# AFOM: ADVANCED FLOW OF MOTION DETECTION ALGORITHM FOR DYNAMIC CAMERA VIDEOS

1st Ifeoluwapo Aribilola*
*Software Research Institute, Technological University of the Shannon: Midlands Midwest, Athlone, Ireland*
i.aribilola@research.ait.ie

2nd Mamoona Naveed Asghar
*School of Computer Science, National University of Ireland*
Galway, Ireland
mamoona.asghar@nuigalway.ie

3rd Nadia Kanwal
*School of Computing and Mathematics, University of Keele*
United Kingdom
n.kanwal@keele.ac.uk

4th Mohammad Samar Ansari
*Faculty of Science and Engineering, University of Chester*
United Kingdom
m.ansari@chester.ac.uk

5th Brian Lee
*Software Research Institute, Technological University of the Shannon: Midlands Midwest, Athlone, Ireland*
blee@ait.ie

2022 33rd Irish Signals and Systems Conference (ISSC) | 978-1-6654-5227-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISSC55427.2022.9826141

***Abstract*—The surveillance videos taken from dynamic cameras are susceptible to multiple security threats like replay attacks, man-in-the-middle attacks, pixel correlation attacks etc. Using unsupervised learning, it is a challenge to detect objects in such surveillance videos, as fixed objects may appear to be in motion alongside the actual moving objects. But despite this challenge, the unsupervised learning techniques are efficient as they save object labelling and model training time, which is usually a case with supervised learning models. This paper proposes an effective computer vision-based object identification algorithm that can detect and separate stationary objects from moving objects in such videos. The proposed Advanced Flow Of Motion (AFOM) algorithm takes advantage of motion estimation between two consecutive frames and induces the estimated motion back to the frame to provide an improved detection on the dynamic camera videos. The comparative analysis demonstrates that the proposed AFOM outperforms a traditional dense optical flow (DOF) algorithm with an average increased difference of 56% in accuracy, 61% in precision, and 73% in pixel space ratio (PSR), and with minimal higher object detection timing.**

***Index Terms*—Accuracy, Computer Vision, Dense Optical Flow (DOF), Object Detection, Pixel Space Ratio (PSR)**

## I. Introduction

Object detection is a computer vision technique that identifies objects at their respective location in a video or image. Object detection can be implemented by both supervised and unsupervised machine learning techniques [1]. Supervised machine learning involves the training on a set of data to master the attributes of an object for later identification [1] and is considered to have accurate detection. Whereas in unsupervised learning, there is no such training i.e., the labelled datasets, therefore, outcomes are unknown on real-time data processing, thus, considered to have low detection accuracy. Furthermore, better accuracy output of the supervised learning models are dependent on fairly large labelled datasets for training, which make it a less preferred option when using supervised learning on real-time data. However, unsupervised learning do not require labelling and training of datasets, making it a low computational cost algorithm for real-time motion detection for dynamic camera videos [2].

The object detection algorithms, specifically in videos, broadly classify the objects into two groups i.e., the foreground (FG) and the background (BG). The FG contains objects in motion, such as cars, humans, or animals, while the BG mostly consists of objects that are in a fixed position. These videos can either be captured with static (CCTV, fixed) or dynamic cameras (drones, unmanned aerial vehicle (UAV), dashboard, pan-tilt-zoom (PTZ) or hand-held).

The use of unsupervised machine learning techniques to detect FG objects [3], [4] in the static camera videos are easy and accurate, because these objects are the only objects in motion; thus separating their information from the BG is straightforward. A dynamic camera, however, presents a challenge for FG detection because the BG also appears to be moving, thus causing inaccurate detection.

To overcome this, the optical flow (OF) method has been widely used with more focus on the dense optical flow (DOF) in the past decade. The researchers [5] analysed the benefits of DOF, hence proving its relevancy even for the recent dynamic applications. Despite the implementation of the DOF, there was little or no improvement in the object detection accuracy for dynamic camera videos [6]. As a proof, the

This work is part of doctoral degree research funded by the Presidential Doctoral Scheme (PDS) of the Technological University of the Shannon: Midlands, Midwest, Athlone, Ireland.

visual results using DOF over dynamic cameras are shown in the section 4 of this paper. However, most of the existing studies (Section 2) focuses on combining DOF with other algorithms for accurate object detection. In this paper, we have proposed an efficient unsupervised object detection algorithm i.e., Advanced Flow Of Motion (AFOM) by applying motion estimation and infusing the results back to the frame to mask out, and to enhance the accuracy of the detected moving objects.

