

# Perancangan Aplikasi Absensi Karyawan Dengan Deteksi Wajah Menggunakan Metode Eigenface

Susi Tamba

Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

Email: susitamba3@gmail.com

Email Penulis Korespondensi: susitamba3@gmail.com

**Abstrak**—Absensi adalah salah satu transaksi repetitif yang sangat penting, karena berkaitan dengan produktivitas dari karyawan dan merupakan salah satu indikator pengontrol Sumber daya manusia (SDM) yang bertujuan meningkatkan potensi sumber daya manusia serta digunakan dalam rangka efisiensi. Perkembangan teknologi saat ini memungkinkan membuat suatu sistem yang dapat membantu manusia dalam pengenalan suatu citra digital. Salah satu bidang yang sekarang ini sudah mulai dikembangkan adalah pengenalan pola. Teknologi ini mengidentifikasi ciri-ciri khusus fisik seseorang. Contoh pengenalan pola misalnya adalah pengenalan wajah (face recognition), Pengenalan iris (iris recognition), dan pengenalan sidik jari (finger recognition). Dalam penelitian pengenalan wajah ini menggunakan sebuah kamera untuk menangkap wajah seseorang kemudian dibandingkan dengan wajah yang sebelumnya telah disimpan di dalam database tertentu. Secara garis besar proses dari pengenalan wajah ini adalah kamera webcam melakukan capture pada wajah. Kemudian didapatkan sebuah nilai R,G,B. Dengan menggunakan pemrosesan awal, setelah itu dilakukan tahap pengolahan wajah dengan menggunakan metode eigenface. Didalam metode eigenface ini terdapat beberapa tahapan inti yaitu: mengubah wajah menjadi matriks, menghitung rata-rata FlatVector, menentukan nilai eigenface dan melakukan proses identifikasi wajah dengan mencari nilai eigenface yang mendekati. Pengenalan wajah ini salah satunya dapat dikembangkan untuk menjadi aplikasi absensi yang dapat diterapkan diperusahaan untuk mencegah manipulasi absen oleh karyawan.

**Kata Kunci:** Eigenface; Eigenvektor; Mean Pengenalan; Wajah; Absensi; Matriks

**Abstract**—Attendance is one of the most important repetitive transactions, because it is related to the productivity of employees and is one of the indicators of controlling human resources (HR) which aims to increase the potential of human resources and is used for efficiency. Current technological developments make it possible to create a system that can assist humans in recognizing a digital image. One area that is currently being developed is pattern recognition. This technology identifies a person's special physical characteristics. Examples of pattern recognition, for example, are face recognition, iris recognition, and fingerprint recognition. specific databases. Broadly speaking, the process of facial recognition is that the webcam camera captures the face. Then obtained a value of R, G, B. By using the initial processing, after that the facial processing stage is carried out using the eigenface method. In this eigenface method there are several core stages, namely: converting faces into matrices, calculating the FlatVector average, determining the eigenface values and carrying out the face identification process by looking for the eigenface values that are close to it. One of these facial recognition can be developed to become an attendance application that can be applied in the company to prevent manipulation of absenteeism by employees.

**Keywords:** Eigenface; Eigenvector; Mean Introduction; Face; absenteeism; Matrix

## 1. PENDAHULUAN

Pengenalan wajah manusia adalah bidang penelitian untuk dikombinasikan dengan berbagai objek. Hasil pengenalan tersebut memunculkan banyak variasi. Hal tersebut mungkin diakibatkan cara representasi oleh perbedaan konsep, sehingga dalam penentuan model komputasi menjadi sulit. Pengolahan citra cukup banyak digunakan dalam kehidupan sehari-hari, sehingga dengan pengolahan citra, proses absensi akan menjadi lebih jelas. Proses pendeteksian wajah memiliki peran yang sangat penting dalam sistem pengenalan wajah, karena proses pengenalan akan lebih akurat setelah wajah dalam suatu gambar telah terdeteksi.

Deteksi wajah ini dibagi menjadi dua bagian yaitu : dikenali dan tidak dikenali. Setelah dilakukan perbandingan dengan pola yang sebelumnya disimpan didalam *database*. Perhitungan model pengenalan wajah memiliki beberapa masalah. Kesulitan muncul ketika wajah dipresentasikan dalam suatu pola yang berisi informasi unik yang membedakan dengan wajah lain [1]. Absensi adalah salah satu transaksi repetitif yang sangat penting karena berkaitan dengan produktivitas dari karyawan dan merupakan salah satu indikator pengontrol Sumber daya manusia (SDM) yang bertujuan meningkatkan potensisumber daya manusia serta digunakan dalam rangka efisiensi[2].

