Fine-Tuning GMM and Total Pixel-Based Drowsiness Detection: A Strategy for Detection Open and Closed Eye

Elindra Ambar Pambudi[*1], Dion Romodhon[2], Ermadi Satriya Wijaya[3]
[1,2,3] Informatics Engineering Study Programs, Engineering and Science Faculty,
Muhammadiyah University of Purwokerto, Indonesia
E-mail: [*1]**elindraambarpambudi@ump.ac.id**, [2]dion.romodon@gmail.com,
[3]ermadi.wijaya@gmail.com

Abstract

Fatigue driving represents a substantial and often unrecognized risk in traffic accidents. A technique that may be employed involves the detection of open and closed eyes. The research on open and closed eye identification use approaches based on haar cascade and complete pixel analysis. Our proposed method employs an adaptive thresholding technique is implemented right before total pixel process. The processing steps involve the application of haar cascade, adaptive thresholding, fine-tuning of Gaussian Mixture Models (GMM), and the calculation of the total pixel count in the image that is utilized to identify the state of the eye using thresholding. The results from Fine-Tuning GMM thresholding for the left and right eyes are as follows: MSE values of 7.02 and 7.96, and PSNR values of 39.24 and 39.21, respectively. The results derived from fine-tuning are comparable to those obtained using Otsu's method.

Keywords — *Fine-Tuning, GMM, Hyperparameter, Eye Detection, Adaptive Thresholding*

1. INTRODUCTION

The National Highway road Safety Administration indicates that in 2016, 37,461 people lost their lives and 3,144,000 were injured in 7,277,000 car accidents that occurred in the US. About 20%-30% of accidents that occur on the road are caused by drivers who are fatigued. So, sleepy drivers pose a serious and sinister threat to road accidents.

Systems for detecting fatigue in drivers, utilizing eye state analysis, have become a significant area of research in recent years [1], [2], [3], [4]. The research to detect drowsiness through the analysis of the eyes and face. Methods predominantly employ deep learning techniques [5], [6], [7], visual features [8], Viola-Jones and Haar cascade algorithms [9], [10], [11], facial texture descriptors [12], and Laplacian operations [13].

In our research will focus on the use of haar cascade. This research applies the haar cascade approach in detecting and extracting eye from face. This project will employ an alternative binarization procedure for open and closed eye identification compared to research [14]. This research aims to enhance perform of identify open and closed eye based on total pixel through utilizing adaptive thresholding.

Image thresholding is important for identification and segmentation object. The establishment of an effective threshold algorithm significantly enhances the accuracy of object detection[15], [16]. The image binarization stage is a critical component in the pixel-based detection of open or closed eyes, as the total pixel count is derived from the binary values of the segmentation results [17]. The research emphasizes enhancing segmentation quality through the application of Gaussian Mixture Models (GMM).

GMM serves as a viable option in the image segmentation process through the application of a clustering approach. GMM serves as a versatile method for both segmentation [18] and classification [19]. Certain studies [19], [20] indicate that GMM, when applied in the context of thresholds, yields satisfactory outcomes.

The critical aspect of GMM for segmentation lies in the effective tuning parameter [21], [22]. Methods that can be employed through hyper-parameter tuning for enhancing the efficiency of the model [23]. In the context of segmentation, the application of fine-tuning significantly enhances the performance of algorithms such as KNN [24], random forests [25]. This study will employ this fine-tuning method to optimize the GMM parameters effectively.

Our contribution to this research is the implementation of fine-tuning Gaussian Mixture Models (GMM) to enhance the quality of picture binarization within the GMM technique, specifically used to eye localization, hence facilitating the subsequent total pixel processing stage effectively. This paper will be organized as follows: Session II of the research methodology will focus on data collection, followed by the proposed implementation of fine-tuning Gaussian Mixture Models (GMM) in the eye image segmentation phase. Session III will present the results of this process along with the evaluation stage of the research proposal, culminating in the conclusion in Section VII.

2. RESEARCH METHOD

As shown in Figure 1, the suggested approach includes many stages for strategic thresholding in closed and open-eye detection. Initially, frame splitting is conducted on the video dataset, followed by the grayscaling process. This is succeeded by the retrieval of the eye ROI, then preprocessing with a Gaussian filter. Next, an adaptive threshold is applied to extract the foreground of the eye image. The morphological process follows, and the final step involves calculating the total pixels of the eye image. In the foreground of the eye image, a morphological process is applied, and the final step involves calculating the total pixel count.