Following an extensive review of the literature related to object detection in dynamic camera videos, the following research questions (RQs) are identified as a target for the research presented in this paper:

**RQ1:** Why is it necessary to use an accurate object detection algorithm for the videos captured with dynamic camera devices?

**RQ2:** What are the benefits of implementing a low computational unsupervised object detection algorithm for low computational camera devices?

To address these RQs, this paper presents an unsupervised machine learning algorithm (AFOM) for surveillance videos captured with dynamic cameras. The research contributions of this paper are:

- The experiments illustrate the effectiveness of AFOM algorithm in a visual representation with high accuracy and low pixel space ratio (PSR) value;
- The high structural similarity index (SSIM) value of tested videos affirms that AFOM based detection produces the similar shape of the object; and
- Results prove that AFOM has a low computational cost because of the minimal variation in detection timing compared with DOF, therefore justifying its high efficiency with constraint devices.

The remainder of the paper is organized as follows: Section 2 describes the related work on object detection techniques. Section 3 presents the adopted methodology of implementing AFOM for videos captured with dynamic camera devices. Section 4 illustrates the comparative visual results and performance analysis to show the efficacy of AFOM for the object detection in moving camera videos. Section 5 discusses the summary and conclusion of this paper.

## II. Related Work

This section reviews the existing literature on unsupervised learning based object detection techniques within dynamic camera videos.

Object detection using unsupervised learning in videos captured with dynamic cameras is challenging [7] due to the simultaneous movement of the cameras and the foreground (FG) objects (non-stationary objects in the video) during recording. There are several different techniques such as compensation method [8], trajectory classification [9], background subtraction [10], robust principal component analysis (RPCA) [11], and motion estimation [12] that might recognize objects in a dynamic camera video [13].

Motion estimation which determines motion vectors of objects from one frame to another has been widely adopted for identifying moving objects in dynamic camera videos. Motion estimation can be achieved using kalman filter (KF) [14], block matching (BM) [15], optical flow (OF) [3].

KF estimates the object's states based on observations or measurements, to determine if there is a change in the state. This algorithm is divided into two phases; the predicted phase which calculates the prediction state of the object and the update phase, to calculate the difference between the true measurement and the previous estimated measurement [14], [16], [17].

BM divides video frames into blocks and the best-matching blocks are selected from a region of the previous frame for each block in the current frame [15]. Motion vectors are estimated for each block independently [18].

OF method, which detects the motion of an object or the camera across two contiguous frames is a popular algorithm to identify objects within a video taken from a dynamic camera. OF is categorised into two classes i.e., sparse optical flow (SOF) and dense optical flow (DOF). SOF measures the motion vector of selected pixels or features of the objects, and demand some pre-processing method to obtain these features on the object, such as using a corner detector algorithm. However, this implies SOF cannot be implemented in isolation. Hence, different researchers [19]–[21] have implemented these two techniques in a combination for object detection.

DOF for this purpose, as developed in [3] used polynomial interpolation to estimate the motion between two frames to measure the motion vectors of each object's pixels, which improves the implementation of SOF itself. DOF provides an improved output when using a static camera [5] however with no significant improvement for dynamic cameras [6]. Due to this reason, researchers have combined DOF with various algorithms for dynamic cameras [22]–[26], which unfortunately increases computational cost during implementation.

Thus, rather than combining motion estimation with a supervised learning technique, [27], [28] this study proposes a low computational unsupervised object detection (AFOM) by applying motion estimation and frame fusion techniques. The proposed AFOM algorithm increases the detection accuracy for dynamic camera videos with nominal increase in computational cost.

## III. The Proposed Algorithm

This section presents the adopted methodology by considering step-by-step implementation of the proposed AFOM algorithm.

AFOM algorithm was developed for a precise detection of moving objects in videos captured from dynamic cameras for real-time processing. AFOM does not require datasets for object classification and training for accurate detection, thus, reducing data pre-processing and implementation cost.

