

Journal of Dinda

Kelompok Keahlian Rekayasa Data
Institut Teknologi Telkom Purwokerto

Vol. 2 No. 1 (2022) 21 - 27

ISSN Media Elektronik: 2809-8064

Penerapan Face Recognition Berbasis GUI Visual Studio 2012 Menggunakan Algoritma Eigenface dan Metode Pengembangan Waterfall Pada Sistem Absensi Mahasiswa IT Telkom Purwokerto

Ilham Fauzi¹, Apri Junaidi², Wahyu Andi Saputra³

¹Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom Purwokerto

²Sains Data, Fakultas Informatika, Institut Teknologi Telkom Purwokerto

³Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom Purwokerto

Email : ¹15102021@st3telkom.ac.id, ²apri@ittelkom-pwt.ac.id, ³andi@ittelkom-pwt.ac.id

Abstract

Every human being has unique characteristics, one of which is the face, which is a normal human feature. The human face, of course, has distinct features that differentiate it from others, allowing it to be recognised by other humans as well as by a machine capable of doing so. Facial recognition is closely linked to human biometrics due to the fact that each person's face is unique information contained in it. The presence system is one example of facial recognition technology that can be used.

The eigenface algorithm is one of the many methods used in the facial recognition process. Eigenface functions are used to measure eigenvalue and eigenvector, which are used as facial recognition features. The image will be represented in a vector combination that is made into a single matrix. From this single matrix, the main feature that distinguishes one face image from another face image will be extracted. To be able to recognize and recognise someone's face, this research includes an additional tool in the form of a web camera, also known as a WebCam, and Visual Studio 2012 as the application to be used.

This facial recognition technology can be used by IT Telkom Purwokerto as a student attendance system. The unsuccessful use of the previous RFID-based attendance system was one of the findings of the assessment of the need to use technology face recognition as a student attendance system. Due to a variety of technical issues with the attendance method, the attendance process was carried out manually using the lecturer's attendance file.

Keywords: *Image, Eigenface, Face recognition, Image Processing, C #, Attendance System*

Abstrak

Setiap manusia memiliki karakter yang berbeda antara satu dengan yang lainnya, salah satunya adalah karakteristik alami yang dimiliki oleh manusia yaitu wajah. Wajah manusia tentu saja memiliki ciri unik yang membedakan satu dengan lainnya, sehingga dapat dikenali oleh manusia lain maupun oleh suatu sistem yang memiliki kemampuan tersebut. Pengenalan wajah berkaitan erat dengan biometrik manusia, hal tersebut dikarenakan terdapat informasi unik yang terkandung di dalamnya. Teknologi pengenalan wajah dapat dimanfaatkan salah satunya pada sistem presensi kehadiran.

Banyak metode yang digunakan pada proses pengenalan wajah, salah satunya dengan menggunakan algoritma *eigenface*. *Eigenface* berfungsi untuk menghitung *eigenvalue* dan *eigenvector* yang akan digunakan sebagai fitur dalam melakukan pengenalan wajah. Citra akan direpresentasikan dalam sebuah gabungan vektor yang dijadikan satu matriks tunggal. Dari matriks tunggal ini akan di ekstrasi suatu ciri utama yang membedakan antara citra wajah satu dengan citra wajah yang lainnya. Untuk dapat mengenali dan mengidentifikasi wajah seseorang maka pada penelitian ini diperlukan sebuah *tools* tambahan berupa *web camera* atau sering kita kenal dengan istilah *WebCam* dan aplikasi yang akan digunakan adalah Visual Studio 2012.

Teknologi pengenalan wajah ini dapat dimanfaatkan oleh IT Telkom Purwokerto sebagai sistem presensi kehadiran mahasiswa. Salah satu hasil evaluasi perlunya pemanfaatan teknologi *face recognition* sebagai sistem presensi kehadiran mahasiswa dikarenakan belum optimalnya pemanfaatan sistem absensi berbasis RFID yang sebelumnya telah digunakan, berbagai permasalahan teknis yang dihadapi oleh sistem absensi tersebut mengakibatkan proses absensi kembali dilakukan secara manual menggunakan kertas absensi yang diberikan oleh Dosen.

