



Prototipe Absensi STMIK Amik Riau Berbasis *Face Recognition* Menggunakan Metode *Eigenface*

Abdul Mutholib
STMIK Amik Riau
Abdulmuntholib16@gmail.com

Susi Erlinda
STMIK Amik Riau
susierlinda@stmik-amik-riau.ac.id

Abstrak

Pengenalan wajah merupakan salah satu teknologi dari komputer yang bisa mengidentifikasi wajah seseorang melalui sebuah gambar digital. Untuk kondisi absensi yang ada saat ini masih menggunakan kartu tanda karyawan, sehingga di lakukan penitipan absensi. Karyawan bisa saja datang terlambat absen sehingga dapat menurunkan tingkat kedisiplinan dan kualitas kerja. Untuk mengatasi hal tersebut, maka di buat sebuah alat sensor wajah dengan menggunakan metode *Eigenface*. Metode *Eigenface* suatu metode pengenalan wajah yang berdasarkan pada algoritma *Principal Component Analysis (PCA)*. Secara singkat prosesnya merupakan citra yang di representasikan dalam sebuah gabungan vektor yang dijadikan satu matriks tunggal. Dari matriks tunggal akan diekstraksi kan suatu ciri utama yang akan membedakan antara sebuah citra wajah satu dengan citra wajah lainnya. Dari hasil pengujian mendapatkan rata-rata presentase kecocokan yaitu 60%.

Kata kunci: Metode *Eigenface*, *Face Recognition*, Absensi, *PCA*

1. Pendahuluan

Mengingat wajah seseorang sangatlah mudah di lakukan oleh manusia. Seseorang dapat mudah mengingat dan menghafal wajah yang sudah lama dilihat sebelumnya walaupun bentuk dan ekspresi wajah yang berbeda-beda. Akan tetapi, bagaimana

supaya komputer dapat mengidentifikasi wajah seseorang. Akan ada kendala bagi sistem tanpa ada sebuah pembelajaran untuk sistem. Pengenalan wajah merupakan salah satu teknologi dari komputer yang bisa mengidentifikasi wajah seseorang melalui sebuah gambar digital. Pengenalan wajah bisa di padukan dengan sistem absensi sehingga menjadi hal yang sangat menarik, dimana sistem absensi juga dapat di lakukan menggunakan wajah. Kampus STMIK Amik Riau adalah perguruan tinggi swasta yang berada di Pekanbaru, Riau. Di kampus STMIK Amik Riau karyawan di haruskan absen dua kali sehari. Dimana jam untuk absen sudah di tentukan yaitu sebelum jam 8:15:59 pagi dan sesudah jam 15:45:00 sore. Apabila karyawan hanya absen satu kali, maka akan dinyatakan tidak hadir atau bisa disebut mangkir. Dengan demikian karyawan akan mendapatkan pengurangan gaji sesuai daftar kehadiran. Adapun kondisi absensi yang ada saat ini, masih menggunakan kartu tanda karyawan. Sering kali nama dosen tercantum pada daftar hadir, namun ternyata dosen tersebut belum masuk. Dosen bisa saja menitip absen kepada rekannya sehingga keamanan pada sistem kurang baik. Untuk mengatasi hal tersebut, di buatlah alat sensor wajah menggunakan metode *Eigenface*. Metode *Eigenface* suatu metode pengenalan wajah yang berdasarkan pada algoritma *Principal Component Analysis (PCA)*. Metode ini sangat akurat dari pada metode lainnya, karena berorientasi dengan algoritma *Principal Component Analysis (PCA)*. *PCA* merupakan metode ekstraksi ciri ataupun kompresi data yang bisa mengidentifikasi ciri-ciri tertentu yang merupakan suatu karakteristik dari citra (maksud dari citra adalah wajah).

