AI-based Efficient Hallway Attendance Monitoring using Confidence-driven Face Recognition

Mokhamad Arfan Wicaksono[1,*], Ibrahim Faizal Burhan[1], Hadi Permana[1], Muhung Anggarawan[1], Bakhriyah Firdausi[1], Muhamad Naufal Azwar Iftikar[2], Ratnasari Nur Rohmah[3] & Yoanes Bandung[4]

[1]Software Engineering Technology Department, Akademi Digital Bandung, Jalan Rancamekar Blok Lio, Kel. Cipamokolan, Kec. Rancasari, Bandung 40292, Indonesia
[2]School of Computer Science, The University of Sydney, 1 Cleveland Street, Darlington, NSW 2008, Australia
[3]Department of Electrical Engineering, Universitas Muhammadiyah Surakarta, Jalan Ahmad Yani, Tromol Pos 1, Pabelan, Kartasura, Sukoharjo 57169, Indonesia
[4]School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Jalan Ganesa No. 10, Bandung 40132, Indonesia
*E-mail: arfan.wicaksono@digitalbdg.ac.id

Abstract. Attendance monitoring deployed at entrance areas offers reliable tracking for institutional environments. We propose an efficient face-recognition-based framework combining YOLOv11-based face detection, motion-based tracking, and FaceNet-based recognition. Our key contribution is a confidence-driven face selection method that collects multiple face images per track and selects the highest-quality image for recognition. Experiments on real-world 60-second hallway footage (1,800 frames at 30 FPS) demonstrated up to 77.7% improvement in processing speed (79.36 FPS vs. 44.66 FPS baseline) while maintaining equivalent recognition accuracy. Collection windows of 5 to 7 images provide optimal balance between latency and throughput (73.67–75.86 FPS), demonstrating that strategic recognition frequency reduction enables real-time attendance monitoring suitable for resource-constrained environments.

Keywords: *attendance monitoring; face detection; face recognition; face tracking; quality-driven selection.*

1 Introduction

Attendance monitoring is an important aspect in various environments, such as educational institutions or workplaces, particularly where security is a concern and attendance records need to be accurate and real-time. Due to advancements in digital and AI technology, computer vision-based attendance monitoring systems provide efficient alternatives to traditional methods [1–3].

One of the attendance monitoring methods is an entrance-based system. Face recognition systems can be deployed at hallway entrance areas to enable such monitoring. However, that implementation faces several challenges: (1) dynamic

Received on December 22nd, 2025, Revised on June 25th, 2026. Accepted for publication on July 31st, 2026.
Copyright © 2026 Published by IRCS-ITB, ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2026.20.3.4

input due to continuous movement of subjects through the camera's field of view and varying angles and distances of faces [3, 4], and (2) significant computational power is needed to process real-time data while handling occlusions for multiple individuals simultaneously [5–7]. Moreover, if the system is to be deployed in resource-constrained environments, it must be optimized for efficiency without sacrificing recognition accuracy. Although some deep learning models are designed for resource-constrained computing machines [8], they are still considered to require much computation unless the model is too simple, leading to inaccurate detection.

The typical workflow of a face recognition-based attendance monitoring system consists of three main stages: face detection, face tracking, and face recognition, which are currently available in today's technology [5, 9, 10]. Face detection identifies and localizes faces in video frames, often using object detection technology such as YOLO [9]. Face tracking maintains identity associations across frames, or it could be said that one track may represent one identity. Face recognition matches detected faces against a reference database, often using an embedded method such as FaceNet [10] to identify individuals. However, the computational cost of these stages, particularly face recognition, can be significant, especially when processing multiple faces in real-time, since the face recognition phase mostly relies on deep learning inference [4, 10]. Moreover, if the entrance area is crowded, the system must efficiently handle multiple simultaneous tracks without overwhelming computational resources [11]. Hence, optimizing the recognition stage is important to preserving computational resources.

Considering the challenges previously mentioned, this paper makes the following contributions:

1. We propose a framework that integrates YOLO-based face detection, motion-based tracking, and FaceNet recognition optimized for hallway attendance monitoring scenarios.
2. We propose a confidence-driven face selection method for the recognition, based on the number of multiple face images per track, selecting the highest-quality image based on detection confidence scores.
3. We conducted comprehensive experiments that compared the immediate recognition (baseline method) versus delayed recognition with quality-based selection (our method) across a range of fixed collection-window sizes.

The rest of this paper is organized as follows: Section 2 discusses related works, Section 3 describes our proposed framework, Section 4 presents the experimental methodology, Section 5 discusses results in comparison with the baseline method, and Section 6 contains the conclusion and future work of this research.