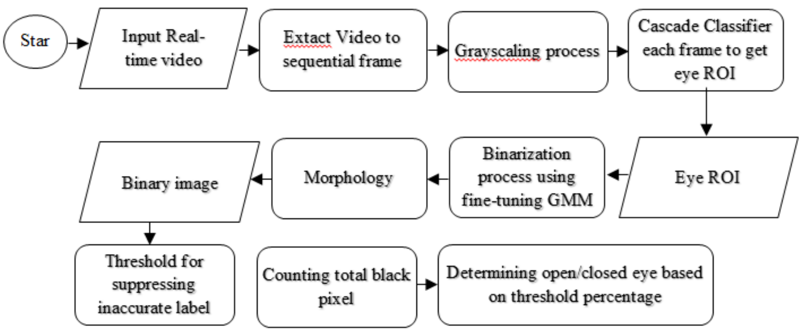

Figure 1. Proposed method

1. Video input occurs when the camera is pointed directly at the driver's object, resulting in a video output when the recording procedure is completed.
2. Video to frame extraction process, every video capture process will be instantly converted into frames so that it is easy to do image processing.
3. The Haar-like feature process is derived by computing the difference between the total pixel values of the dark region and those of the light region. The specific haar feature to be utilized is illustrated in Figure 2.

$$F(haar) = \Sigma F_{white} - \Sigma F_{black} \quad (1)$$

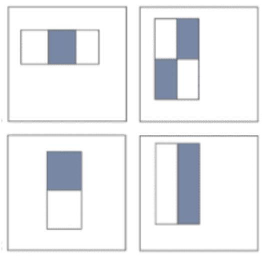

Figure 2. Haar Feature types

4. Integral image process is transformation of the input pixels into a new picture representation is a crucial part of the image processing pipeline. For a high-level representation of the hand-calculation, see Figure 3.

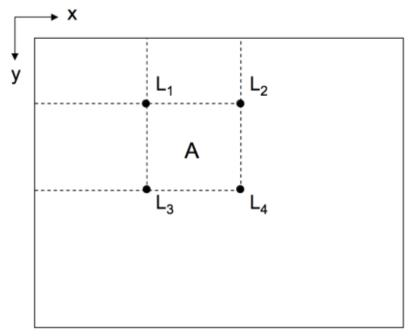

Figure 3. Illustration of the integral image

5. Let gives the sample in pixel (A) in fig. 3, to get the value of the pixel (A) then can use the formula (2). we illustrate how the integral image of the face works in fig. 4.

$$pixel\ (A) = L_4 + L_1 - (L_2 + L_3) \quad (2)$$

6.

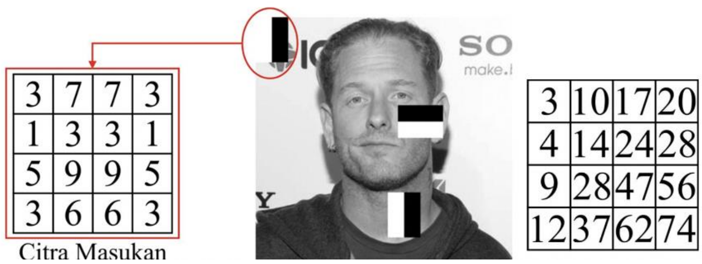

Figure 4. Illustration of the integral image and haar like feature in object

7. Cascade classifiers are used to efficiently connect several features.
8. After getting eye ROI, Fine-Tuning GMM is used to perform the binary segmentation process
   a. Identify the parameters of the Gaussian distribution within the mixture, specifically 2, 3, 4, and 5 as outlined in this study.
   b. Parameter estimation utilizing Maximum Likelihood Estimation (MLE) iterated with the EM algorithm for Gaussian Mixture Models (GMM) with Specify the method employed to set the parameters at the outset.. K-means functions as a variant of the Expectation Maximization (EM)[26]. Set the parameters for mean $\boldsymbol{\mu j}$, covariance $\boldsymbol{\Sigma j}$, and prior distribution $\boldsymbol{\pi j}$ utilizing k-means clustering.
   c. Determine max iteration of 200 iterations. Each iteration do the process:
      i. In the E-Step, the value of zij n equation (3) will be calculated with the initialized parameter values.

