ICIC Express Letters
Part B: Applications
Volume **14**, Number **12**, December **2023**

ICIC International ©2023 ISSN 2185-2766
pp. **1285–1293**

YOLOFOR: YOLO AND OPTICAL FLOW FOR FORENSIC ANALYSIS OF ATTACKED DRONE CASE

Arda Surya Editya, Tohari Ahmad*, Hudan Studiawan
and Swardiantara Silalahi

Department of Informatics
Institut Teknologi Sepuluh Nopember
Kampus ITS, Sukolilo, Surabaya 60111, Indonesia
{ 7025211019; 6025211018 }@mhs.its.ac.id; hudan@its.ac.id
*Corresponding author: tohari@if.its.ac.id

Received February 2023; accepted May 2023

Abstract. *Drone devices have been used for various activities recently. During its operation, a drone may get into an accident, and the authority needs to find the root causes. For this purpose, a forensic analysis is needed to collect relevant information to help and support the investigation. Several evidentiary artifacts can be utilized, such as images or videos captured from the drone's camera. Furthermore, from the video data, it can be seen whether the object causing the incident is moving toward the drone, or the drone is actually moving toward the object by estimating the direction of the object's movement. In this study, we propose YOLO and Optical Flow for Forensics (YOLOFOR), an attack detection method that can perform object detection accompanied by an object's movement direction estimation model to assist forensic investigation on drone video data. The proposed method comprises two main components, specifically YOLOv5 and Lucas Kanade. YOLOv5 is used for object detection, while Lucas Kanade is utilized to calculate the direction of moving objects. The experimental results demonstrated that the proposed method could detect objects with an mAP of 0.633. The estimated direction produced by Lucas Kanade has the least number of the quiver, indicating that the quiver is more focused on the detected moving object.*

Keywords: Digital forensics, Drone forensics, Object detection, Optical flow, YOLO

1. **Introduction.** A drone has become a common technology implemented for various purposes, such as traffic, forest health, flood, and crowd [1, 2, 3]. In its operation, a drone may encounter incidents such as a wall collision, attack, and system crash. In a commercial setting, the data stored in the drone's storage system are highly valuable. For this reason, if an incident happens to a drone, the stakeholders need to know the root cause by conducting a forensics investigation. Generally, several steps are needed for the investigation: preparation, data collection, data analysis, and reporting.

Drone forensics has a significant role in identifying and analyzing drone evidence to find the root causes of an accident [4]. Several studies in drone forensics have been proposing methods for analyzing drone artifacts, such as implementing a convolutional neural network (CNN) to identify attacking objects from a drone camera [5], optical flow to detect movement of drone [6], and YOLO on a drone camera to detect an object captured by the drone camera [7]. A deep learning method, the YOLO [8], can be implemented to detect objects in the images or videos captured by the drone camera suspected of attacking the drone. In the forensic investigation, the information about the type of objects is not sufficient to explain the incident's root cause. Some helpful information can be derived from the direction of the object's movement.

DOI: 10.24507/icicelb.14.12.1285

In specific cases, such as Nepal and Eslamiat [8] and Sahin and Ozer [9], deep learning is being used to detect landing spots and objects while drones are flying. In addition, the study of optical flow into drone was conducted by Editya et al. [6] and Pedro et al. [5]. In their study they used optical flow to estimate the direction when a drone collided and was attacked by another object. Based on the previous study, this study aimed to develop the combination of YOLO and optical flow for faster and more accurate forensic analysis.

This study proposes an attack detection technique by appending an optical flow [10] model to the YOLO [8] architecture, which adds direction estimation to the detected object. Generally, this research proposes a method to recognize the drone attacker's object and draw an estimated attacker's direction. This method provides more information to help the forensic investigation of attacked drone cases. It is intended to make a forensic investigation faster and more accurate, represented by the mAP score for the YOLO and processing time for the optical flow evaluation.

YOLOFOR's results are quite good, with an mAP score of 0.633 and an average of 3,476 direction estimations. It was tested on a whole dataset from ColaNet [5]. The drone collision cases included a bird attack, a drone attack from another drone, and a human attack.

