

SISTEM PENGENALAN WAJAH DENGAN METODE EIGENFACE UNTUK ABSENSI PADA PT FLORINDO LESTARI

Indra

Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta 12260
E-mail : indra@budiluhur.ac.id

ABSTRAK

Absensi adalah salah satu transaksi repetitif yang sangat penting, karena berkaitan dengan produktifitas 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 satunya 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), pengenalan sidik jari (finger recognition), dan lain-lain. 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 kamer webcam melakukan capture pada wajah. Kemudian didapatkan sebuah nilai R,G,B. Dengan menggunakan pemrosesan awal, dilakukan crop, konversi RGB ke Grayscale. Setelah dilakukan proses Grayscale, dilakukan tahap pengolahan wajah dengan menggunakan metode eigenface. Didalam metode eigenface ini terdapat beberapa tahapan inti yaitu: mengubah wajah menjadi matrik, menghitung rata-ran 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, FlatVector, matrik, grayscale, Absensi, Wajah*

1. PENDAHULUAN

1.1 Latar Belakang

Beberapa perusahaan belum memiliki sistem absensi sehingga perusahaan tersebut belum memiliki data tentang persentase kehadiran karyawan tersebut. Kondisi inilah yang terjadi pada PT Florindo Lestari. PT Florindo Lestari merupakan suatu perusahaan yang bergerak dalam bidang *ekspor* dan *import* barang. Dimana perusahaan tersebut cukup mempunyai banyak karyawan. Setiap hari kerja karyawan masuk pada pukul delapan pagi dan pulang pukul empat sore. PT Florindo Lestari belum memiliki sistem absensi sehingga data masuk karyawan belum dapat dicatat dan belum memiliki data tentang tingkat persentase kehadiran para karyawannya.

Umumnya sistem absensi karyawan pada perusahaan dilakukan dengan mengisi buku absen atau yang lebih maju lagi dengan menggunakan mesin absensi. Mesin absensi tersebut menggunakan kartu tempat mencetak tanggal, jam masuk serta jam pulang. Dilihat dari segi keamanan, sistem ini mempunyai banyak kelemahan, antara lain adalah absen karyawan bisa dititipkan oleh karyawan lainnya, dan jika kartu tersebut tertinggal maka karyawan tidak dapat melakukan absensi. Hal ini tentu berakibat kerugian bagi karyawan. Alternatif lain yang bisa dilakukan adalah dengan menggunakan sidik jari sebagai pengganti *barcode*. Akan tetapi, pada absensi dengan mesin sidik jari masih ditemukan beberapa kekurangan seperti ketidak mampuan mesin absensi sidik jari *memverifikasi* jari karyawan apabila jari karyawan itu kotor, terluka ataupun berminyak, dan memerlukan tambahan alat dengan harga yang tidak murah.

1.2 Tujuan Penelitian

Dengan melihat permasalahan di atas, maka dirancang dan dibangun perangkat lunak yaitu :

- Membuat aplikasi berbasis pengenalan wajah (*face recognition*) dengan menggunakan metode eigenface untuk melakukan proses absensi karyawan.
- Mempelajari prinsip dasar dari pengenalan wajah dengan menggunakan metode Eigenface.
- Mengkomputerisasikan data absensi karyawan sehingga lebih mudah dalam pelaporan ke pihak eksekutif.

1.3 Identifikasi Masalah

Bagaimana merancang sistem absensi untuk mendapatkan tingkat keakuratan yang lebih maksimal dan dengan biaya yang lebih murah yaitu dengan menggunakan metode Grayscale.

2. METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam penelitian ini dibagi dalam beberapa tahap, yaitu: mempelajari algoritma eigenface, mengimplementasikan algoritma dalam prototype, melakukan pengujian algoritma dan analisa hasil pengujian algoritma.

2.1 Memahami kebutuhan informasi

Pada tahapan ini dilakukan proses analisis yang mendalam mengenai kebutuhan informasi dari proses absensi mahasiswa. Karena jumlah karyawan yang semakin dengan proses absensi yang masih manual. Maka, absensi manual tersebut dapat menghambat proses bisnis perusahaan.

