

Sistem Pengenalan Wajah Secara *Real-Time* dengan *Adaboost*, *Eigenface* PCA & MySQL

Dodit Suprianto, Rini Nur Hasanah, Purnomo Budi Santosa

Abstrak – Aplikasi sistem pengenalan wajah secara *real time* dapat ditemukan pada sistem pengawasan, identifikasi dan keamanan yang berbasis pada pengenalan wajah. Pengamatan wajah secara langsung oleh manusia mempunyai kelemahan karena kelelahan dan kejenuhan yang mungkin terjadi dapat menyebabkan menurunnya ketelitian. Untuk itu penggunaan komputer dapat menjadi alternatif solusi. Pada penelitian ini pengenalan wajah dilakukan melalui tahap *face detection*, *feature extraction* dan *face recognition*, selanjutnya dicocokkan dengan data profil yang tersimpan di dalam *database*. Pendeteksian wajah menggunakan metode *Adaboost*, pengenalan wajah menggunakan metode *Eigenface PCA* dan *database MySQL* untuk menyimpan informasi profile. Penggunaan metode tersebut untuk pengenalan wajah pada kondisi *real time* dengan perbedaan jarak antara sensor dan wajah, posisi wajah, intensitas cahaya yang mengenai wajah, mimik muka dan atribut wajah dalam penelitian ini memberikan tingkat keberhasilan sebesar 80% dalam mengidentifikasi wajah.

Kata Kunci—*Face detection*, *Face recognition*, *Adaboost*, *Eigenfaces PCA*, *MySQL*.

I. PENDAHULUAN

PENGENALAN wajah adalah teknologi komputer untuk menentukan lokasi wajah, ukuran wajah, deteksi fitur wajah dan pengabaian citra latar, selanjutnya dilakukan identifikasi citra wajah [1]. Pengenalan wajah melibatkan banyak variabel, misalnya citra sumber, citra hasil pengolahan citra, citra hasil ekstraksi dan data profil seseorang. Dibutuhkan juga alat pengindra berupa sensor kamera dan metode untuk menentukan apakah citra yang ditangkap oleh *webcam* tergolong wajah manusia atau bukan, sekaligus untuk menentukan informasi profil yang sesuai dengan citra wajah yang dimaksud.

Pengenalan wajah diperlukan oleh berbagai pihak, antara lain sipil, kepolisian maupun militer untuk verifikasi identifikasi dan kontrol akses fisik [2].

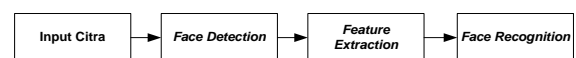
Dodit Suprianto adalah Mahasiswa Program Magister Teknik Elektro, Fakultas Teknik Universitas Brawijaya Malang (email: doditsuprianto@gmail.com)

Rini Nur Hasanah adalah Dosen Teknik Elektro, Fakultas Teknik Universitas Brawijaya Malang (email: rini.hasanah@ub.ac.id)

Purnomo Budi S. adalah Dosen Teknik Mesin, Fakultas Teknik Universitas Brawijaya Malang (email: budi_akademika@yahoo.com)

Pengenalan wajah diperlukan sebagai alat pengawasan, penandaan otomatis (*automatic tagging*) dan interaksi robot dan manusia [3]. Berdasarkan beberapa penelitian sebelumnya maka peneliti mengangkat topik sistem pengenalan wajah secara *real-time*.

Pada gambar 1 ditunjukkan blok diagram metode pengenalan wajah yang tersusun tiga bagian: *face detection*, *feature extraction* dan *face recognition*.

