MATHEMATICAL MODELING AND COMPUTING, Vol. 11, No. 4, pp. 1093–1105 (2024)

Implementation of presence detection with Haar cascade and local binary patterns histograms

Elkari B.[1], Ourabah L.[1], Sekkat H.[1], Farah G.[1], Soufi I.[1],
Baddou A.[1], Hafidi N.[1], El Moutaouakil K.[2]

[1] *EIDIA, Euromed Research Center, Euro-Med University (UEMF), Fez, Morocco*
[2] *Engineering Science Laboratory, Polydisciplinary Faculty of Taza, Sidi Mohamed Ben Abdellah University of Fez, Morocco*

(Received 22 January 2024; Revised 6 November 2024; Accepted 13 November 2024)

School truancy is a significant problem that affects the educational environment and student achievement. This article presents a project to develop an automated absence detection system for classrooms using Haar Cascade and Local Binary Patterns Histogram (LBHP) techniques. The study begins by collecting a large dataset of classroom images, including various lighting scenarios and conditions. Haar Cascade is used to detect human faces in images, followed by LBHP feature extraction for each detected face. Experimental results demonstrate the effectiveness of the proposed system, achieving a high accuracy rate. This project contributes to the field of educational technology by providing a practical solution for monitoring classroom attendance. The integration of Haar Cascade and LBHP techniques provides robust and efficient performance in absence detection.

Keywords: *absence detection; Haar cascade; local binary patterns (LBP); computer vision.*

2010 MSC: 68T45 **DOI:** 10.23939/mmc2024.04.1093

1. Introduction

School attendance plays a pivotal role in the educational process, exerting a direct influence on the learning environment and academic achievements [1–4]. The maintenance of precise attendance records is crucial, serving not only to monitor student progress but also to guarantee their safety and identify potential issues or concerns [5–8]. Traditional manual methods for tracking attendance can prove to be both time-consuming and error-prone. As educational institutions grow in size and complexity, the need for efficient and reliable classroom absence detection becomes increasingly apparent. Hence, there is a pressing necessity to explore automated solutions that not only streamline the attendance-taking process but also enhance its accuracy and effectiveness. Automated systems, such as the one proposed in this article, contribute to creating a more secure and conducive learning environment by addressing the limitations associated with manual methods [9, 10].

In recent years, computer vision methodologies have demonstrated significant promise across diverse domains, encompassing applications like object recognition and detection [11–13]. Capitalizing on the capabilities of these techniques, the objective of this investigation is to create an automated absence detection system for classrooms by employing a synergistic approach involving Haar Cascade and Local Binary Patterns (LBP) [14–17]. Haar Cascade [4], a popular object detection algorithm, is known for its effectiveness in detecting objects based on specific features [18, 19]. LBP, on the other hand, is a texture analysis technique that captures local patterns in images [20–22].

In the following sections, the methodology employed for data collection will be discussed, alongside the implementation details of the absence detection system, the evaluation parameters utilized, and the results of the experimental evaluations. Additionally, the potential implications of this research will be highlighted, and suggestions for future improvements to refine and expand the capabilities of the automated absence detection system will be proposed.

This work was supported by UEMF.

2. Related works

Facial recognition, a biometric technology, uses mathematical techniques to extract facial features and record them as facial fingerprints for unique identification [23]. This technology, which has become widely used in various applications, including security systems and mobile platforms, works by comparing selected facial features with a database of faces [24]. Despite its lower accuracy compared to other biometric technologies, facial recognition is favored for its contactless and non-invasive process [24]. It has also been criticized for ethical concerns, but its use in airports has been shown to improve security and efficiency [24]. The technology has been further analyzed for its effectiveness and weaknesses, with potential applications in India [25]. Facial recognition technology can be classified into two different categories. On one hand, there are facial imaging techniques that focus on general structural features and are applied to the entire facial image or specific parts. On the other hand, there are also feature-based techniques that use geometric aspects of the face, such as the mouth, eyes, and eyebrows, and the geometric relationships between these elements [26]. Currently, facial recognition is used for various purposes, such as managing student attendance in universities [27]. The goal is to improve attendance recording systems by reducing errors associated with manual processes while providing a reliable attendance tracking system. Additionally, the technology makes it possible to regularly report absenteeism while preventing cases of false attendance, thereby improving privacy and security [28].

A range of studies have explored the use of Haar cascade classifiers for face recognition. Safiullina evaluated four types of Haar cascade classifiers and recommended $haarcascade\_frontalface\_alt$ and $haarcascade\_frontalface\_alt2$ for biometric systems [29]. In [30], the authors emphasized the efficient use of resources and high efficiency in their applications of Haar cascade classifiers for face recognition.

The integration of Haar Cascade and Local Binary Pattern Histogram in various applications has shown promising results. Mactal demonstrated the system's potential in smart attendance and intruder detection [31], while [32] achieved a high accuracy level in a door locking system. In [33], Al-Aidid further highlighted the system's ability to differentiate human faces from other objects and recognize faces from a database. In [34], Kumar expanded the system's application to a real-time web-based face recognition system for student attendance, emphasizing its precision and real-time performance. These studies collectively underscore the effectiveness of the Haar Cascade and Local Binary Pattern Histogram integration in presence detection.

In [35], Budiman et al. conducted a comparative analysis of algorithms suitable for academic environments, specifically focusing on CNN and LBPH. The findings revealed that, in contrast to LBPH, CNN exhibited superior accuracy and maintained a more stable performance even in the presence of external factors that could potentially impact accuracy.

In [36], Tej Chinimilli et al. concentrated on the development of facial recognition-based absence systems with the aim of reducing false positive rates. The system incorporated confidence thresholds based on Euclidean distance to enhance stranger recognition and image preservation accuracy. Face detection employed the Haar cascade method, while face recognition utilized the LBPH algorithm as part of an automated attendance management system. They also created a user-friendly graphical interface for image acquisition, dataset formation, and system integration. Consequently, the student facial recognition rate reached 77%, with a false positive rate of 28%. The success rate for facial recognition of an unknown person was nearly 66%, irrespective of threshold application. False positive rates with and without threshold application were 14% and 30%, respectively.