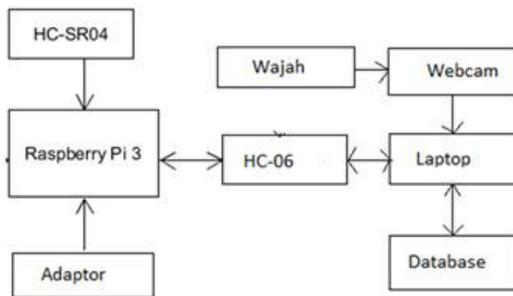
Naskah bisa ditulis dalam bahasa Indonesia atau bahasa Inggris. Panduan ini berisi format tentang jenis

font, spasi yang digunakan dalam pengetikan naskah, dan informasi lainnya yang berkaitan dengan naskah jurnal yang dihasilkan. Silahkan ikuti format yang diberikan dan jika ada pertanyaan silahkan hubungi kami di bagian LPPM, STMIK Amik Riau

2. Metodologi Penelitian

2.1 Pencadangan system

Untuk mengatasi permasalahan dari penitipan absen dan keterlambatan masuk kerja maka bangunlah system absensi berbasis face recognition yang akan membantu meningkatkan keamanan pada absensi tersebut. Berikut merupakan Blok Diagram sistem absensi yang diajukan. Untuk lebih jelasnya dapat dilihat pada gambar 1.



Gambar 1. Blok Diagram yang Diajukan

Dari Blok diagram diatas, dapat dijelaskan wajah merupakan objek pada penelitian ini. Wajah akan di ambil oleh webcam dimana dalam pengambilan wajah di beri jarak antara 30 – 70 cm. Apabila wajah telah diambil oleh webcam selanjutnya di kirim ke Vb serta untuk menjalankan raspberry pi 3 diperlukan adaptor. Adapun laptop berfungsi sebagai tampilan dari sistem absensi dan juga akan sebagai penyimpanan database dimana modul hc-06 yang akan menghubungkan raspi dengan laptop.

2.2 Penulisan Persamaan Eigenface

Eigenface adalah salah satu metode pengenalan pola wajah yang berdasarkan pada Algoritma PrincipalComponent Analysis (PCA). Prinsip dasar dari pengenalan wajah adalah dengan mengutip informasi unik wajah tersebut kemudian di encode dan dibandingkan dengan hasil decode yang sebelumnya dilakukan. Dalam metode *Eigenface*, decoding dilakukan dengan menghitungeigenvector kemudiandirepresentasikan dalam sebuah matriks yang berukuran besar.Eigenvector juga dinyatakan sebagai karakteristik wajah oleh karena itu metode ini disebut dengan *Eigenface*. Setiap wajah direpresentasikan dalam kombinasi linear *Eigenface*. Metode*Eigenface* pertama kali dikembangkan oleh Matthew Turk dan Alex Pentland dari Vision and Modeling Group, TheMedia Laboratory MassachusettsInstitute of

Technology pada tahun1987. Metode ini disempurnakan lagi oleh Turk dan Pentland pada tahun 1991.

Metode pengenalan wajah di mulai dengan membuat matriks kolom dari wajah yang dimasukkan ke dalam database. Rata - rata vector citra (mean) dari matriks kolom dihitung dengan cara membaginya dengan jumlah banyaknya citra yang disimpan di dalam database.

Metode *Eigenface* secara keseluruhan cukup sederhana. *Image* matriks (Γ) direpresentasikan ke dalam sebuah himpunan matriks ($\Gamma_1, \Gamma_2, \dots, \Gamma_M$). Cari nilai rata-rata (Ψ)dan gunakan untuk mengekstraksi eigenvalue(λ) dan eigenvector(v) dari himpunan matriks. Gunakan nilai eigenvector untuk mendapatkan nilai *Eigenface* dari image. Apabila ada sebuah image baru atau test face(Γ_{new}) yang ingin dikenali, proses yang sama juga diberlakukan untuk image (Γ_{new}), untuk mengekstraksi eigenvector (v) dan eigenvalue (λ), kemudian cari nilai *Eigenface* dari image test face (Γ_{new}). Setelah itu barulah image baru (Γ_{new}) memasuki tahapan pengenalan dengan menggunakan metode *euclidean distance*. Tahap perhitungan selengkapnya dapat dilihat seperti berikut ini. Tahapan Perhitungan *Eigenface* :

