

## Analisa Perbandingan Pengenalan Wajah Dengan Principal Component Analysis Dan Bidirectional Associative Memory

**Etini Harefa<sup>1\*</sup>, Anda Yanny<sup>2</sup>**
<sup>1</sup>Teknik Informatika, Univeristas Budi Darma, Indonesia<sup>2</sup>Univeristas Budi Darma, Indonesiae-mail: <sup>1\*</sup>[etiniharefa424@gmail.com](mailto:etiniharefa424@gmail.com)**Keywords:**

*Comparison,  
Face recognition,  
Principal Component Analysis,  
Bidirectional Associative  
Memory,  
Image processing.*

**ABSTRACT (10 PT)**

This research aims to compare two methods in face recognition systems, namely Principal Component Analysis (PCA) and Bidirectional Associative Memory (BAM), to determine the most accurate method in recognizing face images. In this experiment, both methods are tested using the same face image samples, so that the results can be compared objectively. PCA is used to reduce the data dimension and extract facial features through an eigenface approach, while BAM applies a heteroassociative neural network to recognize input patterns based on bidirectional synaptic weights. The results show that the PCA method produces a higher level of accuracy than BAM. This is evidenced by the smallest Euclidean Distance value of 455, which indicates a match between the test image and the training image. In contrast, the BAM method did not successfully identify the test image because the final result did not match the target, which shows its limitation in recognizing facial image patterns. Thus, PCA is more recommended to be applied in image-based face recognition systems.

**Kata Kunci**

*Perbandingan,  
Pengenalan wajah,  
Principal Component Analysis,  
Bidirectional Associative  
Memory,  
Pengolahan citra.*

**ABSTRAK**

Penelitian ini bertujuan untuk membandingkan dua metode dalam sistem pengenalan wajah, yaitu Principal Component Analysis (PCA) dan Bidirectional Associative Memory (BAM), guna menentukan metode yang paling akurat dalam mengenali citra wajah. Dalam eksperimen ini, kedua metode diuji menggunakan sampel citra wajah yang sama, sehingga hasilnya dapat dibandingkan secara objektif. PCA digunakan untuk mereduksi dimensi data dan mengekstrak fitur wajah melalui pendekatan eigenface, sedangkan BAM menerapkan jaringan saraf heteroasosiatif untuk mengenali pola input berdasarkan bobot sinaptik dua arah. Hasil penelitian menunjukkan bahwa metode PCA menghasilkan tingkat akurasi yang lebih tinggi dibandingkan BAM. Hal ini dibuktikan dengan nilai Euclidean Distance terkecil sebesar 455, yang menunjukkan kesesuaian antara citra uji dan citra latih. Sebaliknya, metode BAM tidak berhasil mengidentifikasi citra uji karena hasil akhir tidak sesuai dengan target, yang menunjukkan keterbatasannya dalam mengenali pola citra wajah. Dengan demikian, PCA lebih direkomendasikan untuk diterapkan dalam sistem pengenalan wajah berbasis citra.

**Korespondensi Penulis \*):**

Etini Harefa  
Universitas Budi Darma  
Jalan Sisingamangaraja No. 338 Kota Medan

*Diajukan: 02-08-2024 | Diterima: 02-08-2024 | Diterbitkan: 30-08-2024*
**1. PENDAHULUAN**

Identifikasi merupakan hal yang sangat penting dalam dunia digital, Salah satu sistem pengenalan yang banyak dikembangkan pada saat ini adalah sistem informasi biologis yaitu retina, sidik jari, wajah, retina dan lain-lain. Salah satu informasi pada biologis yang sekarang banyak dikembangkan dan memiliki tingkat keakuratan tinggi yaitu citra wajah. Citra wajah seseorang memiliki ciri yang unik masing-masing yang dapat diidentifikasi. Identifikasi tersebut dapat digunakan untuk sistem pencarian seseorang di dalam sebuah gambar yang berisi wajah. Eigenface merupakan salah satu metode pengenalan wajah yang berdasarkan pada *Principal Component Analysis (PCA)* dan *Bidirectional Associative Memory (BAM)* yang mudah di implementasi.

Sistem face recognition merupakan sebuah solusi dalam melakukan identifikasi wajah seseorang. Banyak metode yang telah diterapkan untuk pengenalan wajah manusia dengan kelebihan dan kekurangan tertentu. Hal ini disebabkan karena wajah manusia mempresentasikan sesuatu yang kompleks, sehingga untuk mengembangkan model komputasi yang ideal untuk pengenalan wajah manusia adalah suatu hal yang sulit, tidak dibantahkan masih banyak pembuat sistem belum mengetahui metode yang lebih akurat untuk diterapkan pada suatu sistem pengenalan wajah.