This paper is organized as follows. Section 2 presents previous research on the YOLO and the use of optical flow in many applications, specifically in drones. Section 3 explains the proposed method in drone forensics. Section 4 provides the result and the analysis. Finally, Section 5 presents this research's conclusion and future work.

2. **Previous Work.** The use of deep learning on drones has been studied in specific cases. Nepal and Eslamiat [8] investigated object detection using YOLO versions on drones, including YOLOv3 [11], YOLOv4 [7], and YOLOv5 [12]. The training is conducted on a Personal Computer (PC) and a Companion Computer (CC), which are analyzed to see each YOLO method's performance in determining landing spots. It is found that YOLOv5 has the highest accuracy. In other research, Sahin and Ozer [9] introduced a YOLO-based method to improve object detection, especially when the drone is flying, called YOLODRONE. It can detect objects accurately even if the objects are seen from a considerable distance.

To increase the performance, research on YOLO is also conducted by modifying parameters, such as that done by Yan and Xu [13] focusing on adding filters to the backbone layer of YOLO to minimize object detection errors, while Zhang et al. [14] improved YOLOv3 by working on the backbone layer of CSP Darknet. The method allows YOLO to detect objects in drone frames with a better performance. Furthermore, Lu et al. [15] developed YOLO with SIFT to detect and track multi small targets. Starting with pruning the YOLO network, we are able to enable the detection of small objects. To optimize detection, YOLO uses SIFT feature extraction to detect small objects. Currently, this method has an mAP score of 0.28. This method can be further developed to increase its mAP score.

In addition, Editya et al. researched optical flow in drone image data [6] by applying Gunnar Farnerback, Lucas Kanade, and Horn & Schunck methods to estimating the drone's movement direction. For the same objective, Pedro et al. [5] used CNN for optical flow analysis to estimate the distance between the object and the drone to avoid collisions. In another study, Minaeian et al. [16] analyzed optical flow applications in a drone. They used optical flow to detect moving objects using a drone camera. With their proposed, they can detect multiple moving objects with the drone camera, although it still has low F-Measure and Gmean values in the case of performance evaluation [17].

3. **Proposed Method.** The method proposed in this research, called YOLOFOR, is developed using YOLOv5 and Lucas Kanade, with a workflow as shown in Figure 1.

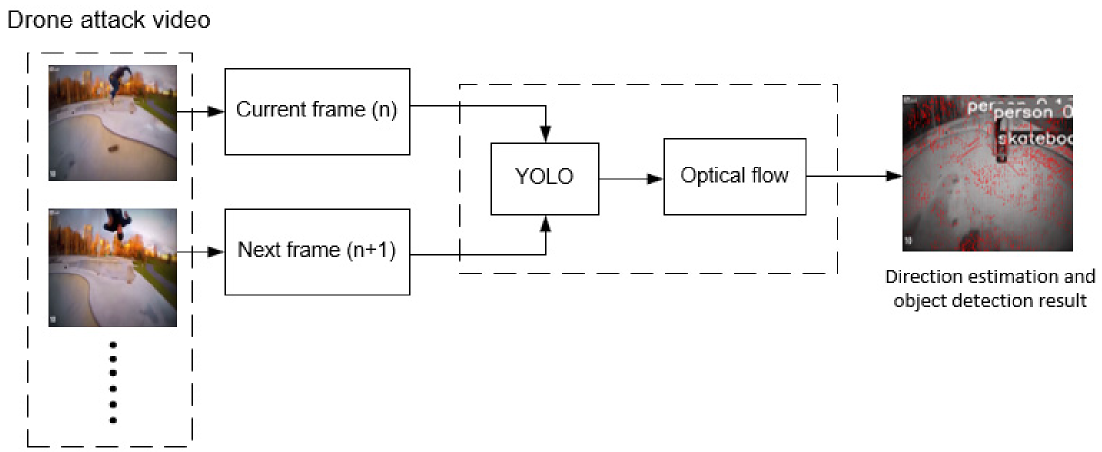

FIGURE 1. The diagram of the proposed method YOLO Optical Flow for Forensics (YOLOFOR)

Generally, the method can be divided into two steps: object detection and direction estimation. This method takes two frames as input, and each frame is processed using YOLOv5 [12] to detect objects within. From this stage, the object type is obtained. Afterward, Lucas Kanade is applied to the two frames to draw the direction of the object's movement. Finally, we have a frame containing the object type and the direction of the object's movement.