*Eigenface* berasal dari Bahasa Jerman "*eigenwert*" dimana "*eigen*" artinya karakteristik dan "*wert*" artinya nilai. Teknik ini pertama sekali dikembangkan oleh Mathew Turk dan Alex Pentland dari *Vision and Modelling Group, The Laboratory, Massachusetts Institute of Technology* pada tahun 1987. Metode ini terus dikembangkan dengan merubah cara perhitungan matriks kovarian melalui perhitungan sebanyak  $N^2 \times N^2$  untuk memperoleh  $N^2$  vektor *eigen* (*eigenvector*) dan nilai *eigen* (*eigenvalue*). Sebaliknya, dengan modifikasi yang diajukan hanya perlu dilakukan  $M \times M$  perhitungan yang lebih efisien dibandingkan perhitungan  $N^2 \times N^2$  [3].

*Eigenfaces* adalah sekumpulan *Standardized Face ingredients* yang diambil dari analisis statistic dari banyak gambar wajah [4]. Metode *Eigenfaces* adalah salah satu metode pengenalan wajah yang didasarkan pada *Principal Component Analysis (PCA)* yang merupakan kumpulan dari *eigenvector* agar dapat digunakan untuk masalah *computer vision* pada pengenalan wajah manusia [5]. Wajah merupakan salah satu bagian dari manusia yang memiliki ciri-ciri berbeda. Wajah dapat digunakan untuk mengenali seseorang, misalnya untuk absensi, pembuatan e-ktip dan sistem pengamanan dengan menggunakan sistem pengenalan wajah. Karena wajah manusia mempresentasikan sesuatu yang kompleks, pengembangan model komputasi yang *ideal* untuk pengenalan wajah adalah sesuatu hal yang sangat penting.

Selain itu sistem pengenalan wajah juga mendapat kesulitan pada orientasi wajah yang berlainan, pencahayaan, latar belakang, potongan rambut, kumis atau jenggot, penutup kepala, kacamata serta perbedaan kondisi misalnya orang tersebut dalam keadaan agak menoleh atau menunduk.

Penggunaan aplikasi absensi akan membantu dalam pengelolaan aplikasi dan terjamin validitasnya karena proses absensi tidak dapat diwakilkan. Pencatatan waktu juga memudahkan proses administrasi laporan sehingga data yang dihasilkan akurat terhadap validitas data dan aktual terhadap waktu pencatatan. Penggunaan kamera sebagai peralatan *input* akan menjadi media yang digunakan sehingga pada proses absensi akan dilakukan verifikasi wajah. Penggunaan kamera ini juga memudahkan *user* karena alat identifikasi berupa wajah yang tidak mungkin tertinggal.

## 2. METODOLOGI PENELITIAN

### 2.1 Citra

Secara harfiah, citra (*image*) adalah gambar pada bidang *dwimatra* (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari *intensitas* cahaya pada bidang dua dimensi. Citra (*image*) adalah suatu *representasi*, kemiripan, atau imitasi dari suatu objek atau benda. Istilah lain untuk citra adalah suatu komponen multimedia yang memegang peranan sangat penting sebagai bentuk informasi *visual*. Citra mempunyai karakteristik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi. Maksudnya sebuah gambar dapat memberikan informasi yang lebih banyak dari pada informasi yang di-sajikan dalam bentuk teks. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat – alat *optik*, misalnya mata pada manusia, kamera, pemindahan (*scanner*), dan sebagainya, sehingga bayangan objek dalam bentuk citra tersebut dapat terekam[6]