### A. AFOM Implementation steps

The AFOM implementation steps, as given in "Fig. 1" are as follows:

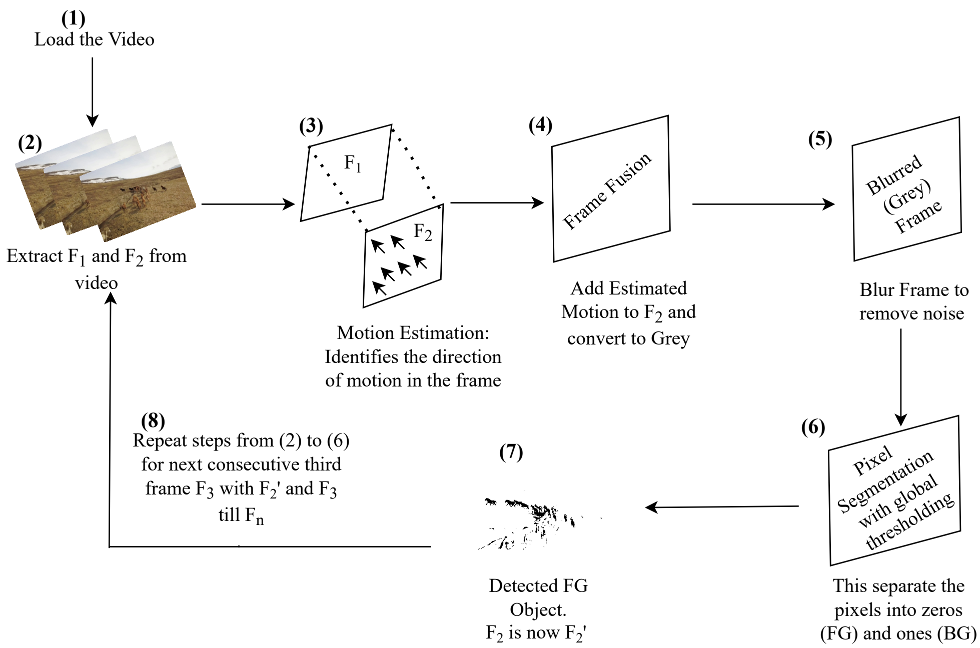



Fig. 1. AFOM implementation steps.

(1) Load the video file from its path and read frame by frame, from the first frame to the last frame, say $F_1$ to $F_n$.
(2) Read initial two (02) consecutive frames $F_1$ and $F_2$ from the loaded video file.
(3) Performs a motion estimation by extracting the coordinate vectors of the motion between these two consecutive frames using the Farneback algorithm [3].
(4) Add the extracted motion back into the frame $F_2$ to detect the object moving in the frame using equation (1) where the $x$ value varies, $y$ value is constant at 0.5 and The $z$ value is constant at 0.

$$Fusion = x \cdot frame + y \cdot estimated\_motion + z \quad (1)$$

(5) Gaussian blur is thereafter used on the output of Frame $F_2$, hence producing a noiseless grey-scaled image suitable for pixel segmentation.
(6) Pixel segmentation is applied using global thresholding to separate FG objects' pixels from BG objects' pixels.
(7) After steps (2) to (6) till the end of file, frame $F_2$ has become $F_2'$, and eventually frame $F_n$ will become $F_n'$

$$F_2' = F_2 + F_1 \quad (2)$$

(8) Repeat steps from (2) to (6) for next consecutive frame ($F_3$) of the video with $F_2'$, and continue the process until the $F_n$.

$$F_n' = F_n + F_{n-1} \quad (3)$$

The pseudo-code describing the implementation of AFOM is given in Algorithm 1.

**Algorithm 1:** Object detection with AFOM

**Input:** Dynamic Camera Video
`/* FG (Moving Object) detection */`
**Output:** Video with FG-Extraction
**Input:** Load Video from path
**while** $video == True$ **do**
  $ret, F1 \leftarrow video.read()$;
  $ret, F2 \leftarrow video.read()$;
  $motion\_estimation \leftarrow calcOpticalFlowFarneback()$;
  $magnitude, angle \leftarrow cartToPolar(motion\_flow)$;
  $motion\_extracted \leftarrow normalize(magnitude, angle)$;
  $fusion \leftarrow add\ motion\_extracted\ to\ F2$;
  $F2\_gray \leftarrow cvtColor(fusion)$;
  $F2\_blur \leftarrow GaussianBlur(F2\_gray)$;
  $ret, thresh \leftarrow apply\ global\ thresholding$;
  `/* thresh contains zeros and one pixels. */`
  `/* The zeros are the FG while the ones are the BG. */`
  **if** $cv2.waitKey(27)\ \&\ 0xFF == ord\ ('q')$ **then**
    $break$
  **end**
**end**
**return** (0);
$video.release()$
**Output:** Pixel Segmented Video: Where zeros (black) are the FG and ones (white) are the BG

