

# Pengenalan Wajah Dengan Algoritma *Canonical Correlation Analysis (CCA)*

Ratna Dwi Kartika Rini, Wirawan dan Hendra Kusuma  
Jurusan Teknik Elektro-FTI, Institut Teknologi Sepuluh Nopember (ITS)  
Jl. Arief Rahman Hakim, Surabaya 60111  
E-mail: wirawan@ee.its.ac.id

**Abstrak**— Seiring dengan teknologi pengenalan wajah yang semakin berkembang dan aplikatif, maka diperlukan suatu metode pengenalan wajah alternatif yang mampu menghasilkan akurasi pengenalan yang tinggi. Beberapa penelitian telah dilakukan untuk mendapatkan akurasi pengenalan yang tinggi, tetapi pada penelitian sebelumnya masih terdapat berbagai kendala. Pada tugas akhir ini akan dilakukan proses pengenalan wajah dengan menggunakan algoritma CCA (*Canonical Correlation Analysis*). CCA digunakan untuk mengukur hubungan linear yang saling berkorelasi antara dua variable multidimensi. CCA dapat juga digunakan untuk menemukan dua set dari basis vector, satu untuk  $x$  dan satu untuk  $y$ , sehingga dapat memaksimalkan korelasi antara dua proyeksi dari variabel yang ada di dalam basis vektor secara bersamaan. Sifat lain yang penting dari *canonical correlation* adalah bahwa metode ini tidak terpengaruh oleh *affine transformation* dari variabel, *affine* disini terdiri dari translasi, rotasi, dan dilasi, sehingga metode CCA cocok digunakan untuk pengenalan wajah yang kendala utamanya adalah PIE (*Pose, Illumination, Expression*). Ini adalah hal terpenting yang membedakan antara CCA dan analisa korelasi biasa yang sangat mengandalkan pada basis di dalam variable yang diuraikan.

Pada tugas akhir ini akan menerapkan algoritma CCA untuk pengenalan wajah, uji coba akan dilakukan dengan menggunakan *database* Yale-B dan ORL. Metode yang digunakan adalah menggabungkan fitur wajah yang berasal dari dua buah *feature extractor* yang berbeda, *feature extractor* disini adalah fitur PCA (*Principal Component Analysis*) secara global dan PCA (*Principal Component Analysis*) per bagian atau *parts*.

**Kata kunci** --- CCA, PCA, Feature Extractor

## I. PENDAHULUAN

Teknik pengenalan wajah (*Face Recognition*) merupakan suatu teknik yang masih sangat diminati oleh para periset, hal ini disebabkan karena teknik ini bisa diaplikasikan ke dalam berbagai bidang, misalnya pada aplikasi untuk sistem keamanan dan pengawasan atau pemantauan (*Surveillance*), mesin absensi, kontrol akses, dan interaksi manusia dan mesin (*Human Machine Interaction*)[1]. Dengan menggunakan teknik pengenalan wajah ini, maka manusia dapat mengingat ratusan bahkan ribuan wajah dalam hidupnya, dan mampu mengenali kembali wajah tersebut dalam kondisi dan perspektif yang

berbeda. Ada dua macam skenario pada proses pengenalan wajah, yang pertama adalah *face verification*, dan yang kedua adalah *face identification*. *Face verification* merupakan verifikasi wajah dimana dilakukan perbandingan satu banding satu dengan database yang ada (*one to one*), sedangkan *face identification* merupakan identifikasi wajah dimana dilakukan perbandingan satu wajah akan dibandingkan dengan sekumpulan suatu database yang ada (*one to many*)[2].

