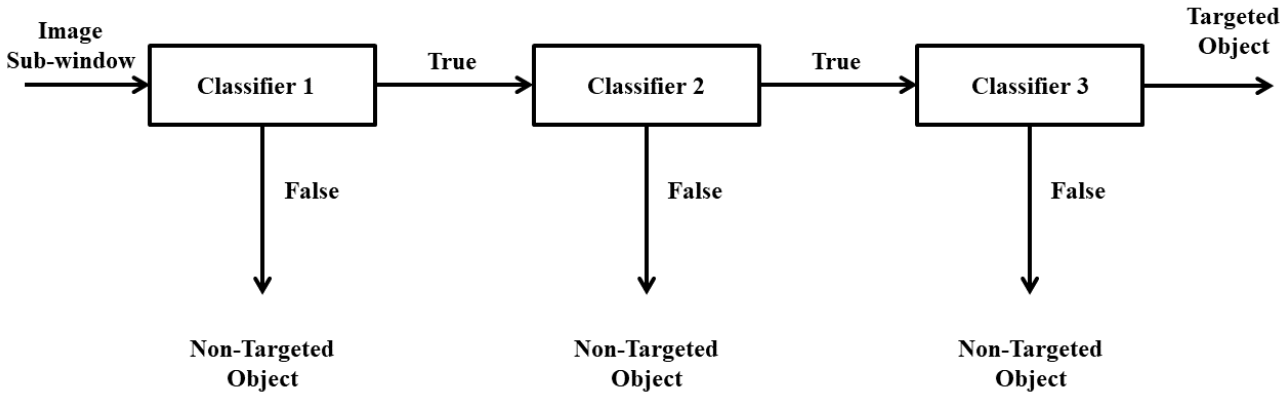

Figure 3. Representation of cascade classifier

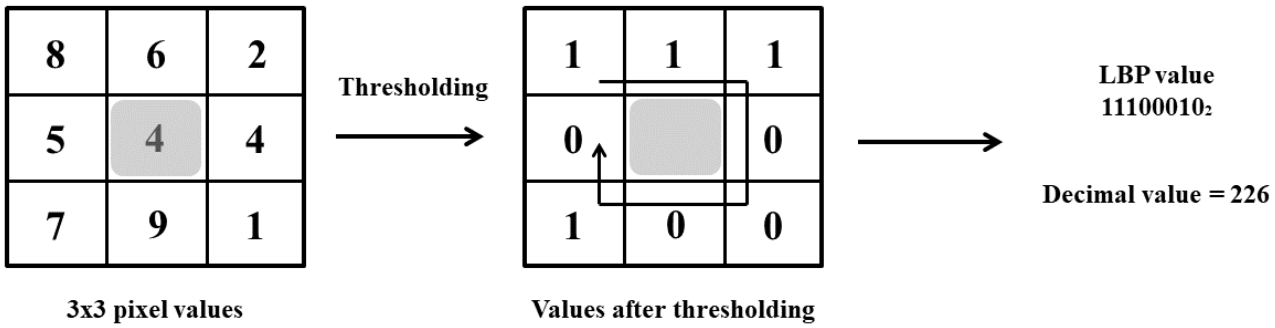

Figure 4. LBP transformation from pixel values

2.7 Algorithm frame structure

After developing the cascade classifier models (Haar and LBP), an algorithm frame was structured with the implementation of the developed cascade classifier model. It helps detect and track the aircraft in recorded videos. Thus, the location of the aircraft is virtually depicted on the screen. The structured algorithm frame is shown in Figure 5. It begins with extracting each frame one by one from the imported video file for further processing. Thereafter, a trained cascade classifier model (Harr/LBP) was implemented to detect the presence of aircraft in the given video file. If the aircraft is detected, then it develops a Virtual Bounding Box (V-BB) over the detected object until it detects the presence of the aircraft. Otherwise, it checks the preceding frame often to detect the presence of aircraft.