3. The proposed approach and methodology

Face detection and recognition, integral to machine learning, involve the extraction and analysis of features from the human body, as illustrated in Figure 1. These features are then compared with test images to either identify specific individuals or ensure their anonymity. This rapidly advancing identification technology, surpassing the reliability of the human eye, employs machine learning algorithms within facial recognition systems capable of discerning faces in both simple images and videos. While certain algorithms specialize in high- or low-resolution images, current research emphasizes capturing various frontal views of faces under different angles and lighting conditions. Despite algorithmic diversity, there exists a shared architectural framework and procedural workflow among these systems.

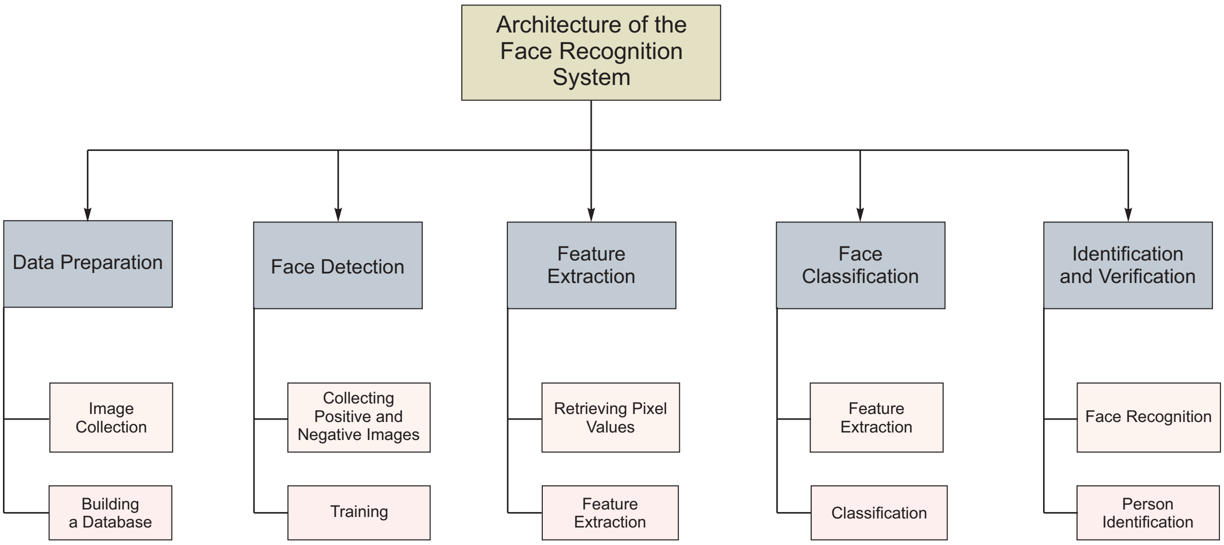

Fig. 1. Facial recognition system architecture.

As seen in Figure 2 the facial recognition process involves several key steps to accurately identify and verify individuals in images or videos. Initially, the facial recognition system undertakes data preparation, wherein it collects representative images covering various lighting conditions, angles, and facial expressions. The collected data is then preprocessed by resizing images, converting them to grayscale, and employing normalization techniques to enhance quality and consistency. Additionally, the data is labeled to associate each face with a corresponding identifier, facilitating model training. Subsequently, face detection becomes crucial, marking the initial phase of automatic face recognition. Specific algorithms, such as "Haar" and the LBHP classifier, are employed to identify and localize faces in real-time video streams. These algorithms analyze distinctive features like general shape, eyes, nose, and face to detect and frame faces within each frame of the video stream.

Feature extraction follows face detection, where key information distinguishing individuals is obtained from recognized faces. This step involves analyzing facial regions, including eyes, nose, mouth, as well as features like facial shape and skin structure. The precise location of key facial features, such as eyes, nose, and mouth, is crucial for normalization in global matching methods like eigenfaces and Fisherfaces.

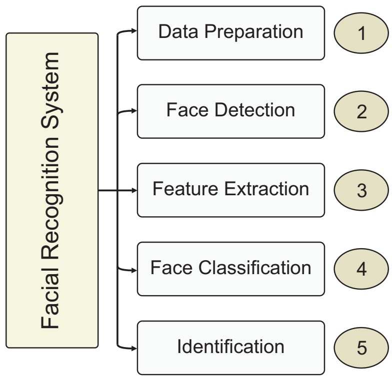

Fig. 2. Facial recognition steps.

Face classification is a pivotal step, where labels or identifiers are assigned to detected faces based on the extracted facial features. This information is then input into a classification algorithm. Facial recognition identification, the final step, associates the recognized face with a specific person. Utilizing machine learning techniques and pre-trained models, the system compares detected faces with those stored in the database, employing mathematical calculations like Euclidean distance and maximum likelihood. Confidence thresholds aid in making decisions about corresponding identities, answering the question "Who does this face belong to?" and determining the most likely identity associated with the detected face.

Comparison of these features with test images aids in identifying individuals or preventing mutual recognition. The utilization of cubes and variable parameters in face detection and recognition accommodates factors such as lighting, diverse poses, facial expressions, and poor-quality input images. Facial recognition systems exhibit varied perspectives, with some projects concentrating solely on high-resolution images and others on low resolution. Notably, recent research has focused on capturing different frontal views of images from diverse angles and lighting conditions. The following sections delineate the methodology for data collection, implementation details of the absence detection system, the evaluation parameters employed, and the results of the experimental assessment.