Pada pengenalan wajah ini juga ada dua macam image based, yaitu *appearance based* dan *model based*. *Appearance based* terdiri dari beberapa algoritma seperti PCA (*Principal Component Analysis*), LDA (*Linear Discriminant Analysis*), dan yang lainnya, sedangkan *model based* terdiri dari algoritma *Elastic Bunch Graph*, *Active Appearance Model*, dan *Gabor*. *Appearance based* yang banyak digunakan dalam sistem pengenalan wajah dan cukup sukses dalam mengenali wajah. Pendekatan ini pertama kali dikenalkan oleh Turk dan Pentland pada tahun 1991 dan juga dikenal dengan metode *eigenface*. Pada metode ini setiap citra wajah dianggap sebagai satu vektor tunggal dengan cara mereduksi vektor pada citra wajah menjadi satu vektor yang panjang. Dan untuk mendapatkan akurasi pengenalan yang tinggi, maka digunakan CCA (*Canonical Correlation Analysis*). CCA digunakan untuk mengukur hubungan linear yang saling berkorelasi antara dua variable multidimensi. CCA dapat juga digunakan untuk menemukan dua set dari basis vector, satu untuk  $x$  dan satu untuk  $y$ , sehingga dapat memaksimalkan korelasi antara dua proyeksi dari variabel yang ada di dalam basis vektor secara bersamaan. Properti lain yang penting dari *canonical correlation* adalah bahwa metode ini tidak terpengaruh oleh *affine transformation* dari variabel, *affine* disini terdiri dari translasi, rotasi, dan dilasi, sehingga metode CCA cocok digunakan untuk pengenalan wajah yang kendala utamanya adalah PIE (*Pose, Illumination, Expression*)[3]. Ini adalah hal terpenting yang membedakan antara CCA dan analisa korelasi biasa yang sangat mengandalkan pada basis di dalam variable yang diuraikan.

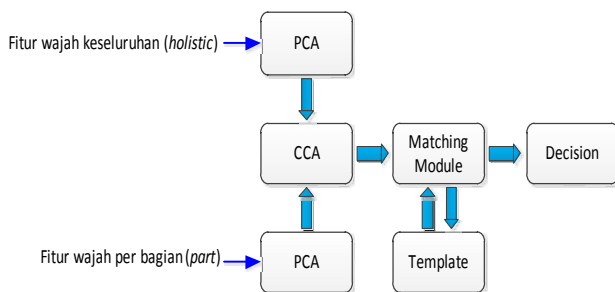
## II. METODOLOGI

### A. Perencanaan Algoritma

Pada tugas akhir ini akan digunakan algoritma PCA dan CCA untuk proses pengenalan wajah, kemudian untuk ekstraksi fiturnya digunakan PCA secara keseluruhan dan

PCA per bagian atau part (mata, hidung, dan mulut). Untuk mengerjakan tugas akhir ini langkah awal yang harus dilakukan adalah pembuatan basis data citra wajah untuk proses training. Basis data menggunakan basis data ORL dengan ukuran 64x64, kemudian dari basis data tersebut dilakukan *cropping* untuk bagian mata, mulut, dan hidung. Proses *training* ini dilakukan untuk mengenalkan citra wajah pada sistem, sehingga pada proses selanjutnya sistem bisa mengenali wajah. Proses *training* disini mencakup proses pembentukan *eigenface* dan *eigenfeature*, kemudian melakukan proses PCA pada keduanya yang nantinya akan diproyeksikan dengan menggunakan CCA. Kemudian untuk proses pengenalannya, yaitu untuk mencari tingkat kemiripan antara citra tes dengan citra dalam basis data, dilakukan dengan metode pengukuran jarak *Euclidean*.

Blok diagram sistem secara umum di dalam tugas akhir ini dapat dilihat pada blok diagram di bawah ini :



Gambar 1. Blok diagram sistem secara keseluruhan

**B. Perancangan Data**

Basis data yang digunakan dalam tugas akhir ini adalah basis data ORL yang berukuran 64x64. Basis data ini terdapat 40 individu dimana masing-masing individu memiliki 10 macam variasi *headpose*. Kemudian basis data dibagi menjadi dua bagian, masing-masing diambil sebanyak 200 untuk *training*, dan 200 untuk *testing*.