Each frame is transformed from RGB to HSV. Generally, RGB is an amalgamation of red, green, and blue colours. Concurrently, the amalgamation of hue, saturation, and value is represented as HSV. The intensity of colour is represented as saturation; the quality of colour is represented by hue, which ranges from 0^o to 360^o; and the contrast of colour directly indicates the value. A threshold approach is used to choose the colour of V-BB in the RGB-to-HSV transformed frame. Thus, this threshold process makes V-BB white and the rest black. Therefore, it is helpful to repeat the entire process by grabbing the V-BB and keeping it in a loop.

Moreover, it checks for the presence of V-BB in the screen plane under the loop. If V-BB is present, then it proceeds with the further process; otherwise, it checks until V-BB is present. The pixel location value of V-BB is obtained as follows:[13]

$$X_c^1 = \frac{\sum_{i=1}^{K_1} X_i^1}{K_1}, \tag{5}$$

$$Y_c^1 = \frac{\sum_{i=1}^{K_1} Y_i^1}{K_1}, \tag{6}$$

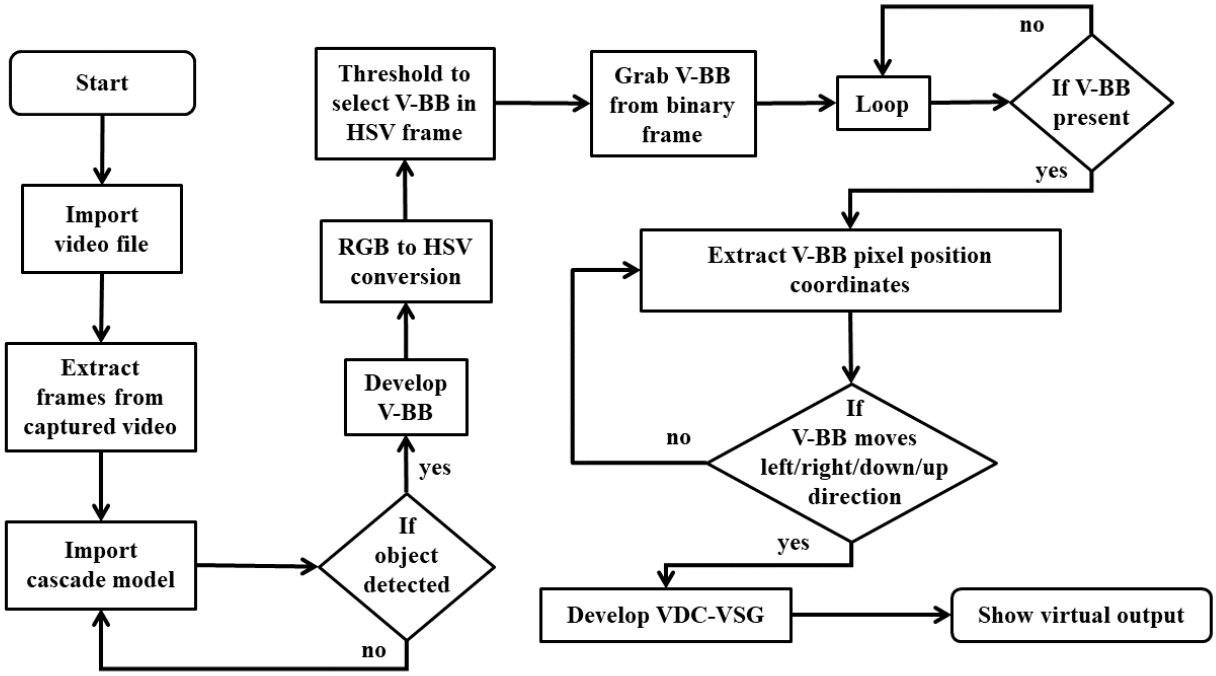

Figure 5. Algorithm flowchart for proposed method

where k_i is the total number of pixels that match the V-BB, and x_i and y_i are the values of position in the x and y directions for the i^{th} pixel in the screen plane. The centre of mass is denoted by x_c and y_c. The obtained V-BB pixel location value is utilised to trace the location of detected aircraft in a video frame with the help of a virtually designed Virtual Dynamic Crossline with a Virtual Static Graph (VDC-VSG). After processing them all, the final virtual output of the structured algorithm frame is shown in Figure 6. To check the accuracy level, those models were compared with each other, which is discussed in the next section.