Kata kunci: Citra, *Eigenface*, *Face recognition*, *Image Processing*, C#, Sistem Absensi

© 2022 Jurnal DINDA

a. Pendahuluan

Pemanfaatan teknologi informasi saat ini telah mengalami perkembangan yang signifikan, hal tersebut dikarenakan oleh kemajuan ilmu pengetahuan dan teknologi yang menunjang kemudahan akses informasi di berbagai aspek kehidupan. Salah satunya di dalam dunia Pendidikan.

Terdapat banyak pemanfaatan teknologi informasi di dalam dunia Pendidikan, salah satunya adalah sistem presensi kehadiran. IT Telkom Purwokerto merupakan salah satu perguruan tinggi swasta yang menerapkan sistem presensi kehadiran mahasiswa menggunakan teknologi *Radio Frequency Identification* (RFID) [1]. Sistem absensi berbasis RFID telah menggantikan sistem absensi manual (menggunakan kertas) pada saat itu.

Terdapat permasalahan yang dihadapi oleh IT Telkom purwokerto terhadap penggunaan teknologi RFID pada sistem presensi mahasiswa, diantaranya adalah masih terdapatnya kekurangan, seperti sistem RFID offline pada saat akan dilakukan proses absensi, sehingga hal tersebut sangat mempengaruhi kelancaran kegiatan mata kuliah.



Gambar 1. 1 Perbandingan Jumlah Mahasiswa dan Jumlah Dosen Institut Teknologi Telkom Purwokerto

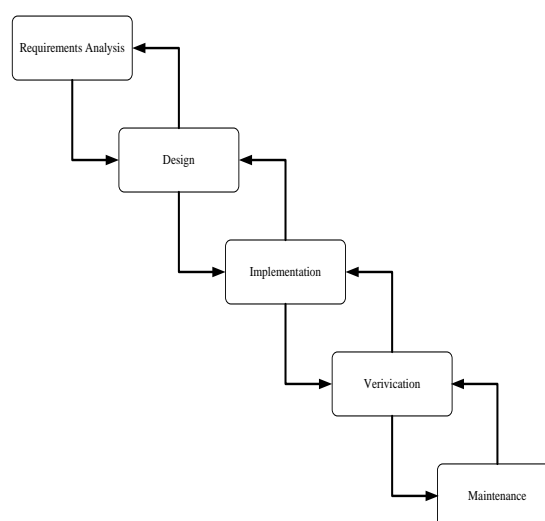
Berdasarkan grafik perbandingan jumlah mahasiswa dan dosen pada tahun 2014 sampai dengan tahun 2017, didapatkan hasil bahwa jumlah mahasiswa IT Telkom Purwokerto semakin meningkat setiap tahunnya. Sebagai perguruan tinggi swasta yang semakin diminati, IT Telkom membutuhkan sistem yang dapat menunjang kelancaran kegiatan mata

kuliah, dengan melihat permasalahan tersebut penulis berusaha mengembangkan sistem presensi dengan menggunakan teknologi *face recognition* [2].

Face recognition banyak diterapkan pada banyak bidang, salah satunya pada sistem presensi kehadiran, tentunya dengan menggunakan berbagai macam metode berbeda, seperti : jaringan syaraf tiruan [3], *principal component analysis* [4], *template matching* [5], *learning vector quantization* [6], dan *euclidean* [7]. Masing-masing metode tentunya memiliki keunggulan dan kelemahan. Dari berbagai macam metode yang ada, penulis menggunakan metode *eigenface* [8]. Penelitian ini bertujuan untuk memberikan solusi terhadap permasalahan sistem presensi kehadiran mahasiswa pada Institut Teknologi Telkom Purwokerto. Solusi yang penulis tawarkan adalah dengan memanfaatkan sistem pengenalan wajah/ *face recognition* pada presensi mahasiswa Institut Teknologi Telkom Purwokerto

b. Metode Penelitian

Pada penelitian ini terdapat tahapan penelitian yang dilakukan oleh penulis dalam mengembangkan sistem presensi mahasiswa menggunakan teknologi *face recognition*.