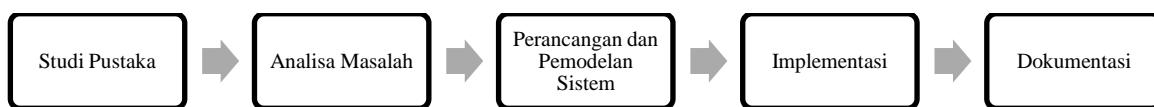
Permasalahan pada penelitian ini adalah mencari pengenalan wajah yang paling akurat untuk diterapkan dengan membandingkan dua hasil dari metode yang berbeda, dimana sampel input di gunakan citra wajah yang sama, sehingga metode Principal Component Analysis Dan Bidirectional Associative Memory akan didapatkan hasil perbandingan pengenalan citra wajah

Perbandingan antara metode *Principal Component Analysis* dengan *Bidirectional Associative Memory* menjadikan solusi dalam menggunakan metode yang paling akurat diterapkan pada sistem pengenalan (identifikasi) sesorang, baik dalam pencarian sesorang maupun pada keamanan data.

Penelitian Terdahulu yang dilakukan oleh ‘Rina Firliana dengan judul Implementasi *Principal Component Analysis (PCA)* Untuk Pengenalan Wajah Manusia dengan menyimpulkan Tingkat akurasi dipengaruhi nilai threshold ukuran pixel, jumlah data set dan noise” [1]. Penelitian yang dilakukan oleh Rizqi Elmuna Hidayah, S.Si, M.Kom dengan judul Implementasi Metode Principal Component Analysis Pada Pengenalan Wajah Berbasis Eigenface menyimpulkan dengan menggunakan citra taining sebanyak 25 citra yang sama terdiri dari 5 orang yang berbeda, dan tiap orang mewakili 5 citra yang berbeda dengan melakukan pengujian presentasi kesuksesan mencapai 81.82% dengan 1 orang mengalami kegagalan indentifikasi [2]. Penelitian yang dilakukan oleh Zainal Arifin dengan judul Jaringan Syaraf Tiruan Bidirectional Associative Memory (BAM) Sebagai Identifikasi Pola Sidik jari Manusia menyimpulkan Bidirectional Associative Memory (BAM) tidak dapat mengenali secara tepat beberapa pola sidik jari manusia yang dikenai noise [3]. sedangkan penelitian yang dilakukan oleh Aperius Giawa dengan judul Implementasi Metode *bidirectional associative memory* pada absensi berbasis identifikasi wajah menyimpulkan sistem yang dengan metode BAM dapat melakukan pengenalan terhadap citra wajah input [4].

## 2. METODE PENELITIAN

## **2.1 Kerangka Kerja Penelitian**



**Gambar 1.** Pelaksanaan Penelitian

1. Studi Pustaka, merupakan salah satu metode pengumpulan data dalam kualitatif yang tidak memiliki pengetahuan mendalam akan literatur dari kemampuan yang digunakan tertentu dari pihak peneliti.
  2. Analisa Masalah, data yang diperoleh dari hasil studi pustaka dianalisa untuk merencanakan pembuatan sistem yang sesuai dengan kebutuhan.
  3. Perancangan dan Pemodelan, setelah analisa masalah, dilakukan pemodelan dan interface, dari pemodelan sistem yang dibuat sesuai dengan tujuan dan batasan masalah yang ada.
  4. Implementasi, sistem yang dirancang diimplementasikan dengan bahasa pemrograman visual basic 2008 dengan menerapkan metode yang diteliti.
  5. Dokumentasi, dari hasil implementasi dilakukan penyusunan laporan dalam format penulisan penelitian.

## 2.2 Principal Component Analysis (PCA)

*Principal Component Analysis* memiliki tahapan yaitu mereduksi atau mengurangi informasi data yang besar dari sebuah citra wajah tanpa menghilangkan informasi yang ada pada sebuah citra wajah. Algoritma PCA melakukan penguraian citra wajah kedalam kumpulan fitur karakteristik yang disebut sebagai “Eigenface”. Hal tersebut yang kemudian dikatakan sebagai Principal Component dalam sebuah data trainingset, fitur utama dari sebuah algoritma PCA adalah merekonstruksi dari beberapa citra asli dari trainingset dengan mengkombinasikan eigenface.