3.1. **Object detection.** The proposed method starts with object detection using the YOLO method. In general, there are three main processes in the YOLO method, i.e., a backbone, a PANet, and an output layer. In the backbone layer, the YOLO process starts by dividing the image into N grids, having an equal dimensional region of $S \times S$. Each grid is responsible for detecting and localizing objects within, and it can also be called BottleNeckCSP. Therefore, these grids calculate the bounding box as well as the object label. In addition, the algorithm determines the probability of an object being presented in the cell. It can be called Spatial Pyramid Pooling (SPP) [18]. Since both detection and recognition of the image are handled by the cells in the image, this process reduces computational time significantly [19]. As a result, it generates a lot of duplicate predictions. This is because multiple cells predict the same object with different bounding boxes and its process in Path Aggregation Net (PANet). To resolve this issue, YOLO uses non-maximal suppression. As a result, YOLO suppresses all bounding boxes having lower probability scores. In YOLO, each decision is assessed based on the object's probability, then selecting the object with the highest probability. In addition, it suppresses the bounding boxes with the largest intersection with the current high probability bounding box, calculated using Equation (1) to Equation (4) [12]:

$$b_x = \sigma(t_x) \times 2 - 0.5 + c_x \tag{1}$$

$$b_y = \sigma(t_y) \times 2 - 0.5 + c_y \tag{2}$$

$$b_w = p_w(\sigma(t_w) \times 2)^2 \tag{3}$$

$$b_h = p_h(\sigma(t_h) \times 2)^2 \tag{4}$$

From the above equation, b is a bounding box drawn in an image to show the detected object, $\sigma(t)$ is the value for each coordinate x and y at each time, and c is the confidence threshold which can be set by the user. Using these equations, YOLOv5 detects objects in static images and in video or real-time videos since the time parameter is included when creating the bounding box. Additionally, in Equation (3) and Equation (4), there are p_w and p_h that are used to predict the weight and height of the output image. So, the method can detect any size images. In YOLOv5, bounding boxes are developed using auto anchor boxes, making the bounding boxes more accurate. Specifically, drawing three or five objects shapes can be done better to decide their type.

3.2. **Direction estimation.** Once the YOLO process has been completed, we take the two frames with different timesteps and convert them into grayscale images to speed up the process. The procedure detects features in those frames using the Shi-Tomasi corner detector [20], groups all the features, and tracks the object using optical flow. By estimating optical flow between frames, we can measure the velocity of any objects in a frame extracted from an image using Equation (5):

$$I_x u + I_y v + I_t = 0 \tag{5}$$

where, I_x, I_y, I_t are the spatiotemporal images brightness derivatives representing values in frame matrices, and u is the horizontal optical flow indicating parameter that multiplies frame matrix values in x-axis. In y-axis, v shows the vertical optical flow depicting the matrix values multiplied by the vertical optical flow parameter.

The moving of each feature in x and y coordinates is obtained from Equation (1). The result from Equation (5) can be used to draw quiver symbols representing the direction of moving object. As a means of specifying the object movement using the optical flow method, Lucas Kanade is employed to localize the direction estimation for each object.

The optical flow generally has five stages of extracting information from video frames. In the process, two frames are taken as input. Then, the frame color space is converted to make it easier to calculate values between them. The next step is blurring the frame. This step is used to separate objects and backgrounds. Having found the object, we need to find a decent feature in the frame to estimate the movement. Once we have a movement value, we can draw the movement vector and calculate the total direction vector within a frame.

In this study, we utilize three optical flow methods that are often applied, namely Lucas Kanade [21], Horn and Schunck [22] and Gunnar Farnerback [23]. The Lucas Kanade method is a method that has been widely applied in optical flow for various domains. Lucas Kanade [21] is considered a simple method that can predict an object's movement by analyzing unique features from the frame. It only uses the gray channel for processing to find the difference between the two frames. As for finding the difference between frames, we use Equation (1). While Equation (6) is used to calculate the localization of direction estimation on each object and written as follows:

$$E_{u,v} = \sum_{x \in \Omega} W^2 \left[I_x u + I_y v + I_t \right]^2 \tag{6}$$

where E is defined as the estimation direction of an object movement in an image, W is a window function emphasizing the constraints at the center of each section; Ω is the optical flow constraint equation for each section. To solve the optical flow imperative for u and v in Equation (6), the Lucas Kanade method separates the original frame into simpler areas and expects a consistent velocity in each segment. Then, it performs the least squares weighting of the constraint equations to the model for $[u, v]$ in each segment. This calculation provides a set of optical flow vectors that are distributed over the frame and provide an estimation of the movement of objects in the frame.