2 Related Work

2.1 Face Detection and Recognition

Face recognition has evolved from traditional approaches like Local Binary Patterns (LBP) [12] and Histogram of Oriented Gradients (HOG) [13] to modern deep learning methods. Convolutional Neural Networks (CNNs) now provide superior accuracy through automatic hierarchical feature learning, as confirmed by Boe et al. [14] in classroom attendance systems. We use YOLO (You Only Look Once) [9] for face detection due to its speed-accuracy balance [15-17], and FaceNet [10] for recognition based on its 98.4% accuracy performance [18].

2.2 Attendance Monitoring Systems

Various face recognition-based attendance systems have been proposed for different deployment scenarios. Rao [2] created AttenFace, which captures classroom images every 10 minutes rather than processing all frames, reducing computational load. Mukhitdinov et al. [19] proposed a lightweight access control system using quantized MobileNetV2 on Raspberry Pi 4, achieving 96.8% recognition accuracy with 850 ms response time. Thai *et al.* [20] proposed mobile-based attendance systems for SME environments using ResNet34, achieving 92% accuracy.

Most existing systems target controlled environments where subjects are relatively stationary, such as classrooms. Our work addresses the more challenging scenario of continuous hallway monitoring, where subjects move freely with varying distances and potential occlusions. While prior work often does not explicitly optimize computational costs for all recognition attempts [21], we optimized for efficiency through confidence-driven selection, similar to the resource-optimization goals of Thai *et al.[*20] and Mukhitdinov *et al.*[19] but tailored for dynamic hallway scenarios.

Compared to recent attendance and face-recognition systems—AttenFace [2] (periodic sampling), Thai *et al.* [20] (ResNet34, 92% accuracy, mobile), and Mukhitdinov *et al.* [19] (quantized MobileNetV2 on Raspberry Pi, 96.8% accuracy, 850 ms response)—our work differs by targeting continuous, unconstrained hallway motion and by reducing recognition frequency through confidence-driven selection rather than compressing the recognition model. This yields substantial throughput gains (up to 79.36 FPS) while preserving recognition completeness.

3 Proposed Framework

3.1 System Architecture

We propose a framework that considers efficiency as the primary goal. Overall, there are four main components in the system. It combines AI-based face detection, motion-based tracking, confidence-driven face selection, and deep learning-based face recognition, as described below:

1. **Face Detection Module:** Real-time face localization where we acquire a bounding box for every detected face along with its confidence score
2. **Tracking Module:** Motion-based identity association across frames to identify existing identity across footage
3. **Quality Evaluation Module:** Confidence-driven best face selection per identity track to be supplied into the face recognition module
4. **Face Recognition Module:** AI-based identifier matching against reference database, recognizing the attendance if the result matches

The pipeline of our proposed system, shown in Figure 1, processes video frames sequentially through detection, tracking, quality evaluation, and recognition stages. The recognition is executed by considering the embedded face dataset previously loaded.

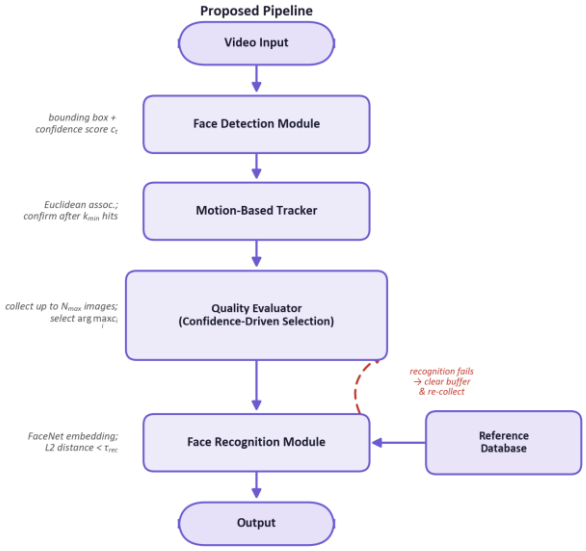


Figure 1 System architecture showing the complete pipeline of face detection, tracking, quality evaluation, and recognition.

3.2 Face Detection

We employ an AI-based object detection model with pre-trained face weights to detect faces in a frame. The model processes input frames and produces bounding boxes with associated confidence scores. These scores are later used in the quality evaluation stage. Several parameters are provided to configure the detection process, including confidence threshold (τ_{det}) and IoU threshold (τ_{iou}). Additionally, the optional device parameter specifies whether to use CPU or GPU for inference, although GPU is preferred for real-time performance, and we assume that the system is mostly applied in environments with GPU availability. Each detection $d_t = (\mathbf{b}_t, c_t, \mathbf{I}_t)$ at frame t includes bounding box $\mathbf{b}_t = (x, y, w, h)$, confidence score $c_t \geq \tau_{\text{det}}$, and cropped face image $\mathbf{I}_t$.