Gambar 2. 1. Alur kerja pengembangan waterfall

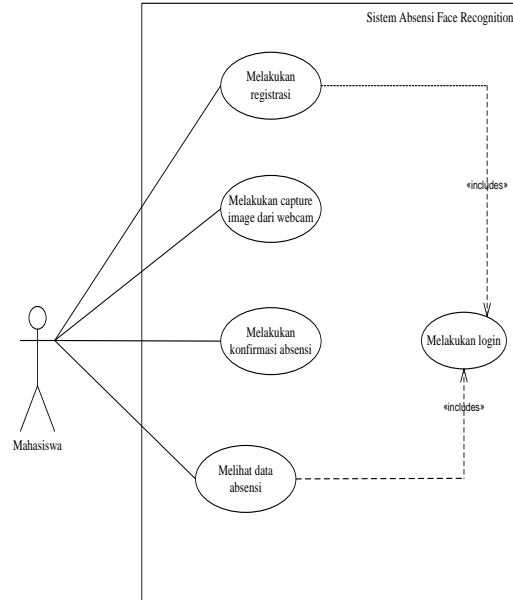
Gambar 2.1 alur kerja pengembangan waterfall yang penulis lakukan dalam penelitian ini, yang secara garis besar akan di jabarkan sebagai berikut :

a. Fase Definisi

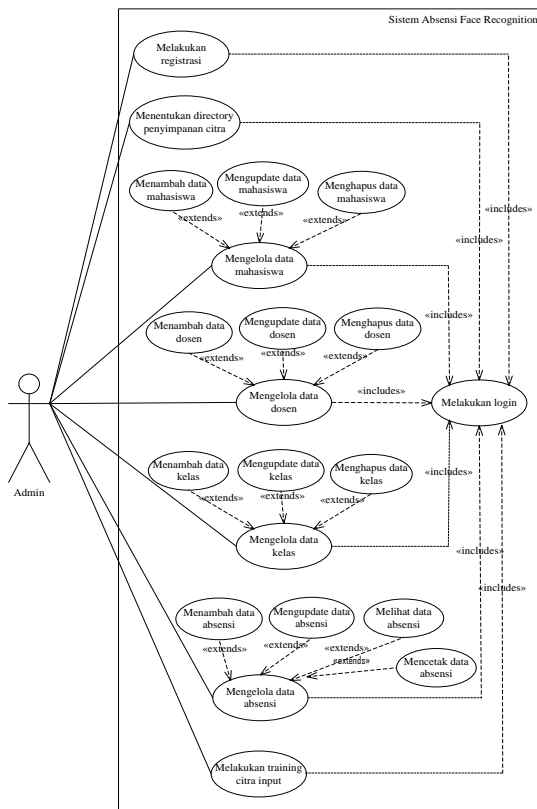
Pada tahapan penelitian ini, penulis melakukan analisis sistem yang sedang berjalan, kemudian dari hasil analisis tersebut didapatkan sebuah permasalahan [9], berdasarkan permasalahan yang ditemukan pada sistem berjalan itulah penulis melakukan analisis terhadap kebutuhan sistem yang akan dirancang, apa saja yang dibutuhkan oleh sistem selanjutnya untuk mengatasi permasalahan yang dihadapi oleh sistem yang sedang berjalan

b. Fase Desain

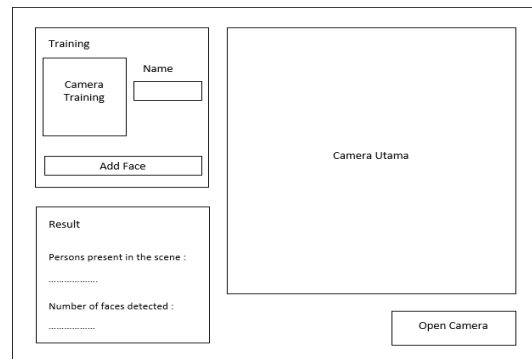
Pada tahapan penelitian ini, penulis mendesain sistem basis data yang akan digunakan, merancang UML diagram [10], serta mendesain struktur menu dan user interface aplikasi yang akan digunakan.