Penyusunan algoritma PCA akan dijelaskan sebagai berikut:

- Pencarian nilai *eigen* terlebih dahulu perlu melakukan perhitungan nilai rata-rata dari sebuah *citra* dengan persamaan sebagai berikut:

### Keterangan :

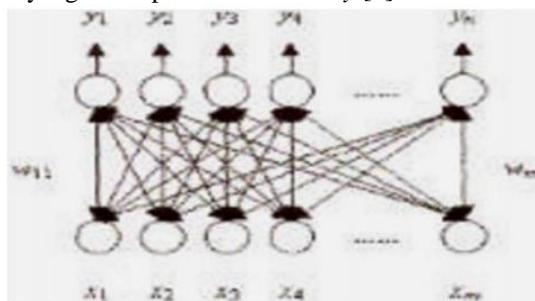
Keterangan :  
 $X_i$ =data ke i dari variabel X

n=jumlah data

Perhitungan nilai rata-rata guna mengurangi dimensi yang akan di hitung pada proses selanjutnya.

### 2.3 Bidirectional Associative Memory

*Bi-directional associative memory* merupakan model *heteroassociative* yang terdiri dari dua lapisan memori (A dan B) yang dapat diakses, kedua pola vector dan menggunakan aliran perambatan informasi arah umpan maju dan arah umpan balik untuk menuju pola yang tersimpan dalam *memory* [4]



**Gambar 2.** Arsitektur Bidirectional Associative Memory (BAM)

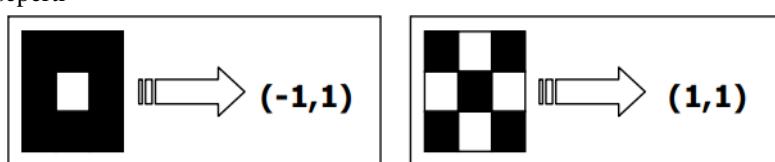
1. Jika jaringan mengirim sinyal dari X-layer ke Y-layer maka bobot jaringan dituliskan sebagai vektor berbentuk W.
  2. Jika jaringan mengirim sinyal dari Y-layer ke X-layer maka bobot jaringan dituliskan sebagai vektor bentuk WT.

Algoritma jaringan syaraf tiruan *Bidirectional Associative Memory* (BAM) adalah sebagai berikut:



### Contoh Kasus :

Misalkan kita memiliki 2 matriks  $3 \times 3$  yang mewakili bilangan huruf O dan X. Tiap matriks berhubungan dengan kode bipolar seperti



**Gambar 3** Matriks 3x3 untuk BAM

1. Atur pola *input* dan pola *output*

Kita bisa membawa tiap-tiap matriks menjadi satu bentuk vektor dengan elemen-elemennya berupa bilangan biner -1 atau 1, sebagai berikut:

Pola 1 : Pola O

$$x_1: 1 \ 1 \ 1 \ 1 \ -1 \ 1 \ 1 \ 1 \quad y_1 \rightarrow -1 \ 1$$

Pila 2 : Pola X

$$x_2: 1 \ -1 \ 1 \ -1 \ 1 \ -1 \ 1 \ -1 \quad y_2 \rightarrow -1 \ 1$$

2. Hitung matriks bobot dari pola X ke pola Y

Matriks bobot untuk pola 1 :

$$W_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} * [-1 \ 1] = \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ 1 & 1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \end{bmatrix}$$

Matriks bobot untuk pola 2 :

$$W_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ 1 \end{bmatrix} * [1 \ 1] = \begin{bmatrix} 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \end{bmatrix}$$

Matriks bobot untuk  $W_{\text{Total}}$  :

$$w_{total} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \\ -1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \\ -1 & -1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 2 & 0 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 0 & 2 \end{bmatrix}$$

Matriks bobot tersebut menghubungkan antara neuron-neuron di lapisan input ke neuron-neuron yang ada di lapisan output. Sedangkan matriks bobot yang menghubungkan antara neuron-neuron di lapisan output ke neuron-neuron yang ada di lapisan input adalah  $W_t$

3. Recall pola input

Tentukan nilai  $y_i$  dengan membandingkan nilai  $y_{in_i}$  dengan *threshold*

Jika  $y_{in_i} \geq \theta$  maka  $y_i = 1$

Jika  $y_{in_i} < \theta$  maka  $y_i = -1$

$$(w_{total})^t = \begin{bmatrix} 0 & -2 & 0 & -2 & 2 & -2 & 0 & -2 & 0 \end{bmatrix}$$