3.3 Motion-based Tracking

The tracking component maintains consistent identities across frames using the following parameters: distance factor (α), minimum hits (k_{min}), aging threshold (a_{max}), and maximum collection size (N_{max}). Our tracker maintains active tracks $\mathcal{T} = \{T_1, T_2, \ldots, T_n\}$, where each track represents a unique individual. Detections are assigned to tracks using Euclidean distance in Eq. (1),

$$\text{dist}(\mathbf{b}_i, \mathbf{b}_j) = \sqrt{\left(\text{cx}_i - \text{cx}_j\right)^2 + \left(\text{cy}_i - \text{cy}_j\right)^2} \tag{1}$$

with threshold in Eq. (2),

$$\tau_{\text{max}} = \alpha \cdot \sqrt{w^2 + h^2}, \tag{2}$$

where (cx, cy) are bounding box centers and (w, h) are dimensions.

Track state management uses the following rules:

1. **Confirmation:** Track confirmed after k_{min} consecutive hits
2. **Aging:** Track deleted after a_{max} frames without detection
3. **Collection:** Track accumulates up to N_{max} face images

3.4 Confidence-driven Face Selection

This component is our main proposed method that differentiates our framework from conventional immediate recognition approaches. The quality evaluator collects multiple face images per track and selects the highest-quality image based solely on the detection confidence scores produced by the face detector. To clarify the distinction of our method, we compared it against the baseline immediate recognition approach.

3.4.1 Baseline Method

The baseline approach performs immediate recognition upon track confirmation (after $k_{\min}$ consecutive hits), using the latest available face image. If recognition fails, further attempts are made on subsequent frames for that track, which can lead to redundant recognition operations and increased computational load.

3.4.2 Proposed Method

We designed our proposed method to reduce computational load while keeping recognition accuracy sufficient. The proposed method delays recognition until the collection window ($N_{\max}$) is filled, then selects the image with the highest detection confidence score for recognition. If recognition fails, the image buffer is cleared and a new set of ($N_{\max}$) images is collected for another attempt. This approach minimizes redundant recognition attempts and leverages the best available image per track. The algorithm of our proposed method is described in Algorithm 1.

Algorithm 1. Our Method – Confidence-driven Selection

Require: Track T with images $\{(\mathbf{I}_1, c_1), \ldots, (\mathbf{I}_n, c_n)\}$
Ensure: recognized identity, or "Unknown" if no match is found.
Append the current detection (I_i, c_i) to the track buffer
if ($n \geq N_{\max}$) OR (Track T is aging out) **then**
  if Track T not yet recognized **then**
    $i^* \leftarrow \arg\max_i c_i$ // Select highest confidence
    $\mathbf{I}_{\text{rec}} \leftarrow \mathbf{I}_{i^*}$
    identity ← Recognize ($\mathbf{I}_{\text{rec}}$)
    if (recognition fails) AND (Track T still active) **then**
      Clear image buffer
      Re-collect images and retry
    end if
  end if
end if

The confidence-only approach offers several advantages. The first advantage is minimal additional computation, since confidence scores are already produced by the face detector. Secondly, confidence scores can be utilized to determine the best-quality face image, since higher confidence usually correlates with better quality. Lastly, the implementation is straightforward, which avoids additional computation to assess image quality. Overall, this method provides a lightweight yet effective means of enhancing recognition efficiency without sacrificing accuracy.

We adopted detection confidence as a principled proxy for face image quality: a face detector assigns higher confidence to faces that are frontal, well-illuminated, in-focus, and unoccluded—precisely the conditions that also yield more discriminative recognition embeddings. Face image quality assessment studies note that detector confidence can itself serve as a quality signal [24], [25] and recent work explicitly calibrates such confidence against objective quality factors, including pose, blur, occlusion, and illumination [26]. Using confidence therefore requires no additional computation, since it is already produced by the detector, whereas dedicated quality networks (e.g., SER-FIQ) add inference cost. We adopt it as a deliberate lightweight approximation and note its limitations below.

3.5 Face Recognition

We employ an AI-based face recognition model in this component to perform face embedding. The embedded face is then compared against a reference database for identification. The comparison is done by computing the L2 distance between embeddings, where a lower L2 distance value indicates a closer match. In this process, the adjustable parameter is a recognition threshold (τ_{rec}), where an acceptable match is one with a distance below this threshold.

To enable face recognition in the system, the identity reference should be placed in the identity reference folder/database to be embedded using an AI model. For each identity in the reference database, the image is loaded and preprocessed, then the embedding is extracted using the AI model. The embeddings are stored along with their associated identity labels for later comparison during the recognition phase.

Recognition is performed for every detected face (baseline method) or every selected face among a collection of images (proposed method). For each face image, the embedding is extracted and compared against all reference embeddings using L2 distance. The identity with the minimum distance below the recognition threshold (τ_{rec}) is returned as the recognized identity; otherwise, the face is labeled as 'Unknown'.