- Langkah pertama adalah menyiapkan data dengan membuat suatu himpunan S yang terdiri dari seluruh training image, ($\Gamma_1, \Gamma_2, \dots, \Gamma_M$)
 $S = (\Gamma_1, \Gamma_2, \dots, \Gamma_M)$
- Langkah kedua adalah ambil nilai tengah atau mean (Ψ)
- $\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$ Langkah ketiga kemudian cari selisih (Φ) antara nilai training image (Γ_i) dengan nilai tengah (Ψ)
- Keempat adalah menghitung nilai matriks kovarian (C)

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T$$

$$L = A^T A = \Phi_m^T \Phi_n$$

- Langkah kelima menghitung eigenvalue (λ) dan eigenvector (v) dari matriks kovarian (C)

$$C x v_i = \lambda_i x v_i$$

- Langkah keenam, setelah eigenvector (v) diperoleh, maka *Eigenface* (μ) dapat dicari dengan

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

Tahapan Pengenalan wajah :

- Sebuah image wajah baru atau test face (Γ_{new}) akan dicoba untuk dikenali, pertama terapkan carapada tahapan pertama perhitungan *Eigenface* untuk mendapatkan nilai eigen dari image tersebut.

$$\mu_{new} = v(\Gamma_{new} - \Psi)$$

$$\Omega = [\mu_1, \mu_2, \dots, \mu_M]$$

- Gunakan metode euclidean distance untuk mencari jarak.(distance) terpendek antara nilai *eigen* dari training image dalam database dengan nilai eigen dari image test face.

$$\varepsilon_k = \| \Omega - \Omega_k \|$$

(Muhammad Rizki Muliawan, Beni Irawan, 2015)

Massachusetts Institute of Technology pada tahun 1987. Metode ini disempurnakan lagi oleh Turk dan Pentland pada tahun 1991.

Metode pengenalan wajah di mulai dengan membuat matriks kolom dari wajah yang dimasukkan ke dalam database. Rata - rata vector citra (mean) dari matriks kolom dihitung dengan cara membaginya dengan jumlah banyaknya citra yang disimpan di dalam database.

Metode *Eigenface* secara keseluruhan cukup sederhana. *Image* matriks (Γ) direpresentasikan ke dalam sebuah himpunan matriks ($\Gamma_1, \Gamma_2, \dots, \Gamma_M$). Cari nilai rata-rata (Ψ) dan gunakan untuk mengekstraksi eigenvector (v) dan eigenvalue (λ) dari himpunan matriks. Gunakan nilai eigenvector untuk mendapatkan nilai *Eigenface* dari image. Apabila ada sebuah image baru atau test face (Γ_{new}) yang ingin dikenali, proses yang sama juga diberlakukan untuk image (Γ_{new}), untuk mengekstraksi eigenvector (v) dan eigenvalue (λ), kemudian cari nilai *Eigenface* dari image test face (Γ_{new}). Setelah itu barulah image baru (Γ_{new}) memasuki tahapan pengenalan dengan menggunakan metode euclidean distance. Tahap perhitungan selengkapnya dapat dilihat seperti berikut ini. Berikut tahapan perhitungan *eigenface* :

- Langkah pertama adalah menyiapkan data dengan membuat suatu himpunan S yang terdiri dari seluruh *training image*, ($\Gamma_1, \Gamma_2, \dots, \Gamma_M$)

$$S = (\Gamma_1, \Gamma_2, \dots, \Gamma_M)$$

- Langkah kedua adalah ambil nilai tengah atau mean (Ψ)