3. EXPERIMENT RESULTS

The precision level of the developed models (Haar and LBP) was tested in two situation-based recorded videos (under static and dynamic frames). The camera was fixed in the first video, and the camera was moving in the second video. Thus, different satisfactory results were obtained from those sample videos. Although there are some occultations such as smoke, sunshine, grass on the ground, etc,. The results of the detected aircraft position and traversed pathway of aircraft are depicted in Figures 9 and 10 for the first and second sample videos, respectively. In addition, the visual output of detection and tracking is shown in Figures 7 and 8.

The position of the detected aircraft is obtained in pixels. Therefore, the result of the x and y axes in Figures 9 and 10 represents the sample video screen resolution (1920 x 1080). The detection of aircraft using the Haar cascade classifier model is depicted in orange and blue for the LBP classifier model in Figures 9 and 10. The aircraft was successfully detected instantly from the beginning to the end of those sample videos using the Haar model. However, the LBP model was unable to detect the aircraft in an abundance of frames, which are shown as a few missing aircraft trajectory lines in Figures 9 and 10. Through this comparison between those developed models, the Haar cascade model performed excellently in sample videos compared to the LBP model.

Moreover, the performance of the developed models was good in fixed video frames. Thus, a clear trajectory of the detected aircraft was obtained, as shown in Figure 9. However, the developed models were not suitable for dynamic video frames. Thus, we obtained the clumsy trajectory of the detected aircraft shown in Figure 10. Although less inefficient and highly precise, it still needs an explicit line of sight to the target. Additionally, in adverse weather conditions like persistent fog, dust, heavy rain, etc., precision suffers.[8] In the presence of

