International Journal of Multilingual Education and Applied Linguistics
Vol.1, No.4 November 2024
e-ISSN :3047-907X; p-ISSN :3047-9061, Page 01-10
DOI: https://doi.org/10.61132/ijmeal.v1i4.94
Avalable online at: https://international.aspirasi.or.id/index.php/IJMEAL

Implementation of Automatic Attendance System Based on Face Recognition Using CNN Method in IPB University Vocational School Environment

Dini Nurul Azizah[1], Raisa Mutia Thahir[2], Luthfi Dika Chandra[3], Muhammad Naufal Ardhani [4], Endang Purnama Giri[5], Gema Parasti Mindara[6]
[1,2,3,4,5] IPB University, Indonesia

Address: Jl. Kumbang No.14, RT.02/RW.06, Bogor - Jawa Barat 16128
Corresponding author : dininurulazizah04@gmail.com

Abstract *. The research focuses on creating an automated attendance system using face recognition through the Convolutional Neural Network (CNN) approach at IPB University's Vocational School. The current manual attendance methods show limitations, such as potential inaccuracies in recording and the risk of cheating, such as attendance proxies. To overcome these challenges, this study applies the CNN approach with Python and OpenCV, enabling automatic face detection and recognition for students. The system accurately logs attendance by identifying faces in real time. Testing indicates that the system records attendance reliably, whether with a single individual or with multiple faces present within a single frame.*

Keywords *Automatic Attendance System, Face Recognition, CNN, Python, OpenCV*

1. BACKGROUND

In educational environments, such as at IPB University's Vocational School, the current attendance recording process is still carried out manually. Students can record attendance through a *mobile application* by manually clicking attendance or using paper-based attendance in class. Although *mobile applications* facilitate the attendance process, this method is still susceptible to recording errors and fraud, such as asking for attendance or students who record attendance even though they are not on campus. This kind of fraud can damage the integrity and accuracy of attendance data.

Advances in technology have made facial recognition an innovative solution to improve the efficiency and accuracy of attendance systems. This technology utilizes image processing algorithms to detect and recognize individual faces in *real-time* , enabling fast and automatic attendance recording. Attendance systems that use facial recognition enable automatic attendance recording without manual interaction. Thus, the risk of fraud is reduced and attendance data processing becomes easier.

This study aims to implement an automatic attendance system that uses facial recognition technology based on the Convolutional Neural Network (CNN) method, by utilizing the Python programming language and the OpenCV library at the IPB University Vocational School. Through this implementation, it is expected that the attendance process can be carried out more efficiently and accurately, and offers a better solution compared to the manual method currently applied.

Received: September 25, 2024 Revised: Oktober 28, 2024 Accepted: November 02, 2024 Published: November 04, 2024

2. THEORETICAL STUDY

1. Automatic Attendance System

The term "Automatic Attendance System" is composed of three main words, namely system, absence, and automatic. A system can be defined as a set of procedures or components that are interconnected and operate together to achieve the same goal (Napitupulu & Ekaputri, 2017). Absence means "not present," but can also be interpreted as the presence or absence of a person in the context of individuals involved in an organization. In this case, it is important to provide information about the condition or presence of the person in the organizational environment (Husain et al., 2017). Automatic is working alone or by itself (Prabowo et al., 2020).

In other words, an automated attendance system is a system that automatically records the presence or absence of individuals within an organization, providing accurate information without requiring manual interaction, and increasing efficiency in managing attendance data.

2. Facial Recognition (*Face Recognition*)

Facial recognition *is* a form of biometric technology that is often used in security systems, providing an alternative to other biometric methods such as iris, fingerprint, and retina scanning (Muliawan et al., 2015). This technology aims to recognize and verify individuals based on the unique characteristics of their faces. The process begins with capturing a facial image, followed by image analysis to extract distinctive features. Once these features are generated, the system compares them to data stored in a database to perform identification. The results are then verified to ensure the accuracy of the match before appropriate access or authorization is granted. Facial recognition has a variety of applications, such as in door security, surveillance systems, and identification, using algorithms and image processing techniques to achieve high levels of accuracy and efficiency (Khair et al., 2024).

