Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM)
Vol. 8, No. 3, November 2025, pp. 591 – 600
p-ISSN: 2614-3372 | e-ISSN: 2614-6150

Real-Time Access Control System with YOLOv11-Based Face and Blink Detection

[1]Namira Nur Rifani, [2]RD. Kusumanto, [3*]Nyayu Latifah Husni
[1,2,3]Electrical Engineering, State Polytechnic of Sriwijaya, Indonesia
Email: [1]namiranurrifani@gmail.com, [2]rd.kusumanto@polsri.ac.id, [3]nyayu_latifah@polsri.ac.id

Article Info
Article history:
Received May 08th, 2025
Revised Sep 23th, 2025
Accepted Oct 22th, 2025

Keyword:
Blink Detection
Face Recognition
Liveness Detection
Smart Access
YOLOv11

ABSTRACT
This study presents a real-time smart access control system that combines facial recognition with blink-based liveness detection to strengthen security and reduce spoofing risks. The primary purpose is to provide a lightweight and efficient method that verifies both identity and physical presence in real-time. The system employs two YOLOv11 models: one for detecting facial regions and another for distinguishing eye states through "open" and "closed" transitions. Identity verification is carried out by comparing facial embeddings using Euclidean distance. A private dataset was collected for facial images, while blink data was obtained from a public source, both of which were annotated in YOLO format. After 100 epochs, the face detection model achieved 0.999 precision, 1.000 recall, 0.995 mAP50, and 0.868 mAP50–90, while the blink detection model recorded 0.959 precision, 0.962 recall, 0.967 mAP50, and 0.678 mAP50–90. These outcomes confirm that the objectives were achieved, demonstrating a practical and reliable biometric authentication solution with integrated liveness verification. The system also offers scalability for future multi-modal applications.

Corresponding Author:
Nyayu Latifah Husni,
Electrical Engineering, State Polytechnic of Sriwijaya,
Srijaya Negara Street, Palembang, South Sumatra, Indonesia.
Email: nyayu_latifah@polsri.ac.id

DOI: http://dx.doi.org/10.24014/ijaidm.v8i3.36812

1. INTRODUCTION

Securing access to buildings such as offices, schools, and public facilities is essential for protecting people and assets. Conventional security tools like access cards and physical keys are becoming less reliable due to their vulnerability to being lost, duplicated, or stolen [1]. As stated by to Motwani et al. [2], these systems are prone to breaches, increasing the risk of unauthorized entry, which makes them increasingly unsuitable in environments where consistent reliability and rapid response are required. In high-risk settings such as educational campuses, office complexes, and public infrastructure, a single failure in access security can compromise not only physical assets but also personal safety, creating widespread consequences that are difficult to mitigate after the fact. Consequently, there is a growing shift toward identity-based authentication technologies that offer greater resilience and improved security in modern access control environments, reflecting the urgent need for solutions that can minimize human error, prevent credential misuse, and address the rising frequency of security incidents.

To address these vulnerabilities, biometric technologies have gained prominence as a more secure and identity-driven approach to access control. Among them, face recognition stands out due to its contactless nature, ease of integration, and minimal user effort during authentication, ensuring both user convenience and operational efficiency. Beyond security, its ability to streamline entry processes also makes it appealing for organizations seeking to balance safety with productivity, particularly in environments that require frequent and rapid verification. Facial features are now considered one of the most convenient and widely accepted biometric modalities, making face recognition an attractive solution for modern security infrastructures [3][4], and reinforcing why its continuous enhancement remains a critical research priority.

Several prior studies have applied face recognition within security systems, employing a range of algorithms and architectural designs, with diverse approaches demonstrating both accuracy and practical feasibility. Classical techniques such as Haar Cascade with LBP descriptors [5] and gradient-based methods like HOG [6] established important foundations in face detection. Building on these, researchers introduced convolutional neural networks (CNN) in combination with Radio Frequency Identification (RFID) [7], which highlighted the benefits of combining deep features with multi-factor authentication. The adoption of deep learning frameworks such as TensorFlow-based CNNs further demonstrated how neural architectures could achieve high precision in real-time scenarios [8]. Parallel efforts explored deployments, such as ESP32-CAM with Multi-task Cascaded Convolutional Networks (MTCNN) [9], which was noted for its balanced parameters and adaptability in embedded and resource-constrained environments.

While these techniques have significantly advanced face recognition, their performance is often constrained either by computational cost or limited adaptability. To overcome these challenges, the growing demand for higher speed and flexibility in real-time applications has encouraged the adoption of You Only Look Once (YOLO) models. Unlike traditional two-stage detectors, YOLO follows a one-stage design that performs object localization and classification simultaneously, eliminating the extra region-proposal step and drastically reducing computation time. At the same time, its integrated feature extraction and multi-scale prediction strategy help preserve accuracy, allowing YOLO to deliver an effective balance of precision and efficiency. These characteristics make it especially suitable for real-time security and access control tasks.