Kita bisa menguji bobot tersebut. Misalkan kita cobakan vektor input pertama yang mewakili Huruf O, maka output yang diperoleh adalah:

$$(w_{total})^t = [1 \ 1 \ 1 \ 1 \ -1 \ 1 \ 1 \ 1] * \begin{bmatrix} 0 & 2 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 2 & 0 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 0 & 2 \end{bmatrix} = [-10 \ 8]$$

Karena  $(y_{in_1}(1) = -10 < 0)$ , maka  $y_1(1) = -1$  dan  $(y_{in_1}(2) = 8 > 0)$ , maka  $y_1(2) = 1$ , maka nilai  $y_1 = [-1 \ 1]$ , sama dengan target yang diharapkan

Untuk vektor input kedua yang mewakili Huruf X, maka output yang diperoleh adalah:

$$(w_{total})^t = [1 \ -1 \ 1 \ -1 \ 1 \ -1 \ 1 \ -1 \ 1] * \begin{bmatrix} 0 & 2 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 2 & 0 \\ -2 & 0 \\ 0 & 2 \\ -2 & 0 \\ 0 & 2 \end{bmatrix} = [-10 \ 8]$$

Karena  $y_{in2}(1) = 10 > 0$ , maka  $y_2(1) = 1$  dan  $(y_{in2}(2) = 8 > 0$ , maka  $y_2(2) = 1)$ , maka nilai  $y_2 = [1 \ 1]$ , sama dengan target yang diharapkan

4. Balik arah, cari nilai  $x_{in}$

Sekarang apabila dibalik,  $y$  digunakan sebagai input untuk mendapatkan  $x$ . Misalkan kita cobakan vektor input pertama  $y_1 = [-1 \ 1]$ , maka output yang diperoleh adalah:

$$(w_{total})^t = [-1 \ 1] * \begin{bmatrix} 0 & -2 & 0 & -2 & 2 & -2 & 0 & -2 & 0 \\ 2 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 2 \end{bmatrix} = 2[2 \ 2 \ 2 \ -2 \ 2 \ 2 \ 2] \Rightarrow [1 \ 1 \ 1 \ -1 \ 1 \ 1 \ 1]$$

Outputnya sama dengan target, yaitu Huruf O.

Untuk vektor input kedua yaitu  $y_2 = [1 \ 1]$ , maka output yang diperoleh adalah:

$$(w_{total})^t = [1 \ 1] * \begin{bmatrix} 0 & -2 & 0 & -2 & 2 & -2 & 0 & -2 & 0 \\ 2 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 2 \end{bmatrix} = 2[-2 \ 2 \ -2 \ 2 \ -2 \ 2 \ -2] \Rightarrow [1 \ 1 \ 1 \ -1 \ 1 \ 1 \ -1]$$

Outputnya sama dengan target, yaitu Huruf X

### 3. HASIL DAN ANALISIS

#### 3.1 Penerapan Principal Component Analysis

Langkah 1 :

Menyusun seluruh *training image* menjadi 1 matriks seperti berikut.

$$C_1 = \begin{bmatrix} 295 & 259 & 408 \\ 484 & 454 & 444 \\ 388 & 409 & 248 \end{bmatrix} = 295 \ 259 \ 408 \ 484 \ 454 \ 444 \ 288 \ 409 \ 248$$

$$C_2 = \begin{bmatrix} 527 & 341 & 350 \\ 461 & 281 & 611 \\ 566 & 350 & 719 \end{bmatrix} = 527 \ 341 \ 350 \ 461 \ 281 \ 611 \ 566 \ 350 \ 719$$

Langkah 2 :

Mencari nilai rata-rata (*mean*) *flatvector* dengan menjumlahkan seluruh baris dengan jumlah *image*, seperti berikut :

$$\text{Mean } (f_v) = \frac{295 \ 259 \ 408 \ 484 \ 454 \ 444 \ 288 \ 409 \ 248 + 527 \ 341 \ 350 \ 461 \ 281 \ 611 \ 566 \ 350 \ 719}{2} = 411 \ 300 \ 379 \ 473 \ 368 \ 528 \ 427 \ 380 \ 484$$

Dengan hasil hitung nilai pada citra *mean flatvector*, maka nilai *eigenface* untuk matriks *flatvector* yang sudah disusun tersebut dapat dihitung nilai *eigenfacenya*.