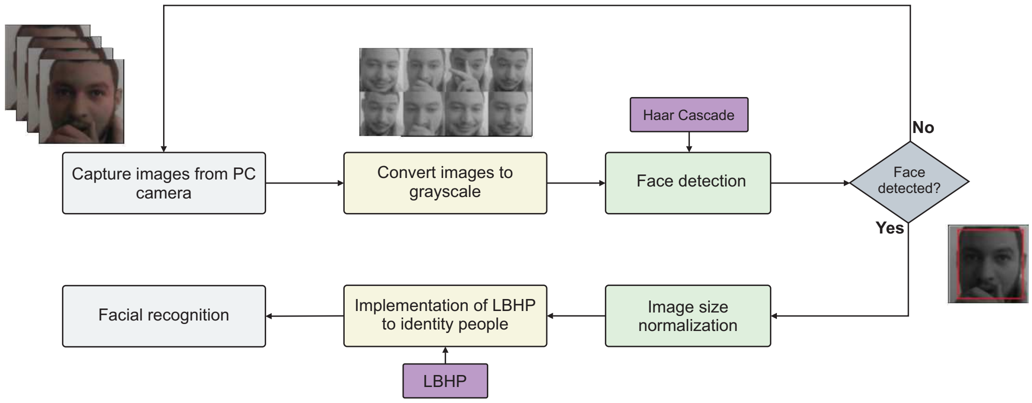

Fig. 3. The proposed approach architecture.

Moreover, the study underscores the potential impact of automatic absence detection systems and suggests avenues for future enhancements to refine and extend their capabilities, as illustrated in Figure 3. Absence detection, also referred to as presence/absence detection, constitutes a significant research area in computer vision, determining the presence or absence of objects or individuals in a given scene. This method incorporates two common absence detection techniques: the Haar Cascade and Local Binary Pattern (LBP) methods.

3.1. Data preparation

For the training of classification files in the machine learning system, a database has been essential. Public databases, such as the LFW database containing over 13 000 facial images from more than 5 700 individuals, have been employed to conduct facial recognition research tests. These images, captured in real-life conditions, showcase variations in poses, facial expressions, lighting, and backgrounds. Faces other than the target face have been treated as "background" [37]. Additionally, a proprietary database

Fig. 4. Database preprocessing for facial recognition: grayscale conversion and normalization.

has been generated, comprising two-dimensional optical projections of faces in a world visually perceived as three-dimensional. Vision inherently adopts an "inverted perspective", requiring the inversion of the 3D-2D projection to discern object properties in image space. It is crucial to note that the strictly mathematical 2D-3D inversion of such projections is not possible. The dataset has undergone preprocessing for the feature extraction process, involving the conversion of images to grayscale and normalization to enhance recognition outcomes (see Figure 4).

3.2. Facial detection

The identification of human faces in images or videos, known as face detection, constitutes a foundational stage in the evolution of facial recognition systems. This process involves the exploration and recognition of the existence of faces within visual content, aiming to pinpoint image areas that contain faces and subsequently extract this pertinent information for subsequent analysis.

3.3. The Haar cascade classifier

Rectangular features are computed by subtracting the sum of pixel intensities in white rectangles R_i^+ from the sum in black rectangles R_i^-. A Haar-like feature $H(x,y)$ is formed by combining these rectangles with corresponding weights w_i. p and q typically represent the horizontal and vertical coordinates, respectively, within the rectangular regions that make up the Haar-like features. The expression represents the sum of pixel intensities in the white rectangle (see equation (1)), and similarly equation (2) represents the sum of pixel intensities in the black rectangle,

$$\sum_{(p,q)\in R_i^+} I(x+p, y+q), \tag{1}$$

$$\sum_{(p,q)\in R_i^-} I(x+p, y+q), \tag{2}$$

where $I(x,y)$ represents the pixel intensity at location (x,y). The Haar-like feature is mathematically expressed as equation:

$$H(x,y) = \sum_i w_i \Bigg(\sum_{(p,q)\in R_i^+} I(x+p, y+q) - \sum_{(p,q)\in R_i^-} I(x+p, y+q) \Bigg) \leqslant T. \tag{3}$$

Here, T is the threshold for classification, and the summation is carried out over the pixels within the defined rectangles.

The Haar Cascade algorithm takes its name from fact that it uses Haar features to detect objects as seen in Figure 5. These features are simple rectangular patterns that detect changes in light intensity in an image. Using a set of Haar features and a classifier like AdaBoost, the Haar cascade algorithm can learn to recognize specific objects efficiently.

The Haar cascade training process involves weak classifiers $h_j(x)$, where x represents the input features, distinguishing between positive $(+1)$ and negative (-1) examples. Each weak classifier is determined by a threshold condition as seen equation (4):

$$h_j(x) = \begin{cases} +1 \text{ if } p_j(x) < \theta_j, \\ -1 \text{ otherwise}, \end{cases} \tag{4}$$

where the $p_j(x)$ is the response of the j-th weak classifier to input x, and θ_j is the corresponding threshold. The strong classifier $H(x)$ is then formed as a linear combination of these weak classifiers:

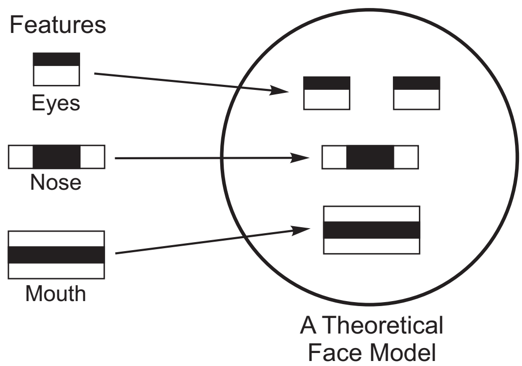

Fig. 5. Haar features.