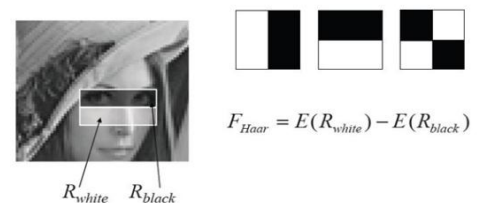


Gambar 1. Blok diagram pengenalan wajah

A. Face Detection

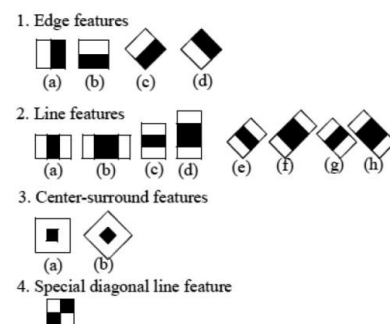
Face detection pada penelitian ini menggunakan metode yang dikemukakan oleh Viola & Jones [1], terbagi menjadi 4 komponen utama: *Haar Like Feature*, *Integral Image*, *Adaptive Boosting* dan *Cascade of Classifier*.

▪ Haar Like Feature



Gambar 2. Skema kerja *Haar Like Feature*

Gambar 2 menunjukkan skema *Haar Like feature* [4] yang memproses citra dalam wilayah kotak-kotak, berisi beberapa *pixel* dari bagian citra. Pixel-pixel dalam satu wilayah tersebut dijumlahkan dan dilakukan proses perhitungan (pengurangan rata-rata nilai pixel di bagian kotak yang terang dan gelap) sehingga diperoleh perbedaan nilai unik disetiap wilayah kotak-kotak tersebut.



Gambar 3. Fitur *Haar Like*

Gambar 3 menunjukkan model fitur *Haar Like* yang

dikembangkan oleh Lienhart, Kuranove dan Pisarevsky untuk mendeteksi objek wajah [5].

▪ *Integral Image*

Citra *integral* adalah citra yang nilai tiap *pixel*-nya merupakan akumulasi dari nilai *pixel* atas dan kirinya. Sebagai contoh, $pixel(a,b)$ memiliki nilai akumulatif untuk semua $pixel(x,y)$ dimana $x \leq a$ dan $y \leq b$.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

Dimana $ii(x,y)$ adalah citra *integral* dan $i(x,y)$ adalah citra asli dengan kondisi:

$$ii(x, y) = i(x, y) + ii(x - 1, y) + ii(x, y - 1) - ii(x - 1, y - 1) \quad (2)$$

▪ *Adaptive Boosting*

Secara umum *boosting* adalah menambahkan *weak learner* ke dalam satu *strong learner*. Setiap perulangan, satu *weak learner* belajar dari suatu data latih. Kemudian *weak learner* tersebut ditambahkan ke dalam *strong learner*. Setelah *weak learner* ditambahkan, data-data kemudian diubah masing-masing bobotnya. Data-data yang mengalami kesalahan klasifikasi akan mengalami penambahan bobot dan data-data yang terklasifikasi dengan benar akan mengalami pengurangan bobot, oleh karena itu *weak learner* pada perulangan selanjutnya akan lebih terfokus pada data-data yang mengalami kesalahan klasifikasi oleh *weak learner* sebelumnya.

$$F(x) = \sum_{t=1}^T \alpha_t h_t(x) \quad (3)$$

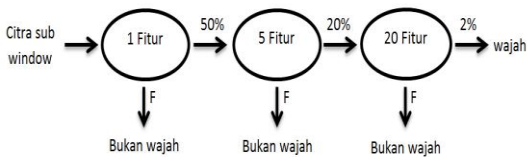
$h_t(x)$ = *weak* atau *basic classifier*

α = tingkat pembelajaran (*learning rate*)

$H(x)$ = dilambangkan dengan $F(x)$ *strong* atau *final classifier*

▪ *Cascade Of Classifier*

Dengan mengkombinasikan pengklasifikasian dalam sebuah *Cascade of Classifier* kecepatan proses pendeteksian dapat meningkat, yaitu dengan memusatkan perhatian pada daerah-daerah dalam citra yang berpeluang saja. Hal ini dilakukan untuk menentukan dimana letak objek yang sedang dicari pada suatu citra.