Early developments with YOLOv3 showed promising results in custom face recognition tasks, achieving a mean Average Precision (mAP) of 63.4 with real-time performance at 45 frames per second, which was significantly faster than R-CNN–based approaches [10]. The subsequent generation, YOLOv4, further advanced this line of research, demonstrating strong accuracy in broader safety monitoring tasks. For instance, YOLOv4 achieved an mAP of 97.64% with an F1 score of 0.96 in face mask and face shield detection, while its lightweight variant YOLOv4-Tiny reached 171 FPS with 96.75% mAP, underscoring its efficiency in real-time environments [11]. Building on these improvements, YOLOv5 was benchmarked against other state-of-the-art deep learning techniques and demonstrated a 94% mAP, while also offering practical advantages such as being lightweight, requiring no preprocessing, and supporting real-time multi-face detection. Although its score was slightly lower than some more complex models, YOLOv5 confirmed the efficiency of YOLO-based approaches for security applications [12]. More recently, YOLOv8 marked a significant leap in the evolution of YOLO, introducing a refined architecture capable of delivering higher accuracy while also supporting advanced tasks such as segmentation and multi-object tracking. In security-oriented applications, it has been integrated into a two-stage framework combining face and license plate detection with a Siamese Neural Network for rider verification, demonstrating its adaptability and reliability in complex surveillance scenarios [13].

Building on this evolution, YOLOv11 introduces architectural refinements such as the C2PSA module for improved feature representation and enhanced multiscale fusion, enabling stronger detection accuracy while maintaining computational efficiency. Benchmark results on the COCO dataset also demonstrate measurable improvements in both precision and recall compared to earlier YOLO versions [14]. Accordingly, this study adopts YOLOv11 as the foundation of its access control framework, leveraging its balance of accuracy and efficiency to support real-time deployment.

Despite its advantages, face recognition systems remain vulnerable to spoofing attacks, where unauthorized users attempt to deceive the system using printed photos, video replays, or digital displays. Several recent studies have emphasized the need for robust liveness detection to ensure the physical presence of the user. For example, Basurah et al. [15] implemented a facial expression–based liveness detection system using TensorFlow.js, which blocked photo-based spoofing by identifying real-time facial movements. This approach highlights the strength of expression-based cues, though it generally requires broad coverage of different user expressions to maintain reliability. Yang Wei et al. [16] proposed combining facial feature analysis with liveness verification based on temporal consistency, showing how frame-to-frame changes can provide stronger protection against replay attempts. While effective, such methods may depend heavily on stable frame rates and consistent video quality, which can be challenging in practical deployments. Another approach used texture analysis to differentiate between live skin and printed images [17], leveraging surface detail as an anti-spoofing feature. However, its accuracy can be influenced by lighting variations or camera resolution, which may limit consistency across different environments.

Building on these diverse directions, this research introduces a face-based access control solution that focuses on blink-driven liveness verification. Unlike expression-, temporal-, or texture-based cues, blinking is a natural, low-effort signal that is easy to capture in real time. In practice, blink detection can be achieved through the simple observation of transitions between “eye open” and “eye closed” states, making it lightweight and reliable without the need for large datasets or complex temporal analysis. The proposed system employs a dual-model design with YOLOv11 to detect both facial regions and eye activity, ensuring

that identity recognition is paired with evidence of physical presence. By leveraging blinking as the liveness factor, the system reduces the risk of spoofing via static photos or screen displays while preserving computational efficiency suitable for constrained deployment scenarios.

2. RESEARCH METHOD

This study was carried out using a development setup consisting of a laptop with 8GB of RAM and a 720p webcam, which served as the primary tools for data acquisition and testing. The system was designed to operate in real time, requiring lightweight yet reliable components that can function smoothly under limited hardware resources. To ensure clarity and systematic implementation, the research methodology was structured into multiple stages, from data collection and annotation to model training, evaluation, and deployment. Each stage is essential for building an access control framework that combines face recognition with blink-based liveness detection. A flowchart illustrating the entire process of real-time facial authentication and verification is presented in Figure 1.