## IV. The Experiment

For the purpose of assessment, the AFOM algorithm was implemented in Python with OpenCV. The system specifications are listed as follows: Intel(R) Core (TM) i7-10510U CPU@1.80GHz 2.30GHz processor, 16GB RAM, 64bit Oper-

TABLE I
THE CHARACTERISTICS OF THE EXPERIMENTAL TESTED VIDEOS

| S/N | Video file | Background type | Video size (MB) | Resolution | Video duration (sec) | Frame rate (FPS) | Frame count |
|---|---|---|---|---|---|---|---|
| 1 | Horse Moving | Dynamic | 4.29 | 860 x 484 | 5 | 23 | 126 |
| 2. | Dashboard_Cam | Dynamic | 6.61 | 1280 x 720 | 6 | 24 | 144 |
| 3. | Safari Moving | Dynamic | 6.00 | 1280 x 720 | 5 | 23 | 120 |
| 4. | Mall | Dynamic | 1.69 | 1280 x 720 | 5 | 23 | 117 |
| 5. | Traffic | Dynamic | 3.66 | 1920 x 1080 | 5 | 46 | 234 |

ating system, x64-based processor system type, Intel(R) UHD Graphics.

This evaluation was performed with a dataset of five (05) publicly available dynamic camera videos, available at Pixabay [29] and Pexels [30] web-pages). All test videos have various characteristics in terms of colour, motion, and spatial information. The characteristics of these test videos are given in "Table I" with the video speed, measured in frames per second (FPS).

The state-of-the-art DOF [3] has been widely deployed by previous studies as an unsupervised learning technique. Therefore, we have compared the proposed AFOM with DOF for different type of evaluation in this paper.

### A. Relative Evaluation of the AFOM and DOF

This sub-section compares the visual results taken with AFOM and DOF algorithms.

*1) Visual Evaluation:* The results of the object detection using AFOM and DOF appear in "Table II". Comparing the results, AFOM leads to a coherent detection of the FG objects in the tested videos, unlike for DOF. This reveals that AFOM performs an accurate detection.

*2) Pixel Space Ratio (PSR) Evaluation:* The PSR was calculated as the ratio (percentage) of the total pixels to the detected FG pixels within the videos. Comparative analysis of the PSR results of both algorithms is presented in "Fig. 2".

"Fig. 2" indicates that DOF results is more, by wrongly identifying more BG pixels as FG pixels. In other words, DOF's false positive rate is higher than that of AFOM.

*3) Accuracy and Precision Evaluation:* The accuracy and precision was calculated and the results are shown in "Table III". From the result, AFOM demonstrated higher accuracy and greater precision than DOF [3], indicating the effectiveness of AFOM in detecting moving objects within dynamic camera videos.

*4) Structural Similarity Index (SSIM) Evaluation:* The SSIM was calculated using equation (4) and the results are presented in the "Fig. 3". It is noted that SSIM ranges from 0 to 1 (0 for low and 1 for high similarity index)

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c1)(2\sigma_{xy} + c2)}{(\mu_x^2\mu_y^2 + c1)(\sigma_x^2 + \sigma_y^2 + c2)}, \tag{4}$$