### 2.2 Metode Eigenface

*Eigenface* adalah salah satu metode pengenalan wajah yang didasarkan pada *Principal Component Analysis (PCA)*. *PCA* digunakan untuk menghitung *eigenvalue* dan *eigenvector* dari citra atau *image*[7]. Dalam metode *eigenface*, *decoding* dilakukan dengan menghitung *eigenvector* kemudian direpresentasikan dalam sebuah matriks yang berukuran besar. Algoritma *Eigenface* secara keseluruhan cukup sederhana. *Image* Matriks ( $\Gamma$ ) direpresentasikan ke dalam sebuah himpunan matriks ( $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ ). Cari nilai rata-rata ( $\Psi$ ) dan gunakan untuk mengekstraksi *eigenvector* ( $v$ ) dan *eigenvalue* ( $\lambda$ ) dari himpunan matriks. Gunakan nilai *eigenvector* untuk mendapatkan nilai *eigenface* dari *image*. Apabila ada sebuah *image* baru atau *test face* ( $\Gamma_{new}$ ) yang ingin dikenali, proses yang sama juga diberlakukan untuk *image* ( $\Gamma_{new}$ ), untuk mengekstraksi *eigenvector* ( $v$ ) dan *eigenvalue* ( $\lambda$ ), kemudian cari nilai *eigenface* dari *image test face* ( $\Gamma_{new}$ ). Setelah itu barulah *image* baru ( $\Gamma_{new}$ ) memasuki tahapan pengenalan dengan menggunakan metode *euclidean distance*. Metode selengkapnya dapat dilihat seperti :

#### 1. Tahap Perhitungan Eigenface :

- a. Langkah pertama Tahap Perhitungan *Eigenface* adalah menyiapkan data dengan membuat suatu himpunan  $S$  yang terdiri dari seluruh *training image* ( $\Gamma_1, \Gamma_2, \dots, \Gamma_m$ ).

$$S = (\Gamma_1, \Gamma_2, \dots, \Gamma_m) \quad (1)$$

- b. langkah kedua adalah mengambil nilai tengah atau *mean* ( $\Psi$ )

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \quad (2)$$

- c. Langkah ketiga kemudian mencari selisih ( $\Phi$ ) antara *training image* ( $\Gamma_i$ )

dengan nilai tengah ( $\Psi$ )

$$\Phi_i = \Gamma_i - \Psi \quad (3)$$

- d. langkah keempat adalah menghitung nilai matriks *covariant* ( $c$ )

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (4)$$

$$A = [\Phi_1, \Phi_2, \dots, \Phi_M] \quad (5)$$

$$L = AA^T \quad L = \Phi_n \Phi_n^T \quad (6)$$

- e. langkah kelima menghitung *eigenvalue* ( $\lambda$ ) dan *eigenvector* ( $v$ ) dari matriks *covariant* ( $c$ )

$$C \times v_i = \lambda_i \times v_i \quad (7)$$

- f. langkah keenam, setelah *eigenvector* ( $v$ ) diperoleh, maka *eigenface* ( $\mu$ ) dapat dicari dengan :

$$\mu_i = \sum_{k=1}^M v_{lk} \Phi_k \quad l = 1, \dots, M \quad (8)$$

#### 2. Tahap Pengenalan :

- a. Sebuah *image* wajah baru atau *test face* ( $\Gamma$ ) akan dicoba untuk dikenali, pertama terapkan cara pada tahapan pertama perhitungan *eigenface* untuk mendapatkan nilai *eigenface* dari *image* tersebut.

$$\mu_{new} = v \cdot (\Gamma_{new} - \Psi) \quad (9)$$

$$\Omega = (\mu_1, \mu_2, \dots, \mu_3) \quad (10)$$

- b. Gunakan metode *Euclidean Distance* untuk mencari jarak (*distance*) terpendek antara nilai *eigenface* dari *training image* dalam *database* dengan *eigenface* dari *image test face*.

$$\varepsilon = \left| |\Omega - \Omega_k| \right| \quad (11)$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisa Masalah

Sistem absensi berbasis *webcam* adalah sistem yang melakukan absensi karyawan berdasarkan *input* citra hasil *capture webcam*. Sistem akan melakukan posting absen karyawan perhari dan membandingkan hasil *capture* wajah karyawan dengan *database* wajah yang telah di-*input* terlebih dahulu oleh seorang administrator (Admin). Proses-proses yang dilakukan dalam sistem absensi karyawan ini adalah:

1. Proses pemasukan biodata karyawan dengan *file* wajah hasil *capture webcam*.
2. Proses absensi harian karyawan dengan *capture webcam* dan hasil *capture* dibandingkan dengan *file* wajah karyawan.