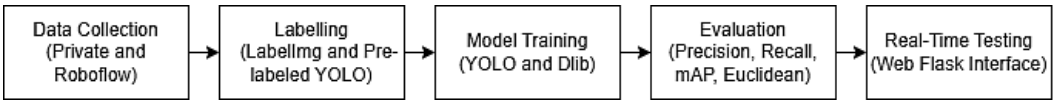


Figure 1. Research Framework

2.1. YOLOv11

YOLOv11 represents the most recent advancement in the Ultralytics YOLO lineup for real-time object detection. It offers notable enhancements in detection precision, inference speed, and computational efficiency. As a refinement of previous YOLO models, YOLOv11 features architectural and training pipeline optimizations, enabling its adaptability across diverse computer vision applications [18]. Figure 2 shows that YOLOv11 is composed of three key components which are the backbone, neck, and head, forming a streamlined pipeline from input to final detection output.

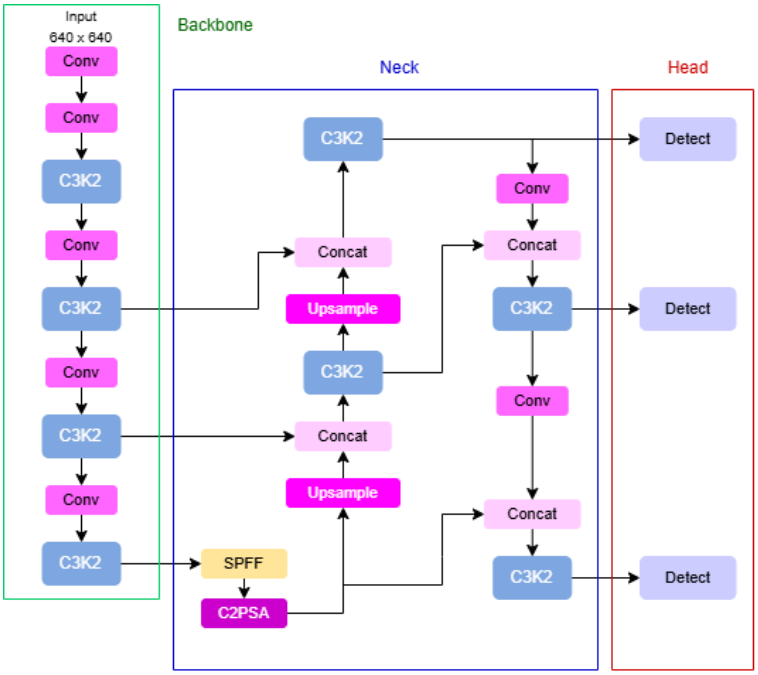


Figure 2. YOLOv11 Architecture [19]

The backbone serves as the primary feature extractor and comprises several modules including Conv, C3K2, SPPF, and C2PSA. The Conv layer begins the process with basic feature extraction, followed by C3K2, a multibranch module designed to balance efficiency and representational power. The SPPF (Spatial Pyramid Pooling Fast) module integrates multiscale context through progressive max pooling, while C2PSA introduces pixel-level attention across spatial scales to enhance fine-grained localization and suppress background noise.

These refined feature maps are passed to the neck, which performs upsampling and feature concatenation to aggregate multiscale features. The neck also incorporates C3K2 again to ensure deep

semantic fusion and richer contextual understanding. Finally, the head processes these features through parallel branches tailored for objects of different scales. Each branch performs classification and bounding box regression independently, producing the final detection outputs: class labels, bounding boxes, and confidence scores. Non-Maximum Suppression (NMS) is applied at the end to eliminate redundant detections [20].

2.2. Face Recognition

Face recognition is a process in computer vision that aims to identify or verify a person's identity by analyzing unique patterns in their facial features. This technique transforms facial regions into numerical representations called embeddings, which can be matched against a stored database. Unlike classification tasks that assign labels, face recognition operates within an embedding space where similarity is measured by comparing vector distances between faces. The goal is to determine whether the detected face matches a stored identity with sufficient similarity.

The face_recognition function in the Dlib library follows a structured sequence of steps: it begins with face detection using CNN or HOG, followed by the localization of 68 facial landmarks such as the eyes, nose, and mouth. The detected face is then aligned using affine transformation to ensure consistent orientation. Next, facial features are extracted through a ResNet-based convolutional neural network [21] and encoded into a 128-dimensional embedding vector. Finally, face comparison is performed using Euclidean distance, where smaller values indicate higher similarity. These distance values typically range between 0 and 1, and a default threshold is applied to determine whether two embeddings represent the same identity [22][23].

2.3. Data Collection

Each dataset used in this study underwent task-specific preprocessing. For the face detection model, facial images were manually captured in varying conditions, including different lighting angles and backgrounds. The annotation process was carried out using the LabelImg tool [24], with a single class labeled as faces, following the YOLO format. The labeling interface and bounding box annotation can be seen in Figure 3.