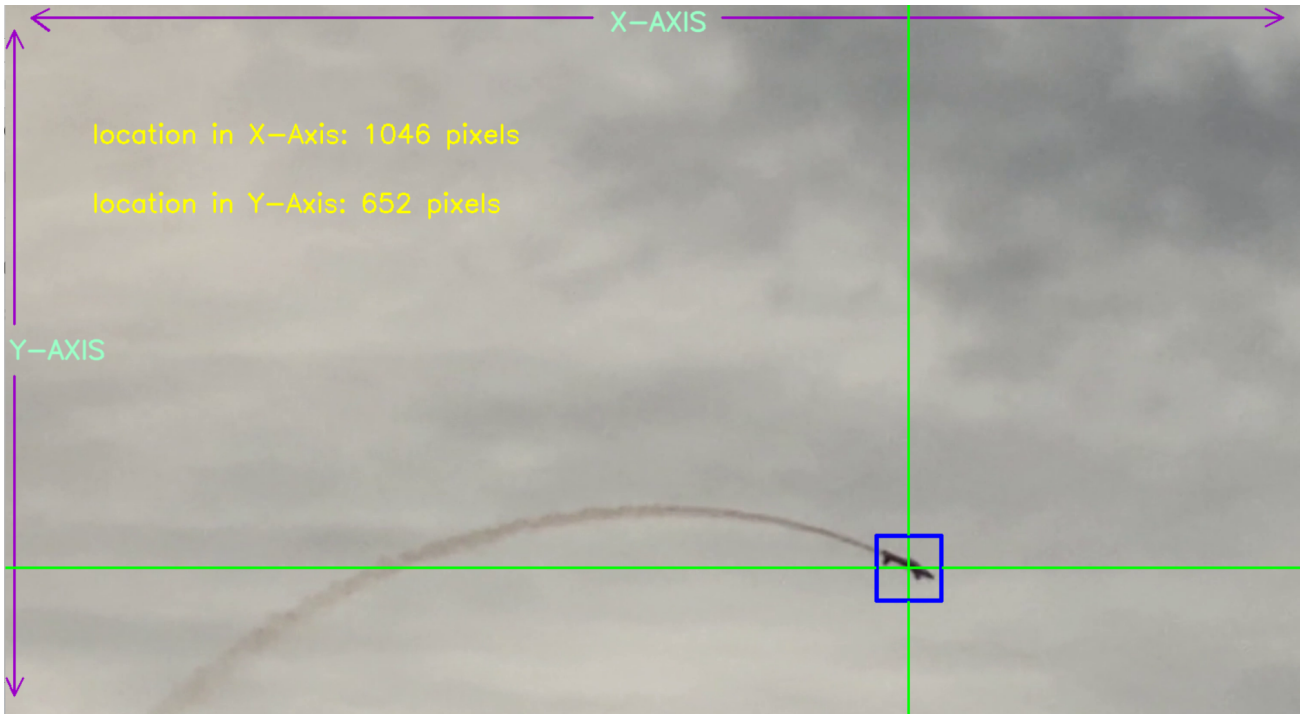

Figure 6. Final output of proposed method

Figure 7. Final output in static video frame

Figure 8. Final output in dynamic video frame

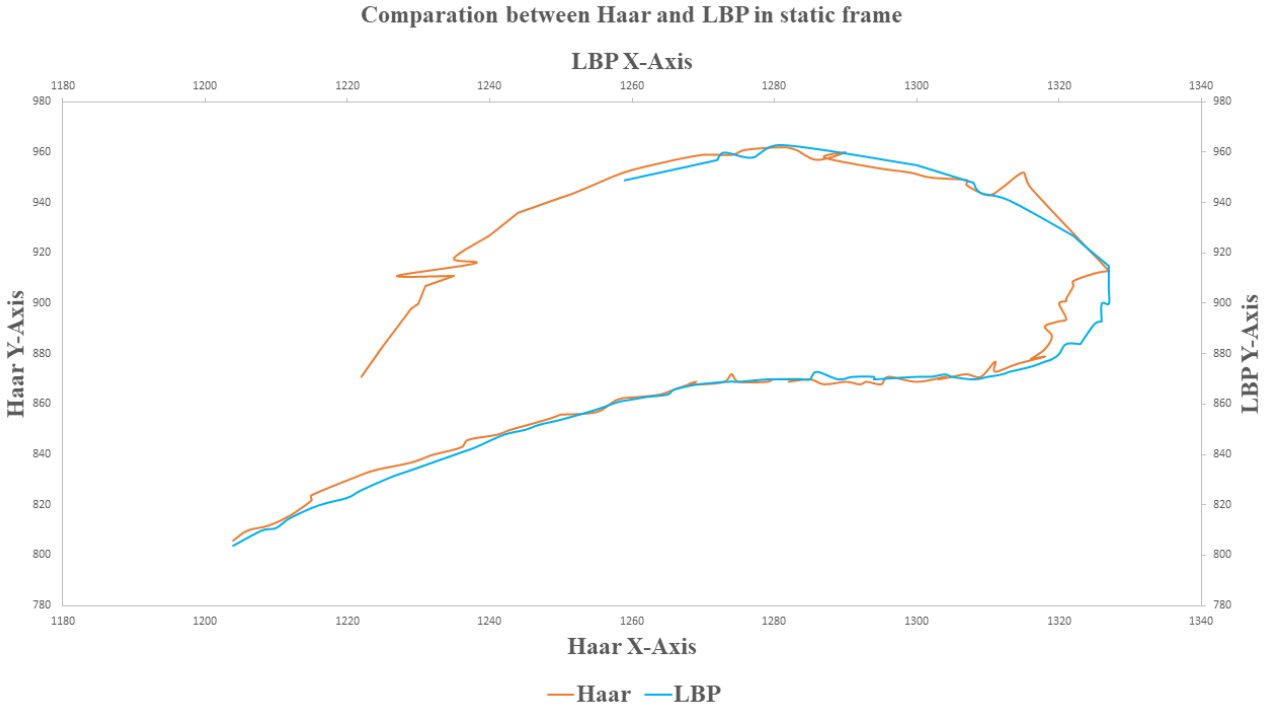

Figure 9. Comparison between Haar and LBP in static video frames

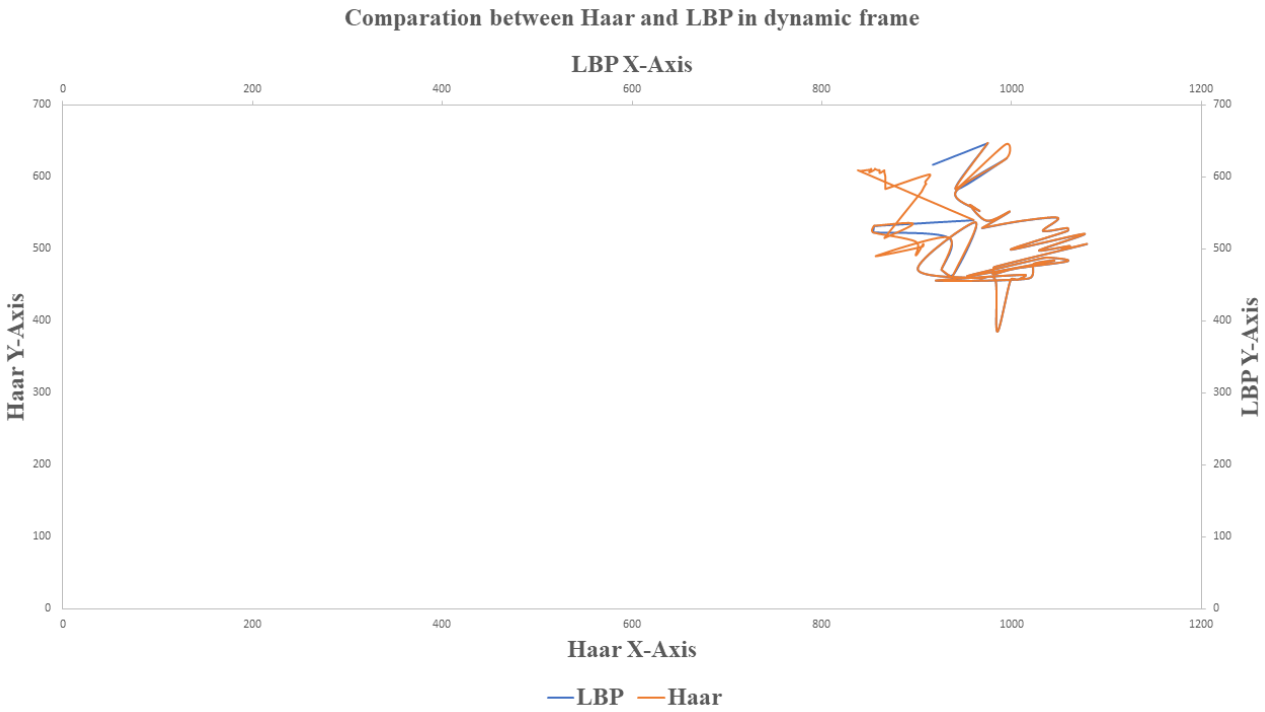

Figure 10. Comparison between Haar and LBP in dynamic video frames

unforeseen occultations, such as jitters, noise, posture, scale, and lighting deviations, several failure occurrences were found during the execution of constructed models. Future research will focus on overcoming these issues, which represent further amelioration required for the developed models to perform in dynamic frames.

The entire research work is operated in Python 3 on an i7 CPU with a processing capacity of 16 GB of RAM and a 500 GB hard drive, along with the related image processing toolkit OpenCV. For this research, 288 positive images, which contain aircraft, and 519 negative images, which contain aerial-related scenarios without aircraft, were extracted from recorded video. These datasets were used to train those models (Haar and LBP). The developed models and the datasets with sample video, and output video are made publicly available at the given GitHub link (https://github.com/rk3839/aerial_object_track/tree/master).