Langkah 3 :

Menghitung nilai *eigenface* dengan cara mengurangi baris-baris pada matriks *flatvector* dengan nilai rata-rata *flatvector*. Jika didapatkan nilai dibawah 0 (nilai minus), maka nilai diganti dengan 0 (nol).

$$C_1 = \frac{295 \ 259 \ 408 \ 484 \ 454 \ 444 \ 288 \ 409 \ 248 - 411 \ 300 \ 279 \ 473 \ 368 \ 528 \ 427 \ 380 \ 484}{2} = \frac{0 \ 0 \ 129 \ 211 \ 86 \ 0 \ 0 \ 29 \ 0}{527 \ 341 \ 350 \ 461 \ 281 \ 611 \ 566 \ 350 \ 719} = \frac{411 \ 300 \ 279 \ 473 \ 368 \ 528 \ 427 \ 380 \ 484 - 116 \ 41 \ 71 \ 0 \ 0 \ 83 \ 139 \ 0 \ 235}{116 \ 41 \ 71 \ 0 \ 0 \ 83 \ 139 \ 0 \ 235}$$

Untuk mengenali *citra tes (testface)* pada saat uji, langkah identifikasinya adalah menghitung nilai *eigenface* untuk matriks *testface* dengan cara seperti sebelumnya yaitu dimulai dari awal penentuan nilai *flatvector* kemudian dilakukan pengurangan dengan nilai rata-rata *flatvector* (dari *citra training*), dan mendapatkan *eigenface* untuk *testface* seperti berikut ini :

$$C_t = \begin{bmatrix} 315 & 258 & 186 \\ 354 & 420 & 447 \\ 300 & 327 & 354 \end{bmatrix}$$

$$C_T = \frac{\begin{array}{ccccccccc} 315 & 258 & 186 & 354 & 420 & 447 & 300 & 327 & 354 \\ 411 & 300 & 279 & 473 & 368 & 528 & 427 & 380 & 484 \end{array}}{0 \quad 0 \quad 0 \quad 0 \quad 52 \quad 0 \quad 0 \quad 0 \quad 0}$$

$$C_t = 0 \quad 0 \quad 0 \quad 0 \quad 52 \quad 0 \quad 0 \quad 0 \quad 0$$

Langkah 4 :

Melakukan identifikasi dengan *euclidean distance* yaitu menentukan jarak terdekat antara nilai *eigenface training* dengan nilai *tesface*.

$$C_1 = \frac{\begin{array}{ccccccccc} 0 & 0 & 129 & 211 & 86 & 0 & 0 & 29 & 0 \\ 0 & 0 & 0 & 0 & 52 & 0 & 0 & 0 & 0 \end{array}}{0 \quad 0 \quad 129 \quad 211 \quad 86 \quad 0 \quad 0 \quad 29 \quad 0}$$

$$C_1 = 0+0+129+211+86+0+0+29+0$$

$$C_1 = 455$$

$$C_2 = \frac{\begin{array}{ccccccccc} 116 & 41 & 71 & 0 & 0 & 83 & 139 & 0 & 235 \\ 0 & 0 & 0 & 0 & 52 & 0 & 0 & 0 & 0 \end{array}}{116 \quad 41 \quad 71 \quad 0 \quad 52 \quad 83 \quad 139 \quad 0 \quad 235}$$

$$C_2 = 116+41+71+0+52+83+139+0+235$$

$$C_2 = 737$$

Dari hasil perhitungan yang dilakukan diatas, diperoleh jarak *citra face* satu memiliki nilai terkecil yaitu 455, maka hasil identifikasi menyimpulkan bahwa *tesface* lebih mendekati pada *face* satu dari pada *face* dua.

### 3.2 Penerapan Principal Component Analysis

Perhitungan menggunakan metode *Principal Component Analysis* menunjukkan *citra 1 tesface* terkecil, maka pada proses metode *bidirectional associative memory* akan digunakan nilai *citra 1*, untuk melihat perbandingan antara dua mete yang berbeda.

Langkah pertama :

- Menentukan *citra input* untuk diproses

**Table 1.** Nilai citra RGB 3x3 pixel

R	G	B
295	259	408
484	454	444
388	409	248

Dari nilai *citra RGB* pada tabel 4.3 di atas, maka dilakukan pencarian nilai *citra grayscale* seperti berikut ini :

$$295 = \frac{117 + 83 + 55}{3} = 98$$

Langkah yang sama dilakukan pada nilai *RGB*, Sehingga didapatkan nilai *citra grayscale* dari *citra RGB* sebagai berikut :

**Tabel 2.** Nilai *citra grayscale*

R	G	B
98	86	136
161	151	148
129	136	83

- Dari hasil nilai *citra grayscale* pada *table 4.4* diatas, maka dilakukan proses pengubahan nilai ke nilai *biner* dengan menggunakan *threshold* sebesar 128, dimana nilai piksel yang kurang dari 128 akan di ubah menjadi 0 sedangkan jika nilai pikselnya sama dengan 128 atau lebih akan dirubah menjadi 1, seperti berikut ini.