2.2 Memahami Alur Metode Eigenface

Pada tahap ini dilakukan analisis yang mendalam mengenai alur dari proses absensi menggunakan metode eigenface. Inti dari algoritma eigenface adalah ketika karyawan melakukan absensi maka gambar foto karyawan akan disimpan dalam database aplikasi, lalu dikalkulasi nilai eigenface gambar karyawan tersebut dan dilakukan proses matching untuk dicari nilai eigen yang paling mendekati.

2.3 Alur Proses Identifikasi dengan Metode Eigenface

Pada tahap ini akan dilakukan proses perhitungan matriks dari setiap wajah yang dicapture oleh webcam. Tahapan yang terjadi adalah penyusunan flatvector matriks citra, menentukan nilai rata-rata flatvector, penghitungan nilai eigenface dan proses identifikasi untuk mendapatkan identitas gambar yang dicapture.

2.4 Pengujian Algoritma pada Metode Eigenface

Pada tahapan ini dilakukan pengujian dengan melakukan absensi kepada lima orang. Percobaan ini didasarkan atas pernyataan dari Ming-Hsuan, Kriegman, dan Ahuja pada 2002 yang mengatakan salah satu faktor dari sistem pengenalan wajah adalah Kondisi pencitraan yang salah satu contohnya adalah pencahayaan dan jarak. Pencahayaan yang dilakukan menggunakan cahaya gelap dan terang. Serta untuk jarak dengan webcam dengan 30 cm dan 60 cm.

2.5 Analisa Hasil Pengujian Algoritma

Pada tahapan terakhir ini, dilakukan proses analisa yang mendalam mengenai hasil dari pengujian algoritma eigenface. Setelah dilakukan pengujian sampai dengan enam kali maka didapatkan kesimpulan sebagai saran dan masukan untuk pengembangan absensi dengan metode eigenface ini kedepannya.

3. PEMBAHASAN METODE EIGENFACE

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 menghitung *eigenvector* kemudian direpresentasikan 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, The Media Laboratory, Massachusetts Institute of Technology pada tahun 1987. Metode ini disempurnakan lagi oleh Turk dan Pentland pada tahun 1991. Prinsip dasar dari metode *eigenface* adalah bagaimana caranya untuk mengekstrak informasi yang relevan dari sebuah citra wajah lalu mengubahnya ke dalam satu set kode yang paling efisien, dan membandingkan kode wajah ini dengan database berisi beragam wajah yang telah dikodekan secara serupa [9].

3.1 Algoritma Eigenface

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

3.1.1. Penyusunan Flatvector matriks citra

Langkah pertama adalah menyusun seluruh *training image* menjadi 1 matrix tunggal. Misalnya *image* yang kita simpan berukuran $H \times W$ piksel dan jumlahnya N buah, maka memiliki *flatvector* dengan dimensi $N \times (H \times W)$. representasikan semua matriks *training* menjadi matriks dengan bentuk $N \times 1$ atau matriks linier seperti yang ditunjukkan berikut ini:

$$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix} \rightarrow [a \ b \ c \ x \ y \ z]$$

Gambar 1 : Skema Matriks Wajah

Misalnya didalam *training image* terdapat tiga image ukuran 3x3 piksel maka kita akan mempunyai *eigenvector* ukuran 2x9. Contoh dibawah ini menggunakan empat wajah citra yang telah diubah menjadi matrix, lalu matrix tersebut diubah kedalam bentuk rataa *FlatVector*.

$$C_1 \rightarrow \begin{bmatrix} 10 & 10 & 10 \\ 10 & 10 & 10 \\ 10 & 10 & 10 \end{bmatrix} \rightarrow [10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10]$$

$$C_2 \rightarrow \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix} \rightarrow [2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2]$$

$$C_3 \rightarrow \begin{bmatrix} 11 & 11 & 11 \\ 11 & 11 & 11 \\ 11 & 11 & 11 \end{bmatrix} \rightarrow [11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11 \ 11]$$

$$C_4 \rightarrow \begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 9 & 9 & 9 \end{bmatrix} \rightarrow [9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9 \ 9]$$

Gambar 2: Skema *FlatVector* Wajah

3.1.2. Hitung Rataan *FlatVector*

Dari *FlatVector* yang diperoleh, jumlahkan seluruh barisnya sehingga kita peroleh matrix berukuran 1 x (H x W).

$$C_1 + C_2 + C_3 + C_4 = \begin{bmatrix} 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 \\ 9 & 9 & 9 & 9 & 9 & 9 & 9 & 9 & 9 \end{bmatrix} \quad (1)$$

$$C_1 + C_2 + C_3 + C_4 = [32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32] \quad (2)$$

Setelah itu bagi matriks tadi dibagi dengan jumlah image N untuk mendapatkan Rataan *FlatVector*.

$$\frac{[32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32 \ 32]}{4} = [8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8] \quad (3)$$

Nilai *flatvector* citra akan digunakan untuk menghitung nilai *eigenface* citra wajah untuk *training image*.