Kemudian basis data ORL dilakukan *cropping* untuk bagian mata, hidung, dan mulut, dengan ukuran masing-masing bagian adalah mata 64x24 piksel, hidung 28x20 piksel, dan mulut 37x17 piksel. Untuk basis data mata, hidung, dan mulut juga dilakukan pembagian sebanyak 200 untuk citra *training*, dan 200 untuk citra *testing*.



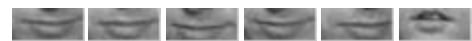
Gambar 2. Contoh citra wajah basis data ORL



Gambar 3. Contoh citra mata basis data ORL



Gambar 4. Contoh citra hidung basis data ORL



Gambar 5. Contoh citra mulut basis data ORL

**C. Pemodelan Sistem**

Pemodelan sistem pada tugas akhir ini meliputi bagaimana proses dan tahap-tahap yang dilakukan dalam pengenalan wajah.

Tahap awal yang dilakukan pada proses pengenalan wajah ini adalah *cropping* dan *grayscale*, *cropping* dilakukan untuk memotong bagian-bagian mata, hidung, dan mulut dari suatu wajah, yang nantinya akan digunakan untuk basis data PCA *part* atau PCA per bagian. Kemudian citra yang sudah di *cropping* akan dilakukan proses normalisasi *grayscale*, tujuan dari proses *grayscale* ini adalah mendapatkan komputasi yang sekecil mungkin dengan mengubah nilai 24 bit RGB menjadi 8 bit *grayscale*. Selanjutnya adalah mendapatkan *eigenface* dan *eigenfeature* dari basis data yang sudah dirancang. *Eigenface* didapatkan dari basis data wajah keseluruhan, sedangkan *eigenfeature* merupakan gabungan dari basis data mata, hidung, dan mulut. Prosedur untuk mendapatkan *eigenface* adalah sebagai berikut [4] :

1. Terdapat citra basis data wajah dengan ukuran  $N_x \times N_y$  ( $64 \times 64$ ), dimana  $N_x$  menyatakan banyak region dan  $N_y$  menyatakan panjang *histogram*. Kemudian matriks citra dengan ukuran  $(N_x \times N_y)$  dikonversikan ke suatu vector citra  $T$  dengan ukuran  $(N \times 1)$  dimana nilai  $N$  adalah (banyaknya region x panjang *histogram*) yaitu sebesar 4096. Dimana  $T$  berukuran  $(N \times P)$  dan  $P$  merupakan jumlah citra *training* dalam basis data yaitu sebanyak 200 citra.

$$T = \begin{bmatrix} T_{1,1} & T_{1,2} & \dots & T_{1,200} \\ T_{2,1} & T_{2,2} & \dots & T_{2,200} \\ \dots & \dots & \dots & \dots \\ T_{4096,1} & T_{4096,2} & \dots & T_{4096,200} \end{bmatrix}$$

2. Kemudian mencari rata-rata *mean* semua piksel dari citra-citra *training*. Dapat dinyatakan dengan persamaan berikut :

$$\bar{T} = \frac{1}{200} \sum_{i=1}^{200} T_i$$

3. Sehingga didapatkan matriks rata-rata total PCA dengan dimensi  $(1 \times N)$  yaitu :

$$\bar{T} = \begin{bmatrix} \bar{T}_{1,1} \\ \dots \\ \bar{T}_{1,4096} \end{bmatrix}$$

- Setelah mendapatkan *mean* dari setiap citra *training*, langkah berikutnya adalah mencari *zero mean* yang merupakan pengurangan citra *training* dan *mean* yang dinyatakan dengan persamaan berikut :

$$\text{Mean subtracted image } \Phi = T - \bar{T}$$

Vector *zero mean* ini mempunyai ukuran (P x N) piksel. Setelah mendapat vektor *zero mean*, tiap vektor yang diperoleh dimasukkan ke dalam suatu matriks A, dimana matriks A ini pada tiap kolomnya berisi semua vektor citra *training* yang sudah dikurangi dengan *mean*. Matriks A ini disebut dengan *Difference Matrix*.