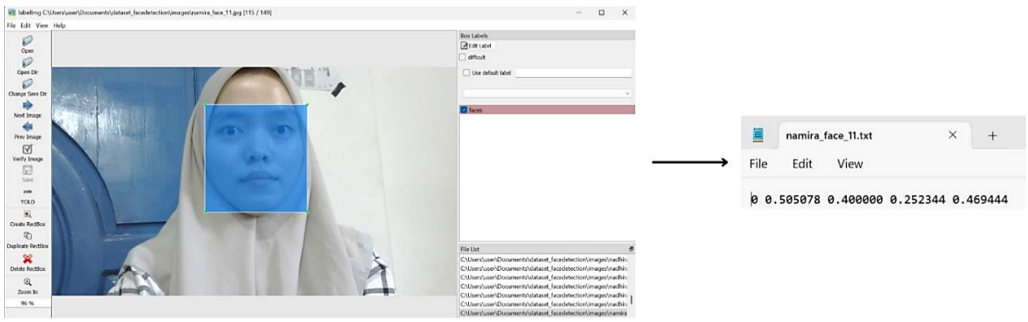


Figure 3. Face Bounding Box Annotation with LabelImg

The blink detection dataset was sourced from Roboflow and came pre-labeled in YOLO format. Annotations and folder structure were already compatible with the training pipeline, so the dataset was used without modification. It includes two object classes, "blink" and "attentive", which represent closed and open eyes respectively, as shown in Figure 4.

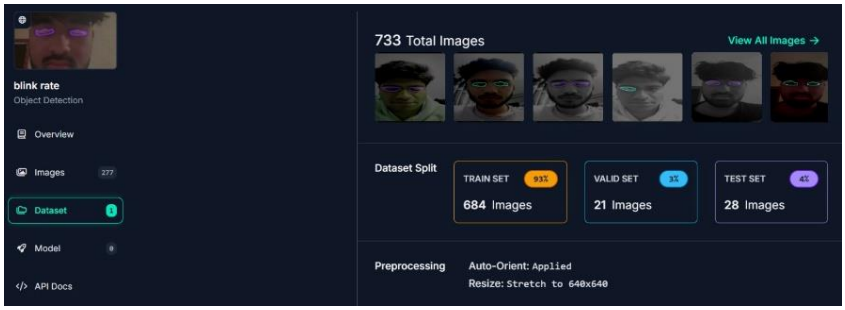


Figure 4. Blink Rate Dataset Roboflow

For face recognition, individual images were captured using a webcam in natural, unconstrained settings. Each image was then cropped to the face region using the detection model, and directly processed with the face_recognition library to generate a 128-dimensional embedding vector. These embeddings were stored as identity references [25], enabling real-time face matching without further training. The dataset structure is illustrated in Figure 5.

Figure 5. Cropped Face Images Used for Recognition

2.4. Model Evaluation

This study evaluates the model's performance using four key indicators: precision, recall, mAP_{50}, and mAP_{50-95}, each of which provides a distinct perspective on detection accuracy and reliability. As described in Equation (1), precision is calculated by comparing the number of correctly predicted positive samples to the total number of predicted positives. This metric helps determine how well the model avoids false positives by identifying only relevant objects..

Meanwhile, recall, as presented in Equation (2), measures the number of actual positive instances that were successfully detected. A high recall indicates that the model missed fewer target objects. Mean Average Precision at IoU 0.5 (mAP_{50}), described in Equation (3), is the mean of the average precision (AP) values computed at an Intersection over Union (IoU) threshold of 0.5 for each class. It summarizes the trade-off between precision and recall at a moderate overlap threshold. Meanwhile, mAP_{50-95}, defined in Equation (4), evaluates the model across a wider range of IoU thresholds, from 0.5 to 0.95 in increments of 0.05. This metric provides a more rigorous and comprehensive assessment by accounting for varying degrees of localization accuracy, particularly in challenging scenarios.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{2}$$

$$\text{mAP}_{50} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}} \text{AP}_{50,\text{i}} \tag{3}$$

$$\text{mAP}_{50-95} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}} \frac{1}{\text{K}}\sum_{\text{j=1}}^{\text{K}} \text{AP}_{\text{i,j}} \tag{4}$$

To support these metrics, the following definitions apply [20] : True Positive (TP): the number of positive objects correctly identified by the system; False Positive (FP): the number of objects mistakenly predicted as positive; False Negative (FN): the number of actual positives that the model failed to detect. N: the total number of object categories evaluated; AP_i: the average precision score for class i, derived from the area under the precision-recall curve; K: the total number of IoU thresholds used in mAP_{50-95}, usually set from 0.5 to 0.95 in 0.05 increments; AP_{ij}: the average precision for class i at IoU threshold j, representing accuracy under specific overlap constraints.