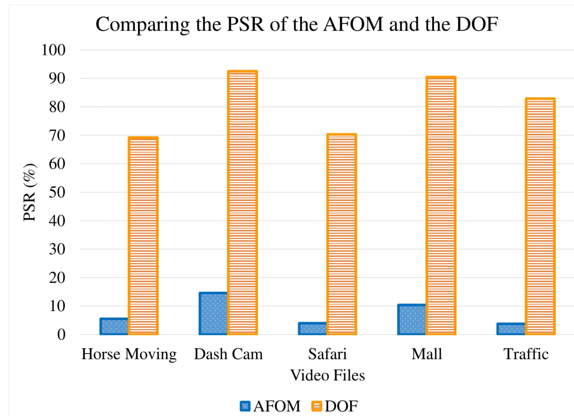



Fig. 2. PSR comparison of AFOM and DOF

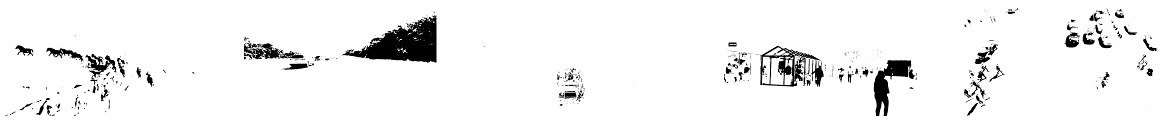
where x = original tested videos, y = encrypted tested videos, $\mu_x$ = average of x, $\mu_y$ = average of y, $\sigma_x^2$ = variance of x, $\sigma_y^2$ = variance of y, $\sigma_{xy}$ = covariance of x and y, c1 = $(K_1L)^2$ c2= $(K_2L)^2$, L = dynamic range, $(K_1)$ = 0.01, $(K_2)$ = 0.03

"Fig. 3" shows that the SSIM values for the AFOM are 0.5 above which is closer to 1 and significantly better than the DOF values [3]. Results indicate that the structure/shapes of the detected objects by AFOM are matched with the objects present in the original tested videos. The SSIM values for the DOF were very low, indicating that DOF poorly recognized the particular shape of the detected object. These SSIM results provide evidence that the AFOM is competent at detecting moving objects within dynamic camera videos. The visual results given in "Fig. 2" also confirms the structural similarity of the objects detected with AFOM.

### B. Computational Analysis

"Fig. 4" compares the computational cost, which is defined as the time (in $\mu$s) required to perform object detection on the test videos. From "Fig. 4", AFOM took longer to detect objects in the tested videos than the original DOF [3]. However, across the videos tested, in absolute timing terms, the differences in detection times are negligible and should be weighed against the significantly improved accuracy and precision resulting

TABLE II
VISUAL REPRESENTATION COMPARING THE EXISTING DOF AND PROPOSED AFOM ON DYNAMIC VIDEOS OBJECT DETECTION

| | Horse_moving Dashboard_Cam Safari Mall Traffic |
|---|---|
| Input image |  |
| DOF_Detection |  |
| AFOM_Detection |  |

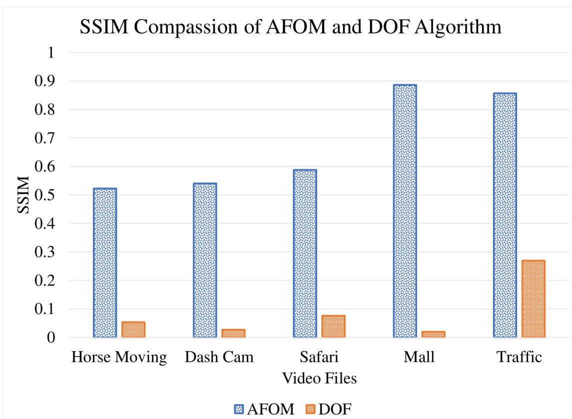



Fig. 3. SSIM comparison of AFOM and DOF

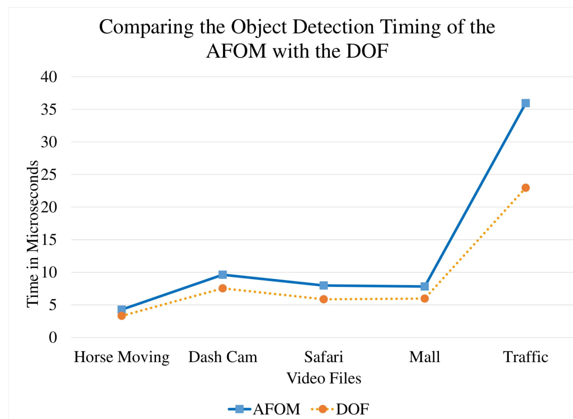



Fig. 4. Comparing AFOM and DOF detection timing

TABLE III
ACCURACY FOR AFOM AND DOF ALGORITHMS

| | Accuracy (%) | | Precision (%) | |
|---|---|---|---|---|
| Video file | AFOM | DOF | AFOM | DOF |
| Horse_moving | 84.615 | 30.769 | 93.333 | 17.647 |
| Dashboard Cam | 63.636 | 20.000 | 66.667 | 12.500 |
| Safari | 90.000 | 22.222 | 87.500 | 16.667 |
| Mall | 85.000 | 15.000 | 81.250 | 12.500 |
| Traffic | 85.294 | 41.176 | 86.667 | 50.000 |

from using AFOM, see “Table III” as well as the SSIM result in “Fig. 3”. Thus, the trade-off is in the favour of AFOM.