$$P(\,j \mid i) = P(y = j|X_n) = \frac{P(j)P(X_n|j)}{P(X_n)} = \frac{P(j)P(X_n|j)}{\sum_{j=i}^{K} P(j)P(X_n|j)} \quad (3)$$

      ii. In the maximization step (M-Step), the parameters will be updated. {$\boldsymbol{\mu j}$, $\boldsymbol{\Sigma j}$, πj}
      iii. Evaluate the log-likelihood function in equation (4) and It is important to verify that the log-likelihood function and the parameter values are convergence.

$$L(\pi, \boldsymbol{\mu}, \boldsymbol{\Sigma}|y) = \sum_{i=1}^{N} \log\left(\sum_{j=1}^{K} \frac{\pi_j}{2\pi^{d/2}\left|\Sigma_j\right|^{1/2}} exp\left(-\frac{1}{2}\left(y_i - \mu_j\right)^T \Sigma_j^{-1}(y_i - \mu_j)\right)\right) \quad (4)$$

   d. Establish the threshold for assessing the convergence of the log-likelihood, specifically: <0.0001.
   e. Assess the Gaussian Mixture Model (GMM) by varying hyperparameters while implementing cross-validation with a value of 3.
9. The outcomes of fine-tuning are subsequently incorporated into the GMM variables. After that, do the morphology process using opening to get representation of shape object.
10. Binarized image of GMM result is assigned a value designated as the threshold that is determined by the total number of white pixels and black pixels. This study continues to utilize the threshold value derived from GMM fine-tuning binarization results to identify the optimal ROI eye location. Applying the criteria of more than 2200 white pixels and more than 1500 black pixels for suppressing ROI that appears.
11. Ascertain the state of the eye as either closed or open by evaluating the percentage of total black pixels against the specified threshold.

3. RESEARCH RESULTS AND DISCUSSION

This section presents the results of image thresholding utilizing Fine-tuning GMM. The results are explained, commented upon, and a comparison is made between Fine-tuning GMM and OTSU. the available video in this research is then extracted into 20 frames. All project examined using python from macbook pro 2017 Intel Core I5 8 GB RAM. The video was captured right inside the car.

3.1. Haar Cascade Processing

Video comprises multiple frames, from which a specific frame is selected. The selected frame is then converted into a grayscale image, facilitating the conversion of images into binary form, thereby simplifying object detection within the image. The system will subsequently detect the face using Haar `cascade. The system will identify the face within the specified frame, subsequently crop the frame, and proceed to further processing. Subsequently, the acquired frame will utilize Haar cascade for eye detection and perform cropping accordingly. The Haar cascade used for eye detection is referred to as the eye cascade. Fig. 5 shows the results after the eye extraction process using haar cascade.

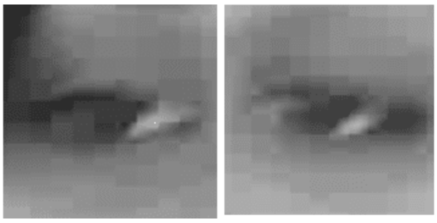

Figure 5. extracted eye using haar cascade

3.2. *Fine-Tuning GMM Process*

The haar cascade results will identify regions deemed as noise. Typically, these noise region are diminutive in scale. A Gaussian smoothing procedure is employed to eradicate them. Gaussian smoothing, sometimes referred to as Gaussian blur, will obscure an unnecessary area. Then perform the GMM Fine Tuning process and then finally closed by using morphology opening. The result of Fine-Tuning GMM can be seen in fig. 6.

Figure 6. Fine-tuning GMM result

3.3. Implementation Threshold Value for Total Counts of white and Black Pixels

After Fine-tuning GMM processing result, It requires adjustment to optimize ROI by establishing a threshold value for the total number of white and black pixels. This study establishes a threshold of 2200 for total white pixels and 1500 for complete black pixels.

if *$total_{whitepixels}$ > 2200 and $total_{blackpixels}$ > 1500* ***then***
    if *(totalpixels_nonZero / roi_eye_size)* ***then***
        eye_state = "eye is open"
    else
  eye_state = "eye is closed"
else
    output ("not detected");

Before doing the total pixel process, giving the threshold value of $total_{whitepixels}$ and $total_{blackpixels}$ is used to suppress the bounding box that appears on top of each other. Final process is Determining whether the eye is closed or open using the percentage of the total number of black pixels that appear. The final process of suppress bounding box and result of total pixel can be seen in fig 7.