2.5. System Planning

This system is deployed as a real-time web interface. As illustrated in Figure 6, the process begins with the activation of the webcam to capture continuous video input. Each frame is analyzed using a YOLOv11-based face detection model to identify and crop the face region of interest. Once detected, the cropped face is simultaneously forwarded to two modules, face recognition and blink detection, ensuring that both identity and liveness are processed in parallel for faster verification.

The recognition module generates a 128-dimensional vector using the face_recognition library and compares it against stored embeddings in the local database. Meanwhile, the blink detection model evaluates the eye region to determine liveness. If both recognition and liveness are verified, the system logs the event with a timestamp and captured image, then grants access. The entire workflow runs in real time and is accessible through a web interface, offering both transparency for administrators and practicality for end users in everyday access control scenarios.

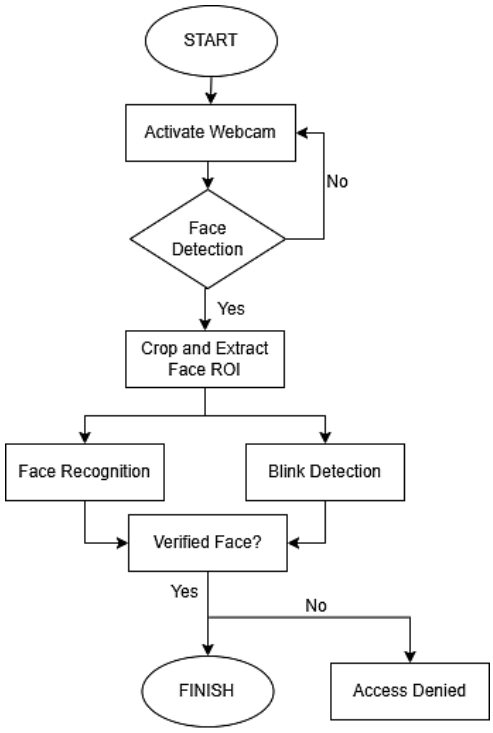


Figure 6. System Flowchart of Face and Blink-Based Access Verification

3. RESULTS AND ANALYSIS

This section presents the experimental results and analysis of the proposed system. Evaluation is based on detection accuracy, robustness across lighting conditions, and system responsiveness during real-time deployment.

3.1. Result and Analysis

To evaluate the performance of the implemented models, both the face detection and blink detection systems were trained and tested using the YOLOv11n architecture for 100 epochs. The training was conducted on an NVIDIA Tesla T4 GPU to ensure efficient computation.

The face detection algorithm was trained using a custom dataset consisting of 450 annotated facial images. To improve generalization, augmentation methods like horizontal mirroring and brightness adjustment were applied, followed by dividing the dataset into training and validation subsets. The resulting model delivered outstanding results, achieving a precision of 0.999, recall of 1.000, mAP_{50} of 0.995, and $mAP_{50–90}$ of 0.868. These metrics indicate a highly accurate and robust ability to detect facial regions from webcam input, with only a slight drop at higher IoU thresholds, showing that the model remains consistent even when stricter bounding box overlap is required.

For the blink detection model, training was performed on an open-source Roboflow dataset annotated with two classes: attentive and blink. After 100 epochs of training, the model achieved a precision of 0.959, recall of 0.962, mAP_{50} of 0.967, and $mAP_{50–90}$ of 0.678. While the overall detection remains strong, the relatively lower $mAP_{50–90}$ highlights the increased challenge of accurately detecting small eye regions at higher IoU thresholds. This effect is consistent with known limitations of object detectors when applied to small-scale features, where slight misalignments or lighting changes can impact bounding box accuracy. The evaluation results for both models are shown in Table 1.