Gambar 4. Alur Kerja Klasifikasi Bertingkat

Gambar 4 menunjukkan klasifikasi tingkat pertama, tiap sub-citra akan diklasifikasi menggunakan satu fitur. Klasifikasi ini kira-kira akan menyisakan 50% sub-citra untuk diklasifikasi ditahap kedua. Jumlah sub-citra yang lolos klasifikasi akan berkurang hingga mencapai jumlah sekitar 2%.

Struktur *cascade* berguna untuk menolak *sub-window*

negatif sebanyak mungkin dan secepat mungkin di level yang memungkinkan. Sementara contoh positif akan memicu evaluasi setiap *classifier* pada *cascade*. Pengklasifikasian selanjutnya dilatih menggunakan contoh-contoh yang melewati semua tahap sebelumnya.

B. *Face Recognition*

Prinsip dasar pengenalan wajah adalah mengutip informasi unik wajah, kemudian di-*encode* dan dibandingkan dengan hasil *decode* yang sebelumnya dilakukan. Pada metode *eigenface*, *decoding* dilakukan dengan menghitung *eigen vector* kemudian direpresentasikan dalam sebuah matriks. *Eigen vector* juga dinyatakan sebagai karakteristik wajah oleh karena itu metode ini disebut dengan *eigenface* [6]. Setiap wajah direpresentasikan dalam kombinasi linear *eigenface*. Metode *eigenface* adalah bagaimana cara menguraikan informasi yang relevan dari sebuah citra wajah, kemudian mengubahnya ke dalam satu set kode yang paling efisien dan membandingkan kode wajah tersebut dengan database berisi beragam wajah yang telah dikodekan secara serupa.

Eigenfaces PCA digunakan untuk mereduksi dimensi sekumpulan atau ruang gambar sehingga basis atau sistem koordinat yang baru dapat menggambarkan model yang khas dari kumpulan tersebut dengan lebih baik. Model yang diinginkan merupakan sekumpulan wajah yang dilatihkan. Fitur yang baru tersebut akan dibentuk melalui kombinasi linear. Komponen fitur ruang karakter ini tidak akan saling berkorelasi dan akan memaksimalkan perbedaan yang ada pada variabel aslinya.

Secara garis besar langkah-langkah metode PCA adalah sebagai berikut:

- Cari matriks u
- Cari matriks *covariance* : $C = u^T x u$
- Cari eigen values (λ) dan *eigen vector* (V) dari matriks C
- Cari matriks *eigenface*

Matriks *eigenface* dapat digunakan untuk pengenalan citra.

II. METODE PENGENALAN WAJAH SECARA REAL TIME

Pada gambar 5 menunjukkan diagram alir aplikasi sistem pengenalan wajah yang dibangun, yang dapat dijelaskan sebagai berikut ini:

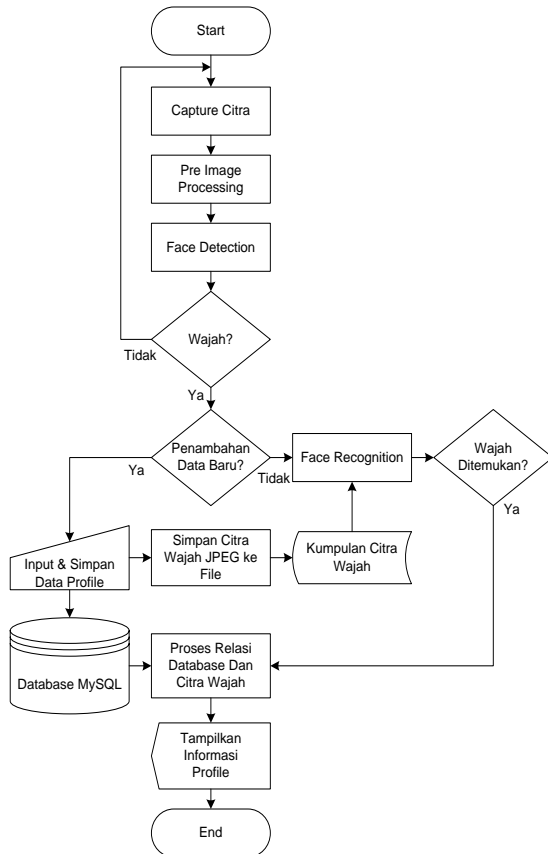
Citra diperoleh dari *sensor webcam*, citra warna RGB dikonversi menjadi citra *grayscale* untuk mengurangi kedalaman warna.