3.1.3. Tentukan Nilai *Eigenface*

Dengan memakai rataa *flatvector* citra di atas nilai *eigenface* untuk matriks *flatvector* yang sudah disusun tersebut dapat dihitung nilai *eigenfacenya*. Caranya dengan mengurangi baris-baris pada matrix *flatvector* dengan rataa *flatvector*. Jika didapatkan nilai dibawah nol, maka nilainya diganti dengan nol.

$$C_1 = \frac{10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10 \ 10}{2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2} - \quad C_2 = \frac{2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \ 2}{8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8 \ 8} - \frac{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0}{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0}$$

$$C_3 = \begin{matrix} 1111111111 \\ 888888888 \\ 333333333 \end{matrix} \quad C_4 = \begin{matrix} 999999999 \\ 888888888 \\ 111111111 \end{matrix}$$

(4)

3.1.4. Proses Identifikasi

Untuk mengenali citra tes (*testface*), langkah identifikasinya adalah hitung nilai *eigenface* untuk matriks *testface* dengan cara sebelumnya untuk penentuan nilai *eigenface* dan *flatvector* citranya.

$$Ct \rightarrow \begin{bmatrix} 9 & 9 & 9 \\ 9 & 7 & 9 \\ 9 & 7 & 9 \end{bmatrix} \rightarrow (999979979)$$

$$\rightarrow \begin{bmatrix} 9 & 9 & 9 & 9 & 7 & 9 & 9 & 7 & 9 \\ 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} -$$

Nilai eigen dari testface = (1 1 1 1 0 1 1 0 1) (5)

Nilai *eigen* (*eigenvalue*) dari *testface* digunakan untuk identifikasi dengan menentukan jarak terpendek dengan *eigenface* dari *eigenvector training* dengan cara menentukan nilai *absolut* dari pengurangan baris I pada matriks *eigenface training* citra dengan *eigenface* dari *testface* dan jumlahkan dengan elemen penyusun *vector* yang dihasilkan dari pengurangan dan didapat jarak d indeks I dan cari nilai d yang paling kecil.

Setelah nilai *eigenface* untuk *testface* diperoleh maka kita bisa melakukan identifikasi dengan menentukan jarak terpendek dengan *eigenface* dari *eigenvector training image*. Pertama tentukan nilai *absolut* dari pengurangan baris pada matrix *eigenface training image* dengan *eigenface* dari *testface*, kemudian jumlahkan elemen-elemen penyusun *vector* yang dihasilkan dari pengurangan tadi dan ditemukan jarak indeks. Lakukan untuk semua baris. Cari nilai yang paling kecil.

Nilai Eigenface $C_1 = (222222222)$

$$C_1 = \frac{\begin{matrix} 222222222 \\ 111101101 \\ 111121121 \end{matrix}}{111121121} -$$

$C_1 = 1+1+1+1+2+1+1+2+1 = 11$ (6)

Nilai Eigenface $C_2 = (000000000)$

$$C_2 = \frac{\begin{matrix} 000000000 \\ 111101101 \\ -1-1-1-10-1-10-1 \end{matrix}}{-1-1-1-10-1-10-1} -$$

$C_2 = 1+1+1+1+0+1+1+0+1 = 7$ (7)

Nilai Eigenface $C_3 = (333333333)$

$$C_3 = \frac{\begin{matrix} 333333333 \\ 111101101 \\ 222232232 \end{matrix}}{222232232} -$$

$C_3 = 2+2+2+2+3+2+2+3+2 = 20$ (8)

Nilai Eigenface $C_4 = (111111111)$

$$C_4 = \begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{matrix}$$

$$C_4 = 0+0+0+0+1+0+0+1+0= 2 \quad (9)$$

Dari hasil perhitungan, diperoleh jarak citra wajah empat memiliki nilai yang terkecil yaitu dua. Karena jarak *eigenface face* satu dengan *testface* yang paling kecil, maka hasil identifikasi menyimpulkan bahwa *testface* lebih mirip dengan *face* empat dari pada *face* satu, *face* dua, dan *face* tiga.