Figure 7. detection result using threshold to suppress bounding box

To evaluate the quality of the proposed threshold, one can utilize MSE and PSNR[27] The proposed method will be compared with the OTSU threshold for each right and left eye across the entire frame. MSE comparison between Fine-tuning GMM and OTSU can be seen in table 1.

Table 1. Comparison of The MSE between FT-GMM and OTSU

Sequential Frame	MSE			
	Left Eye		Right Eye	
	FT-GMM	OTSU	FT-GMM	OTSU
1	7,21	7,00	10,49	11,82
2	7,41	8,23	9,49	8,79
3	6,71	5,03	8,67	8,14
4	6,70	7,47	6,56	8,81
5	10,12	8,44	10,05	9,96
6	6,09	6,03	9,21	9,20
7	7,93	8,32	9,14	9,03
8	7,98	7,33	7,64	6,42
9	7,05	7,23	9,65	10,52
10	6,61	6,05	7,63	8,27
11	4,58	5.65	6,16	6,64
12	5,40	5,77	7,69	7,39
13	7,85	5,86	7,27	7,57
14	7,22	5,46	Undetected	Undetected
15	6,68	5,61	Undetected	Undetected
16	7,39	4,92	5,43	6,39
17	6,69	5,83	5,27	5,47
18	7,60	6,70	8,15	7,48
19	8,32	7,44	8,65	6,89
20	4,83	4,85	6,18	5,82
Average	**7,02**	**6,50**	**7,96**	**8,03**

The MSE parameter quantifies the standard deviation of the cumulative error between the segmented image and the ground truth image. A lower MSE value indicates superior image segmentation results. the average MSE value for the Fine-tuning GMM algorithm is 7.02 for left eye and 7.96 for the right eye., whereas the MSE value for the FCM method is 6.50 and 8.03. This indicates that the Fine-tuning yields comparable quality to OTSU in the segmentation process.

Table 2. Comparison of The PSNR between FT-GMM and OTSU

Sequential Frame	PSNR (dB)			
	Left Eye		Right Eye	
	FT-GMM	OTSU	FT-GMM	OTSU
1	39,55	39,68	37,92	37,41
2	39,43	38,98	38,36	38,69
3	39,86	41,12	38,75	39,02
4	39,87	39,40	39,96	38,68
5	38,08	38,87	38,11	38,15
6	40,29	40,33	38,49	38,49
7	39,14	38,93	38,52	38,58
8	39,11	39,48	39,30	40,05
9	39,65	39,54	38,29	37,91
10	39,93	40,32	39,31	38,95
11	41,52	40.61	40,24	39,91
12	40,81	40,52	39,27	7,39
13	39,18	40,45	39,52	7,57
14	39,54	40,76	Undetected	Undetected
15	39,88	40,64	Undetected	Undetected
16	39,45	41,21	40,78	40.08
17	39,88	40,47	40,91	5,47
18	39,32	39,87	39,02	7,48
19	38,93	39,42	38,76	6,89
20	41,29	41,27	40,22	5,82
Average	**39,74**	**40,07**	**39,21**	**39,17**

To the contrary of MSE, the PSNR parameter determined the segmented image closer to the ground truth image when its value increases and vice versa as shown in table 2. A lower PSNR value indicates that the segmented image is of lower quality. The Fine-Tuning GMM method's obtain 39.74 and 39.21 for average PSNR, then otsu method's value is 40.07 and 39.17. The results demonstrate that the Fine-tuning and otsu approach achieves segmentation quality that closely resembles the ground truth. The reason this research was compared with the Otsu threshold, cause Otsu is recognized as the most effective method for thresholding[28].