$$H(x) = \sum_{j=1}^{T} \propto_j h_j(x), \tag{5}$$

where T is the total number of weak classifiers, and $\propto_j$ represents the weight assigned to the j-th weak classifier based on its performance. The boosting process adjusts the weights of individual examples (see equation (6)):

$$\omega_i^{(t+1)} = \omega_i^{(t)} \cdot \exp\big(-\alpha_t y_i h_t(x_i)\big). \tag{6}$$

The y_i is the true label of example i, and t denotes the iteration of the boosting process. The Adaboost algorithm provides weights for each weak classifier as mentioned in equation (7),

$$\alpha_t = \frac{1}{2}\ln\frac{1-\varepsilon_t}{\varepsilon_t}, \tag{7}$$

where ε_t represents the weighted error of the j-th weak classifier. The cascade, trained in stages (S), is implemented with a sliding window during detection, scanning the image to identify faces:

$$H(x) = \sum_{s=1}^{S}\sum_{j=1}^{T_s} \alpha_j^{(s)} h_j^{(s)}(x). \tag{8}$$

Here, S is the total number of stages, and T_s denotes the number of weak classifiers in the s-th stage.

The strong classifiers are arranged in a cascade, so that each classifier only processes those regions of the image that have passed the previous stages. This helps speed up the detection process by quickly eliminating regions that do not contain probably not the object of interest as seen in Figure 6.

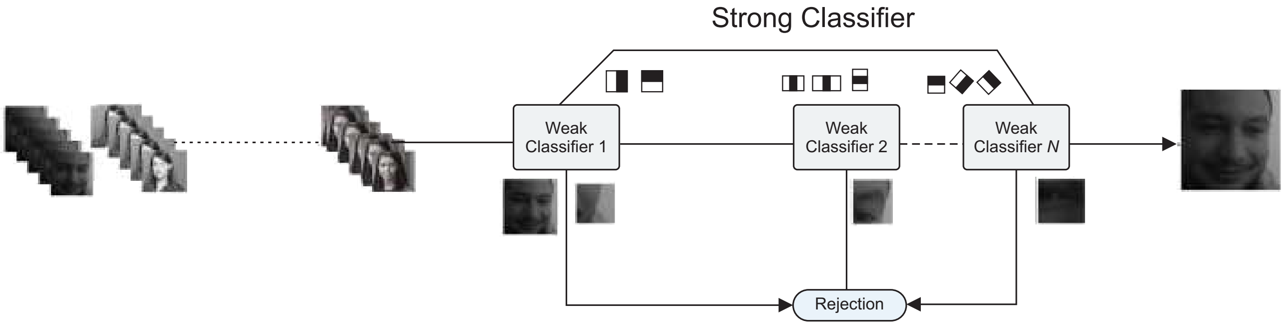

Fig. 6. Cascade of classifier.

The Sliding window, as illustrated in Figure 7, has been employed in the Haar Cascade object detection method. It has facilitated the examination of an image by analyzing subregions of varied sizes and positions to identify the presence of a target object. This technique operates by scanning the image with a fixed-size rectangular window, commonly referred to as a "detection window". The window systematically slides across the image pixel by pixel, executing Haar feature calculations at each position.

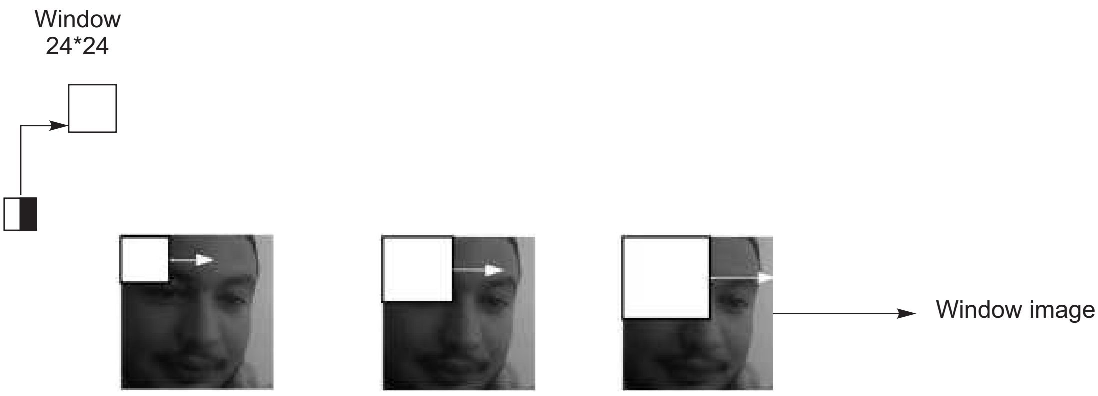

Fig. 7. Illustration of sliding window technique in Haar cascade object detection.

The Haar features have been calculated and compared to a previously trained model at each window position, employing a supervised learning algorithm (Figure 8). If the match proves close enough, it signifies the potential presence of the searched object, such as a face. In such instances, the detection window is marked as positive.

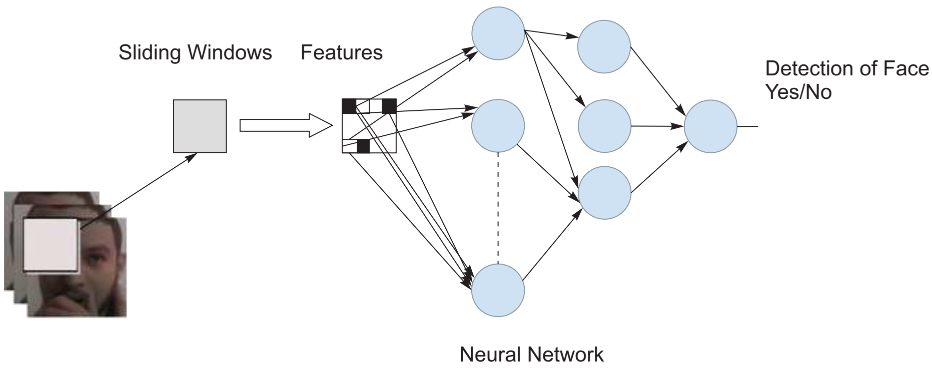

Fig. 8. Illustration of Haar features calculation and comparison in window positions for object detection.