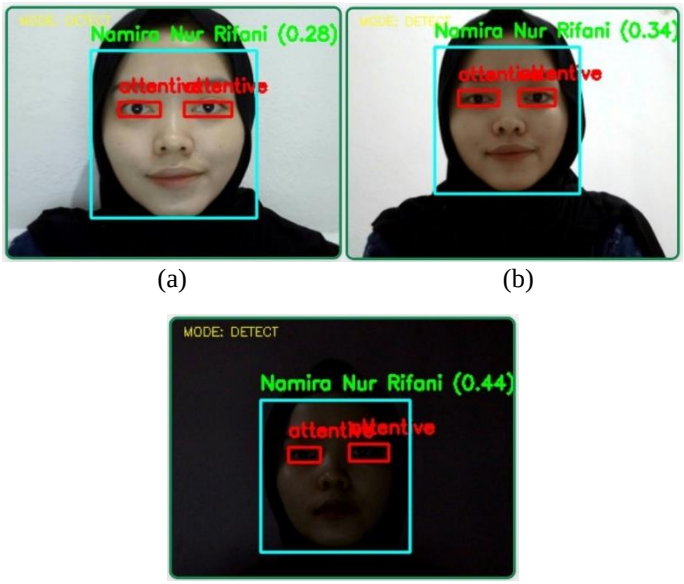
Table 1. YOLOv11 Evaluation Results

Model	Epoch	Precision	Recall	mAP_{50}	mAP_{50-90}
Face Detection	100	0.999	1	0.995	0.868
Blink Detection	100	0.959	0.962	0.967	0.678

To evaluate the consistency of facial recognition under different lighting conditions, Euclidean distance values between the detected face embeddings and registered identity vectors were analyzed. The system assigns a similarity score based on this distance, where lower values indicate a stronger match.

Figure 7 shows the detection results captured under three different lighting conditions: bright, moderate, and low. In each scenario, the Euclidean distance between the detected face and the stored embedding remained within the acceptable threshold, with values of 0.28 for bright light, 0.34 for moderate light, and 0.44 for low light. Despite the variation in illumination, all values stayed well below the standard threshold of 0.6, indicating successful recognition.

The slight increase in distance under low-light conditions reflects reduced feature clarity but remains within acceptable bounds for verification. These findings confirm that the system is reliable and accurate even in suboptimal lighting conditions, demonstrating its robustness for real-world indoor environments where lighting consistency cannot always be ensured.



(a) (b)

(c)

Figure 7. (a) Bright Light (b) Moderate Light (c) Low Light

In addition to the quantitative metrics, confusion matrices were generated for both models to provide further insight into classification performance. For the face detection model, the confusion matrix demonstrates perfect classification, with all 49 facial instances correctly detected and no background samples misclassified (Figure 8). This confirms the near-perfect performance already reflected in the precision and recall metrics, highlighting the robustness of the face detection pipeline in practical scenarios.

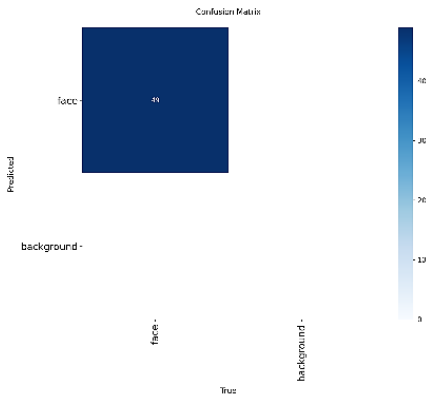


Figure 8. Confusion Matrix for Face Detection

For the blink detection model, the confusion matrix shows that the system correctly identified 16 attentive states and 25 blink states, with only 1 case of misclassification in each category (Figure 9). This indicates that the model is able to consistently differentiate between open and closed eye states with very few false positives or false negatives, further supporting its reliability for liveness verification.

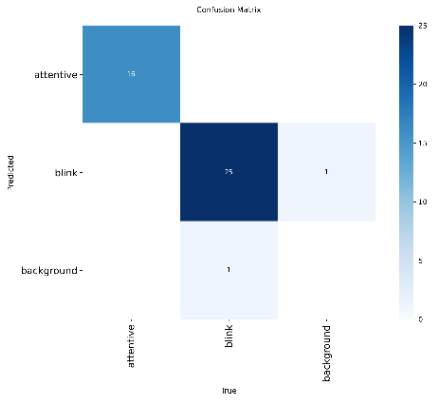


Figure 9. Confusion Matrix for Blink Detection

These confusion matrix results reinforce the numerical evaluation, demonstrating that both models not only achieve high accuracy in aggregate metrics but also maintain strong per-class consistency, which is critical for the dual verification process in real-time access control.

3.2. Deployment

To demonstrate the system's real-time functionality, a scenario was conducted where a registered user was detected and a valid blink was identified. As illustrated in Figure 10(a), the system successfully recognized the user's face, confirmed liveness through blink detection, and granted access. The interface displayed the user's name, marked both eyes as "blink," and recorded a snapshot along with a timestamp. This confirms that the system can accurately perform dual verification in real-time conditions.

Figure 10(b) shows the detection history interface, which logs each access attempt with relevant details including the user's name, time of detection, facial snapshot, and access result. This feature supports traceability and helps administrators review activity, monitor usage patterns, and perform post-event audits. The logging mechanism reinforces the system's reliability and provides a transparent overview of user interactions.