3.6 Performance Monitoring

To enable comparison of our proposed method against the baseline, we included a comprehensive performance monitoring feature that tracks metrics related to processing speed, resource utilization, and recognition performance. Metrics are logged per-frame to CSV files and aggregated to JSON summaries for analysis.

4 Experimental Methodology

4.1 Dataset and Setup

4.1.1 Data Acquisition

Experiments were conducted on a 60-second hallway video captured at 30 FPS (1,800 total frames) with resolution 1920 x 1080 pixels. The video contains natural hallway traffic with multiple individuals walking through the camera's field of view at various distances and angles. The data was captured using a Raspberry Pi 5 with Camera Module 3 Wide, positioned to monitor a typical hallway entrance area with fixed placement, as can be seen in Figure 2. Example footage can be seen in Figure 3.

The dataset is self-collected and not publicly available. It comprises 1,800 frames (60 s at 30 FPS, 1920 × 1080) yielding 23 person tracks across 11 registered identities. The footage was acquired by the authors at a single hallway entrance; reference images were captured manually from the same footage under controlled framing.

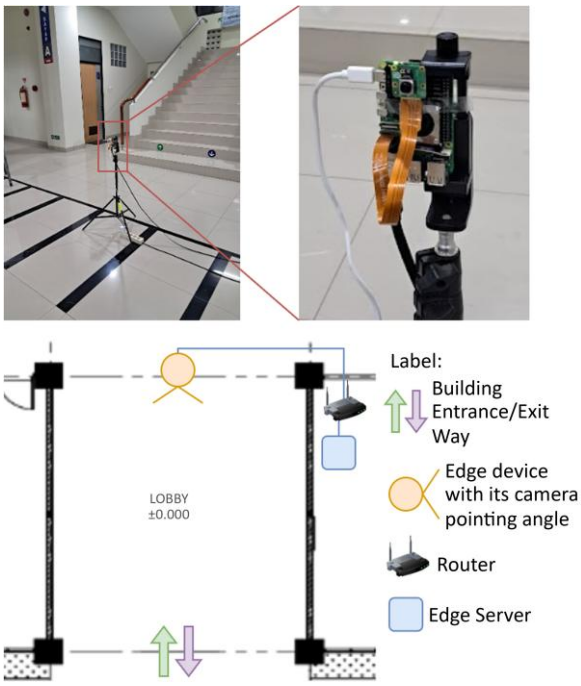


Figure 2 Camera placement.

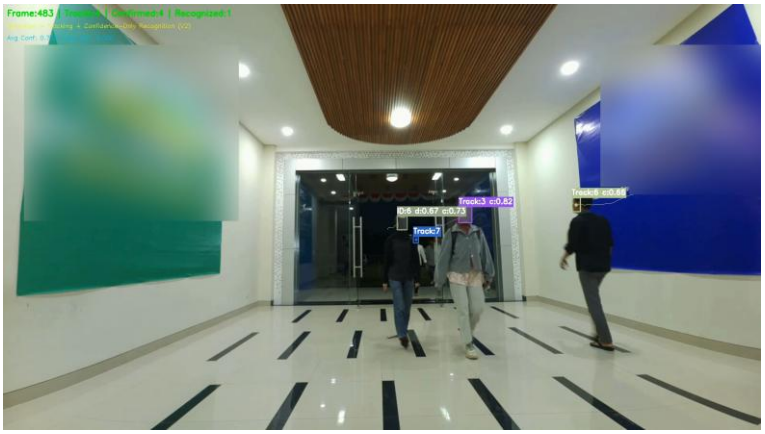


Figure 3 Example of hallway footage.

4.1.2 Reference Database

A reference database of known identities was constructed from pre-registered face images taken manually from the video footage. The total number of identities registered in the database was 11. Each identity has one reference photo captured under controlled conditions.

4.1.3 Hardware Configuration

All experiments were conducted on the following hardware:

1. GPU: NVIDIA RTX 5060 TI (CUDA-enabled)
2. CPU: AMD Ryzen 5 4600G
3. Memory: 32 GB DDR4 RAM
4. OS: Ubuntu 24.04 LTS

4.1.4 AI Models Used

The two main AI-dependent components, i.e., face detection and face recognition, utilized in the experiments are the YOLOv11s model pre-trained on the WIDER FACE dataset [22] for face detection and FaceNet (InceptionResnetV1) pre-trained on the VGGFace2 dataset [10, 23] for face recognition. As explained in Section 2, YOLOv11s was chosen due to its speed-accuracy trade-off, while FaceNet is selected for its high recognition accuracy.

4.2 Experimental Design

We conducted 10 experiments to compare the baseline method against our method with varying collection window sizes. The experiments were as follows:

1. Experiment 1: Baseline method
2. Experiments 2-10: Our method with $N_{\max} \in \{2,3,4,5,6,7,8,9,10\}$

We focused on evaluating how the collection window size ($N_{\max}$) affects recognition distance, processing speed, and resource utilization. All other parameters were held constant: $\tau_{\text{det}} = 0.25$, $\tau_{\text{iou}} = 0.45$, $\alpha = 1.5$, $a_{\max} = 30$ frames, $k_{\min} = 3$ frames, and $\tau_{\text{rec}} = 0.7$.