**Table 3.** nilai *biner*

0	0	1
1	1	1
1	1	0

- Dari nilai *citra biner* yang diperoleh dari tabel 4.5 diatas, maka dilakukan proses perubahan ke nilai *bipolar*, dimana proses ini jika *citra* bernilai 0 maka dirubah menjadi -1 sedangkan nilai 1 nilainya tetap, maka diperoleh nilai bipolar seperti pada tabel berikut :

**Tabel 4.** nilai *bipolar*

-1	-1	1
1	1	1
1	1	-1

- Proses menentukan pola *input* ( $x_{11}$ ) dan *output* berdasarkan *Bidirectional Associative memory*.  
target [1,9] = {-1, -1, -1, -1, 1, 1, -1, -1, -1}

- Menghitung matriks bobot ( $w_{total}$ ) atau  $W_{11} = X_{11} * target1$ , hasil perkalian diperoleh seperti berikut ini :  

$$W_{total}(W11) = \{-1, -1, 1, 1, 1, 1, 1, -1\}$$
  - Melakukan recal ( $x_{in11}$ ) pola pada *inputan*, nilai *transpose* dari  $W_{total}$  yang dihasilkan pada tahap ke 5 akan dikalikan dengan seluruh target yaitu variabel *target1*. Proses ini dilakukan untuk membuktikan hasil perkalian antara  $W_{total}$  dengan targer1  

$$W\_in_{11} = \{1, -1, 1, 1, 1, 1, 1, -1\}$$

### Kesimpulan :

Jika nilai  $y_{in11}$  sama dengan nilai target maka identifikasi berhasil dilakukan atau sesuai dengan target, sebaliknya jika nilai  $y_{in11}$  berbeda maka identifikasi gagal dilakukan atau tidak mengenali target. Dari hasil perhitungan antara dua metode yang di bandingkan, maka penulis menyimpulkan metode BAM kurang efektif dalam pengenalan wajah seseorang dibuktikan dengan hasil yang didapat pada langkah 6 nilai  $y_{in11}$  tidak sama dengan nilai target.

### 3.3 Hasil Pengujian PCA

Pada pengujian dengan metode *Principal Component Analysis* ini akan diberikan *citra* wajah sebagai *inputan* seperti berikut :

### 1. Citra inputan wajah training



Gambar 4. *citra inputan sampel metode PCA*

Dari *citra input* wajah sampel pada gambar 4.19 diatas, di dapatkan jumlah nilai *Euclidean Distance* =1,791.647 dapat dilihat pada gambar berikut :