### *C. Comparative Analysis*

To further highlight the efficacy of AFOM for mobile camera videos, a comparison was also made with existing studies that implemented motion estimation with unsupervised learning. The object detection accuracy was used as a comparison criterion. The results in “Table IV” show that DOF has the lowest value of accuracy, which, hence, confirms the inadequacy of DOF for object detection within dynamic camera videos. Despite good accuracy results in the studies [5], [24], [25], which combined DOF with other techniques, the proposed AFOM has the highest accuracy, at around 82%.

## V. CONCLUSION

In this paper, an effective motion detection technique using AFOM algorithm is proposed to identify the moving objects in the videos captured by mobile or moving cameras. AFOM algorithm has been implemented by performing motion estimation and motion fusion to the frames to achieve high accuracy in the detection of FG objects. Performance analysis confirmed that AFOM also exhibits more precise detection than the state-of-the-art DOF, while remained competitive with the DOF in the computational time to perform its operations. The comparative analysis in “Table IV”, where DOF

TABLE IV
ACCURACY COMPARISON BETWEEN AFOM ALGORITHM WITH EXISTING ALGORITHMS

| Existing Techniques | Proposed Model | Method | Accuracy (%) |
|---|---|---|---|
| [3] Initial Method (2003) | Dense Optical Flow (DOF) | Motion vector of camera and objects | 25.83 |
| [25] (2020) | Dense optical flow based background subtraction technique | Homography matrix, single Gaussian and DOF | 52.49 |
| [5] (2019) | Integration of Optical Flow and Action Recognition | Shuffled images with DOF for Recognition | 59.55 |
| [24] (2013) | Anticipated geometry pixel deviation | DOF and fundamental matrix | 65.03 |
| Proposed (2022) | AFOM | Motion estimation and Frame fusion, | 81.71 |

is integrated with other techniques for improving detection accuracy also shows the highest accuracy of AFOM algorithm. Nevertheless, the results reveal the AFOM algorithm to be effective and reasonably efficient for the constraint dynamic camera devices for surveillance videos.

## REFERENCES


[1] A. R. Pathak, M. Pandey, and S. Rautaray, "Application of Deep Learning for Object Detection," *Procedia Computer Science*, vol. 132, no. Iccids, pp. 1706–1717, 2018. [Online]. Available: https://doi.org/10.1016/j.procs.2018.05.144

[2] D. Cavaliere, V. Loia, and S. Senatore, "Towards an ontology design pattern for uav video content analysis," *IEEE Access*, vol. 7, pp. 105 342–105 353, 2019.

[3] G. Farneb, "Two-frame motion estimation based on polynomial expansion," *Lecture Notes in Computer Science*, vol. 2749, no. 1, pp. 363–370, 2003.

[4] J. Janai, F. Guney, A. Ranjan, M. Black, and A. Geiger, "Unsupervised learning of multi-frame optical flow with occlusions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[5] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, and M. J. Black, "On the integration of optical flow and action recognition," in *Pattern Recognition*, T. Brox, A. Bruhn, and M. Fritz, Eds. Cham: Springer International Publishing, 2019, pp. 281–297.

[6] L. Kurnianggoro, A. Shahbaz, and K.-H. Jo, "Dense optical flow in stabilized scenes for moving object detection from a moving camera," in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, Oct 2016, pp. 704–708.

[7] M. N. Chapel and T. Bouwmans, "Moving objects detection with a moving camera: A comprehensive review," *Computer Science Review*, vol. 38, p. 100310, 2020. [Online]. Available: https://doi.org/10.1016/j.cosrev.2020.100310

[8] C. H. Yeh, C. Y. Lin, K. Muchtar, H. E. Lai, and M. T. Sun, "Three-pronged compensation and hysteresis thresholding for moving object detection in real-time video surveillance," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4945–4955, 2017.