4.3 Evaluation Metrics

We evaluated system performance across three main dimensions: recognition performance, processing performance, and resource utilization. The specific metrics tracked were as follows:

Recognition Performance

1. Total tracks detected
   a. Number of recognized identities
   b. Recognition distance (L2 distance)
   c. Recognition rate (proportion of registered identities recognized)

Processing Performance

2. Average FPS (frames per second)
   a. Frame processing time (ms)
   b. Detection time (ms)
   c. Tracking time (ms)
   d. Quality evaluation time (ms, our method only)
   e. Recognition time (ms)

Resource Utilization

3. CPU usage (average, max, min)
   a. Memory consumption (MB)
   b. GPU utilization (average)
   c. GPU memory usage (MB)

5 Results and Discussion

5.1 Recognition Performance

There were 23 tracks detected in the video, containing 11 identities. All the experiments detected a total of 11 recognized identities, indicating that confidence-driven selection maintains recognition accuracy equivalent to immediate recognition. This suggests that even with a high number of collection windows, e.g., ($N_{\max} = 10$), selecting the highest-confidence image provides sufficient quality for accurate recognition.

The consistent recognition rate in our method indicates that it successfully identifies correct face images. Based on the video footage, the 12 unrecognized tracks likely represent tracks with insufficient quality across all collected images, including tracked distant persons, persons walking in the opposite direction, and occluded persons. Based on the footage example shown in Figure 3, Track 6 represents a person walking in the opposite direction, which is later occluded by Track 3. Moreover, Track 7 shows a distant face detected in the footage, which is later occluded as well, before the face appears again.

In terms of recognition rate—the proportion of registered identities successfully recognized—both the baseline and the proposed method achieved 11/11 (100%) identity recognition, while 12 of the 23 tracks remained unrecognized (distant, opposite-direction, or occluded subjects). This confirms that the proposed method preserves recognition completeness while substantially improving throughput. A full per-track precision/recall evaluation is left for future work.

As shown in Figure 4, recognition distance distributions remained similar across methods, indicating that our method can match the baseline method in recognizing faces. This is also shown in Table 1 by looking at the statistical result. Based on the table, the baseline method shows a mean L2 recognition distance of 0.656 (min 0.602, max 0.694; n=11). Our method ranges between 0.600 and 0.669 in mean distance across $N_{\max} = 2 \ldots 10$ (mins 0.408-0.515, maxes 0.694-0.697; all n = 11). Figure 5 shows the recognition timing, where our method recognizes later in a track lifecycle, because the face selection is previously conducted before face recognition occurs but still remains within an acceptable time in the attendance system.

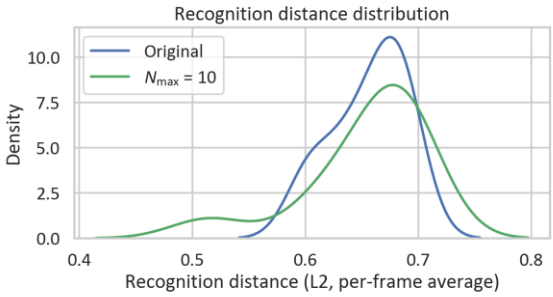


Figure 4 Recognition distance distributions (L2) remain similar across methods.

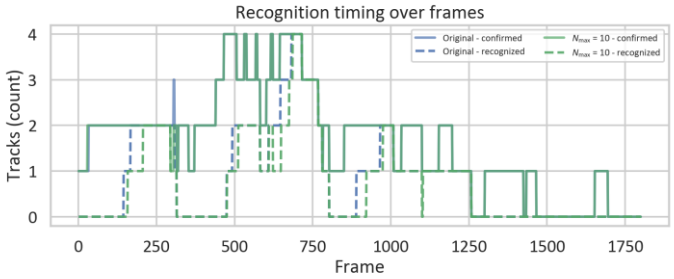


Figure 5 Recognition timing over frames.

Table 1 Recognition distance (L2) by configuration.

Method	N_{max}	Count	Mean	Min	Max
Baseline	-	11	0.656	0.602	0.694
Our method	2	11	0.666	0.588	0.694
Our method	5	11	0.655	0.515	0.694
Our method	7	11	0.605	0.449	0.694
Our method	10	11	0.654	0.515	0.697

5.2 Processing Performance

Table 2 presents detailed processing performance metrics.

Table 2 Average processing performance comparison for every identity.