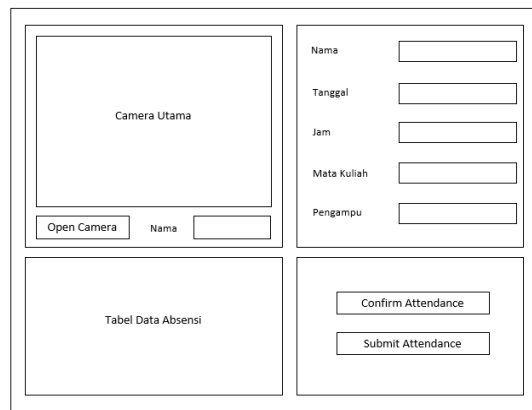
Gambar 2. 3. Use Case Mahasiswa



Gambar 2. 2. Use Case Admin



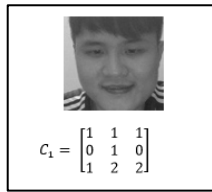
Gambar 2. 4. Mokeup form training citra



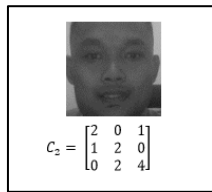
Gambar 2. 5. Mokeup form absensi mahasiswa

c. Fase Implementasi

Pada tahapan penelitian ini, penulis melakukan proses perancangan aplikasi (melakukan pengkodean) dengan mengimplementasikan perancangan model desain yang telah di buat pada proses sebelumnya, serta mengimplementasikan algoritma eigenface pada sistem.



Gambar 2. 7. Citra Training 1



Gambar 2. 6. Citra Training 2

Nilai tengah (*mean*)

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n$$

$$\Psi = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 2 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 1 \\ 1 & 2 & 0 \\ 0 & 2 & 4 \end{bmatrix}$$

$$\Psi = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix}$$

Untuk mendapatkan nilai eigenface, maka diperlukan nilai eigenvector. Untuk mendapatkan nilai eigenvector, maka diperlukan nilai eigenvalue.

Menghitung Nilai *Eigenvalue* (λ)

$$(L - \lambda I) = 0 \text{ atau } (\lambda I - L) = 0$$

$$0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

$$\det \begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} = 0$$

$$\begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} \begin{bmatrix} \lambda - 2 & -1 \\ -1 & \lambda - 2 \\ 0 & 0 \end{bmatrix} = 0$$

$$= \lambda^3 - 7\lambda^2 + 15\lambda - 9$$

Didapatkan hasil eigenvalue sebagai berikut :

$$\lambda_1 = 3, \lambda_2 = 1, \lambda_3 = 3$$

Menghitung Nilai *Eigenvector* (v)

Eigenvector dihasilkan dengan mensubstitusikan nilai *eigenvalue* ke dalam persamaan

$$(\lambda I - L)v = 0$$

Untuk $\lambda_1 = 3$

$$\begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$$

Untuk $\lambda_2 = 1$

$$\begin{bmatrix} -1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$$

Untuk $\lambda_3 = 3$

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Menghitung Nilai *Eigenface* (μ)

$$\mu_i = \sum_{k=1}^M v_{ik} \Phi_k$$

$\mu = \text{eigenface}$

$v = \text{eigenvector}$

$\Phi =$ Selisih antara *training* image dengan nilai tengah

$M =$ Jumlah data wajah

$$\mu_1 = v \times \Phi_1$$

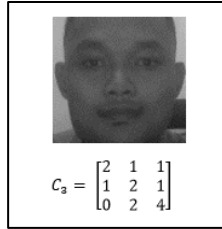
$$\mu_1 = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

$$\mu_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mu_2 = v \times \Phi_2$$

$$\mu_2 = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} x \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} 0 & -1 & 0 \\ -2 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