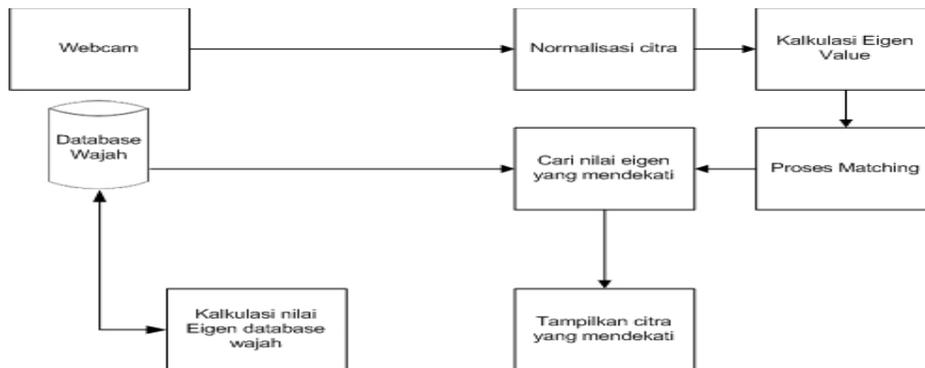
4. ANALISA DAN IMPLEMENTASI METODE EIGENFACE

Pada tahapan ini dilakukan analisa dari hasil pengujian pengenalan wajah dengan menggunakan metode eigenface. Terdapat beberapa tahapan dalam melakukan pengujian dan analisa terhadap metode ini.

4.1 Sistem Pemecahan Masalah

Dalam hasil absensi yang akurat, dibutuhkan sebuah sistem absensi cukup mudah digunakan oleh karyawan, dan memiliki kunci yang unik untuk setiap orang. Maka dari itu dikembangkanlah sebuah sistem yang memanfaatkan wajah manusia yang dimiliki setiap orang, sifatnya unik. Sehingga karyawan tidak perlu repot membawa kartu karena memanfaatkan wajah sebagai media absensi. Sistem pengenalan wajah dengan algoritma *eigenface* adalah sebuah solusi identifikasi wajah dan pengenalan wajah. Sistem ini dapat diterapkan baik dalam aplikasi desktop yang menggunakan wajah sebagai autentikasinya atau pengenalan dan identifikasi wajah otomatis.

Untuk menghasilkan *eigenface*, sekumpulan besar citra digital dari wajah karyawan diambil pada kondisi pencahayaan yang sama melalui *webcam* kemudian dinormalisasikan dan kemudian diolah pada resolusi yang sama. kemudian diperlukan sebagai vector dimensi di mana komponennya diambil dari nilai dari pikselnya. Setelah itu, dilakukan proses kalkulasi Eigen value untuk didapatkan nilai matriks dengan melalui beberapa tahap yaitu: mencari nilai Matriks FlatVector, Mendapatkan nilai rata-rata FlatVector, dan mendapatkan nilai eigenface. Setelah didapatkan nilai eigenface maka dilakukan proses matching dengan mencari nilai eigenface dengan database wajah untuk mendapatkan nilai eigenface yang paling mendekati. Tahapan terakhir adalah menampilkan gambar/citra yang mendekati dari hasil nilai eigenface yang sudah didapatkan . Untuk lebih jelasnya dapat dilihat pada Gambar 3.1 berikut :



Gambar 3 : Langkah-Langkah Proses Identifikasi

4.2 Metode Kerja Sistem Aplikasi

Proses pengenalan wajah manusiadari sisi aplikasi ini terdiri dari beberapa tahap. Tahap pertama dimulai dengan pengambilan gambar dari wajah karyawan melalui alat pengambilan gambar (*webcam*). Setelah itu, gambar diproses untuk menghasilkan database wajah. Penjelasan dari tiap-tiap tahapan proses adalah sebagai berikut :

- 1) Pengambilan Data
Pada tahap ini, setiap data karyawan menginput data masing masing dengan lengkap di form master karyawan, setelah lengkap maka karyawan difoto melalui *webcam*. Posisi karyawan harus benar dalam pengambilan gambar. Penerangan diruangan tempat pengambilan gambar harus sama.
- 2) Proses Absensi
Pada tahap ini karyawan berdiri tegak dan menyesuaikan wajahnya agar tepat pada posisi *webcam*. Setelah tepat maka karyawan menekan tombol ambil yang terdapat pada form. Jika wajah teridentifikasi maka selanjutnya karyawan

menekan tombol absen. Jika wajah karyawan berbeda, maka karyawan harus menekan tombol batal dan lalu mencari posisi kembali yang sesuai sampai data karyawan tersebut tepat.