$$A = \begin{bmatrix} \Phi_{1,1} & \Phi_{1,2} & \dots & \dots & \Phi_{1,200} \\ \Phi_{2,1} & \Phi_{2,2} & \dots & \dots & \Phi_{2,200} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \Phi_{4096,1} & \Phi_{4096,2} & \dots & \dots & \Phi_{4096,200} \end{bmatrix}$$

- Setelah didapatkan *different matrix* selanjutnya mencari matriks kovarians, matriks kovarians merupakan perkalian matriks A dengan transposenya sehingga akan dihasilkan matriks dengan ukuran (PxP), sesuai persamaan berikut :

$$L = \frac{1}{200} \sum_{i=1}^{200} \Phi_i \Phi_i^T$$

- Selanjutnya dilakukan dekomposisi *eigen* sehingga diperoleh matriks *eigenvalue* ( $\lambda$ ) dan *eigenvector* ( $V$ ) yang masing-masing berdimensi (PxP) dimana P merupakan jumlah dari citra *training*.

$$V = \begin{bmatrix} V_{1,1} & V_{1,2} & \dots & V_{1,200} \\ V_{2,1} & V_{2,2} & \dots & V_{2,200} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ V_{4096,1} & V_{4096,2} & \dots & V_{4096,200} \end{bmatrix} \quad \lambda = \begin{bmatrix} \lambda_{1,1} & 0 & 0 & 0 & 0 \\ 0 & \lambda_{2,2} & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \lambda_{200,200} \end{bmatrix}$$

(a) (b)

Sedangkan prosedur untuk mendapatkan *eigenfeature* adalah sebagai berikut :

- Terdapat tiga citra basis data yaitu mata, hidung, dan mulut dengan ukuran masing-masing untuk mata  $N_x \times N_y$  (64x24), hidung  $N_x \times N_y$  (28x20), dan mulut  $N_x \times N_y$  (37x17), dimana  $N_x$  menyatakan banyak region dan  $N_y$  menyatakan panjang *histogram*, kemudian ketiga citra tersebut digabung menjadi satu sehingga diperoleh nilai  $N_x \times N_y$  adalah 2725. Kemudian matriks citra dengan ukuran ( $N_x \times N_y$ ) dikonversikan ke suatu vector citra T dengan ukuran (N x 1) dimana nilai N adalah (banyaknya region x panjang *histogram*) yaitu sebesar 2725. Dimana T berukuran (N x P) dan P merupakan

jumlah citra *training* dalam basis data yaitu sebanyak 200 citra.

$$T = \begin{bmatrix} T_{1,1} & T_{1,2} & \dots & T_{1,200} \\ T_{2,1} & T_{2,2} & \dots & T_{2,200} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ T_{2725,1} & T_{2725,2} & \dots & T_{2725,200} \end{bmatrix}$$

- Kemudian mencari rata-rata *mean* semua piksel dari citra-citra *training*. Dapat dinyatakan dengan persamaan berikut :

$$\bar{T} = \frac{1}{200} \sum_{i=1}^{200} T_i$$

- Sehingga didapatkan matriks rata-rata total PCA dengan dimensi (1xN) yaitu

$$\bar{T} = \begin{bmatrix} \bar{T}_{1,1} \\ \vdots \\ \bar{T}_{1,2725} \end{bmatrix}$$

- Setelah mendapatkan *mean* dari setiap citra *training*, langkah berikutnya adalah mencari *zero mean* yang merupakan pengurangan citra *training* dan *mean* yang dinyatakan dengan persamaan berikut :

$$\text{Mean subtracted image} = T - \bar{T}$$

Vector *zero mean* ini mempunyai ukuran (P x N) piksel. Setelah mendapat vektor *zero mean*, tiap vektor yang diperoleh dimasukkan ke dalam suatu matriks A, dimana matriks A ini pada tiap kolomnya berisi semua vektor citra *training* yang sudah dikurangi dengan *mean*. Matriks A ini disebut dengan *Difference Matrix*.