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$$

- Langkah ketiga kemudian cari selisih (Φ) antara nilai training image (Γ_i) dengan nilai tengah (Ψ)

$$\Phi_i = \Gamma_i - \Psi$$

- Langkah keempat adalah menghitung nilai matriks kovarian (C)

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T$$

$$L = A^T A = \Phi_m^T \Phi_n$$

- Langkah kelima menghitung eigenvalue (λ) dan eigenvector (v) dari matriks kovarian (C)

$$C x v_i = \lambda_i x v_i$$

- Langkah keenam, setelah eigenvector (v) diperoleh, maka *Eigenface* (μ) dapat dicari dengan :

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

Tahapan Pengenalan wajah :

- Sebuah image wajah baru atau *test face* (Γ_{new}) akan dicoba untuk dikenali, pertama terapkan cara pada tahapan pertama perhitungan *Eigenface* untuk mendapatkan nilai eigen dari image tersebut.

$$\mu_{new} = v(\Gamma_{new} - \Psi)$$

$$\Omega = [\mu_1, \mu_2, \dots, \mu_M]$$

- Gunakan metode euclidean distance untuk mencari jarak (distance) terpendek antara nilai eigen dari training image dalam database dengan nilai eigen dari image test face.

$$\varepsilon_k = \| \Omega - \Omega_k \|$$

(Muhammad Rizki Muliawan, Beni Irawan, 2015)

3. Hasil Pembahasan

3.1 Analisa Proses *Eigenface*

Eigenface merupakan metode yang berdasarkan pada algoritma Principal Component Analysis (PCA), dimana *Eigenface* akan dibantu oleh pca untuk dalam pengambilan ciri-ciri khusus pada wajah manusia. *Eigenface* memiliki tahapan untuk proses pengenalan wajah, dimana tahap pertama menyiapkan data himpunan matriks, setelah itu ambil nilai tengah atau bisa disebut dengan mean, kemudian cari nilai selisih training image dengan nilai tengah, jika sudah cari nilai kovarian, apabila sudah maka hitunglah eigenvalue dengan eigenvector jika dapat nilaitersebut maka nilai *Eigenface* dapat dicari. Untuk tahap selanjutnya yaitu mengidentifikasi wajah baru dengan yang ada di database dengan menyamakan data yang sudah ada. Di bawah ini adalah tahapan untuk lebih jelasnya lagi.

3.2 Perhitungan *Eigenface*

- Penyusunan Flatvector Matriks Citra

Langkah pertama adalah menyusun suatu himpunan S matriks yang terdiri dari seluruh training image ($\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_m$). Misalnya, training image terdapat satu data wajah seperti terlihat pada gambar 3.1, yang memiliki nilai matriks.



$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

Gambar 2. Citra wajah 1

- Hitung Nilai Tengah atau Mean

Dari himpunan matriks yang telah diperoleh, langkah selanjutnya adalah mencari nilai tengah atau mean Ψ .

Jumlahkan nilai matrik wajah 1 kemudian bagi dengan jumlah data wajah yang ada di database, pada penelitian ini jumlah data wajah yang ada di database adalah dua data wajah.

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

Cari nilai tengah atau mean .

$$\Psi = \frac{1}{2} \sum_{n=1}^2 \Gamma_n = \frac{1}{2} \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 0 \\ 1 & 1 & 4 \end{bmatrix}$$

3. $\Psi = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix}$ Hitung Nilai Tengah atau Mean

Dari himpunan matriks yang telah diperoleh, langkah selanjutnya adalah mencari nilai tengah atau mean Ψ . Jumlahkan nilai matrik wajah 1 kemudian bagi dengan jumlah data wajah yang ada di database, pada penelitian ini jumlah data wajah yang ada di database adalah dua data wajah.