Dari citra *grayscale* dilakukan pendeteksian wajah dengan menggunakan metode *Adaboost*, jika citra tidak terdeteksi sebagai wajah maka lakukan penangkapan citra oleh sensor webcam secara berulang.

Jika bagian dari citra terdeteksi sebagai wajah oleh mesin *face detection* maka lanjutkan dengan dua opsi berikutnya:

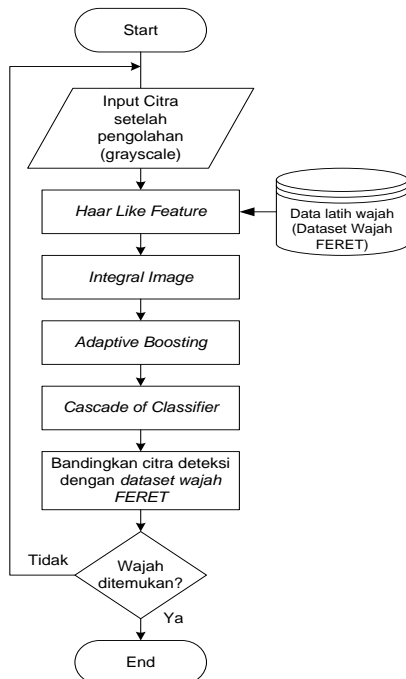
- Lakukan penyimpanan data profile personal ke dalam database sekaligus menyimpan citra wajah ke *file* sebagai bahan perbandingan.

- Jika sebelumnya citra wajah dan data profile telah tersimpan di dalam database maka lakukan identifikasi wajah dengan metode *Eigeface PCA*.

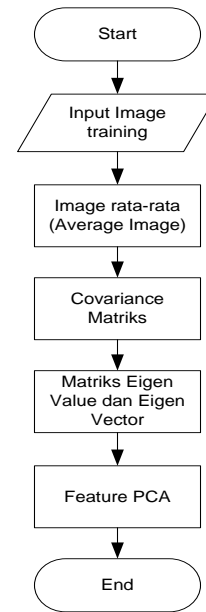


Gambar 5. Alur Kerja Pengenalan Wajah

Gambar 6 menunjukkan diagram alir pendeteksian wajah (*face detection*) dilakukan dengan mengklasifikasikan sebuah citra, setelah sebelumnya sebuah pengklasifikasi dibentuk dari data latih.



Gambar 6. Algoritma Face Detection



Gambar 7. Algoritma Face Recognition

Gambar 7 memberikan algoritma *Face Recognition* [4], yang merupakan langkah pertama untuk mendapatkan satu set S dengan M citra wajah. Sebagai contoh: $M = 25$. Setiap gambar ditransformasi menjadi sebuah vektor berukuran N dan ditempatkan ke dalam set.

$$S = \{\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M\} \quad (4)$$

Setelah mendapatkan set, maka akan diperoleh gambar rata-rata (*mean image*) Ψ .

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \quad (5)$$

Kemudian diperoleh perbedaan Φ antara citra input dan citra rata-rata.

$$\phi_i = \Gamma_i - \Psi \quad (6)$$

Selanjutnya mencari set M vektor ortonormal, u_n , yang paling menggambarkan distribusi data. Vektor K_{th} , u_k , dipilih sedemikian rupa.

$$\lambda_k = \frac{1}{M} \sum_{n=1}^M (u_k^T \phi_n)^2 \quad (7)$$

Adalah maksimal untuk subject

$$u_1^T u_k = \delta_{lk} = \begin{cases} 1 & \text{jika } l = k \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Catatan: u_k and λ_k adalah *eigenvectors* dan *eigenvalues* dari kovarian matriks C .