Gambar 2. 8 Citra Testing

Menghitung selisih *testface* dengan nilai tengah (*mean*)

$$\Phi_{new} = \mu_{new} - \Psi$$

$$\Phi_{new} = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 0 & 2 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix}$$

$$\Phi_{new} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Menghitung nilai *eigenface* pada *testface*

$$\mu_{new} = v \times \Phi_{new}$$

$$\mu_{new} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} x \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mu_{new} = \begin{bmatrix} 0 & 0 & -1 \\ -2 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

Menghitung selisih terkecil antara *eigenface* training face dengan *eigenface* testing face menggunakan euclidean distance

$$\varepsilon_k = ||\Omega - \Omega_{new}||$$

$$\varepsilon_1 = \left| \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 & -1 \\ -2 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \right|$$

$$\varepsilon_1 = \left| \begin{bmatrix} 0 & 1 & -1 \\ 2 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \right|$$

$$\varepsilon_1 = \sqrt{(0)^2 + (1)^2 + (-1)^2 + (2)^2 + (1)^2 + (-1)^2 + (0)^2 + (0)^2 + (0)^2}$$

$$\varepsilon_1 = \sqrt{8}$$

$$\varepsilon_1 = 2,82$$

$$\varepsilon_2 = \left| \begin{bmatrix} 0 & -1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \right|$$

$$\varepsilon_2 = \sqrt{(0)^2 + (-1)^2 + (-1)^2 + (0)^2 + (1)^2 + (-1)^2 + (0)^2 + (0)^2 + (0)^2}$$

$$\varepsilon_1 = \sqrt{4}$$

$$\varepsilon_1 = 2$$

Karena jarak (*distance*) *eigenface* pada *image 2* dengan *eigenface* pada *testface* memiliki nilai paling kecil, maka dapat disimpulkan bahwa *testface* memiliki kemiripan ciri dengan *image 2*.

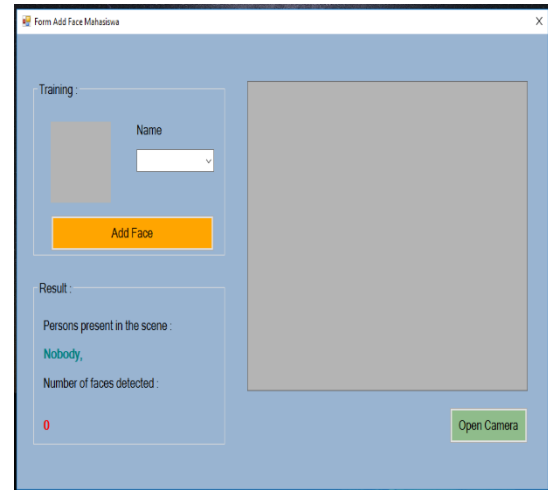
Semakin kecil *distance* *eigenface* antara citra uji dan citra latih, maka akan semakin sesuai citra yang dikenali.

d. Fase Pengujian

Pada tahapan penelitian ini, penulis melakukan proses pengujian terhadap sistem yang telah dirancang. Responden yang digunakan adalah mahasiswa IT Telkom Purwokerto secara acak., pengujian ini dilakukan untuk mengetahui apakah seluruh komponen pada sistem dapat berfungsi dengan baik.

3. Hasil dan Pembahasan

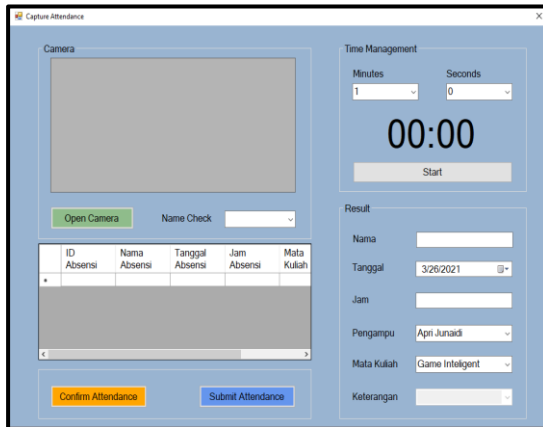
3.1 Implementasi Mokeup Form Citra Training



Gambar 3. 1. Implementasi form training citra

Gambar 3.1 merupakan implementasi desain form training citra. Setiap mahasiswa akan diambil sebanyak 10 macam citra training.