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

Cari nilai tengah atau mean .

$$\Psi = \frac{1}{2} \sum_{n=1}^2 \Gamma_n = \frac{1}{2} \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 0 \\ 1 & 1 & 4 \end{bmatrix}$$

4. Hitung Selisih Antara Training Image dengan Nilai Tengah

Dengan memakai nilai tengah citra di atas, langkah selanjutnya adalah mencari selisih (Φ) antara training image (Γ) dengan nilai tengah (Ψ), dengan mengurangi training image (Γ) dengan nilai tengah (Ψ).

$$\Phi_i = \Gamma_i - \Psi$$

$$\Phi = \Gamma - \Psi = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 0 \\ 1 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 2 \end{bmatrix}$$

5. Hitung Nilai Matriks Kovarian

Nilai matriks kovarian (C) digunakan untuk menghitung eigenvalue (λ) dan eigenvector (v).

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T$$

$$L = A^T A = \Phi_m^T \Phi_n$$

Hitung nilai matriks kovarian (C)

$$L = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 2 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 2 \end{bmatrix}$$

$$L = \begin{bmatrix} 2 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 6 \end{bmatrix} \quad \text{matriks kovarian}$$

6. Hitung Nilai Eigenvalue dan Eigenvector

Langkah selanjutnya adalah menghitung nilai eigenvalue (λ) dan eigenvector (v) dari matriks kovarian (C).

$$Cx v_i = \lambda_i x v_i$$

Cari nilai nilai eigenvalue (λ) dan eigen-vector (v).

$$Lx v = \lambda x v$$

$$Lx v = \lambda I x v$$

$$L - \lambda I = 0 \text{ at } -L = 0$$

$$0 = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 6 \end{bmatrix}$$

$$0 = \det \begin{bmatrix} \lambda - 2 & 0 & 3 \\ 1 & \lambda - 1 & 1 \\ \lambda - 3 & 0 & \lambda - 6 \end{bmatrix}$$

Maka eigenvalue yang dihasilkan adalah $\lambda_1 = 3, \lambda_2 = 1, \lambda_3 = 5$

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

Eigenvector (v) dihasilkan dengan mensubstitusikan nilai eigenvalue (λ) kedalam persamaan $\lambda I - = 0$. Eigenvector dari masing-masing eigenvalue didapat berdasarkan masing-masing kolom eigenvalue dan kemudian dihimpun kembali menjadi satu matriks.

a. Untuk $\lambda_1 = 3$, maka :

$$\begin{bmatrix} \lambda - 2 & 0 & 3 \\ 0 & \lambda - 1 & 1 \\ \lambda - 3 & 0 & \lambda - 6 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan eigenvector v_1 adalah $\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$,

b. Untuk $\lambda_2 = 1$, maka :

$$\begin{bmatrix} \lambda - 2 & 0 & 3 \\ 0 & \lambda - 1 & 1 \\ \lambda - 3 & 0 & \lambda - 6 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan eigenvector v_1 adalah $\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$,

c. Untuk $\lambda_3 = 3$, maka :

$$\begin{bmatrix} \lambda - 2 & 0 & 3 \\ 0 & \lambda - 1 & 1 \\ \lambda - 3 & 0 & \lambda - 6 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan eigenvector v_3 adalah $\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$

Setelah didapat eigenvector $v_1, v_2, \text{ dan } v_3$, maka eigenvector yang dihasilkan dari matriks L adalah

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}$$

7. Nilai Eigenface

Langkah selanjutnya, setelah eigenvector (v) diperoleh, maka nilai Eigenface (μ) dapat dicari dengan :

$$\mu_i = \sum_{k=1}^M v_{ik} \Phi_{kt} = 1, \dots, M$$

Cari nilai Eigenface (μ) :

$$\begin{aligned} \mu_i &= \sum_{k=1}^M v_{ik} \Phi_k \\ \mu &= v \times \Phi_1 \\ &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \mu &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

8. Proses Identifikasi

Pertama cari selisih (Φ) antara test face dengan nilai tengah (Ψ). nilai matriks test face dari koordinat tersebut terlihat pada Gambar 3.