#### 3.1.1 Penerapan Metode Eigenface

Algoritma pengenalan wajah dimulai dengan membuat matriks kolom dari wajah yang di-*input* ke dalam *database*. Rata-rata *vector* citra (*mean*) dari matriks kolom dihitung dengan cara membaginya dengan jumlah banyaknya citra yang disimpan di dalam *database*.

1. Adapun perhitungan metode *Eigenface* adalah sebagai berikut :
  - a. Langkah pertama adalah dengan menyiapkan variabel  $S$  dari seluruh citra *training*  $S = (\Gamma_1, \Gamma_2, \dots, \Gamma_m)$  dimana setiap citra wajah adalah  $3 \times 3$  *pixel* :



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

Citra Wajah-1



$$\begin{bmatrix} 4 & 4 & 3 \\ 2 & 4 & 4 \\ 3 & 0 & 0 \end{bmatrix}$$

Citra Wajah-2



$$\begin{bmatrix} 4 & 1 & 1 \\ 1 & 2 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

Citra Wajah-3

- b. Langkah kedua adalah perhitungan rata-rata *flatvector mean* ( $\Psi$ ) matriks citra. Dari *flatvector* yang diperoleh, jumlahkan seluruh barisnya sehingga diperoleh matriks berukuran  $1 \times (H \times W)$ . Setelah itu bagi matriks dengan jumlah citra ( $N$ ) yang dalam contoh adalah dua untuk mendapatkan rata-rata *flatvector* (*mean*) dengan rumus sebagai berikut:

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

$$\Psi = \frac{1}{3} \left( \begin{bmatrix} 2 & 3 & 1 \\ 2 & 1 & 2 \\ 1 & 0 & 2 \end{bmatrix} + \begin{bmatrix} 4 & 4 & 3 \\ 2 & 4 & 4 \\ 3 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 4 & 1 & 1 \\ 1 & 2 & 3 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

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

- c. Langkah ketiga mencari selisih ( $\Phi$ ) antara *training image* ( $\Gamma_i$ ) dengan nilai tengah ( $\Psi$ ). Jika diperoleh nilai *negatif*, maka ganti nilainya dengan nol. Proses dapat dihitung dengan persamaan berikut.

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

$$\Phi_1 = \Gamma_1 - \Psi = \begin{bmatrix} 2 & 3 & 1 \\ 2 & 1 & 2 \\ 1 & 0 & 2 \end{bmatrix} - \begin{bmatrix} 3 & 2 & 1 \\ 1 & 2 & 3 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Phi_2 = \Gamma_2 - \Psi = \begin{bmatrix} 4 & 4 & 3 \\ 2 & 4 & 4 \\ 3 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 3 & 2 & 1 \\ 1 & 2 & 3 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 2 & 0 & 0 \end{bmatrix}$$

$$\Phi_3 = \Gamma_3 - \Psi = \begin{bmatrix} 4 & 1 & 1 \\ 1 & 2 & 3 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 3 & 2 & 1 \\ 1 & 2 & 3 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

d. Langkah keempat adalah menghitung nilai matriks *covariant* (c)

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

$$A = [\Phi_1, \Phi_2, \dots, \Phi_M]$$

$$L = AA^T \quad L = \Phi_n \Phi_n^T$$

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

$$L = \begin{bmatrix} 11 & 7 & 2 \\ 7 & 7 & 2 \\ 2 & 2 & 5 \end{bmatrix}$$

e. Langkah kelima menghitung *eigenvalue* ( $\lambda$ ) dan *eigenvector* (v) dari matriks *covariant* (c)

$$L \cdot v_i = \lambda_i \cdot v_i$$

$$L \cdot v = \lambda I \cdot v$$

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

maka *eigenvalue* dapat dihitung dengan rumus :

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

$$\det = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 11 & 7 & 2 \\ 7 & 7 & 2 \\ 2 & 2 & 5 \end{bmatrix} = 0$$

$$\det = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} - \begin{bmatrix} 11 & 7 & 2 \\ 7 & 7 & 2 \\ 2 & 2 & 5 \end{bmatrix} = 0$$

$$\det = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} - \begin{bmatrix} 11 & 7 & 2 \\ 7 & 7 & 2 \\ 2 & 2 & 5 \end{bmatrix} = \begin{bmatrix} \lambda - 11 & 7 & 2 \\ 7 & \lambda - 7 & 2 \\ 2 & 2 & \lambda - 5 \end{bmatrix}$$

akan dihasilkan nilai  $\lambda = 11$ ,  $\lambda = 7$ , dan  $\lambda = 5$ .