3) Pembuatan Laporan

Pada tahap ini user perlu mengintu tanggal awal dan tanggal akhir yang diinginkan lalu masukan jumlah hari libur diluar hari libur minguan. dan lalu menekan tombol cetak.

4.3 Spesifikasi *Hardware* dan *Software*

Agar sistem ini berjalan dengan baik, maka dibutuhkan beberapa spesifikasi minimal yang harus dipenuhi.

1. *Hardware*

Dibawah ini merupakan spesifikasi minimal *hardware* yang harus dipenuhi untuk mendukung dalam pengoperasian sistem pengenalan wajah yang dibuat.

- 1) Processor Intel Pentium 4 (2,8 Ghz)
- 2) Memori 1GB RAM
- 3) *Keyboard*
- 4) *Optical Mouse*
- 5) Hardisk 20 GB
- 6) *Webcam*
- 7) Monitor
- 8) VGA On Board 32 MB

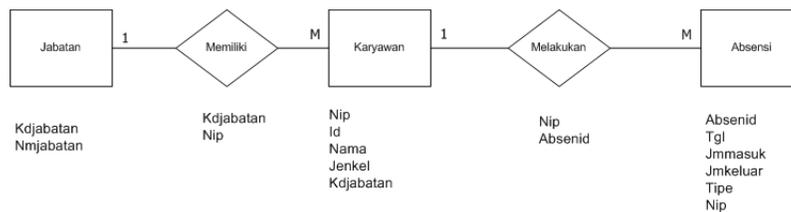
2. *Software*

Dibawah Ini merupakan spesifikasi *software* yang dibutuhkan agar sistem pengenalan wajah dapat berjalan dengan baik.

- 1) Sistem Operasi Microsoft Windows XP *Profesional Edition* SP 2
- 2) connector odbc 5.1.9 win32
- 3) XAMPP Ver 1.6.8

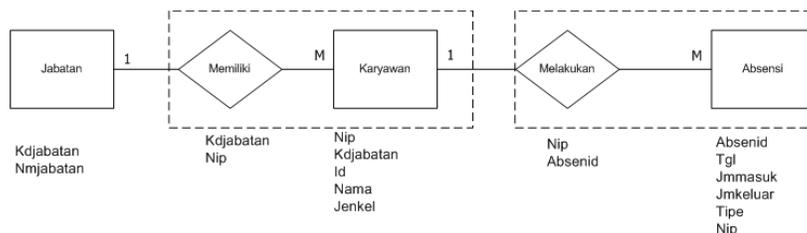
4.4 Rancangan Struktur Basis Data

4.4.1. *Entity Relation Diagram (ERD)*



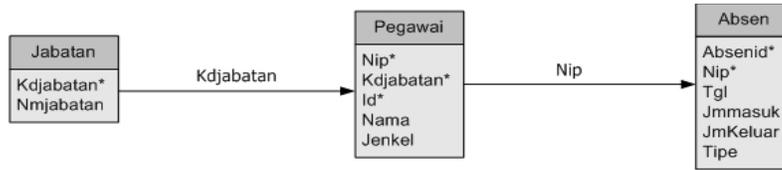
Gambar 4 : Rancangan *Entity Relation Diagram*