$$A = \begin{bmatrix} 1,1 & 1,2 & \dots & \dots & 1,200 \\ 2,1 & 2,2 & \dots & \dots & 2,200 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 2725,1 & \Phi_{2725,2} & \dots & \dots & 2725,200 \end{bmatrix}$$

- Setelah didapatkan *different matrix* selanjutnya mencari matriks kovarians, matriks kovarians merupakan perkalian matriks A dengan transposenya sehingga akan dihasilkan matriks dengan ukuran (PxP), sesuai persamaan berikut :

$$L = \frac{1}{200} \sum_{i=1}^{200} \Phi_i \Phi_i^T$$

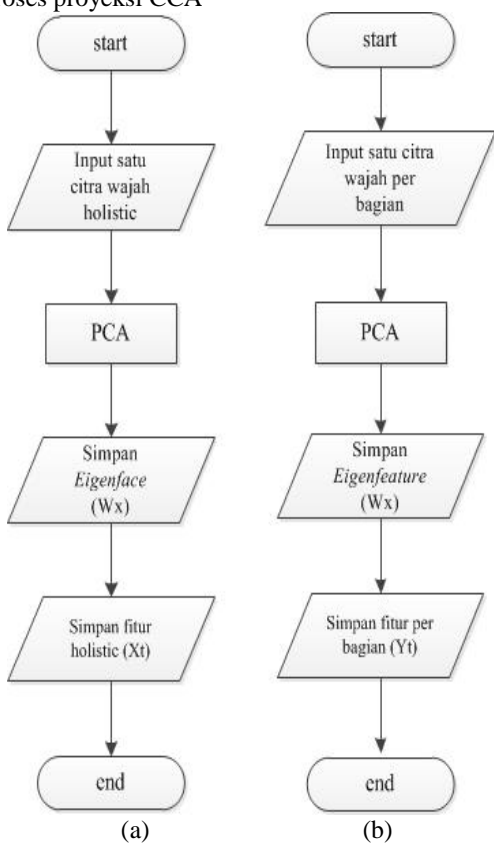
6. Selanjutnya dilakukan dekomposisi *eigen* sehingga diperoleh matriks *eigenvalue* ( $\lambda$ ) dan *eigenvector* ( $V$ ) yang masing-masing berdimensi ( $P \times P$ ) dimana  $P$  merupakan jumlah dari citra *training*.

$$V = \begin{bmatrix} V_{1,1} & V_{1,2} & \dots & V_{1,200} \\ V_{2,1} & V_{2,2} & \dots & V_{2,200} \\ \dots & \dots & \dots & \dots \\ V_{2725,1} & V_{2725,2} & \dots & V_{2725,200} \end{bmatrix} \quad \lambda = \begin{bmatrix} \lambda_{1,1} & 0 & 0 & 0 & 0 \\ 0 & \lambda_{2,2} & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \lambda_{200,200} \end{bmatrix}$$

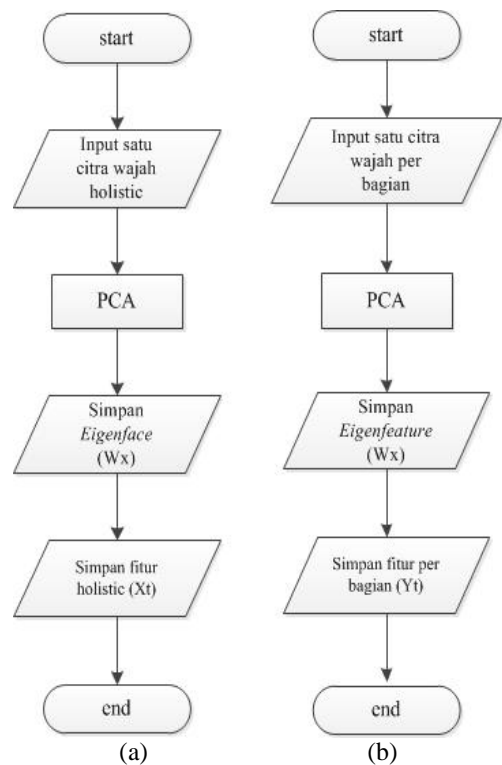
(a) (b)