[9] X. Yin, B. Wang, W. Li, Y. Liu, and M. Zhang, "Background subtraction for moving cameras based on trajectory-controlled segmentation and label inference," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 10, pp. 4092–4107, 2015.

[10] H. Sajid and S.-C. S. Cheung, "Background subtraction for static amp; moving camera," in *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4530–4534.

[11] S. Javed, A. Mahmood, J. Dias, and N. Werghi, "CS-RPCA: Clustered sparse RPCA for moving object detection," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 3209–3213. [Online]. Available: https://ieeexplore.ieee.org/document/9190734/

[12] A. Sarrafi, Z. Mao, C. Niezrecki, and P. Poozesh, "Vibration-based damage detection in wind turbine blades using phase-based motion estimation and motion magnification," *Journal of Sound and Vibration*, vol. 421, pp. 300–318, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022460X18300725

[13] A. Mohanty and S. Sanjivani, "A survey on moving object detection using background subtraction methods in video," *International Journal of Computer Applications*, vol. 1, no. 2, pp. 5–10, 2015.

[14] M. Bersani, M. Vignati, S. Mentasti, S. Arrigoni, and F. Cheli, "Vehicle state estimation based on kalman filters," in *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*, July 2019, pp. 1–6.

[15] E. Cuevas, D. Zaldívar, M. Pérez-Cisneros, H. Sossa, and V. Osuna, "Block matching algorithm for motion estimation based on artificial bee colony (abc)," *Applied Soft Computing*, vol. 13, no. 6, pp. 3047–3059, 2013, swarm intelligence in image and video processing. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494612004401

[16] D. Lunni, G. Giordano, E. Sinibaldi, M. Cianchetti, and B. Mazzolai, "Shape estimation based on kalman filtering: Towards fully soft proprioception," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, April 2018, pp. 541–546.

[17] M. Lin, J. Yoon, and B. Kim, "Self-driving car location estimation based on a particle-aided unscented kalman filter," *Sensors*, vol. 20, no. 9, p. 2544, Apr 2020. [Online]. Available: http://dx.doi.org/10.3390/s20092544

[18] R. Szeliski, "Motion Estimation," pp. 443–482, 2022.

[19] B. Leibe, K. Schindler, and L. Van Gool, "Coupled detection and trajectory estimation for multi-object tracking," *Proceedings of the IEEE International Conference on Computer Vision*, 2007.

[20] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, "Optical flow estimation for flame detection in videos," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2786–2797, 2013.

[21] R. Ke, Z. Li, J. Tang, Z. Pan, and Y. Wang, "Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 54–64, 2019.

[22] Y. Xiang, F. Wang, L. Wan, N. Jiao, and H. You, "OS-Flow: A robust algorithm for dense optical and SAR image registration," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6335–6354, 2019.

[23] T. Che, Y. Zheng, Y. Yang, S. Hou, W. Jia, J. Yang, and C. Gong, "SDOF-GAN: Symmetric Dense Optical Flow estimation with Generative Adversarial Networks," *IEEE Transactions on Image Processing*, vol. 30, pp. 6036–6049, 2021.

[24] J. Maier and M. Humenberger, "Movement detection based on dense optical flow for unmanned aerial vehicles," *International Journal of Advanced Robotic Systems*, vol. 10, pp. 1–11, 2013.

[25] A. Kushwaha, A. Khare, O. Prakash, and M. Khare, "Dense optical flow based background subtraction technique for object segmentation in moving camera environment," *IET Image Processing*, vol. 14, no. 14, pp. 3393–3404, aug 2020. [Online]. Available: https://digital-library.theiet.org/content/journals/10.1049/iet-ipr.2019.0960

[26] D. Gibson and M. Spann, "Robust optical flow estimation based on a sparse motion trajectory set," *IEEE Transactions on Image Processing*, vol. 12, no. 4, pp. 431–445, 2003.

[27] W. Bao, W.-S. Lai, X. Zhang, Z. Gao, and M.-H. Yang, "Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 933–948, March 2021.

[28] M. Cheng, K. Cai, and M. Li, "Rwf-2000: An open large scale video database for violence detection," in *2020 25th International Conference on Pattern Recognition (ICPR)*, Jan 2021, pp. 4183–4190.

[29] "Pixabay Webpage." [Online]. Available: https://pixabay.com/videos/

[30] "Pexels Webpage." [Online]. Available: https://www.pexels.com/