3. CNN (*Convolutional Neural Network*)

Convolutional Neural Network (CNN) is a deep learning algorithm specifically designed to analyze visual data, especially images. It works by applying learned weights and biases to various elements in an image, to recognize important characteristics that distinguish one object from another. CNN become

one of the neural network architectures that is widely used in image data processing (Azmi et al., 2023).

4. Python

   Python is a programming language with a high level of complexity, developed by Guido van Rossum and first introduced to the public in 1991 (Alfarizi et al., 2023). In recent years, Python's popularity has increased rapidly and is now considered one of the leading programming languages. Python is included in the category of scripting programming languages that can operate on various platforms, such as Mac and Windows. This language is open source, so it can be accessed and modified by anyone. Python allows programmers to write code with simple syntax, similar to human language. Various Python applications include web development, numerical programming, game development, and access to serial ports, which often interact with programming languages such as C and C++ (Setiono et al., 2020).

5. OpenCV

   Open Computer Vision (OpenCV) is an open source library specifically designed for image processing. The goal of OpenCV is to provide computers with visual capabilities that resemble the way humans process visual information. This library offers a variety of basic algorithms for computer vision and is equipped with an object detection module that utilizes computer vision techniques (Zulkhaidi et al., 2020). OpenCV offers more than 2500 optimization algorithms that allow efficient image manipulation, processing, and editing. In addition, this library supports various programming languages such as C, C++, Python, Java, and PHP. OpenCV can also be applied in various fields, including Human-Computer Interaction (HCI), object identification and recognition, face detection and recognition, gesture recognition, movement tracking, image processing, and applications in the field of mobile robotics. (Santoso et al., 2020).

3. RESEARCH METHODS

This study focuses on testing the performance and accuracy of a Convolutional Neural Network (CNN)-based facial recognition system. The data used is a collection of facial images of students who voluntarily participated as research objects, taken directly from various angles and lighting conditions to ensure the diversity of data that supports the accuracy of facial recognition. After that, the facial data is processed through a

preprocessing stage to detect facial areas and perform basic image processing, without involving additional attributes such as masks. The series of research stages are presented in the following diagram.

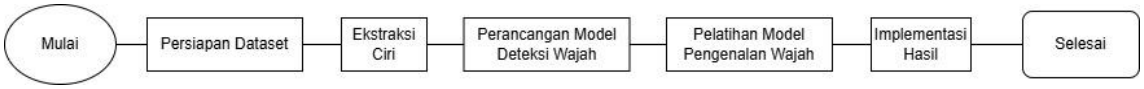

Figure 1. Research Stages Diagram.

This study utilizes various tools and software. One of the software used is Visual Studio Code, which functions as an Integrated Development Environment (IDE). Various Python libraries are utilized in this study, including OpenCV which is used for basic image processing, face_recognition for the purpose of face detection and recognition , and datetime which functions to record the time and date when the attendance process is carried out. In addition, pandas is chosen to manage and analyze the attendance data obtained, while pickle is used to store the data. The csv format is used to export attendance data for further processing. This study uses hardware, namely a laptop equipped with a 2 GHz Quad-Core Intel Core i5 processor, an Intel Iris Plus Graphics 1536 MB graphics card and 16 GB 3733 MHz LPDDR4X memory.

4. RESULTS AND DISCUSSION

System Design

In the system design phase, the visualization and application development process for this system will be carried out. In the data input stage, the process begins when the user activates the application to input student facial data using a webcam. The user enters the student's name and Student Identification Number (NIM) to be recorded. Next, the system accesses the database to store the data. The student's name and NIM are added to the database, and the data that has been entered is loaded for processing. The system then takes a frame from the webcam, detects the facial area in the frame, and processes the detected face into a unique encoding. This facial encoding is then stored in the database along with the student's name and NIM. After all processes are complete, the webcam is removed and the data input stage is declared complete. The following is a diagram of the data input stage in the system.