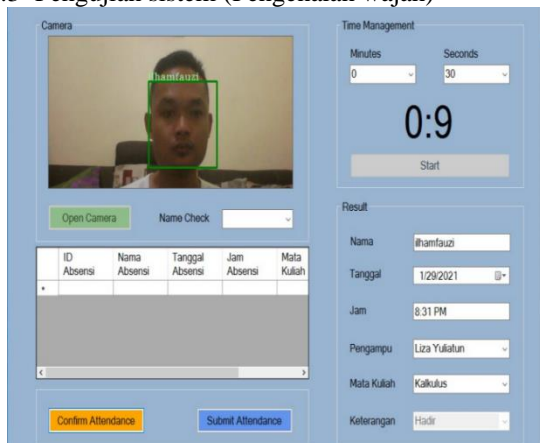
3.2 Implementasi Mokeup Form Absensi Mahasiswa



Gambar 3. 2. Implementasi form absensi mahasiswa

Gambar 3.2 merupakan implementasi desain form absensi mahasiswa. Di dalam form absensi mahasiswa terdapat waktu yang digunakan untuk memberikan batas waktu keterlambatan absensi mahasiswa setiap kegiatan perkuliahan.

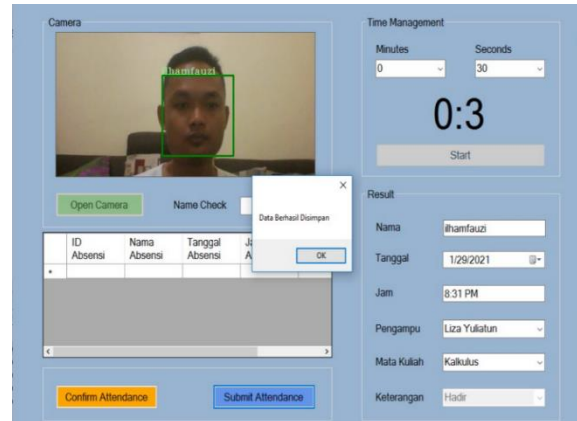
3.3 Pengujian sistem (Pengenalan wajah)



Gambar 3. 3. Pengujian Sitem Mengenali Wajah

Gambar 3.3 merupakan hasil pengujian sistem yang dilakukan dengan wajah tanpa ekspresi, didapatkan hasil bahwa sistem dapat mengenali wajah sesuai dengan identitas mahasiswa.

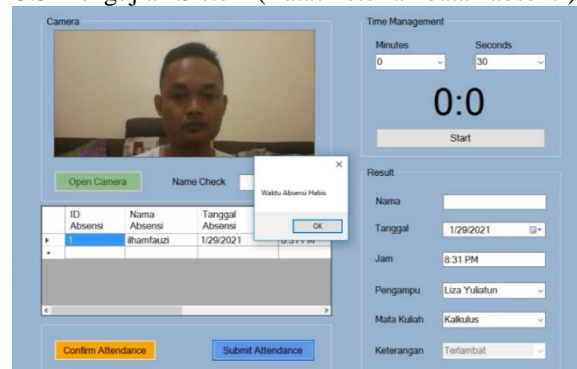
3.4 Pengujian Sistem (Menyimpan data absensi)



Gambar 3. 4. Pengujian Menyimpan Data Absensi

Gambar 3.4 merupakan hasil pengujian sistem yang dilakukan untuk menyimpan data absensi setelah sistem berhasil mengenali citra input mahasiswa

3.5 Pengujian Sistem (Batas keterlambatan absensi)



4. Kesimpulan

4.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis, maka didapatkan kesimpulan sebagai berikut :

1. Sistem absensi mahasiswa dengan menerapkan teknologi face recognition dapat berfungsi dengan baik, sistem dapat mengenali wajah mahasiswa yang terdeteksi pada webcam.
2. Algoritma eigenface yang digunakan pada sistem absensi mahasiswa dapat berfungsi dengan baik, hal tersebut dibuktikan dengan keberhasilan sistem mendeteksi dan mengenali wajah mahasiswa.