t-SNE-PCA - Notepad																							
File	Edit	Format	View	Help																			
97	78	86	91	96	83	106	125	149	153	155	151	147	147	149	151	152	158	156	149	156	156	142	144
123	85	95	100	96	73	92	106	126	143	151	152	151	150	153	157	149	152	152	149	152	152	142	149
144	131	137	131	95	116	131	135	115	127	134	145	154	155	149	143	148	152	153	152	147	146	145	137
144	135	135	111	93	92	123	137	135	127	134	145	154	155	149	143	148	152	153	152	147	146	145	137
144	145	135	118	87	100	95	122	157	160	158	150	158	150	158	151	140	148	149	147	134	132	135	119
144	149	145	139	130	100	95	122	157	160	158	151	146	147	147	145	146	141	139	134	132	135	105	106
122	120	125	149	149	132	100	101	123	137	147	146	152	151	144	144	134	121	108	100	98	92	78	47
122	125	95	122	100	149	155	136	119	123	146	156	155	152	153	154	155	152	154	155	152	154	155	149
106	102	99	133	136	168	161	151	150	135	151	155	149	149	152	148	141	132	105	76	56	49	49	40
95	101	124	149	148	155	149	151	148	149	148	144	144	144	144	139	119	87	57	42	38	37	31	28
124	124	143	157	157	155	149	154	155	155	155	155	155	155	155	155	155	155	155	155	155	155	155	155
145	156	156	152	145	145	138	123	113	137	153	161	152	159	139	123	84	71	51	38	33	31	30	29
154	156	149	143	124	121	101	81	85	124	149	157	148	146	115	58	44	39	39	34	29	30	36	
144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144
66	77	75	79	95	95	107	109	111	136	148	151	147	146	145	113	59	28	27	32	32	29	30	32
95	91	55	55	85	97	136	134	144	147	146	145	146	145	126	96	57	28	21	24	30	31	30	37
141	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142
144	151	147	88	87	82	82	139	139	136	127	148	148	148	148	86	48	36	27	27	26	29	32	34
144	137	138	129	128	145	149	148	148	149	148	142	136	123	99	68	48	33	39	28	27	30	34	37
144	143	152	161	149	151	153	147	127	141	141	126	106	83	57	39	33	29	29	30	32	36	34	41
137	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141	141
137	141	151	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148	148
138	144	138	141	141	149	144	143	138	158	126	80	42	39	49	33	28	26	32	34	34	38	46	51
144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144
145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145
139	143	143	144	144	147	144	143	144	143	144	143	126	79	50	32	36	27	33	32	32	37	37	44
144	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145
144	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145	145
147	140	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147	147
138	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143
143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143
143	138	139	143	143	146	141	129	75	44	38	37	41	43	48	53	55	58	71	92	113	118	115	119
144	143	144	144	144	144	149	150	132	81	94	42	36	44	47	50	56	58	72	103	115	117	116	117
144	143	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144	144
138	139	141	141	144	144	149	152	142	76	70	41	38	48	52	57	61	62	71	89	109	118	117	116
138	139	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140
144	142	142	142	142	142	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143
144	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142	142
144	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143
143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143	143
142	141	140	142	142	147	142	131	67	42	44	46	53	65	72	77	86	95	106	112	113	114	116	118
140	144	139	141	141	146	149	140	129	69	46	43	46	56	65	73	82	91	108	112	114	114	116	118

Gambar 5. Nilai *citra inputan* sampel metode PCA

Nilai *Euclidean Distance* 1,791.647 dari *citra* sampel digunakan sebagai target pada *citra* wajah yang akan dikenali nantinya.

2. Melakukan identifikasi menggunakan metode PCA, dengan memberi *input citra* wajah sebagai berikut :



Gambar 6. Citra wajah *input* identifikasi

3. Dari *citra* wajah input identifikasi di dapatkan nilai *citra* wajah seperti berikut

**Gambar 7.** Nilai citra wajah *input* identifikasi

Dari hasil proses pada citra uji didapatkan  $Euclidean\ Distance=1,584.278$ , karena jarak nilai  $Euclidean\ Distance$  medekati nilai taget maka wajah dikenali, hasil pengujian yang telah dilakukan dapat dilihat pada tabel berikut :

Tabel 5. Hasil Pengenalan Dengan Merode PCA

No	Citra Sampel	Citra Input Target	Nilai Euclidean Distance	Keterangan
1			1,584.278	Etini Harefa
2			1,684.278	Etini Harefa

### **3.4 Hasil pengujian Metode BAM**

Dalam pengujian ini, sistem diberikan 2 (dua) *inputan* dengan gambar wajah yang sebelumnya digunakan pada metode PCA sehingga perbandingan pengenalan wajah dengan menggunakan metode berbeda dapat ditentukan, berikut ini adalah hasil pengujian dengan metode BAB.

## 1. *Input citra latih*



Gambar 8. *Citra input latih* `rgb`

2. Dari gambar 4.25 di atas mengalami proses digitalisasi *citra*, sehingga menghasilkan *citra grayscale* seperti berikut ini :



Gambar 9. citra grayscale latih

Dari gambar 4.26 diatas *citra grayscale* memiliki nilai yang tersimpan pada database *notepad* sebagai berikut :