The main advantage of this computation is that it keeps a fixed-size environment, and the number of computations in each frame tends to be constant. Therefore, the complexity of the algorithm is linear. The difference calculation is based on the number of pixels analyzed in the frame.

4. **Result and Analysis.** In this section, the test results from the existing methods and the results from applying the modified method, namely YOLOFOR, are presented. Three scenarios are designed for the experiment, which is described as follows. First, we apply

the YOLO method to performing object detection on the drone attack dataset. Second, we test the optical flow method to estimate the movement direction of objects on the drone attack dataset, and finally, we perform experiments on the YOLOFOR method to perform both object detection and direction estimation on the drone attack dataset.

4.1. **Material and preprocessing data.** At this stage, the video dataset obtained from ColaNet [5] and the DroneCrashes Youtube Channel (https://www.youtube.com/@DroneCrashes) are processed. The dataset consists of videos recorded from a drone during an attack on the drone. Thirty-five videos are converted into frames with a size of 416×416. As for the conversion of the video dataset, it produces 5,110 frames. All frames are divided into train, validation, and test datasets.

4.2. **YOLO (You Only Look Once).** In this experiment, three variants of YOLO were used, namely YOLOv3, YOLOv4, and YOLOv5. These methods were tested to determine the object detection performance and find the best one. The evaluation metric used in this experiment is mean Average Precision (mAP). The results of this test are shown in Figure 2. According to Figure 2, YOLOv5 has the highest mean Average Precision (mAP) score. The mAP is a parameter used to measure the average precision of object detection in YOLO based on Intersection over Union (IoU) as shown in Equation (7) and Equation (8):

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$mAP = \frac{\sum_{q=1}^{Q} AvP(q)}{Q} \tag{8}$$

Equation (7) explains how to get the precision value in a test that is obtained from the True Positive (TP) and False Positive (FP). After obtaining the precision value, we proceed to Equation (8), which explains how to get the mAP score. Q is the number of queries in the dataset, and $AvP(q)$ is the average precision value for a particular query.

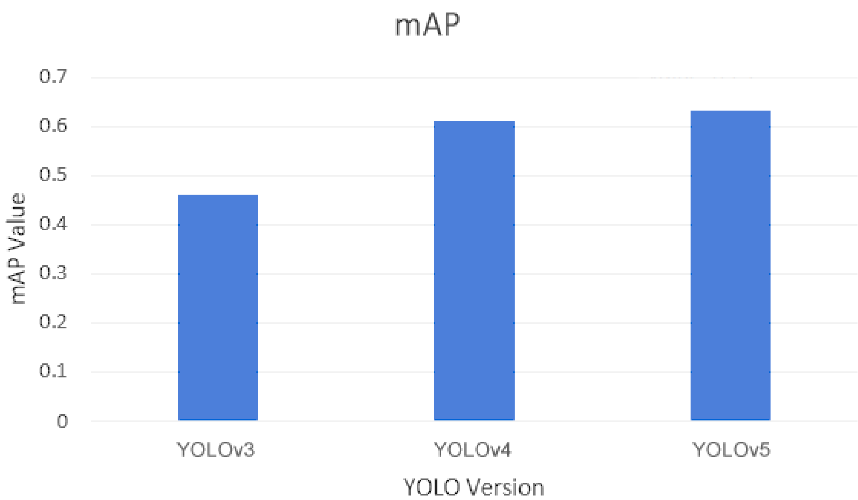

Figure 2. mAP values of object detection using three versions of YOLO

In this case, YOLOv5 is the most effective method for object detection, especially on datasets of attacked drones. It is because YOLOv5 uses the auto anchor feature to determine the bounding box in the three-dimensional YOLO test. YOLOv4 has the second-highest mean Average Precision (mAP) score. The YOLOv4 backbone from the DarkNet has many layers in the process and has a better mAP, but it takes more time than YOLOv5. While YOLOv3 still uses traditional methods to detect the object, which is K-Means clustering, to determine the bounding box, which affects the mAP score.