**D. Proyeksi PCA (Principal Component Analysis)**

pada proyeksi PCA ini akan dilakukan untuk proses pembentukan  $Z_{training}$  dan  $Z_{testing}$  yang nantinya akan dibandingkan untuk mendapatkan nilai yang *match*. Gambar flowchart berikut merupakan proses pembentukan  $Z_{training}$  dan  $Z_{testing}$ . Gambar (a) merupakan *flowchart* dari proses pembentukan  $A_{model.train}$  dan  $B_{model.train}$ , kemudian untuk gambar (B) merupakan *flowchart* dari proses pembentukan AA dan BB. Output dari masing-masing proses tersebut nantinya akan menjadi input dari proses proyeksi CCA



**Gambar 6.** Flowchart Proyeksi PCA Basis Data Training



**Gambar 7.** Flowchart Proyeksi PCA Basis Data Testing

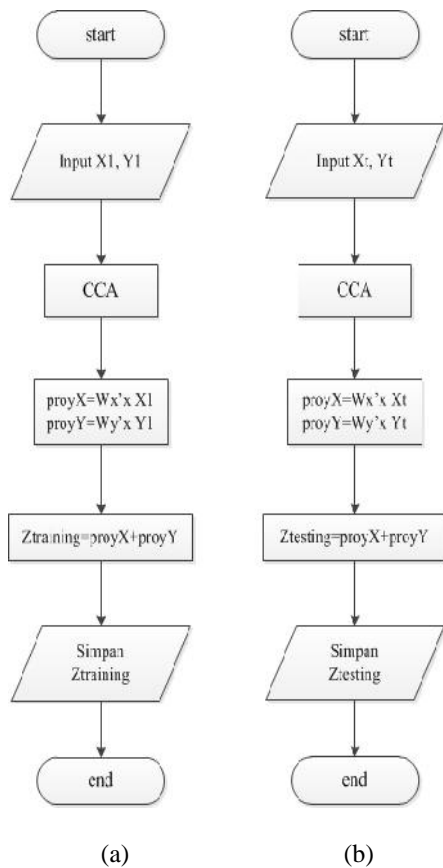
**E. Proyeksi CCA (Canonical Correlation Analysis)**

Pada tugas akhir ini CCA digunakan sebagai tujuan dari pengenalan wajah. Metode yang diterapkan adalah menggabungkan output dari dua *feature extractor*. Kedua output tersebut adalah berasal dari *feature extractor* PCA (*holistic*) atau keseluruhan dan *feature extractor* PCA (*part*) atau per bagian. Tujuannya adalah untuk mendapatkan informasi output maksimum dari dua *feature extractor* tersebut[5]. Input pada proyeksi CCA ini berupa model.train dari *eigenface* dan *eigenfeature* pada proses proyeksi PCA. Tahap dari proyeksi CCA dapat dilihat pada *flowchart* berikut. Gambar (a) merupakan *flowchart* dari proses pembentukan  $Z_{training}$ , dan gambar (b) merupakan *flowchart* dari proses pembentukan  $Z_{testing}$ . Proses pembentukan  $Z_{training}$  dan  $Z_{testing}$  sendiri ada dua cara, yaitu yang pertama menggunakan (*Feature Fusion Method-1*), maka proyeksiX dan proyeksiY akan disusun ke bawah sesuai dengan persamaan (1).

$$z_1 = \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} w_x^T x \\ w_y^T y \end{pmatrix} = \begin{pmatrix} w_x & 0 \\ 0 & w_y \end{pmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix} \quad (1)$$

Kemudian yang kedua adalah menggunakan metode FFM-2 (*Feature Fusion Method-2*), maka proyeksiX dan proyeksiY akan dijumlah sesuai dengan persamaan (2).