3.4. Local binary pattern (LBP)

The Local Binary Pattern (LBP) method serves as a robust technique for characterizing textures within images, finding extensive applications in diverse fields of computer vision, including facial recognition and object detection. It operates by comparing the intensity values of pixels neighboring a central pixel, generating binary patterns that depict local variations in intensity. These patterns are subsequently aggregated into a histogram, providing a concise representation of the image's texture.

LBP computes a binary pattern for each pixel by comparing its gray value g_c with the gray values of its p neighboring pixels (g_p). The binary pattern is then converted into a decimal representation. The LBPH value at pixel (x, y) is given by the following equation:

$$L(x,y) = \sum_{p=1}^{P-1} 2^p \cdot \delta(g_p - g_c), \tag{9}$$

where P represents the number of neighboring pixels, and $\delta(\cdot)$ is the Kronecher delta function, ensuring that the binary comparison results in either 0 or 1. This process is applied to each pixel in the image, generating a spatial representation of local texture patterns. The resulting histograms of LBPH values capture the frequency distribution of these patterns, enabling effective texture-based analysis and object recognition.

For effective implementation with color images, an initial conversion to grayscale is necessary, as LBP operates exclusively on pixel intensities rather than colors. Subsequently, the image is divided into smaller cells, typically 8×8 pixels in size. This division facilitates local processing, enabling the capture of distinct local characteristics.

The LBP operator is applied to each cell by selecting a central pixel and comparing its intensity with that of its neighboring pixels within a predefined neighborhood. As an example, a circular neighborhood might consist of eight neighboring pixels as seen in Figure 9. The binary values obtained from these comparisons are concatenated to form an 8-bit binary number, which is then converted to a decimal value, representing the Local Binary Pattern Histogram (LBPH) of the central pixel. This process is repeated for all central pixels within the cell.

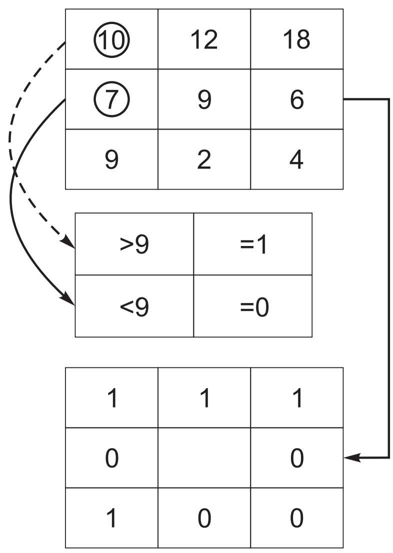

Fig. 9. Calculated binary values based on condition of pixel neighbors.

As seen in Figure 10, to create an overall image descriptor, the histograms of all cells are concatenated, forming a feature vector suitable for tasks like classification or object recognition. In practical terms, this involves repeating the entire process for each cell in the image. The resultant feature vectors, once extracted for all images in a dataset, can be utilized for training classifiers (e.g., SVM, k-NN) or for direct comparisons, enabling the recognition of similar objects or textures.

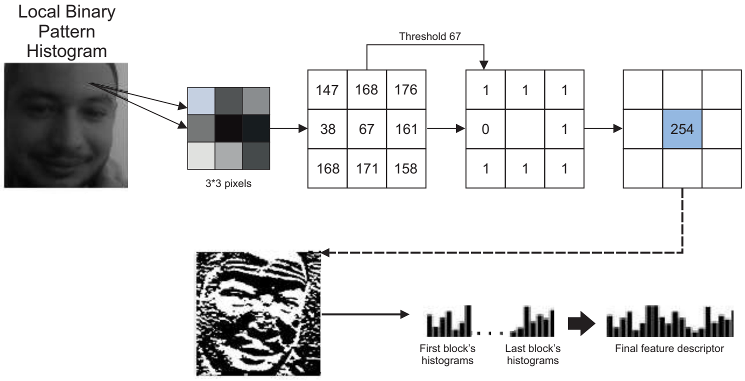

Fig. 10. Local binary patterns.

LBPH, owing to its simplicity and effectiveness, finds widespread application in various computer vision tasks, including but not limited to texture recognition, face recognition, and object classification. Its adaptability and performance make it a valuable tool in the arsenal of image analysis techniques.

4. Results and discussion

The proposed facial recognition approach was implemented in Python, utilizing the Opencv image processing library, the Haar classifier, and the LBPH algorithm with the computer's camera. The face detection project employed a pre-trained Haar Cascade model, which learned to recognize distinctive facial features from a diverse dataset of positive (face) and negative (background) examples. Python served as the primary programming language in our working environment due to its simplicity, flexibility, and rich selection of image processing libraries and modules. Specifically, the OpenCV (Open Source Computer Vision) library was employed for advanced image processing functions, including object detection, motion tracking, and more.

Utilizing the Haar Cascade model involved importing the OpenCV library and loading the xls file containing pre-trained model information. The model was then applied to each image or video frame for face detection, enabling subsequent tasks like facial recognition and expression identification. To ensure optimal performance, an integrated development environment (IDE) such as 'PyCharm' was employed.

The algorithm processed live video input with visible student faces, producing an attendance sheet in Excel format. The steps included applying the Haar cascade classifier to detect faces, extracting features using the LBPH algorithm, and comparing these features with those in the database for identification. If a match was found, student attendance was recorded on the attendance sheet. The process was repeated for each frame, resulting in the generation and export of the attendance sheet.