4.3. **Optical flow results.** In this experiment, three types of optical flow methods were tested, namely Lucas Kanade, Horn & Schunck, and Gunnar Farnerback. As for this test, three evaluation metrics were used: the number of direction estimations generated, CPU usage, and processing time. To get the score of each evaluation metric, a test was conducted using the video dataset of the attacked drone. Then the video was converted into frames. After that, the frames were processed using three optical flow methods. During the processing, computer resources were recorded, including CPU usage, processing time, and the number of directions estimated after processing. All of these experiments used Python 3.8 as the programming language.

Based on Table 1, it can be concluded that the most effective optical flow method is Lucas Kanade. This method has the least number of direction estimation vectors and consumes small CPU usage. This simple direction estimation indicates that the resulting information has a high level of focus. Furthermore, the small CPU usage indicates that the Lucas Kanade method can be used on processors with lower frequency and power.

Table 1. Comparison of optical flow methods

Method	Direction estimation	Processor work (%)	Time (second)
Lucas Kanade	3,391	9.834	17.972
Horn & Schunck	88,649	12.841	10.715
Gunnar Farnerback	157,980	13.018	0.342

Horn & Schunck has more direction estimation and takes less time than Lucas Kanade. However, its processor usage is greater than Lucas Kanade because this method has two terms: the data attachment term, which is given by the optical flow constraint, and the consistency term. In addition, this method assumes that frame noise is expected to be a Gaussian distribution. This term implies that the method can be affected by noise. Therefore, Horn & Schunck used a filter-based approach.

Gunnar Farnerback produces more direction estimations than any other method. The CPU usage is also higher than the others, but this method takes a short time to process. It is because Gunnar Farnerback applies quadratic polynomials to estimating the movement between the matrix values of frames. This can be processed effectively by applying the polynomial expansion transformation. Gunnar Farnerback deals with quadratic polynomials modeled as local signals in a local coordinate system.

4.4. **Combined approach: YOLOFOR.** At this stage, we tested the proposed method using various configurations of YOLO and optical flow techniques. Figure 3 shows a test of the YOLOFOR method on the attack dataset on drones. Each combination of YOLO versions and optical flow methods is evaluated in this experiment.

As shown in Table 2, the most effective configuration for making YOLOFOR is YOLOv5 + Lucas Kanade, where the mAP results are 0.633, and the resulting direction estimation is 3,476. Note that each direction estimation value in Table 2 is a mean of values from all frames in the datasets. YOLOv5 uses novel methods to draw the bounding boxes using auto anchors, so the mAP value in this experiment for YOLOv5 was 0.633. It shows that YOLOv5 achieved a high precision value in detecting objects. Lucas Kanade can accurately show direction estimation in frames in terms of visualization. Lucas Kanade's direction estimation has a smaller number of quivers than other estimations. This lets the quivers focus more on the object and shows the object's movement.

YOLOFOR uses YOLOv5's performance to detect moving objects with blurred images. YOLOv5 can handle this case because it has an auto anchor to determine detected objects. This ability is not available in other versions of YOLO. Lucas Kanade also provides the most accurate performance for estimating the direction of objects. Lucas Kanade used the grayscaled image to estimate the direction. This method works by speculating the