(a)

(b)

Figure 10. (a) Access Granted (b) Log History

3.3. Discussion

The proposed YOLOv11-based dual-model system demonstrated strong performance in both face and blink detection. The nearly perfect face detection results (precision 0.999, recall 1.000, mAP_{50} 0.995) confirm that YOLOv11 is highly effective for larger, well-defined features such as the human face. Meanwhile, the lower $mAP_{50–90}$ in blink detection (0.678) highlights the inherent difficulty of detecting

small-scale eye regions, where subtle variations in lighting, occlusion, or movement can impact bounding box accuracy. These outcomes emphasize the trade-off between feature size and detection precision in real-time systems.

Compared with prior YOLO-based studies, which reported mAP scores of 63.4 for YOLOv3 in custom face tasks and around 94–97% for YOLOv4 and YOLOv5 in safety monitoring scenarios, this research demonstrates that YOLOv11 maintains high accuracy while enabling real-time performance [10–13]. Importantly, the integration of blink-driven liveness verification addresses spoofing vulnerabilities noted in previous work [15–17], showing that simple "eye open/eye closed" cues can provide reliable protection without heavy computational cost. This combination positions the system as both practical and secure for modern access control applications.

4. CONCLUSION

This research developed a real-time access control system that combines YOLOv11-based face detection with blink-driven liveness verification. The novelty of the study lies in adopting a simple yet effective "eye open/eye closed" mechanism to ensure physical presence, thereby addressing spoofing vulnerabilities while maintaining computational efficiency. Experimental results confirmed high accuracy and robustness across varying lighting conditions, supported by a web-based interface for transparency and traceability. These findings demonstrate that the system is practical for smart indoor environments where both speed and security are critical. For future work, expanding the dataset, evaluating performance in outdoor or mobile contexts, and integrating multi-modal biometrics are recommended to further enhance reliability and scalability.

REFERENCES

[1] S. M. Arman, T. Yang, S. Shahed, A. Al Mazroa, A. Attiah, and L. Mohaisen, "A Comprehensive survey for privacy-preserving biometrics: Recent approaches, challenges, and future directions," Comput. Mater. Contin., vol. 78, no. 2, pp. 2087–2110, 2024, doi: 10.32604/cmc.2024.047870.

[2] Y. Motwani, S. Seth, D. Dixit, A. Bagubali, and R. Rajesh, "Multifactor door locking systems: A review," Elsevier, vol. 46, no. March, pp. 7973–7979, 2021, doi: 10.1016/j.matpr.2021.02.708.

[3] R. M. Ibrahim, M. M. Elkelany, and M. I. El-Afifi, "Trends in Biometric Authentication: A review," Nile J. Commun. Comput. Sci., vol. 6, no. December, pp. 1–12, 2023, [Online]. Available: https://njccs.journals.ekb.eg

[4] A. I. Awad, A. Babu, E. Barka, and K. Shuaib, "AI-powered biometrics for Internet of Things security: A review and future vision," J. Inf. Secur. Appl., vol. 82, no. March, p. 103748, 2024, doi: 10.1016/j.jisa.2024.103748.

[5] H. Hadi, H. Radiles, R. Susanti, and M. Mulyono, "Human Face Identification Using Haar Cascade Classifier and LBPH Based on Lighting Intensity," Indones. J. Artif. Intell. Data Min., vol. 5, no. 1, p. 13, 2022, doi: 10.24014/ijaidm.v5i1.15245.

[6] F. M. Sarimole and A. E. Septianto, "Implementation of IoT-Based Facial Recognition for Home Security System Using Raspberry Pi and Mobile Application," Int. J. Softw. Eng. Comput. Sci., vol. 4, no. 2, pp. 453–462, 2024, doi: 10.35870/ijsecs.v4i2.2554.

[7] D. P. Sari, M. A. C. Putra, and R. Kusumanto, "Implementasi Pengenalan Wajah Berbasis Cnn Dan Rfid Untuk Area Akses Aman Di Fasilitas Ruang," J. Teliska, vol. 18, no. Ii, pp. 23–31, 2024.

[8] M. Beldi, "Face Recognition using Deep Learning and TensorFlow framework," J. Comput. Sci. Inst., vol. 29, no. June, pp. 366–373, 2023.

[9] V. Gaikwad, D. Rathi, V. Rahangdale, R. Pandita, K. Rahate, and R. S. Rajpurohit, "Design and Implementation of IOT Based Face Detection and Recognition," Data Sci. Intell. Comput. Tech., pp. 923–933, 2024, doi: 10.56155/978-81-955020-2-8-78.

[10] S. M. M, A. Geroge, A. N, and J. James, "Custom Face Recognition Using YOLO.V3," 3rd Int. Conf. Signal Process. Commun., no. May, pp. 454–458, 2021.