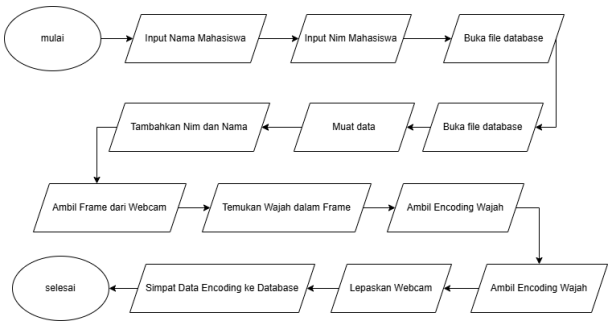

Figure 2. Data Input Stage Diagram.

After the data input stage is successfully completed, the application continues to the attendance process. This process begins by opening the system for attendance purposes. The system reads the facial encoding data from the previously stored database and prepares a list of facial encodings for matching. The current date is recorded for attendance recording purposes, and the webcam is activated to capture real-time images of students' faces. The previous attendance data is read from a CSV file, and the system creates a new CSV file with the current date to store that day's attendance data. Variables are prepared to store the location of the detected face and student name, and then the system enters a loop to continuously capture images from the video. Frames are captured from the video repeatedly to monitor the presence of faces. The system then matches the facial encoding from the video frames with the encoding in the database, and when a face is recognized, the attendance data is stored in a CSV file along with the current time and date. After the attendance process is complete, the webcam is removed, and the automatic attendance process is declared complete. The following is a diagram of the stages of the attendance process.

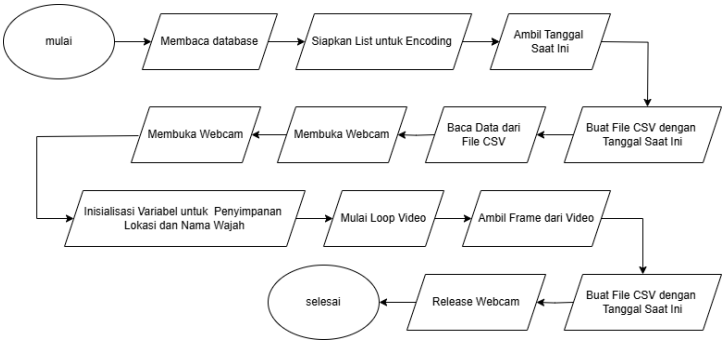

Figure 3. Diagram of the Attendance Process Stages.

System Implementation

At this stage, the collection of student facial image *databases* for the attendance system is carried out. Students will input their facial data using a webcam. In this process, they will follow the application's instructions to enter their name and Student Identification Number (NIM) before their faces are captured by the webcam.

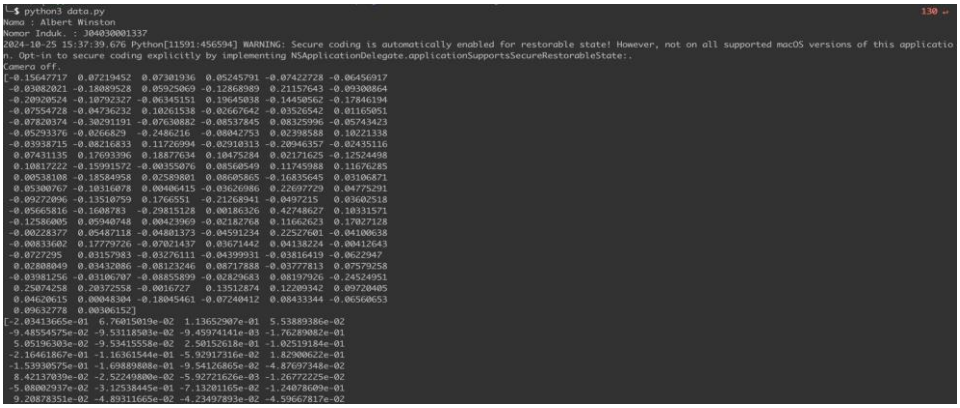

Figure 4. Student Data Input Stage.