$$\text{kemudian mencari nilai } \textit{eigenvector} \text{ dengan } v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

*eigenvector* (v) dapat dihasilkan dengan mensubstitusikan nilai *eigenvalue* ( $\lambda$ ) kedalam persamaan  $(\lambda I - L) v = 0$ .

untuk  $\lambda = 11$ , maka  $\begin{bmatrix} \lambda - 11 & 7 & 2 \\ 7 & \lambda - 7 & 2 \\ 2 & 2 & \lambda - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$

$$\begin{bmatrix} 11 - 11 & 7 & 2 \\ 7 & 11 - 7 & 2 \\ 2 & 2 & 11 - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} 0 & 7 & 2 \\ 7 & 4 & 2 \\ 2 & 2 & 6 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} v_1 & 7v_2 & 2v_3 \\ 7v_1 & 4v_2 & 2v_3 \\ 2v_1 & 2v_2 & 6v_3 \end{bmatrix} = 0$$

Dihasilkan *eigenvector*  $\begin{bmatrix} 0 \\ 7 \\ 2 \end{bmatrix}$

untuk  $\lambda = 7$ , maka

$$\begin{bmatrix} \lambda - 11 & 7 & 2 \\ 7 & \lambda - 7 & 2 \\ 2 & 2 & \lambda - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} 7 - 11 & 7 & 2 \\ 7 & 7 - 7 & 2 \\ 2 & 2 & 7 - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} -4 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & -2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -4v_1 & 7v_2 & 2v_3 \\ 7v_1 & v_2 & 2v_3 \\ 2v_1 & 2v_2 & -2v_3 \end{bmatrix} = 0$$

Dihasilkan *eigenvector*  $\begin{bmatrix} 7 \\ 0 \\ 2 \end{bmatrix}$ .

untuk  $\lambda = 5$ , maka

$$\begin{bmatrix} \lambda - 11 & 7 & 2 \\ 7 & \lambda - 7 & 2 \\ 2 & 2 & \lambda - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} 5 - 11 & 7 & 2 \\ 7 & 5 - 7 & 2 \\ 2 & 2 & 5 - 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} -6 & 7 & 2 \\ 7 & -2 & 2 \\ 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -6v_1 & 7v_2 & 2v_3 \\ 7v_1 & -2v_2 & 2v_3 \\ 2v_1 & 2v_2 & 0v_3 \end{bmatrix} = 0$$

Dihasilkan *eigenvector*  $\begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix}$

sehingga *eigenvector* yang dihasilkan dari matriks  $L$  adalah  $\begin{bmatrix} 0 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix}$

f. Langkah keenam, setelah *eigenvector* ( $v$ ) diperoleh, maka *eigenface* ( $\mu$ ) dapat dicari dengan :

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

$$\mu_1 = v \cdot \Phi_1 = \begin{bmatrix} 0 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mu_2 = v \cdot \Phi_2 = \begin{bmatrix} 0 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 2 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 14 & 4 \\ 7 & 0 & 2 \\ 4 & 0 & 0 \end{bmatrix}$$

$$\mu_3 = v \cdot \Phi_3 = \begin{bmatrix} 0 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

2. Tahap Pengenalan :

a. Cari selisih ( $\Phi_{new}$ ) antara citra *testface* ( $\Gamma_{new}$ ) dengan nilai tengah ( $\Psi$ ), apabila ditemukan nilai dibawah nol, ganti nilainya dengan nol.



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

Citra *testface*

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

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

b. mencari nilai *eigenface* dari citra *testface* ( $\Gamma_{new}$ ).