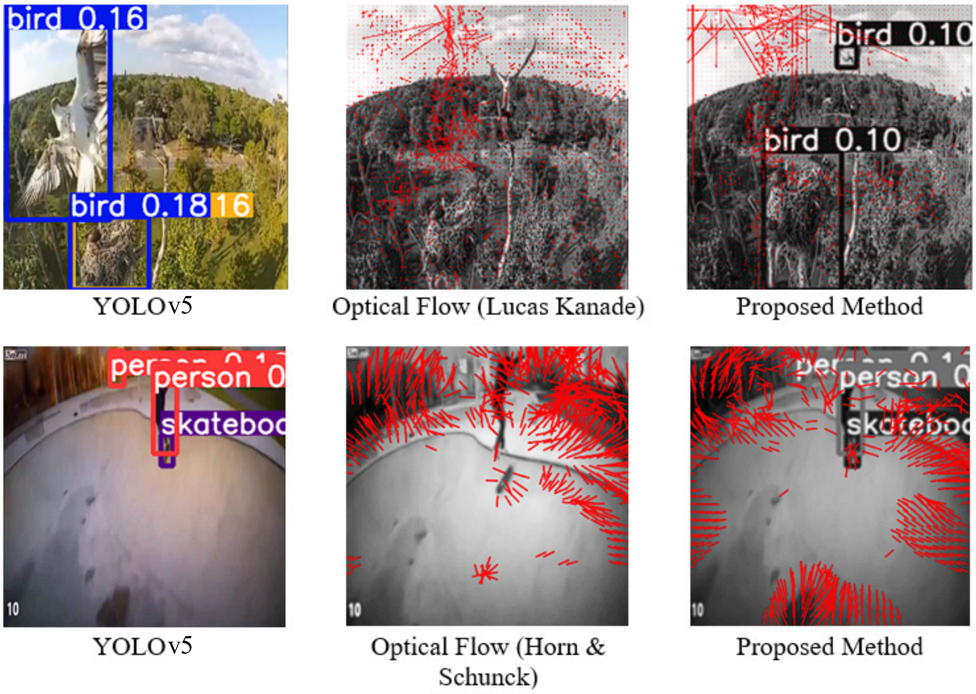

FIGURE 3. Visualization of the proposed method

TABLE 2. Results on combination of object detection and optical flow methods

Method	mAP	Direction estimation
YOLOv3 + Lucas Kanade	0.462	3,301
YOLOv4 + Lucas Kanade	0.611	3,370
YOLOv5 + Lucas Kanade	**0.633**	**3,476**
YOLOv3 + Horn & Schunck	0.465	87,913
YOLOv4 + Horn & Schunck	0.612	88,696
YOLOv5 + Horn & Schunck	0.631	89,166
YOLOv3 + Gunnar Farnerback	0.467	155,431
YOLOv4 + Gunnar Farnerback	0.616	144,365
YOLOv5 + Gunnar Farnerback	0.637	170,106

direction of the movement of objects in images. In addition, it can explain local changes in image intensity. It produces more accurate direction estimation than Horn & Schunck and Gunnar Farnerback.

5. **Conclusion and Future Work.** This work proposes YOLOFOR, an attack detection method to support drone forensic investigations. The YOLOFOR method combines YOLO and optical flow to detect and estimate the object and its direction suspected of attacking the drone. YOLOFOR can assist investigators in determining drone incident root causes, especially in an attacked drone case.

In this paper, we find that the YOLOv5 and Lucas Kanade achieve high performance in object detection and direction estimation on attacked drone datasets. As a result, YOLOFOR has a high mAP score of 0.633 and the least number of average direction estimations of 3,476. However, the mAP score may be improved using more recent YOLO variants. The proposed method also can be implemented on drones to detect real-time attacks.

REFERENCES

[1] T. Rahman, M. A. L. Siregar, A. Kurniawan, S. Juniastuti and E. M. Yuniarno, Vehicle speed calculation from drone video based on deep learning, *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, pp.229-233, 2020.

[2] B. Bender, M. E. Atasoy and F. Semiz, Deep learning-based human and vehicle detection in drone videos, *2021 6th International Conference on Computer Science and Engineering*, pp.446-450, 2021.

[3] S. Wu, C. Du, H. Chen and N. Jing, Coarse-to-fine UAV image geo-localization using multi-stage Lucas-Kanade networks, *2021 2nd Information Communication Technologies Conference (ICTC)*, Nanjing, China, pp.220-224, 2021.

[4] A. L. P. S. Renduchintala, A. Albehadili and A. Y. Javaid, Drone forensics: Digital flight log examination framework for micro drones, *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp.91-96, 2017.

[5] D. Pedro, J. P. Matos-Carvalho, J. M. Fonseca and A. Mora, Collision avoidance on unmanned aerial vehicles using neural network pipelines and flow clustering techniques, *Remote Sensing*, vol.13, no.13, 2021.