Figure 5. Student Data Input Stage.

System Testing

The system testing phase is carried out to evaluate the performance of the facial recognition-based attendance application. This testing covers various scenarios, including cases where students' faces are detected properly (successfully) and situations that cause attendance to fail. The results of this testing will provide an overview of the effectiveness of the system in recognizing students' faces.

1. Attendance Successful

   Several examples of successful attendance show that the system is able to detect students' faces accurately and record their data in the database. In Figures 6 and 7, the attendance process was successful even though Figure 7 shows many

individuals in one frame, the system still managed to detect all faces and record their attendance in the database.

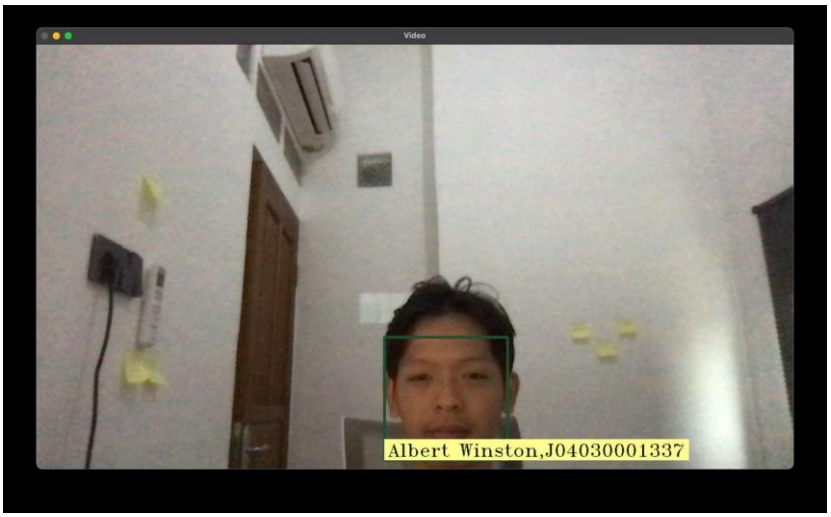

Figure 6. *Tracking* and Detection of Single Face Recognition

Figure 7. *Tracking* and Detection of Face Recognition with *Multiply Face*

Figure 8. *Database* when attendance is successful

2. Attendance Failed

   Some of the images below show examples of attendance failures caused by several factors, including the system's inability to detect faces due to obstructions such as paper, distances that are too far for faces to be detected, and attendance failures due to unregistered students.