This algorithm facilitated face detection, feature extraction, and comparison with a database for live video scenarios and student face detection in photos. Trust assessment was conducted using the LBPH Recognizer method. However, facial recognition's complexity and potential ambiguity, especially in similar facial features, were acknowledged. To enhance system reliability, a diverse training database was proposed, representing various facial variations. Increasing the database's diversity improved the model's predictive accuracy.

To bolster the facial recognition system, adjustments were made to simulate real-world conditions. Varying image brightness and rotation levels increased the system's resilience to lighting changes and enhanced its ability to recognize faces in different orientations. These improvements significantly enhanced system performance and reliability, providing more accurate predictions by combining a diverse training database with brightness and rotation variations.

The enhancement of our facial recognition system's performance and reliability is notably achieved through the effective adjustment of brightness and rotation in images. Our approach involves integrating a substantial and diverse training database, incorporating variations in brightness and rotation.

This strategy facilitates improved generalization and enhances the accuracy of predictions, as evidenced by the findings presented in Table 1.

Table 1. The increase in precision compared to increase the database.

	Number of images	Confidence
First Experience	50	37
Second Experience	1000	77
Third Experienceg	3000	99

To create the application's graphical user interface (GUI), the tkinter library has been utilized [38]. This library enables the creation of a graphical interface featuring buttons, text boxes, and various graphical tools. This design allows users to interact with the application easily, without the need to input or comprehend lines of code. The face recognition application's GUI is designed with simplicity, incorporating interaction buttons for multiple tasks, an access control system with a keyword, and a material selection functionality.

Equipped with an access control system, the application ensures secure space entry through facial recognition to identify authorized individuals. A noteworthy system feature involves the ability to designate a specific subject before initiating facial recognition, facilitating the association of recognition results with a particular subject. This feature proves beneficial for attendance management at academic or professional events.

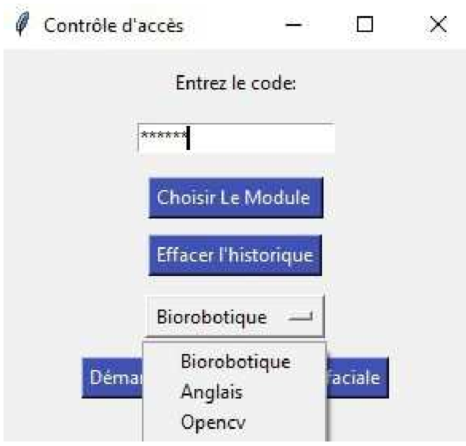

Fig. 11. The graphical interface developed.

Upon prompting the user to input an access code for material selection, a correct entry triggers the display of a drop-down list containing available subjects. As seen in Figure 11, this allows users to easily select the corresponding subject, enhancing the overall functionality of the system.

Observing Figure 12, it is evident that when a student is recognized, their name is presented on the screen alongside a confidence percentage reflecting the accuracy of the recognition.

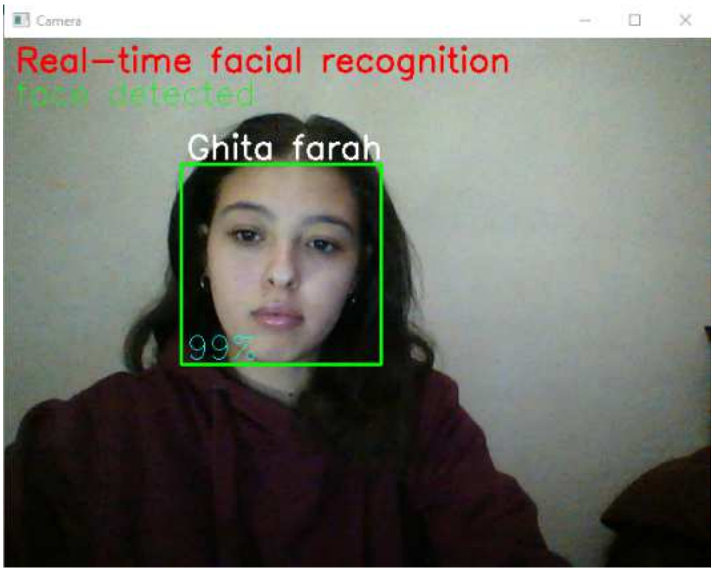

Fig. 12. Student facial recognition.

The outcomes of facial recognition, encompassing details such as date, time, the recognized person's name, presence status, and associated subject, are stored in an Excel file. Figure 13 provides a visual representation of this process. The system autonomously generates the Excel file if it is not already in existence; otherwise, it appends new results to the existing spreadsheet. This feature facilitates the maintenance of an attendance history, enabling subsequent analyses as needed.

	A	B	C	D	E	F
1	Date	Hour	Name	Last Name	Presence	Subject
2	30/11/2023	08:30:03	Ghita	Farah	Present	Opencv
3	30/11/2023	08:30:00	Mouaad	Oujabour	Absent	Opencv
4	30/11/2023	08:30:00	Aymane	Baddou	Present	Opencv
5	30/11/2023	08:30:00	Ilhame	Soufi	Absent	Opencv
6	30/11/2023	08:30:10	Nouhaila	Hafidi	Present	Opencv
7	30/11/2023	08:30:00	Chaimae	Dehhani	Absent	Opencv
8	30/11/2023	08:30:01	Mouad	Ouhasni	Present	Opencv
9	30/11/2023	08:30:07	Achraf	Berriane	Present	Opencv
10	30/11/2023	08:30:01	Achraf	Rachid	Present	Opencv
11	30/11/2023	08:30:00	Imad	Naciri	Absent	Opencv
12	30/11/2023	08:30:03	Chaimae	khani	Present	Opencv
13	30/11/2023	08:30:00	Hassana	Zenkouar	Absent	Opencv
14	30/11/2023	08:30:00	Ayoub	Hsaine	Present	Opencv
15	30/11/2023	08:30:00	Yassine	Bouargane	Absent	Opencv
16	30/11/2023	08:30:03	Chama	Essaiouad	Present	Opencv
17	30/11/2023	08:30:04	Oumaima	Mousaoui	Present	Opencv
18	30/11/2023	08:29:45	Zineb	Tadlaoui	Present	Opencv
19	30/11/2023	08:30:00	Hanaa	Aqboub	Present	Opencv
20	30/11/2023	08:30:01	Imane	Diouri	Present	Opencv
21						

Fig. 13. Saving as an Excel file.