$$z_2 = \begin{pmatrix} x' \\ y' \end{pmatrix} = (W_x^T x + W_y^T y) = (W_x^T \ W_y^T) \begin{pmatrix} x \\ y \end{pmatrix} \quad (2)$$



Gambar 8. Flowchart Proyeksi CCA

III. ANALISA DATA

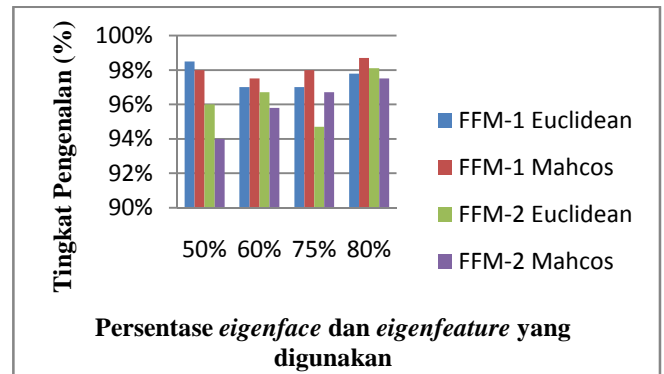
A. Hasil Pengujian

Pada tugas akhir ini, database yang digunakan untuk pengujian bervariasi dari 40 kelas, yaitu 5 database training dan 5 database test, 7 database training dan 3 database test, serta 8 database training dan 2 database test.

Pengujian dilakukan dengan dua cara yaitu menggunakan FFM-1 (Feature Fusion Method-1) dengan metode pengukuran jarak euclidean dan mahcos, serta menggunakan FFM-2 (Feature Fusion Method-2) dengan metode pengukuran jarak euclidean dan mahcos.

Tabel 1 Pengujian dengan Algoritma CCA Menggunakan Database ORL (5 citra training, 5 citra testing)

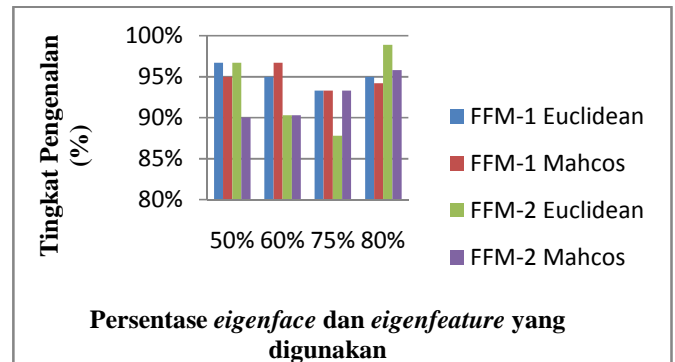
Pemilihan jumlah eigenface dan eigenfeature	FFM-1		FFM-2	
	Euclidean	Mahcos	Euclidean	Mahcos
50%	98.5%	98%	96%	94%
60%	97%	97.5%	96.7%	95.8%
75%	97%	98%	94.7%	96.7%
80%	97.8%	98.7%	98.1%	97.5%
Rata - Rata	97.6%	98.1%	96.4%	96%



Gambar 8. Grafik Rata-Rata Tingkat Pengenalan Wajah dengan Menggunakan Algoritma CCA

Tabel 2 Pengujian dengan Algoritma CCA Menggunakan Database ORL (7 citra training, 3 citra testing)