Table 1 and Table 2 indicate that the developed system effectively detects open and closed eyes, nevertheless, for certain frames in right eye, cannot be detected. It caused error from haar cascade method. Haar cascade is failed to detect right eye in frame 14 and 15. The error arising from the binarization process in this study is due to the similarity between numerous background pixels and quality foreground pixels, resulting in suboptimal MSE and PSNR outcomes for both the suggested approach and Otsu's method

4. CONCLUSION

This study concludes that employing fine-tuning GMM as a threshold approach yields satisfactory results, with an average MSE and PSNR of 7.02 and 39.74 for the left eye, and 7.96 and 39.21 for the right eye for entire frame. The results obtained can be considered to be equivalent to Otsu. Errors arise in specific frames due to the haar cascade method's inability to detect eye location. The thresholding process established following the Fine-tuning GMM segmentation results effectively suppresses overlapping bounding boxes, ensuring that the total pixel process focuses solely on the bounding boxes within the eye area.

5. SUGGESTED

For future research, it is suggested to use the development of other methods besides GMM tuning, such as machine learning tuning such as K-Means, SOM, and others, as well as with the use of deep learning in the case of binarization. Future research can also highlight noise or lighting that can interfere with accuracy. In the evaluation, there appears to be a failure of the haar cascade in frames 14 and 15, so it is necessary to review the development of the haar cascade.

6. REFERENCES

[1] Y. Albadawi, M. Takruri, and M. Awad, “A Review of Recent Developments in Driver Drowsiness Detection Systems,” Mar. 01, 2022, *MDPI*. doi: 10.3390/s22052069.

[2] K. Li, Y. Gong, and Z. Ren, “A Fatigue Driving Detection Algorithm Based on Facial Multi-Feature Fusion,” *IEEE Access*, vol. 8, pp. 101244–101259, 2020.

[3] W. Alkishri, A. Abualkishik, and M. Al-Bahri, “Enhanced Image Processing and Fuzzy Logic Approach for Optimizing Driver Drowsiness Detection,” *Applied Computational Intelligence and Soft Computing*, vol. 2022, 2022.

[4] C. H. Choi, J. Kim, J. Hyun, Y. Kim, and B. Moon, “Face Detection Using Haar Cascade Classifiers Based on Vertical Component Calibration,” *Human-centric Computing and Information Sciences*, vol. 12, 2022.

[5] A. Quddus, A. Shahidi Zandi, L. Prest, and F. Comeau, “Using long short term memory and convolutional neural networks for driver drowsiness detection,” *Accid Anal Prev*, vol. 156, p. 106107, Jun. 2021, doi: 10.1016/j.aap.2021.106107.

[6] D. Sruthi, A. A. Reddy, G. S. S. Reddy, and Mrs. S. Shesham, “Driver Drowsiness Detection System using Deep Learning,” *Int J Res Appl Sci Eng Technol*, vol. 11, no. 4, pp. 1390–1399, Apr. 2023, doi: 10.22214/ijraset.2023.50345.

[7] B. Mokashi, A. Tigadi, B. Chilad, R. Gujanatti, and S. Jadhav, “Deep Learning Approach for Drowsiness Detection Using Facial Features,” in *5th International Conference for Emerging Technology (INCET)*, May 2024, pp. 1–7. doi: 10.1109/INCET61516.2024.10593414.

[8] Y. Albadawi, A. AlRedhaei, and M. Takruri, “Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features,” *J Imaging*, vol. 9, no. 5, May 2023, doi: 10.3390/jimaging9050091.

[9] M. Chua, C. Joy Onquit, E. Galupo, G. Jeffrey Romano, J. Calim, and R. Macatangga, “ENHANCED VIOLA-JONES DETECTION ALGORITHM IN A MOBILE-BASED DROWSINESS DETECTION WITH FACIAL LANDMARKS, ANALYSIS AND DETECTION,” in *The 14th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Yogyakarta: IEEE, Oct. 2022.

[10] R. Kumar, H. Rathore, P. Agrawal, and P. Gupta, "Drowsiness Detection Using Viola–Jones Object Detection Algorithm for Real-Time Data," in *Data Engineering and Intelligent Computing*, V. Bhateja, S. C. Satapathy, C. M. Travieso-González, and V. N. M. Aradhya, Eds., Singapore: Springer Singapore, May 2021, pp. 369–380.

[11] S. Hatim, "Drowsy Driver Detection Using Viola-Jones Algorithm," *Mathematical Sciences and Informatics Journal*, vol. 2, pp. 51–56, Nov. 2021, doi: 10.24191/mij.v2i2.13926.

[12] A. Moujahid, F. Dornaika, I. Arganda-Carreras, and J. Reta, "Efficient and compact face descriptor for driver drowsiness detection," *Expert Syst Appl*, vol. 168, Apr. 2021, doi: 10.1016/j.eswa.2020.114334.