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

$$\mu_{new} = \begin{bmatrix} 0 & 7 & 2 \\ 7 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- c. Gunakan metode *Euclidean Distance* untuk mencari jarak (*distance*) terpendek antara nilai *eigenface* dari *training image* ( $\Gamma_i$ ) dalam *database* dengan *eigenface* dari *image testface* ( $\Gamma_{new}$ ).

$$\varepsilon - \|\Omega - \Omega_k\|$$

$$\varepsilon_1 - \|\Omega_1 - \Omega_{new}\| = \left\| \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\| = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 0$$

$$\varepsilon_2 - \|\Omega_2 - \Omega_{new}\| = \left\| \begin{bmatrix} 0 & 14 & 4 \\ 7 & 0 & 2 \\ 4 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\| = \begin{bmatrix} 0 & 7 & 4 \\ 0 & 0 & 2 \\ 4 & 0 & 0 \end{bmatrix} = 15$$

$$\varepsilon_3 - \|\Omega_3 - \Omega_{new}\| = \left\| \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\| = \begin{bmatrix} 0 & 7 & 0 \\ 7 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 14$$

Hasil identifikasi lebih mirip dengan *image 1* dari pada *image 2* atau *image 3* karena jarak *eigenface image 1* dengan citra *testface* yang paling kecil, dari hasil persentasi perhitungan diatas dapat disimpulkan tingkat kemiripan mencapai 80%.

#### 4. KESIMPULAN

Berdasarkan hasil analisa yang dilakukan terhadap permasalahan dan aplikasi yang dikembangkan, maka dapat ditarik kesimpulan sebagai berikut Teknik pembuatan absensi menggunakan wajah ini dirancang menggunakan program matlab dengan mendeteksi wajah melalui *webcam*. Penerapan metode *eigenface* untuk mendeteksi wajah dengan membandingkan nilai matriks dari wajah kemudian menormalisasikannya sampai perhitungan *weight* dan jarak dari wajah (*image*) tersebut mendekati dan dapat ditemukan. Perancangan aplikasi absensi dengan deteksi wajah menggunakan metode *eigenface* dapat bekerja dengan baik pada *webcam*, baik dengan resolusi kecil maupun tinggi, sedangkan pada *webcam internal* program bekerja secara otomatis dalam mengenali *webcam*.

#### REFERENCES

- [1]. Turk, M and Pentlant, A. (1991) Face Recognition Using Eigenfaces, *Proc. IEEE Conf. of Computer Vision and Pattern Recognition*, 13,586-591.
- [2]. Rinaldi Munir, *Pengolahan Citra Digital dengan Pendekatan Algoritmik*, penerbit Informatika, Bandung, 2004
- [3]. Al Fattah.H.2009. "Rekayasa Sistem Pengenalan Wajah". Yogyakarta : ANDI.
- [4]. Indra, (2012), Sistem Pengenalan Wajah Dengan Metode Eigenface, *Journal pendidikan Teknologi-Informasi & Komunikasi Terapan*, 138,2012.
- [5]. Jogyanto (2005:196), *Analisis dan Desain Sistem Informasi*.
- [6]. Tata Sutabri (2004:9), *Analisis Sistem Informasi*.
- [7]. Drs.H. Malayu S.P. Hasibuan, (2008 : 84), *Manajemen Sumber Daya Manusia*.
- [8]. Yambor, W.S. (2000) Analysis of PCA-Based and Fisher Discriminant-Based Image Recognition Algorithms, Tesis of Master, Colorado State University. Yourdon, Edward. 1989. *Modern Structured Analysis*. New Jersey : Prentice-Hall International, Inc.
- [9]. Hermawati, Fajar Astuti (2013 : 4-5), "Pengolahan Citra Digital" Yogyakarta, Penerbit ANDI.
- [10]. Ni Wayan Marti, Pengenalan Citra Wajah Manusia Menggunakan metode Eigenface, *Journal Pendidikan Teknik dan Kejuruan*, 11,2010.
- [11]. [http:// www.ilmukomputer.com](http://www.ilmukomputer.com) 15 April 2013.
- [12]. [http:// www.wikipedia.org/wiki/Webcam](http://www.wikipedia.org/wiki/Webcam), 19 Sept 2009 12 Jan 2013.
- [13]. <http://www.pages.drexel.edu/~sis26/Eigenface%20Tutorial.htm> 12 Jan 2013.
- [14]. Beni Utomo, *Matematika Eigenface Menggunakan Metrik Euclidean*, *journal pendidikan karakter dalam pembelajaran*,2011.
- [15]. Kholistianingsih, *Sistem Pengenalan Wajah menggunakan Metode Eigenface*, *Journal Pendidikan*, 49-57, Desember 2012.
- [16]. Nopriyansyah, Muhammad Akbar, dan Helda Yudiastuti, *Analisis Dan Perancangan Aplikasi Pengenalan Wajah Pada Citra Digital*, *Journal Ilmiah 1-20*, April 2012.