Method	FPS	Frame (ms)	Det. (ms)	Track (ms)	Qual. (ms)	Rec. (ms)
Baseline	44.66	22.39	9.05	0.03	-	18.59
Our method ($N_{max} = 2$)	62.49	16.00	9.38	0.03	4.02	15.81
Our method ($N_{max} = 5$)	73.67	13.57	9.23	0.03	1.73	14.49
Our method ($N_{max} = 7$)	75.86	13.18	9.19	0.03	1.39	14.57
Our method ($N_{max} = 10$)	79.36	12.60	9.08	0.03	0.93	13.61

Our method achieved substantial FPS improvements: 62.49 FPS (N_2, +39.9%), 73.67 FPS (N_5, +64.9%), and 79.36 FPS (N_{10}, +77.7%) versus 44.66 FPS baseline. This improvement stems from delayed recognition strategy, where our method performs recognition only when $N_{\max}$ images are collected, versus per-frame recognition in the baseline. Detection times remain stable (9.05–9.38 ms) across methods, while quality evaluation overhead decreases with larger $N_{\max}$ (4.02 ms at N_2 to 0.93 ms at N_{10}). Figures 6–8 demonstrate steady throughput improvement and more stable per-frame FPS with increasing $N_{\max}$.

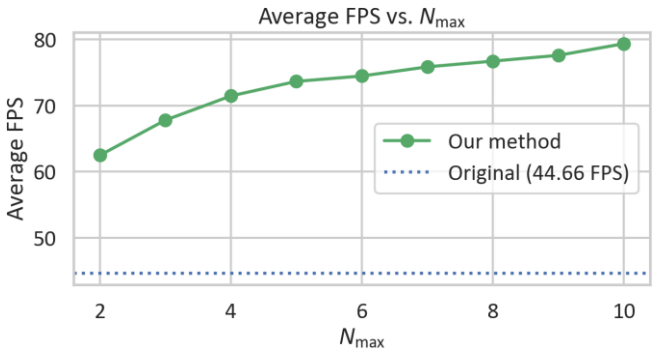


Figure 6 Average FPS vs $N_{\max}$.

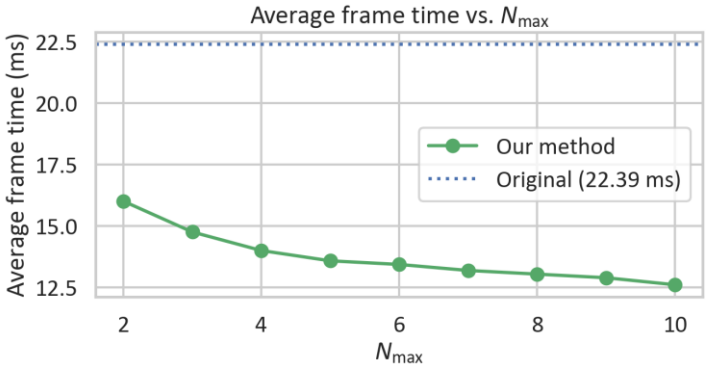


Figure 7 Average frame time vs $N_{\max}$.

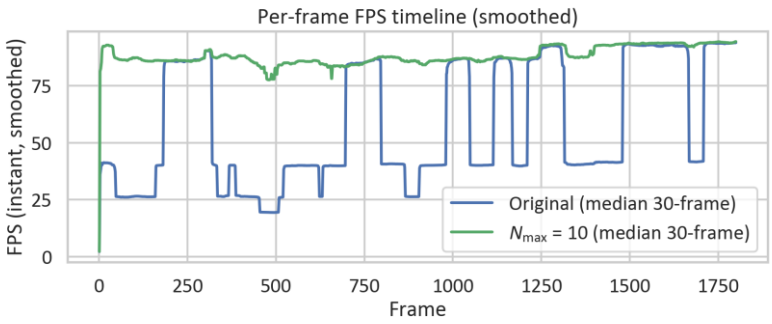


Figure 8 Per-frame FPS over time: baseline vs our method (e.g., $N_{\max} = 10$).

5.3 Resource Utilization

Table 3 presents resource utilization metrics.

Table 3 Resource utilization comparison.

Method	CPU (%)	Mem (MB)	GPU (%)	VRAM (MB)
Baseline	116.3	1845.1	16.5	177.4
Our method (N_2)	119.4	1837.1	15.0	179.3
Our method (N_5)	120.6	1836.4	14.9	179.2
Our method (N_{10})	123.1	1835.6	14.2	178.5

Resource utilization remained remarkably consistent across methods:

1. CPU usage: 116–123% (slightly higher for our method due to increased throughput)
2. Memory: 1835–1845 MB (stable across sampled methods)
3. GPU usage: 14.2–16.5% (slightly lower for our method)
4. GPU memory: 177.4–179.3 MB (stable across sampled methods)

The insignificant resource overhead confirms that confidence-driven selection is extremely lightweight. The small increase in CPU usage for our method is primarily due to processing more frames per second rather than per-frame overhead.