Selanjutnya diperoleh matriks kovarians C dengan cara berikut

$$C = \frac{1}{M} \sum_{n=1}^M \phi_n \phi_n^T \quad (9)$$

$$C = AA^T$$

$$A = \{\phi_1, \phi_2, \phi_3, \dots, \phi_n\} \quad (10)$$

$$A^T \quad (11)$$

$$L_{mn} = \phi_m^T \phi_n \quad (12)$$

Sehingga diperoleh *eigenvector*, v_l, u_l

$$u_1 = \sum_{k=1}^M v_{lk} \phi_k \quad l = 1, \dots, M \quad (13)$$

Prosedur pengenalan wajah pada *Eigenfaces Principal Component Analysis* adalah sebagai berikut:

- Mengubah wajah baru menjadi komponen eigenface.

Pertama, gambar masukan dibandingkan dengan gambar rata-rata dan perbedaan mereka dikalikan dengan masing-masing vektor *eigen* dari matriks *L*. Setiap nilai akan mewakili bobot dan akan disimpan pada vektor Ω .

$$\omega_k = u_k^T(\Gamma - \Psi) \quad \Omega^T = [\omega_1, \omega_2, \dots, \omega_M] \quad (14)$$

- Menentukan kelas wajah mana yang memberikan gambaran terbaik untuk citra masukan.

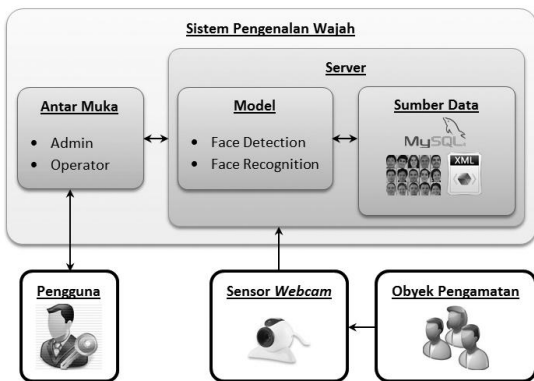
Hal ini dilakukan dengan meminimalkan jarak *Euclidean*.

$$\epsilon_k = \|\Omega - \Omega_k\|^2 \quad (15)$$

Wajah input mempertimbangkan penggolongan kelas. Jika ϵ_k berada di bawah ambang/*threshold* θ_e maka citra merupakan wajah yang dikenali. Jika perbedaan berada di atas *threshold* yang diberikan, tetapi di bawah *threshold* kedua, maka citra merupakan wajah yang tak dikenali. Jika citra masukan bukan bagian dari kedua *threshold* di atas maka citra merupakan bukan wajah.

III. HASIL DAN ANALISIS

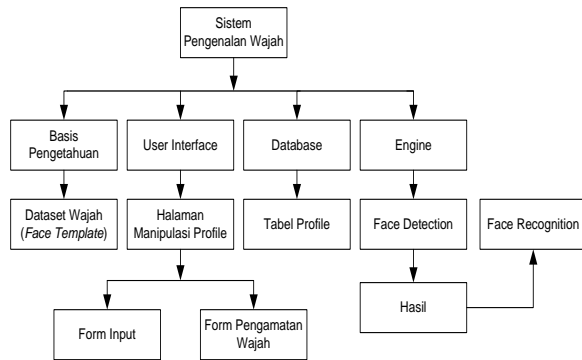
Pada Gambar 8 ditunjukkan arsitektur pengenalan wajah, yang terdiri dari dua bagian utama, yaitu antar muka dan server. Antar muka adalah tampilan awal program berupa otoritas hak akses bagi pengguna. Dalam *server* terdapat dua bagian utama yaitu model dan sumber data, model adalah suatu metode atau algoritma pengenalan wajah, terdiri dari *face detection* dan *face recognition* yang berhubungan dengan sumber data berupa *dataset* wajah FERET (berupa file XML) saat dilakukan pendeteksian wajah, data profile yang disimpan dalam *database MySQL* dan berkas wajah berupa citra wajah berformat JPEG.