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



$$\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 0 & 2 & 4 \end{bmatrix}$$

Gambar 3. Citra wajah baru (testface)

Selanjutnya, setelah selisih (Φ) antara tesface dengan nilai tengah (Ψ) di-dapat, maka nilai *Eigenface* dapat dicari.

$$\begin{aligned} \mu_{new} &= v \times \Phi_{new} \\ \mu_{new} &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \\ 3 & 5 & 3 \end{bmatrix} \times \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 2 & 2 \end{bmatrix} \\ \mu_{new} &= \begin{bmatrix} 3 & 3 & 3 \\ 2 & 6 & 4 \end{bmatrix} \end{aligned}$$

Gunakan euclidean distance untuk mencari selisih terkecil antara *Eigenface* training image (Γ_i) dalam

Proses identifikasi adalah proses dimana wajah baru masuk dan akan dicocokkan dengan wajah yang ada di dalam database untuk mengetahui wajah mana yang cocok antara wajah yang baru masuk dengan wajah yang ada di dalam database. Untuk mengenali wajah baru yang masuk (test face), langkah yang dilakukan sama dengan data wajah yang ada di database, untuk mendapatkan nilai *Eigenface* dari wajah baru.

$$\begin{aligned} \mu_{new} &= v(\Gamma_{new} - \Psi) \\ \Omega &= [\mu_1, \mu_2, \dots, \mu_M] \end{aligned}$$

database dengan *Eigenface* test face (Γ_{new}), setelah itu jumlahkan matriks dari masing-masing euclidean distance.

$$\begin{aligned} \epsilon_k &= \|\Omega - \Omega_k\| \\ \epsilon_k &= \|\Omega - \Omega_{new}\| \\ \epsilon &= \|\Omega - \Omega_{new}\| \\ &= \left\| \begin{bmatrix} 3 & 5 & 3 \\ 3 & 3 & 3 \\ 2 & 6 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right\| \\ &= \sqrt{(2)^2 + (4)^2 + (2)^2 + (2)^2 + (2)^2 + (3)^2 + (2)^2 + (6)^2 + (3)^2} \\ &= \sqrt{66} \\ \epsilon &= 8.124 \end{aligned}$$

Dalam hasil perhitungan, diperoleh jarak *Eigenface* citra wajah satu memiliki jarak yang terkecil. Karena jarak *Eigenface* wajah satu dengan testface yang paling kecil, maka hasil identifikasi menyimpulkan bahwa testface lebih mirip dengan wajah satu dibandingkan dengan wajah dua.

3.3 Gambar Aplikasi Absensi

Pada gambar aplikasi absensi ini, akan memberikan penjelasan seperti form, fungsi dari form tersebut dan akan memberikan tampilan gambar pada form visual basic yang akan digunakan. Untuk dapat lebih jelas lagi dapat dilihat pada Tabel 1.

Tabel 1. Gambar aplikasi absensi

Form	Fungsi	Gambar
Absensi	Sebagai tampilan utama pada sistem absensi dimana akan menampilkan daftar kehadiran dari karyawan selain itu juga akan menampilkan pengumuman dan hasil dari karyawan yang telah selesai absen.	
Daftar	Sebagai pendaftaran karyawan yang nantinya akan disimpan data wajahnya di database untuk absensi selanjutnya.	

		
Login	Sebagai untuk keamanan pada sistem absensi dimana admin saja yang akan mengetahui apa nama pengguna dan password untuk masuk ke sistem absensi.	
Laporan	Sebagai penyimpan data dari karyawan	

Untuk pengujian kamera terhadap objek akan menjelaskan bagaimana posisi wajah atau objek terhadap kamera dalam pengambilannya. Dimana akan

terdapat juga keberhasilan dalam pengenalan wajah. Untuk lebih jelasnya akan dapat dilihat seperti Tabel 2.