4.4.2. Transformasi ER-Diagram ke *Logical Record Structure*



Gambar 5 : Rancangan ER-Diagram ke *Logical Record Structure*

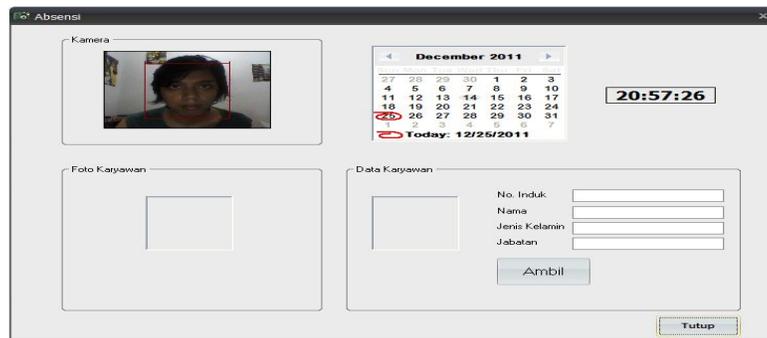
4.4.3. Logical Record Structure (LRS)



Gambar 6 : Rancangan Logical Record Structure

4.5 Tampilan Layar

Form Absensi digunakan untuk karyawan melakukan absensi masuk dan keluar. Pertama posisi wajah karyawan harus berada di dalam kotak merah yang terdapat di form. Lalu karyawan menekan tombol ambil. Jika wajah karyawan cocok maka akan tampil foto dan data karyawan yang sesuai. Jika bukan dirinya maka karyawan menekan tombol batal dan lalu mencari posisi yang pas dan melakukan absensi ulang. Jika sudah benar maka karyawan menekan tombol absensi. Bentuk tampilannya dapat dilihat pada Gambar 4.15.



Gambar 7 : Tampilan Layar Absensi

8. KESIMPULAN

Berdasarkan hasil analisa yang dilakukan terhadap permasalahan dan aplikasi yang dikembangkan, maka dapat ditarik kesimpulan sebagai berikut :

- Aplikasi Absensi dapat dibuat menggunakan pengenalan wajah dengan menggunakan metode eigenface untuk pengenalan wajah.
- Faktor-faktor yang mempengaruhi hasil pengenalan pada sistem ini adalah pencahayaan, jarak *capture* antara objek. Kesimpulan ini didapat dari hasil perbandingan data pengamatan yang dicoba pada 5 orang pada intensitas cahaya yang berbeda dan jarak pengambilan gambar. Terdapat perbedaan keakuratan.
- Aplikasi presensi ini belum menunjukkan hasil maksimal (100%), sehingga belum sepenuhnya dapat menggantikan sistem absensi dengan menggunakan sidik jari dan dengan kartu namun dapat dijadikan alternatif dalam melakukan absensi.
- Metode eigenface merupakan salah satu metode untuk pengenalan wajah dengan memperhitungkan matriks setiap wajah yang dicapture dengan mengkalkulasi nilai eigenface untuk didapatkan nilai eigenface yang mendekati.

DAFTAR PUSTAKA

- [1] Li Stan Z., Jain Anil K. ,2005, *Handbook of Face Recognition*, New York: Springer
- [2] Privida Kristiono, 2008, *Pemrograman Database Tingkat Lanjut dengan VB6*, PT Elex Media Komputindo
- [3] Putra Darma, 2009, *Sistem Biometrika: Konsep Dasar, Teknik Analisis Citra, dan Tahapan Membangun Aplikasi Sistem Biometrika*, Yogyakarta : Andi
- [4] Siswanto, Ir., 2005, *Kecerdasan Tiruan*, Yogyakarta : Graha Ilmu
- [5] Supardi, Ir., Yuniar, 2011, *Semua Bisa Menjadi Programmer VB6 Hingga VB 2008 Basic*, PT Elex Media Komputindo.
- [6] Sutoyo T., Mulyanto Edy S.Si., M.Kom., Dr. Vincent Suhartono, 2009, *Teori Pengolahan Citra Digital*, Yogyakarta: Andi
- [7] Swastika, Windra, 2007, *VB & MYSQL Seri 2*, Dian Rakyat.
- [8] Wikipedia the free encyclopedia, *Eigenface*, <http://en.wikipedia.org/wiki/Eigenface>, diakses pada : Oktober 2011
- [9] Wikipedia the free encyclopedia, *Pengenalan Pola* http://id.wikipedia.org/wiki/Pengenalan_pola, diakses pada : Oktober 2011
- [10] Turk & Pentland, 1991, *Face Recognition Using Eigenfaces* ,http://www.cs.tau.ac.il/~shekler /Seminar2007a/PCA%20and% 20Eigenfaces/eigenfaces_cvpr.pdf, diakses pada: Desember 2011