Gambar 8. Arsitektur Sistem Pengenalan Wajah

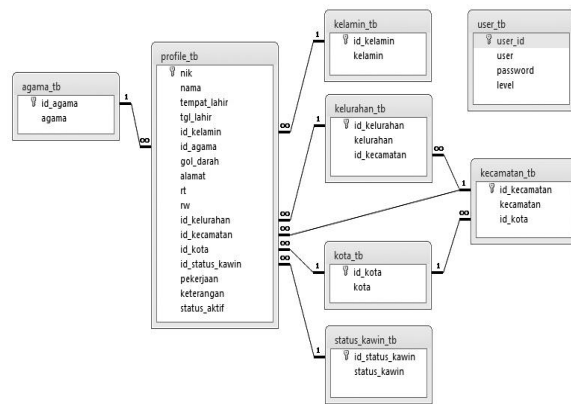
Pada Gambar 9 ditunjukkan sistem pengenalan wajah yang terdiri atas 4 lingkungan, yaitu: basis pengetahuan, *user interface*, *database* dan *engine*. Dalam Basis Pengetahuan terdapat suatu data wajah *template* yang bersumber dari *Database FERET*. *Database* berisi tabel profil. *Engine* terbagi menjadi 2 bagian, yaitu algoritma *Face Detection* dan algoritma *Face Recognition*. Terdapat hubungan antara *face detection* dan *face recognition* yang terintegrasi oleh data *input* dan *output*, dimana kedua proses tersebut harus berurutan. *User Interface* dapat berdialog langsung dengan sistem melalui Halaman Manipulasi Profile melalui *Form*

input dan *Form* Pengamatan Wajah.

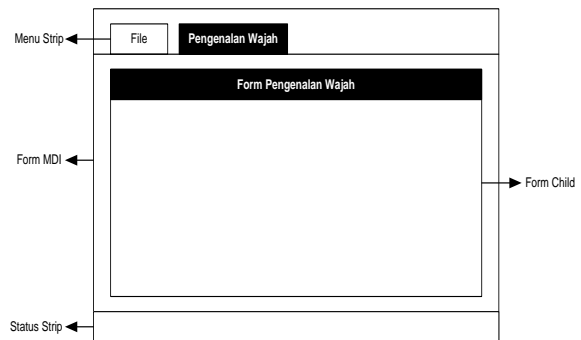


Gambar 9. Sub sistem *model base*

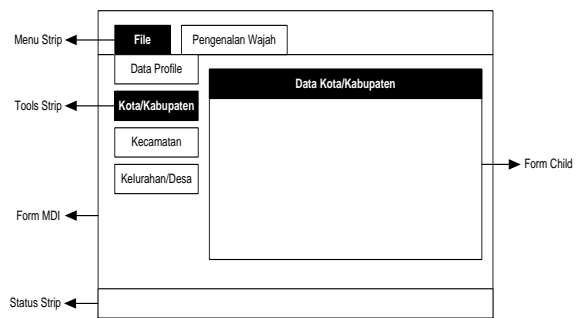
Gambar 10. merupakan desain fisis *database profile* yang sudah ternormalisasi dan berhubungan antara satu tabel dengan yang lain sebagai bentuk integritas data.



Gambar 10. *Physical data model database* pengenalan wajah



Gambar 12. *Form* utama untuk *user Operator*



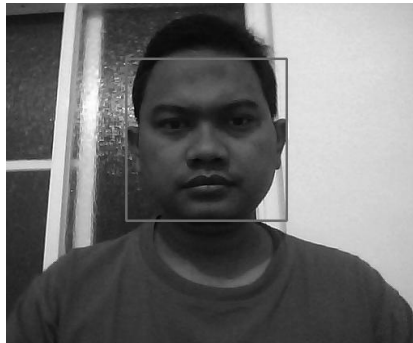
Gambar 11. *Form* utama untuk *user Admin*

Hasil pembangunan aplikasi sistem pengenalan wajah disusun atas: *Face Detection*, *Face Recognition* dan

manipulasi *Database*.