Figure 9 presents resource time series that remained stable throughout processing for both methods.

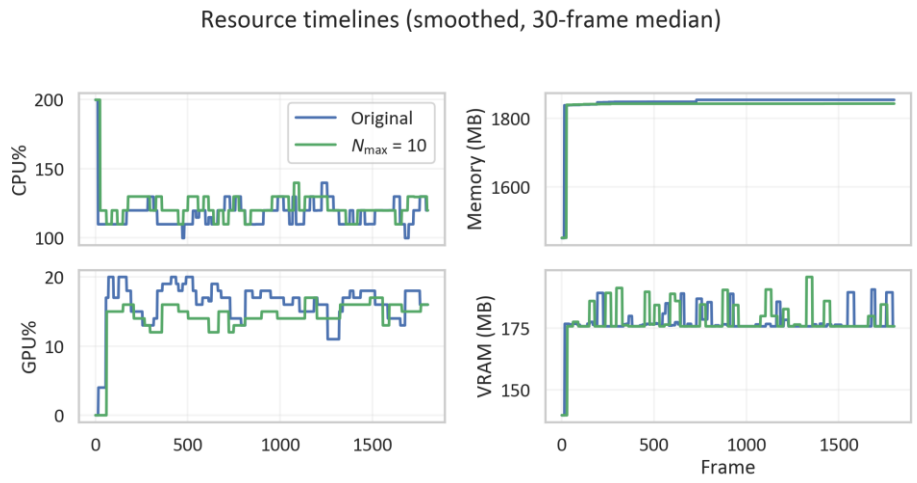


Figure 9 Resource timelines over frames: baseline vs our method ($N_{\max} = 10$).

5.4 Analysis and Discussion

Our method achieved better performance by delaying recognition until $N_{\max}$ images are collected, performing fewer recognition operations per frame while selecting the highest-confidence image. The confidence-only selection introduces negligible computational overhead compared to recognition savings. The results suggest that $N_{\max} = 5$—7 provides an optimal balance between recognition latency and processing throughput, achieving 73.67—75.86 FPS.

This study had several limitations. The evaluation used a single 60-second hallway video with 11 identities and one reference image per identity, captured at one location and lighting condition. Detection confidence was used as an approximate quality signal rather than a learned metric, and recognition was validated on identity completeness rather than full precision/recall ground truth. These factors constrain generalizability; broader and more diverse datasets and learned quality metrics are left to future work.

6 Conclusions and Future Work

This paper proposes a framework for hallway attendance monitoring that combines deep learning-based face detection, motion-based tracking, and confidence-driven face selection. Through comprehensive experiments on real-world hallway video footage that we collected, we showed that our recognition approach with confidence-driven selection achieves up to 77.7% improvement in

processing speed (79.36 FPS at $N_{\max} = 10$ vs. 44.66 FPS baseline) while maintaining equivalent recognition accuracy (between 0.600 and 0.669 for $N_{\max} = 2 - 10$ recognition distance vs 0.656 baseline recognition distance).

Based on our experiments, we found that confidence-driven selection provides an effective, lightweight mechanism for face quality assessment, introducing minimal computational overhead (0.93 ms at $N_{\max} = 10$). Thus, collection windows of 5–7 images offer an optimal balance between recognition latency and throughput (73.67–75.86 FPS). Lastly, the framework shows that strategic recognition frequency reduction enables real-time attendance monitoring suitable for resource-constrained environments.

Future research directions include evaluating the framework on larger and more diverse datasets with varying lighting conditions and crowd density, implementing dynamic collection window size adjustment based on tracking confidence and scene characteristics, and deploying the system on embedded platforms to assess real-world performance within practical resource constraints.

Acknowledgments

This research was supported by Bandung Digital Academy. We thank all participants who contributed to the dataset collection. This work was supported by the Ministry of Education, Culture, Research, and Technology (Kemdiktisaintek) through the Basis Informasi Penelitian dan Pengabdian kepada Masyarakat (BIMA) platform.

References

[1] Turk, M. & Pentland, A., *Eigenfaces for Recognition*, J. Cogn. Neurosci., **3**(1), pp. 71-86, 1991.

[2] Rao, A., *AttenFace: A Real Time Attendance System using Face Recognition*, Proc. IEEE 6th Conf. Inf. Commun. Technol. (CICT), pp. 1-5, 2022.

[3] Zhao, W., Chellappa, R., Phillips, P.J. & Rosenfeld, A., *Face Recognition: A Literature Survey*, ACM Comput. Surv., **35**(4), pp. 399-458, 2003.

[4] Taigman, Y., Yang, M., Ranzato, M. & Wolf, L., *DeepFace: Closing the Gap to Human-level Performance in Face Verification*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1701-1708, 2014.