5. Conclusion

The absence detection methodology introduced in this paper leverages the Haar Cascade and Local Binary Patterns (LBP) techniques, both proven effective in detecting objects and patterns in images. Through the integration of these approaches, a robust absence detection system has been developed, capable of accurately discerning the presence or absence of objects or individuals within a given scene. Notwithstanding its efficacy, it is imperative to acknowledge the inherent limitations of this methodology. Its performance is intricately tied to the quality of the training data and the precise selection of parameters during classifier learning. Achieving optimal results necessitates a comprehensive collection of representative positive and negative images. Furthermore, the fine-tuning of classifier parameters and LBP features demands extensive experimentation to optimize system performance.

Looking toward the future, two compelling perspectives emerge for further exploration. Firstly, the integration of deep learning techniques, specifically convolutional neural networks, holds significant promise for enhancing absence detection accuracy. These models, with their capacity to autonomously extract discriminative features from data, offer considerable advantages in addressing the complexities of diverse scenarios. Secondly, the augmentation of the system with contextual and temporal information emerges as a key enhancement. Techniques such as object tracking, which considers the spatial and temporal coherence of detected objects, can effectively reduce false alarms and enhance system robustness in dynamic environments.

Exploring these perspectives not only contributes to the evolution of absence detection technology but also opens avenues for novel applications. This methodology, despite its challenges and limitations, serves as a foundational framework. It provides a stepping stone for future research endeavors aimed at refining the accuracy and efficiency of absence detection systems. In doing so, it catalyzes advancements in the field of computer vision, paving the way for innovative applications and progress in absence detection methodologies.

[1] Durán-Narucki V. School building condition, school attendance, and academic achievement in New York City public schools: A mediation model. Journal of Environmental Psychology. **28** (3), 278–286 (2008).

[2] Tang Y. M., Chen P. C., Law K. M. Y., Wu C. H., Lau Y., Guan J., He D., Ho G. T. S. Comparative analysis of Student's live online learning readiness during the coronavirus (COVID-19) pandemic in the higher education sector. Computers & Education. **168**, 104211 (2021).

[3] Law K. M. Y., Geng S., Li T. Student enrollment, motivation and learning performance in a blended learning environment: The mediating effects of social, teaching, and cognitive presence. Computers & Education. **136**, 1–12 (2019).

[4] Chemlal Y., Azouazi M. Implementing quality assurance practices in teaching machine learning in higher education. Mathematical Modelling of Computing. **10** (3), 660–667 (2023).

[5] George B., Wooden O. Managing the strategic transformation of higher education through artificial intelligence. Administration Sciences. **13** (9), 196 (2023).

[6] Kearney C. A., Childs J. Improving school attendance data and defining problematic and chronic school absenteeism: The next stage for educational policies and health-based practices. Preventing School Failure: Alternative Education for Children and Youth. **67** (4), 265–275 (2023).

[7] Mishra S., Tyagi A. K. The role of machine learning techniques in internet of things-based cloud applications. Artificial Intelligence–based Internet of Things Systems. 105–135 (2022).

[8] Alam A. Platform utilising blockchain technology for eLearning and online education for open sharing of academic proficiency and progress records. Smart Data Intelligence. 307–320 (2022).

[9] Khoroshchuk D., Liubinskyi B. B. Machine learning in lung lesion detection caused by certain diseases. Mathematical Modelling of Computing. **10** (4), 1084–1093 (2023).

[10] Bellaj K., Benmir M., Boujena S. Enhancing image inpainting through image decomposition and deep neural networks. Mathematical Modelling of Computing. **10** (3), 720–732 (2023).

[11] Li Y., Mao H., Girshick R., He K. Exploring plain vision transformer backbones for object detection. Computer Vision – ECCV 2022. 280–296 (2022).

[12] Joseph K. J., Khan S., Khan F. S., Balasubramanian V. N. Towards open world object detection. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 5830–5840 (2021).

[13] Zou Z., Chen K., Shi Z., Guo Y., Ye J. Object detection in 20 years: A survey. Proceedings of the IEEE. **111** (3), 257–276 (2023).

[14] Shetty A. B., Bhoomika, Deeksha, Rebeiro J., Ramyashree. Facial recognition using Haar cascade and LBP classifiers. Global Transitions Proceedings. **2** (2), 330–335 (2021).

[15] Anand A., Jha V., Sharma L. An improved local binary patterns histograms techniques for face recognition for real time application. International Journal of Recent Technology and Engineering. **8** (2S7), 524–529 (2019).

[16] Pravallika D. S., Sai V. M., Sahithi D. L., Rishitha K., Ismail M. B. Face Expression Recognition by Hybrid Local Binary Pattern with Haar Cascade Method. Solid State Technology. **63** (6), 12919–12927 (2020).

[17] Sharma A., Shah K., Verma S. Face recognition using Haar cascade and local binary pattern histogram in OpenCV. 2021 Sixth International Conference on Image Information Processing (ICIIP). **6**, 298–303 (2021).

[18] Lopez-Tejeida S., Soto-Zarazua G. M., Toledano-Ayala M., Contreras-Medina L. M., Rivas-Araiza E. A., Flores-Aguilar P. S. An Improved Method to Obtain Fish Weight Using Machine Learning and NIR Camera with Haar Cascade Classifier. Applied Sciences. **13** (1), 69 (2022).