▪ *Face Detection*

Dari citra wajah *grayscale* akan dideteksi dengan *face detection* seperti tampak pada Gambar 13.



Gambar 13. *Face Detection* Adaboost

▪ *Face Recognition*

Langkah selanjutnya adalah melakukan *cropping* citra untuk diidentifikasi menggunakan metode *face recognition Eigenface PCA*. Citra wajah pada Gambar 14. akan menjadi citra data latih yang disimpan ke dalam *database* wajah.



Gambar 14. Citra diidentifikasi dengan *Eigenfaces PCA*

▪ *Antarmuka*

Antarmuka program berfungsi sebagai jembatan penghubung antara *user* dan sistem untuk berinteraksi.

Gambar 15. *Form* data profile

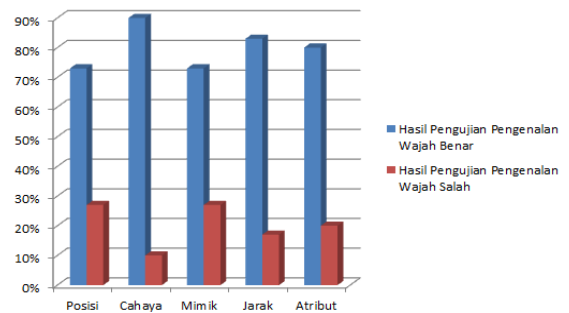
Gambar 15. adalah *form* data profile, di dalamnya terdapat beberapa masukkan berupa *textbox*, *option* dan *combobox* untuk diisi sesuai dengan data wajah yang sedang di-*capture*. Sedangkan pada bagian atas *form* terdapat *toolbar* navigasi yang berfungsi untuk menyimpan, mengubah dan menghapus data profile.

Pengujian dilakukan atas 10 responden yang terdiri dari 5 laki-laki dan 5 perempuan. Tabel I. adalah hasil dari pengujian yang dilakukan.

TABEL I
HASIL PENGUJIAN

Jenis Pengujian	Tingkat Kebenaran	Tingkat Kesalahan
Pengenalan wajah berdasarkan posisi wajah	73%	27%
Pengenalan wajah berdasarkan tingkat pencahayaan	90%	10%
Pengenalan wajah berdasarkan mimik muka	73%	27%
Pengenalan wajah berdasarkan jarak objek dan sensor webcam	83%	17%
Pengenalan wajah berdasarkan atribut berbeda, misalnya: kumis, kacamata, model rambut dan jenggot	80%	20%
Nilai rata-rata validitas	80%	20%

Dari pengujian di atas dapat disimpulkan sesuai dengan Gambar 16. berikut ini:



Gambar 16. Grafik validitas sistem pengenalan wajah

Dari pengujian sampel-sampel object citra wajah didapatkan hasil pemabahasan antara lain:

- Jika intensitas cahaya pada objek citra wajah di atas 1700 lux maka objek citra wajah dapat dikenali dengan baik, namun jika intensitas berkisar antara 0 sampai 1700 lux maka citra wajah tidak dikenali sama sekali atau salah dalam mendeteksi wajah karena kurangnya pencahayaan.
- Bila pencahayaan terhadap objek citra wajah pada kondisi normal (lebih dari 1700 lux) maka penambahan atribut wajah, antara lain: jilbab, cambang, model rambut dan jenggot, objek citra wajah masih dapat dikenali dengan baik. Sedangkan untuk penambahan atribut kacamata dan kumis dapat berpengaruh terhadap penurunan kualitas.
- Pengenalan wajah ditinjau dari posisi rotasi sangat berpengaruh terhadap akurasi pengenalan wajah. Rotasi posisi wajah terdapat 3 jenis kemungkinan, antara lain: menoleh ke kiri atau kanan, menggeleng ke kiri atau kanan, dan mendongak ke atas atau menunduk ke bawah. Secara ideal wajah yang dapat diidentifikasi dengan baik adalah wajah yang menghadap ke sensor kamera secara tegak lurus. Kemiringan yang bisa di tolerir berkisar 10 derajat dari posisi normal. Namun secara normal (tidak cacat fisik) kebanyakan wajah berada posisi tegak lurus dengan sensor kamera web sehingga aplikasi tetap mendeteksi wajah dengan baik.
- Pengenalan wajah juga dipengaruhi oleh mimik wajah, mimik yang dicoba adalah tersenyum (sudut bibir agak naik ke atas), tertawa (tampak

gigi), dan sedih (cemberut dan sudut bibir datar atau turun ke bawah). Dari uji mimik ternyata cukup signifikan mempengaruhi aplikasi dalam mengenali wajah.