Figure 9. Face Recognition Not Detected

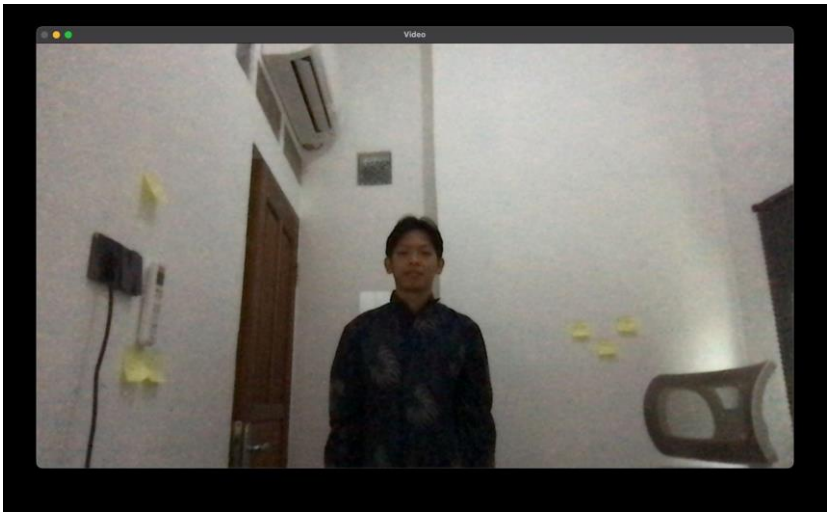

Figure 10. Remote Face *Tracking and Recognition*

Figure 11. *Tracking* and Facial Recognition of Unregistered Students

5. CONCLUSION

This research successfully designed and implemented an automatic attendance system that uses facial recognition with the Convolutional Neural Network (CNN) approach. From the test results, this system demonstrated its ability to record student attendance effectively, both in scenarios where there is only one student and when many faces are detected in one frame. The presence of this system is expected to minimize the risk of fraud, such as asking for attendance, and increase the accuracy and integrity of attendance data. In addition, this system offers a more practical solution compared to the manual attendance method which is still often used.

6. REFERENCE

Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Using Python as a programming language for machine learning and deep learning. *Karimah Tauhid*, 2(1), 1-6. https://doi.org/10.30997/karimahtauhid.v2i1.7518

Azmi, K., Defit, S., & Sumijan, S. (2023). Implementation of convolutional neural network (CNN) for classification of West Sumatra clay batik. *Unitek Journal*, 16(1), 28-40. https://doi.org/10.52072/unitek.v16i1.504

Husain, A., Prastian, A. H. A., & Ramadhan, A. (2017). Design of online attendance system using Android to accelerate employee attendance process at PT. Sintech Berkah Abadi. *Technomedia Journal*, 2(1), 105-116. https://doi.org/10.33050/tmj.v2i1.319

Khair, M. A., Aldiyuda, P., Zukhrufa, M. Z., & Adrezo, M. (2024). Design of student attendance system based on face recognition in UPN Veteran Jakarta environment. *Informatik: Journal of Computer Science*, 20(1), 35-42. https://doi.org/10.52958/iftk.v20i1.6696

Muliawan, M. R., Irawan, B., & Brianorman, Y. (2015). Implementation of facial recognition with the Eigenface method in the attendance system. *Coding Journal of Computers and Applications*, 3(1).

Mustofa, I. H., & Winarno, E. (2023). Masked face recognition system with convolutional neural network method. *Pixel: Scientific Journal of Computer Graphics*, 16(1), 55-66. https://doi.org/10.51903/pixel.v16i1.1062

Napitupulu, F., & Ekaputri, C. (2017). Design and implementation of microcontroller-based motorcycle security system. *eProceedings of Engineering*, 4(2).

Prabowo, R. R., Kusnadi, K., & Subagio, R. T. (2020). Automatic monitoring and feeding system in fish farming using WEMOS with the Internet of Things (IoT) concept. *Jurnal Digit: Digital of Information Technology*, 10(2), 185-195. http://dx.doi.org/10.51920/jd.v10i2.169

Santoso, B., & Kristianto, R. P. (2020). Implementation of OpenCV use in face recognition for student lecture attendance system. *Sistemasi: Journal of Information Systems*, 9(2), 352-361. https://doi.org/10.32520/stmsi.v9i2.822

Setiono, P. R., Sompie, S. R., & Najoan, M. E. (2020). Face recognition application for Raspberry Pi-based class attendance system. *Journal of Informatics Engineering*, 15(3), 179-188. https://doi.org/10.35793/jti.v15i3.31290

Zulkhaidi, T. C. A. S., Maria, E., & Yulianto, Y. (2020). Facial pattern recognition with OpenCV. *Journal of Information Technology Engineering (JURTI)*, 3(2), 181-186. http://dx.doi.org/10.30872/jurti.v3i2.403