4.2 Saran

Berdasarkan hasil penelitian, terdapat beberapa saran yang diberikan oleh penulis untuk penelitian selanjutnya, antara lain :

1. Menambah jumlah data training yang akan digunakan, semakin banyak dan beragam data training yang diambil, maka hasil yang didapatkan akan semakin baik.
2. Melakukan pengembangan aplikasi face

- recognition dengan menggunakan bahasa pemrograman lain, seperti python dan java
3. Menerapkan algoritma lain pada penerapan face recognition, sehingga didapatkan variasi baru dalam pengembangan teknologi face recognition.

Daftar Rujukan

- [1] H. Pranoto and O. Kusumawardani, “Real-time Triplet Loss Embedding Face Recognition for Authentication Student Attendance Records System Framework,” *JOIV Int. J. Informatics Vis. Vol 5, No 2 (2021)DO - 10.30630/joiv.5.2.480*, May 2021, [Online]. Available: <http://joiv.org/index.php/joiv/article/view/480>.
- [2] F. D. Adhinata, N. Annisa, F. Tanjung, W. Widayat, and G. R. Pasfica, “Comparative Study of VGG16 and MobileNetV2 for Masked Face Recognition,” vol. 7, no. 2, pp. 230–237, 2021, doi: 10.26555/jiteki.v7i2.20758.
- [3] A. Fadlil and S. Yeki, “Sistem Verifikasi Wajah Menggunakan Jaringan Saraf Tiruan Learning Vector Quantization,” *J. Inform.*, vol. 4, no. 2, pp. 480–487, 2012, doi: 10.12928/jifo.v4i2.a5279.
- [4] K. Gunadi and S. R. Pongsitaan, “Pembuatan Perangkat Lunak Pengenalan Wajah Menggunakan Principal Components Analysis,” *J. Inform.*, vol. Vol 2, no. Perangkat Lunak Pengenalan Wajah Menggunakan Principal Components Analysis, pp. 57–61, 2001.
- [5] M. A. Rahman and S. Wasista, “Sistem Pengenalan Wajah Menggunakan Webcam Untuk Absensi Dengan Metode Template Matching,” *EEPIS Repos.*, vol. Vol 9, no. Pengenalan Wajah Menggunakan Webcam Untuk Absensi Dengan Metode Template Matching, pp. 1–6, 2010.
- [6] S. Heranurweni, “Pengenalan Wajah Menggunakan Learning Vector Quantization (LVQ),” *Pros. SNST Fak. Tek.*, vol. 1, no. 1, 2010.
- [7] H. Kurniawan and T. Hidayat, “Perancangan Program Pengenalan Wajah Menggunakan Fungsi Jarak Metode Euclidean Pada Matlab,” *Semin. Nas. Apl. Teknol. Inf. 2008 (SNATI 2008) Yogyakarta, 21 Juni 2008*, vol. Vol 1, no. Snati, pp. 15–18, 2008.
- [8] N. K. A. Wirdiani, P. Hridayami, N. P. A. Widiari, K. D. Rismawan, P. B. Candradinata, and I. P. D. Jayantha, “Face Identification Based on K-Nearest Neighbor,” *Sci. J. Informatics*, vol. 6, no. 2, pp. 150–159, 2019, doi: 10.15294/sji.v6i2.19503.
- [9] F. Z. Ramadhan, G. Aditya, P. D. Y. Nainggolan, and F. D. Adhinata, “Sistem Pakar Diagnosa Penyakit pada Hewan Kucing Berbasis Web,” *J. Komtika (Komputasi dan Inform.*, vol. 5, no. 2, pp. 122–131, 2021.
- [10] V. Sahfitri, “Perancangan sistem reservasi dan promosi hotel berbasis website,” *J. Inform.*, vol. 20, no. 1, pp. 54–66, 2020.