Tabel 2. Pengujian kamera terhadap objek

Citra Wajah	Posisi	Hasil
	Tampak Samping Tampak Samping Tampak atas Tampak bawah Tampak depan Tampak depan Tampak depan Tampak atas Tampak depan Tampak depan	Tidak terdeteksi Terdeteksi Terdeteksi Terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi
	Tampak Samping Tampak Samping Tampak depan Tampak depan Tampak depan Tampak depan Tampak depan Tampak depan Tampak depan	Tidak terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi
	Tampak Samping Tampak Samping Tampak Samping Tampak Samping Tampak Samping Tampak depan Tampak depan Tampak Samping	Tidak terdeteksi Tidak terdeteksi Tidak terdeteksi Terdeteksi Tidak terdeteksi Tidak terdeteksi Terdeteksi Terdeteksi

	Tampak depan Tampak depan	Tidak terdeteksi Terdeteksi
	Tampak depan Tampak depan	Tidak terdeteksi Terdeteksi Terdeteksi Terdeteksi Tidak terdeteksi Tidak terdeteksi Terdeteksi Terdeteksi Tidak terdeteksi Terdeteksi

Pada pengujian akurasi objek ini akan di cari berapa presentase keberhasilan dalam pengenalan wajah atau objek. Pada pengujian ini akan di buat tiga pengujian akurasi dengan masing-masing objek

memiliki 10 wajah yang sudah disimpan pada database. Untuk lebih jelasnya dapat dilihat pada Tabel 3.

Tabel 3. Pengujian akurasi objek

Objek	Terdeteksi	Tidak Terdeteksi	Banyak Pengujian	Presentase
	7	3	10	$7/10 \times 100 \% = 70\%$
	6	4	10	$6/10 \times 100 \% = 60\%$
	4	6	10	$4/10 \times 100 \% = 40\%$
	6	4	10	$6/10 \times 100\% = 60\%$

Dari pengujian pada tabel diatas bahwa faktor cahaya dan latar belakang sangat mempengaruhi proses pengenalan wajah. Dari hasil pengujian diatas memiliki nilai rata-rata presentase sebesar 60%.

4. Simpulan

Berdasarkan pada uraian yang telah dibahas pada bab sebelumnya dapat disimpulkan bahwa :

1. Dari hasil pengujian bahwa terdapat nilai rata-rata presentase 60% dan tingkat keberhasilan saat pengenalan wajah dengan memiliki 10 data wajah yang sudah disimpan didatabase.
2. Absen dalam kondisi ruangan kurang cahaya dapat mempengaruhi keakuratan.
3. Kamera akan aktif jika jarak antara 30 cm - 70 cm telah terpenuhi.
4. Sensor ultrasonik berfungsi dengan baik dan memiliki nilai akurasi 99.7%, 99.5% dan 98%

karena dekat dengan 30 cm kemudian memiliki akurasi sebesar 92% dan 93% karena semakin jauh dengan jarak 30 cm.

5. Referensi

- Muhammad Rizki Muliawan, Beni Irawan, Y. B. (2015). Implementasi Pengenalan Wajah Dengan Metode Eigenface Pada Sistem Absensi. *Jurnal Coding, Sistem Komputer Untan*, 03(1), 41–50.
- Muliawarman, RM. Implementasi Pengenalan wajah dengan metode eigenface pada system absensi, *Jurnal Coding Sistem Komputer, Untan*, ISSN : 2338-493x
- Suprianto, D, 2013 Sistem Pengenalan wajah secara real time, dalam adobost, eigenface PCA & My SQL, Malang, Universitas Brawijaya.
- Rahim, MA. 2013, Perancangan Aplikasi menggunakan metode Viola Jonnes, Medan : STMIK Budi Dharma
- Brigida, 2012. Pengenalan Pola, http://Informatika.Web.id/pengenalan_pola.htm
- Brigida, 2012. Komponen Sistem Pengenalan Pola, http://Informatika.Web.id/komponen_system_pengenalan_pola.htm