[13] S. Sougandh, N. Abraham, M. J. V, S. M. Sabu, and R. Krishna, "Driver Drowsiness Detection System," *International Journal of Research in Engineering and Science (IJRES) ISSN*, vol. 10, pp. 319–323, 2022, [Online]. Available: www.ijres.org

[14] K. Adi, A. P. Widodo, C. E. Widodo, A. B. Putranto, S. Naqiyah, and H. N. Aristia, "Detecting driver drowsiness using total pixel algorithm," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2019. doi: 10.1088/1742-6596/1217/1/012036.

[15] J. Ding, Z. Zhang, X. Yu, X. Zhao, and Z. Yan, "A Novel Moving Object Detection Algorithm Based on Robust Image Feature Threshold Segmentation with Improved Optical Flow Estimation," *Applied Sciences (Switzerland)*, vol. 13, no. 8, Apr. 2023.

[16] K. Zhang, S. Tong, H. Shi, G. Yue, and J. Zhao, "Moving object detection of assembly components based on improved background subtraction algorithm," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing Ltd, Jan. 2021.

[17] W. Luo, J. Cao, K. Ishikawa, and D. Ju, "A Human-Computer Control System Based on Intelligent Recognition of Eye Movements and Its Application in Wheelchair Driving," *Multimodal Technologies and Interaction*, vol. 5, p. 50, Aug. 2021, doi: 10.3390/mti5090050.

[18] A. Naseer *et al.*, "Efficient Multi-Object Recognition Using GMM Segmentation Feature Fusion Approach," *IEEE Access*, vol. PP, p. 1, Jan. 2024, doi: 10.1109/ACCESS.2024.3372190.

[19] M. A. Soeleman, I. Haq, Muslih, W. Karis, and P. R. Anggi, "The Use of Gaussian Mixture Model for Counting Human Object on Video," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Apr. 2021.

[20] M. A. Soeleman *et al.*, "Adaptive threshold for moving objects detection using gaussian mixture model," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 1122–1129, 2020, doi: 10.12928/TELKOMNIKA.V18I2.14878.

[21] P. E. Jebarani, N. Umadevi, H. Dang, and M. Pomplun, "A Novel Hybrid K-Means and GMM Machine Learning Model for Breast Cancer Detection," *IEEE Access*, vol. 9, pp. 146153–146162, Oct. 2021.

[22] R. Chavan, *Object detection by automatically tuning GMM training parameters using Cuckoo Search Optimization technique*. 2021.

[23] K. Kousalya and T. Saranya, "Improved the detection and classification of breast cancer using hyper parameter tuning," *Mater Today Proc*, vol. 81, pp. 547–552, 2023, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214785321028091

[24] H. Salem, M. Y. Shams, O. M. Elzeki, M. A. Elfattah, J. F. Al-amri, and S. Elnazer, "Fine-Tuning Fuzzy KNN Classifier Based on Uncertainty Membership for the Medical Diagnosis of Diabetes," *Applied Sciences (Switzerland)*, vol. 12, no. 3, Feb. 2022, doi: 10.3390/app12030950.

[25] F. A. Dael and H. Avvad, "Optimizing Random Forest Hyperparameters for Enhanced Stock Price Prediction," in *2025 International Conference on New Trends in Computing Sciences (ICTCS)*, 2025, pp. 460–465. doi: 10.1109/ICTCS65341.2025.10989295.

[26] S. K. Appiah, K. Wirekoh, E. N. Aidoo, S. D. Oduro, and Y. D. Arthur, "A model-based clustering of expectation–maximization and K-means algorithms in crime hotspot analysis," *Research in Mathematics*, vol. 9, no. 1, 2022, doi: 10.1080/27684830.2022.2073662.

[27] E. A. Pambudi and M. I. Nurhidayat, "Impact of Wolf Thresholding on Background Subtraction for Human Motion Detection," *Compiler*, vol. 13, no. 1, p. 39, May 2024.

[28] Y. Guo, Y. Wang, K. Meng, and Z. Zhu, "Otsu Multi-Threshold Image Segmentation Based on Adaptive Double-Mutation Differential Evolution," *Biomimetics*, vol. 8, no. 5, Sep. 2023, doi: 10.3390/biomimetics8050418.