[5] Bewley, A., Ge, Z., Ott, L., Ramos, F. & Upcroft, B., *Simple Online and Realtime Tracking*, Proc. IEEE Int. Conf. Image Process. (ICIP), pp. 3464-3468, 2016.

[6] Wojke, N., Bewley, A. & Paulus, D., *Simple Online and Realtime Tracking with a Deep Association Metric*, Proc. IEEE Int. Conf. Image Process. (ICIP), pp. 3645-3649, 2017.

[7] Anwar, A. & Raychowdhury, A., *Masked Face Recognition for Secure Authentication*, arXiv:2008.11104, 2020.
[8] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. & Adam, H., *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*, arXiv:1704.04861, 2017.
[9] Redmon, J., Divvala, S., Girshick, R. & Farhadi, A., *You Only Look Once: Unified, Real-time Object Detection*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 779-788, 2016.
[10] Schroff, F., Kalenichenko, D. & Philbin, J., *FaceNet: A Unified Embedding for Face Recognition and Clustering*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 815-823, 2015.
[11] Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W. & Wang, X., *ByteTrack: Multi-object Tracking by Associating Every Detection Box*, Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 1-21, 2022.
[12] Ojala, T., Pietikäinen, M. & Mäenpää, T., *Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns*, IEEE Trans. Pattern Anal. Mach. Intell., **24**(7), pp. 971-987, 2002.
[13] Dalal, N. & Triggs, B., *Histograms of Oriented Gradients for Human Detection*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), **1**, pp. 886-893, 2005.
[14] Boe, C.-H., Ng, K.-W., Haw, S.-C., Naveen, P. & Anaam, E.A., *An Automated Face Detection and Recognition for Class Attendance*, Int. J. Inf. Vis., **8**(3), pp. 1146-1153, 2024.
[15] Hua, Z., Aranganadin, K., Yeh, C.-C., Hai, X., Huang, C.-Y., Leung, T.-C., Hsu, H.-Y., Lan, Y.-C. & Lin, M.-C., *A Benchmark Review of YOLO Algorithm Developments for Object Detection*, IEEE Access, **13**, pp. 123515-123545, 2025.
[16] Lateef, A.S. & Kamil, M.Y., *Effortless Student Attendance: A Smart Human-computer Interactive System using Real Time Facial Recognition*, An-Najah Univ. J. Res. - A (Natural Sciences), **39**(2), pp. 213-224, 2025.
[17] Manongga, W.E., Chen, R.-C. & Jiang, X., *Enhancing Road Marking Sign Detection in Low-light Conditions with YOLOv7 and Contrast Enhancement Techniques*, Int. J. Appl. Sci. Eng., **21**(3), 2023515, 2024.
[18] Serengil, S. & Özpınar, A., *A Benchmark of Facial Recognition Pipelines and Co-usability Performances of Modules*, Bilişim Teknolojileri Dergisi, **17**(2), pp. 95-107, 2024.
[19] Mukhitdinov, O., Bakirov, P., Atashikova, N., Makhkamova, S., Sapaev, I.B., Karimova, Z., Jo'rayeva, M. & Kholmurodova, O., *A Lightweight AI-based Access Control System for IoT-integrated Smart Classrooms*, J. Internet Serv. Inf. Secur., **15**(2), pp. 30-42, 2025.
[20] Thai, H.-D., Seo, Y.-S. & Huh, J.-H., *Enhanced Efficiency in SMEs Attendance Monitoring: Low Cost Artificial Intelligence Facial*

Recognition Mobile Application, IEEE Access, **12**, pp. 184257-184274, 2024.

[21] Habila, M., Francisca, F.N., Ishaya, L., Ahmed, M.K., Muhammad, U.A. & Charles, H.P., *Smart Real-time Attendance System for Nigerian Universities*, J. Inf. Organ. Sci., **49**(1), pp. 121-138, 2025.

[22] Yang, S., Luo, P., Loy, C.C. & Tang, X., *WIDER FACE: A Face Detection Benchmark*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 5525-5533, 2016.

[23] Cao, Q., Shen, L., Xie, W., Parkhi, O.M. & Zisserman, A., *VGGFace2: A Dataset for Recognising Faces across Pose and Age*, Proc. IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), pp. 67-74, 2018.

[24] Schlett, T., Rathgeb, C., Henniger, O., Galbally, J., Fierrez, J. & Busch, C., *Face Image Quality Assessment: A Literature Survey*, ACM Comput. Surv., **54**(10s), pp. 1-49, 2022.

[25] Terhörst, P., Kolf, J.N., Damer, N., Kirchbuchner, F. & Kuijper, A., *SER-FIQ: Unsupervised Estimation of Face Image Quality based on Stochastic Embedding Robustness*, Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 5650-5659, 2020.

[26] Ou, F.-Z., Li, C., Wang, S. & Kwong, S., *CLIB-FIQA: Face Image Quality Assessment with Confidence Calibration*, Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1694-1704, 2024.