[11] M. Muhaimin and T. W. Sen, "Real-Time Detection of Face Masked and Face Shield Using YOLO Algorithm with Pre-Trained Model and Darknet," Indones. J. Artif. Intell. Data Min., vol. 4, no. 2, pp. 97–107, 2021.

[12] F. Majeed et al., "Investigating the efficiency of deep learning based security system in a real-time environment using YOLOv5," Sustain. Energy Technol. Assessments, vol. 53, no. April 2023, 2022, doi: 10.1016/j.seta.2022.102603.

[13] Y. Y. Pane et al., "Motorcycle License Plate and Driver Face Verification Using Siamese Neural Network Model," vol. 8, no. 1, pp. 219–228, 2025.

[14] G. Jocher and J. Qiu, "Ultralytics YOLO11," 2024. https://docs.ultralytics.com/models/yolo11/

[15] M. Basurah, W. Swastika, and O. H. Kelana, "Implementation Of Face Recognition And Liveness Detection System Using Tensorflow.JS," JIP (Jurnal Inform. Polinema), pp. 509–516, 2023.

[16] Y. Wei, I. K. D. Machica, C. E. Dumdumaya, J. C. T. Arroyo, and A. J. P. Delima, "Liveness Detection Based on Improved Convolutional Neural Network for Face Recognition Security," Int. J. Emerg. Technol. Adv. Eng., vol. 12, no. 8, pp. 45–53, 2022, doi: 10.46338/ijetae0822_06.

[17] C. Gao, X. Li, F. Zhou, and S. Mu, "Face liveness detection based on the improved CnN with context and texture information," Chinese J. Electron., vol. 28, no. 6, pp. 1092–1098, 2019, doi: 10.1049/cje.2019.07.012.

[18] A. F. Rasheed and M. Zarkoosh, "YOLOv11 Optimization for Efficient Resource Utilization," 2024, [Online].

Available: http://arxiv.org/abs/2412.14790
[19] A. T. Khan and S. M. Jensen, "LEAF-Net: A Unified Framework for Leaf Extraction and Analysis in Multi-Crop Phenotyping Using YOLOv11," Agric., vol. 15, no. 2, pp. 0–10, 2025, doi: 10.3390/agriculture15020196.
[20] L. Deng, Y. Tan, D. Zhao, and S. Liu, "Research on object detection in remote sensing images based on improved horizontal target detection algorithm," Earth Sci. Informatics, vol. 18, no. 3, pp. 1–25, 2025, doi: 10.1007/s12145-025-01814-z.
[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
[22] A. A. F. Poeloemgam, Hendrawan, E. Mulyana, and W. Hermawan, "Web-based Face Detection and Recognition using YOLO and Dlib," Proceeding 2023 17th Int. Conf. Telecommun. Syst. Serv. Appl. TSSA 2023, pp. 1–6, 2023, doi: 10.1109/TSSA59948.2023.10366984.
[23] D. Zhang, J. Li, and Z. Shan, "Implementation of Dlib deep learning face recognition technology," Proc. - 2020 Int. Conf. Robot. Intell. Syst. ICRIS 2020, pp. 88–91, 2020, doi: 10.1109/ICRIS52159.2020.00030.
[24] B. Pande, K. Padamwar, S. Bhattacharya, S. Roshan, and M. Bhamare, "A Review of Image Annotation Tools for Object Detection," Proc. - Int. Conf. Appl. Artif. Intell. Comput. ICAAIC 2022, no. Icaaic, pp. 976–982, 2022, doi: 10.1109/ICAAIC53929.2022.9792665.
[25] N. Passalis et al., "Leveraging Active Perception for Improving Embedding-based Deep Face Recognition," 2021.

BIBLIOGRAPHY OF AUTHORS

Namira Nur Rifani, was born in Bandar Lampung, Indonesia, a fresh graduate from Politeknik Negeri Sriwijaya majoring in Electrical Engineering. Her research interests encompass the development of IoT systems and the implementation of machine learning in real-world applications.

Dr. RD Kusumanto, S.T., M.M., is a lecturer at the Department of Electrical Engineering, Politeknik Negeri Sriwijaya, Indonesia. He earned his Doctoral degree in Management Science from Universitas Persada Indonesia YAI. His research interests include electronics engineering, digital image processing, IoT systems, and management of technology.

Dr. Nyayu Latifah Husni, S.T., M.T., is currently a lecturer with the Department of Electrical Engineering, Politeknik Negeri Sriwijaya, Indonesia. She received her Doctoral degree in Engineering Science from Universitas Sriwijaya. Her research areas cover robotics, artificial intelligence, wireless sensor networks, and embedded systems