4. CONCLUSION

Various issues, such as size and lighting variations, partial and total occlusion, comparable objects moving in close proximity to one another, random jitters, moving objects coming to a stop, and noise, have been addressed when it comes to object tracking in aerial picture sequences. In the present research, two cascade-based classifier models—Haar and LBP—were developed and deployed into use in an algorithm that is essentially designed to follow a single object, an aircraft. To evaluate the accuracy of such models, preliminary tests were carried out. As the ideal model, this Haar cascade-based classifier is proposed. The proposed model is also empirically evaluated and successfully identifies an appropriate aircraft in fixed point-of-view-based video frames. By incorporating cutting-edge neural network models and applying them through extensive testing on datasets made up of several aircraft and drone models with various viewpoints, sizes, and environmental circumstances, visual identification can be further enhanced in the future.

REFERENCES

[1] Viola, P. and Jones, M., "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*], **1**, I–I, Ieee (2001).

[2] Xiao, J., Yang, C., Han, F., and Cheng, H., "Vehicle and person tracking in aerial videos," in [*International Evaluation Workshop on Rich Transcription*], 203–214, Springer (2007).

[3] Malagi, V. P., DR, R. B., and Rangarajan, K., "Multi-object tracking in aerial image sequences using aerial tracking learning and detection algorithm," *Defence Science Journal* **66**(2), 122–129 (2016).

[4] Björklund, S., "Target detection and classification of small drones by boosting on radar micro-doppler," in [*2018 15th European Radar Conference (EuRAD)*], 182–185, IEEE (2018).

[5] Mohiuddin, K., Alam, M. M., Das, A. K., Munna, M. T. A., Allayear, S. M., and Ali, M. H., "Haar cascade classifier and lucas–kanade optical flow based realtime object tracker with custom masking technique," in [*Advances in Information and Communication Networks: Proceedings of the 2018 Future of Information and Communication Conference (FICC), Vol. 2*], 398–410, Springer (2019).

[6] Shi, Z., Chang, X., Yang, C., Wu, Z., and Wu, J., "An acoustic-based surveillance system for amateur drones detection and localization," *IEEE transactions on vehicular technology* **69**(3), 2731–2739 (2020).

[7] Morris, P. J. B. and Hari, K., "Detection and localization of unmanned aircraft systems using millimeter-wave automotive radar sensors," *IEEE Sensors Letters* **5**(6), 1–4 (2021).

[8] Khan, M. A., Menouar, H., Khalid, O. M., and Abu-Dayya, A., "Unauthorized drone detection: Experiments and prototypes," in [*2022 IEEE International Conference on Industrial Technology (ICIT)*], 1–6, IEEE (2022).

[9] Basak, S., Rajendran, S., Pollin, S., and Scheers, B., "Drone classification from rf fingerprints using deep residual nets," in [*2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)*], 548–555, IEEE (2021).

[10] Zheng, Y., Chen, Z., Lv, D., Li, Z., Lan, Z., and Zhao, S., "Air-to-air visual detection of micro-uavs: An experimental evaluation of deep learning," *IEEE Robotics and automation letters* **6**(2), 1020–1027 (2021).

[11] Adeshina, S. O., Ibrahim, H., Teoh, S. S., and Hoo, S. C., "Custom face classification model for classroom using haar-like and lbp features with their performance comparisons," *Electronics* **10**(2), 102 (2021).

[12] Satti, S. K., Devi, K. S., Dhar, P., and Srinivasan, P., "Detecting potholes on indian roads using haar feature-based cascade classifier, convolutional neural network, and instance segmentation," *Soft Computing* **26**(18), 9141–9153 (2022).

[13] Su, F., Fang, G., Kwok, N. M., et al., "Adaptive colour feature identification in image for object tracking," *Mathematical Problems in Engineering* **2012** (2012).