- Ditinjau dari jarak antara objek wajah dan sensor kamera web maka aplikasi secara efektif dapat mengenali wajah pada jarak 0,30 meter sampai 2,5 meter jika menggunakan kamera web standar yang terdapat pada laptop (webcam built-in). Pengenalan wajah dapat dilakukan lebih jauh lagi (kurang lebih 6 meter sampai 10 meter) tergantung dari kualitas resolusi citra sensor webcam dan kemampuan *zooming* ketika objek wajah berada sangat jauh.
- Waktu yang dibutuhkan komputer untuk mengidentifikasi wajah (pengenalan sampai identifikasi/pelabelan wajah) secara real time sangat cepat, yaitu kurang dari 1 detik. Sebagai uji coba, aplikasi menggunakan 10 citra wajah sebagai data latih dan setiap objek memiliki 5 posisi citra wajah berbeda dan akurasi kebenaran mengenali wajah mencapai 80%.

IV. KESIMPULAN

Dari hasil perancangan, analisis, implementasi dan pengujian sistem maka dapat disimpulkan hal-hal sebagai berikut :

- Perancangan dan implementasi pengenalan wajah dengan metode *Adaboost* dan *Eigenface PCA* telah berhasil dilakukan dalam penelitian.
 - Rata-rata tingkat keberhasilan pengenalan wajah dengan metode *Adaboost* dan *Eigenfaces PCA* mencapai 80% pada berbagai kondisi berbeda (jarak objek dengan sensor, pencahayaan, posisi, atribut, dan mimik muka).
- Bila dibandingkan dengan metode-metode lainnya maka pengenalan wajah dengan metode *Adaboost* dan *Eigenfaces PCA* memiliki kelebihan pada proses kecepatan mengambil keputusan untuk mengenali wajah di kondisi *real time*.
 - Pengenalan wajah yang diintegrasikan dengan data profil bermanfaat di berbagai sektor, misalnya bidang keamanan, pengawasan, kontrol akses, robotika, intelejen, militer, presensi dan lain-lain.
 - Pengenalan wajah secara *real time* layak dikembangkan lebih lanjut oleh peneliti-peneliti lainnya dengan menggabungkan beberapa metode secara bertingkat untuk memperoleh akurasi yang lebih baik.

DAFTAR PUSTAKA

- [1] Viola, P and Jones, M. "Rapid object detection using a boosted cascade of simple features". In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 511–518, Kauai, HI, 2001.
- [2] Sharif M., "Face Recognition using Gabor Filters", Journal of Applied Computer Science & Mathematics, no. 11, Suceava, May. 2011.
- [3] Hashim F. "A Face Recognition System Using Template Matching And Neural Network Classifier", 1st International Workshop on Artificial Life and Robotics, pp 1-6, 2003
- [4] Freund Y. & Schapire R., "A Short Introduction to Boosting", Journal of Japanese Society for Artificial Intelligence, 14(5)-780, September, 1999.
- [5] Lienhart, R & Maydt, J. "An extended Set of Haar-like Features for Rapid Object Detection", IEEE ICIP 2002, Vol 1, pp. 900-903, Sep.2002.
- [6] Turk, M. & Pentland, A. "Eigenfaces for Recognition". Journal of Cognitive Neuroscience. Vol 3, No. 1. 71-86, 1991.
- [7] Laganier, Robert, "OpenCV 2 Computer Vision Application Programming Cookbook", Packt Publishing, 2011.