[19] Yeh J.-F., Lin K.-M., Chang C.-C., Wang T.-H. Expression Recognition of Multiple Faces Using a Convolution Neural Network Combining the Haar Cascade Classifier. Applied Sciences. **13** (23), 12737 (2023).

[20] Vu H. N., Nguyen M. H., Pham C. Masked face recognition with convolutional neural networks and local binary patterns. Applied Intelligence. **52** (5), 5497–5512 (2022).

[21] Tasci B., Tasci G., Ayyildiz H., Kamath A. P., Barua P. D., Tuncer T., Dogan S., Ciaccio E. J., Chakraborty S., Acharya U. R. Automated schizophrenia detection model using blood sample scattergram images and local binary pattern. Multimedia Tools and Applications. **83**, 42735–42763 (2024).

[22] Zhang Z., Wang M. Multi-feature fusion partitioned local binary pattern method for finger vein recognition. Signal, Image and Video Processing. **16** (4), 1091–1099 (2022).

[23] Nigam H., Abbas M. N., Tiwari M., Shalaj H. M., Hasib M. N. Review of Facial Recognition Techniques. International Journal of Research and Applied Sciences Engineering and Technology. **10** (1), 1740–1743 (2022).

[24] Petrescu R. V. V. Face recognition as a biometric application. Journal of Mechatronics and Robotics. **3**, 237–257 (2019).

[25] Thorat S. B., Nayak S. K., Dandale J. P. Facial recognition technology: An analysis with scope in India. Preprint arXiv:1005.4263 (2010).

[26] Kamencay P., Benco M., Mizdos T., Radil R. A new method for face recognition using convolutional neural network. Advances in Electrical and Electronics Engineering. **15** (4), 663–672 (2017).

[27] Agarwal H., Verma G., Gupta L. Student attendance system based on the face recognition. Asian Journal of Convergence in Technology. **7** (2), 70–73 (2021).

[28] Hartanto R., Adji M. N. Face recognition for attendance system detection. 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE). 376–381 (2018).

[29] Safiullina L. Kh., Gabdullin A. S., Anikin I. V. Face recognition in biometric systems using HAAR cascade classification. 2021 Dynamics of Systems, Mechanisms and Machines (Dynamics). 1–5 (2021).

[30] Madan A. Face recognition using Haar cascade classifier. International Journal of Modern Trends in Science and Technology. **7** (01), 85–87 (2021).

[31] Mactal T. M. M., Paglinawan C. C., Bantegui J. A. K. M. Application for Integration of Haar Cascade and Local Binary Pattern Histogram. CNIOT'23: Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things. 81–86 (2023).

[32] Ian Haikal Amir A., Raihan Nugroho P., Rahayu W. R., Febrianty D. F., Farihah N., Azizah W. N., Setiadi I. C., Gultom Y. D. Face detection and recognition in real-time photos with Haar cascade and local binary pattern histogram for automatic door locking system. 4th International Seminar on Photonics, Optics, and Its Applications (ISPhOA 2020). **11789**, 1178908 (2021).

[33] Al-Aidid S., Pamungkas D. Sistem Pengenalan Wajah dengan Algoritma Haar Cascade dan Local Binary Pattern Histogram. Jurnal Rekayasa Elektrika. **14** (1), 62–67 (2018).

[34] Kumar A., Singh D. Comprehensive approach of real-time web-based face recognition system using Haar Cascade and LBPH algorithm. 2023 International Conference on Device Intelligence, Computing and Communication Technologies (DICCT). 371–376 (2023).

[35] Budiman A., Yaputera R. A., Achmad S., Kurniawan A. Student attendance with face recognition (LBPH or CNN): Systematic literature review. Procedia Computer Science. **216**, 31–38 (2023).

[36] Chinimilli B. T., Anjali T., Kotturi A., Kaipu V. R., Mandapati J. V. Face recognition-based attendance system using Haar cascade and local binary pattern histogram algorithm. 2020 4th international conference on trends in electronics and informatics (ICOEI). 701–704 (2020).

[37] Mayr A., Binder H., Gefeller O., Schmid M. The evolution of boosting algorithms. Methods of Information in Medicine. **53** (06), 419–427 (2014).

[38] Hunt J. Tkinter GUI Library. Advanced Guide to Python 3 Programming. 155–168 (2023).

Реалізація виявлення присутності за допомогою каскаду Хаара та гістограм локальних бінарних шаблонів

Елькарі Б.[1], Ураба Л.[1], Секкат Х.[1], Фарах Г.[1], Суфі І.[1], Бадду А.[1], Хафіді Н.[1], Ель Мутауакіл К.[2]

[1] *EIDIA, Дослідницький центр Euromed, Університет Euro-Med (UEMF), Фес, Марокко*
[2] *Лабораторія інженерних наук, Полідисциплінарний факультет Таза, Університет Сіді Мохамеда Бен Абделла, Фес, Марокко*

Прогули школи є значною проблемою, яка впливає на освітнє середовище та успішність учнів. У цій статті представлено проєкт розробки автоматизованої системи виявлення відсутності в класах із використанням методів каскаду Хаара та гістограми локальних бінарних шаблонів (LBHP). Дослідження починається зі збору великого набору даних зображень класа, включаючи різні сценарії та умови освітлення. Каскад Хаара використовується для виявлення людських облич на зображеннях з подальшим виділенням ознак LBHP для кожного виявленого обличчя. Результати експериментів демонструють ефективність запропонованої системи та досягнення високого показника точності. Цей проєкт робить внесок у сферу освітніх технологій, надаючи практичне рішення для моніторингу відвідуваності класа. Інтеграція методів каскаду Хаар та LBHP забезпечує надійну та ефективну роботу для виявлення відсутності.

Ключові слова: *виявлення відсутності; каскад Хаара; локальні бінарні шаблони (LBP); комп'ютерний зір.*