[6] A. S. Editya, T. Ahmad and H. Studiawan, Direction estimation of drone collision using optical flow for forensic investigation, *2022 10th International Symposium on Digital Forensics and Security*, pp.1-6, 2022.

[7] A. Mishra and S. Panda, Drone detection using YOLOV4 on images and videos, *2022 IEEE 7th International Conference for Convergence in Technology (I2CT)*, pp.1-4, 2022.

[8] U. Nepal and H. Eslamiat, Comparing YOLOv3, YOLOv4 and YOLOv5 for autonomous landing spot detection in faulty UAVs, *Sensors*, vol.22, no.2, 2022.

[9] O. Sahin and S. Ozer, YOLODrone: Improved YOLO architecture for object detection in drone images, *The 44th International Conference on Telecommunications and Signal Processing*, pp.361-365, 2021.

[10] N. Ibrahim, W. M. D. Wan Zaki, A. Hussain and M. Marzuki Mustafa, Optical flow improvement towards real time and natural rigid human motion estimation, *2009 IEEE International Conference on Signal and Image Processing Applications*, pp.322-325, 2009.

[11] H. R. Alsanad, A. Z. Sadik, O. N. Ucan, M. Ilyas and O. Bayat, YOLO-V3 based real-time drone detection algorithm, *Multimedia Tools Appl.*, vol.81, no.18, pp.26185-26198, 2022.

[12] J. K. D. Lagman, A. B. Evangelista and C. C. Paglinawan, Unmanned aerial vehicle with human detection and people counter using YOLO v5 and thermal camera for search operations, *2022 IEEE International Conference on Automatic Control and Intelligent Systems*, pp.113-118, 2022.

[13] F. Yan and Y. Xu, Improved target detection algorithm based on YOLO, *2021 4th International Conference on Robotics, Control and Automation Engineering (RCAE)*, pp.21-25, 2021.

[14] Z. Zhang, X. Lu, G. Cao, Y. Yang, L. Jiao and F. Liu, ViT-YOLO: Transformer-based YOLO for object detection, *IEEE/CVF International Conference on Computer Vision Workshops*, pp.2799-2808, 2021.

[15] P. Lu, Y. Ding and C. Wang, Multi-small target detection and tracking based on improved YOLO and SIFT for drones, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.205-224, 2021.

[16] S. Minaeian, J. Liu and Y.-J. Son, Effective and efficient detection of moving targets from a UAV's camera, *IEEE Transactions on Intelligent Transportation Systems*, vol.19, no.2, pp.497-506, 2018.

[17] O. D. Mora Granillo and Z. Zamudio Beltrán, Real-time drone (UAV) trajectory generation and tracking by optical flow, *2018 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)*, pp.38-43, 2018.

[18] Y. Joshi and H. Mewada, Object tracking in occlusion and contrast conditions using patch-wise sparse method, *International Journal of Intelligent Engineering and Systems*, vol.13, no.5, pp.295-306, 2020.

[19] C. Kumar B., R. Punitha and Mohana, YOLOv3 and YOLOv4: Multiple object detection for surveillance applications, *2020 3rd International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, pp.1316-1321, 2020.

[20] H. A. Kadhim and W. A. Araheemah, A method to improve corner detectors (Harris, Shi-Tomasi & FAST) using adaptive contrast enhancement filter, *Periodicals of Engineering and Natural Sciences*, vol.8, no.1, 2020.

[21] F. T. Ishii, F. C. Flores and L. Rittner, Tensorial Lucas-Kanade: An optical flow estimator based on tensorial color representation and tensorial algebra, *2018 IEEE Symposium on Computers and Communications (ISCC)*, pp.00633-00639, 2018.

[22] J. Kanawathi, S. S. Mokri, N. Ibrahim, A. Hussain and M. M. Mustafa, Motion detection using Horn Schunck algorithm and implementation, *2009 International Conference on Electrical Engineering and Informatics*, vol.1, pp.83-87, 2009.

[23] C. Yuan, Z. Liu and Y. Zhang, Vision-based forest fire detection in aerial images for firefighting using UAVs, *2016 International Conference on Unmanned Aircraft Systems*, pp.1200-1205, 2016.