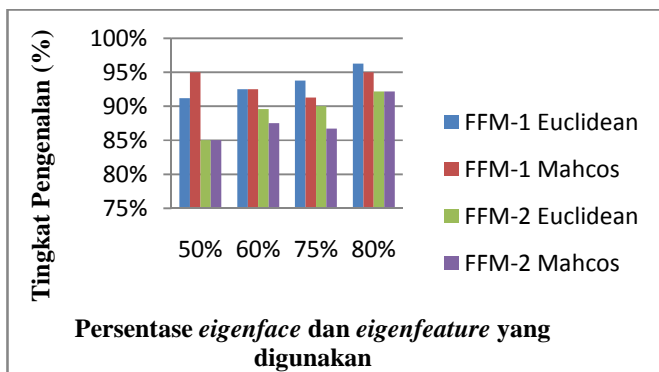
Pemilihan jumlah eigenface dan eigenfeature	FFM-1		FFM-2	
	Euclidean	Mahcos	Euclidean	Mahcos
50%	96.7%	95%	96.7%	90%
60%	95%	96.7%	90.3%	90.3%
75%	93.3%	93.3%	87.8%	93.3%
80%	95%	94.2%	98.9%	95.8%
Rata - Rata	95%	94.8%	93.4%	92.3%



Gambar 9. Grafik Rata-Rata Tingkat Pengenalan Wajah dengan Menggunakan Algoritma CCA

Tabel 3 Pengujian dengan Algoritma CCA Menggunakan Database ORL (8 citra training, 2 citra testing)

Pemilihan jumlah eigenface dan eigenfeature	FFM-1		FFM-2	
	Euclidean	Mahcos	Euclidean	Mahcos
50%	91.2%	95%	85%	85%
60%	92.5%	92.5%	89.6%	87.5%
75%	93.8%	91.3%	90%	86.7%
80%	96.3%	95%	92.2%	92.2%
Rata - Rata	93.5%	93.5%	89.2%	87.9%



Gambar 10. Grafik Rata-Rata Tingkat Pengenalan Wajah dengan Menggunakan Algoritma CCA

#### IV KESIMPULAN

Berdasarkan serangkaian pembahasan dan analisa yang telah dilakukan pada tugas akhir yang berjudul Pengenalan Wajah Dengan Menggunakan Algoritma CCA (*Canonical Correlation Analysis*), maka didapatkan kesimpulan sebagai berikut :

1. Pengenalan wajah dengan menggunakan algoritma CCA memiliki tingkat akurasi yang lebih tinggi dibandingkan dengan algoritma pada penelitian sebelumnya (PCA).
2. Pemilihan *eigenface* dan *eigenfeature* sebesar 80% memiliki rata-rata tingkat pengenalan lebih tinggi dibandingkan dengan pengambilan *eigenface* dan *eigenfeature* sebesar 50%, 60%, dan 75%.
3. Proses pembentukan Ztraining dan Ztesting dengan menggunakan FFM-1 (*Feature Fusion Method-1*) memiliki tingkat pengenalan yang lebih tinggi dibandingkan dengan menggunakan FFM-2 (*Feature Fusion Method-2*).
4. Rata-rata pengenalan wajah dengan menggunakan metode penghitungan jarak *euclidean* akan menghasilkan persentase yang lebih tinggi dibandingkan dengan menggunakan metode penghitungan jarak *mahalanobis cosinus*.

#### DAFTAR PUSTAKA

- [1] Quan Sen Hun, Sheng Gen Zeng, Yan Liu, Pheng Ann Heng, De Shen Xia. A new method of feature fusion and its application in image recognition. *Pattern Recognition* 38 (2005) 2437 – 2448.
- [2] Xiaoyuan Jing, Sheng Li, Chao Lan, David Zhang, Jingyu Yang, Qian Liu. Color image canonical correlation analysis for face feature extraction and recognition, Elsevier, *Signal Processing* (2011).
- [3] Gianluca Donato, Marian Stewart Bartlett, Joseph C. Harger, Paul Ekman, Terrence J. Sejnowski. Classifying facial actions. *Pattern analysis and machine intelligence Vol. 21, No. 10* (1999).
- [4] David Zhang, Xiaoyuan Jing, Jian Yang, "Biometric Image Discrimination Technologies", (2006).
- [5] Q. Sun, Z. Liu, P. Heng and D. Xia, "A theorem on the generalized canonical projective vectors", in *Pattern Recognition*, vol. 38, (2005), 449 – 452.