	File	Edit	Format	View	Help
0.0	78	96	91	96	83
0.1	78	77	100	86	80
0.2	93	95	99	92	92
0.3	111	111	99	122	101
0.4	148	148	92	122	161
0.5	148	148	122	122	135
0.6	148	148	122	122	135
0.7	148	148	122	122	135
0.8	148	148	122	122	135
0.9	148	148	122	122	135
1.0	148	148	122	122	135
1.1	148	148	122	122	135
1.2	148	148	122	122	135
1.3	148	148	122	122	135
1.4	148	148	122	122	135
1.5	148	148	122	122	135
1.6	148	148	122	122	135
1.7	148	148	122	122	135
1.8	148	148	122	122	135
1.9	148	148	122	122	135
2.0	148	148	122	122	135
2.1	148	148	122	122	135
2.2	148	148	122	122	135
2.3	148	148	122	122	135
2.4	148	148	122	122	135
2.5	148	148	122	122	135
2.6	148	148	122	122	135
2.7	148	148	122	122	135
2.8	148	148	122	122	135
2.9	148	148	122	122	135
3.0	148	148	122	122	135
3.1	148	148	122	122	135
3.2	148	148	122	122	135
3.3	148	148	122	122	135
3.4	148	148	122	122	135
3.5	148	148	122	122	135
3.6	148	148	122	122	135
3.7	148	148	122	122	135
3.8	148	148	122	122	135
3.9	148	148	122	122	135
4.0	148	148	122	122	135
4.1	148	148	122	122	135
4.2	148	148	122	122	135
4.3	148	148	122	122	135
4.4	148	148	122	122	135
4.5	148	148	122	122	135
4.6	148	148	122	122	135
4.7	148	148	122	122	135
4.8	148	148	122	122	135
4.9	148	148	122	122	135
5.0	148	148	122	122	135
5.1	148	148	122	122	135
5.2	148	148	122	122	135
5.3	148	148	122	122	135
5.4	148	148	122	122	135
5.5	148	148	122	122	135
5.6	148	148	122	122	135
5.7	148	148	122	122	135
5.8	148	148	122	122	135
5.9	148	148	122	122	135
6.0	148	148	122	122	135
6.1	148	148	122	122	135
6.2	148	148	122	122	135
6.3	148	148	122	122	135
6.4	148	148	122	122	135
6.5	148	148	122	122	135
6.6	148	148	122	122	135
6.7	148	148	122	122	135
6.8	148	148	122	122	135
6.9	148	148	122	122	135
7.0	148	148	122	122	135
7.1	148	148	122	122	135
7.2	148	148	122	122	135
7.3	148	148	122	122	135
7.4	148	148	122	122	135
7.5	148	148	122	122	135
7.6	148	148	122	122	135
7.7	148	148	122	122	135
7.8	148	148	122	122	135
7.9	148	148	122	122	135
8.0	148	148	122	122	135
8.1	148	148	122	122	135
8.2	148	148	122	122	135
8.3	148	148	122	122	135
8.4	148	148	122	122	135
8.5	148	148	122	122	135
8.6	148	148	122	122	135
8.7	148	148	122	122	135
8.8	148	148	122	122	135
8.9	148	148	122	122	135
9.0	148	148	122	122	135
9.1	148	148	122	122	135
9.2	148	148	122	122	135
9.3	148	148	122	122	135
9.4	148	148	122	122	135
9.5	148	148	122	122	135
9.6	148	148	122	122	135
9.7	148	148	122	122	135
9.8	148	148	122	122	135
9.9	148	148	122	122	135
10.0	148	148	122	122	135
10.1	148	148	122	122	135
10.2	148	148	122	122	135
10.3	148	148	122	122	135
10.4	148	148	122	122	135
10.5	148	148	122	122	135
10.6	148	148	122	122	135
10.7	148	148	122	122	135
10.8	148	148	122	122	135
10.9	148	148	122	122	135
11.0	148	148	122	122	135
11.1	148	148	122	122	135
11.2	148	148	122	122	135
11.3	148	148	122	122	135
11.4	148	148	122	122	135
11.5	148	148	122	122	135
11.6	148	148	122	122	135
11.7	148	148	122	122	135
11.8	148	148	122	122	135
11.9	148	148	122	122	135
12.0	148	148	122	122	135
12.1	148	148	122	122	135
12.2	148	148	122	122	135
12.3	148	148	122	122	135
12.4	148	148	122	122	135
12.5	148	148	122	122	135
12.6	148	148	122	122	135
12.7	148	148	122	122	135
12.8	148	148	122	122	135
12.9	148	148	122	122	135
13.0	148	148	122	122	135
13.1	148	148	122	122	135
13.2	148	148	122	122	135
13.3	148	148	122	122	135
13.4	148	148	122	122	135
13.5	148	148	122	122	135
13.6	148	148	122	122	135
13.7	148	148	122	122	135
13.8	148	148	122	122	135
13.9	148	148	122	122	135
14.0	148	148	122	122	135
14.1	148	148	122	122	135
14.2	148	148	122	122	135
14.3	148	148	122	122	135
14.4	148	148	122	122	135
14.5	148	148	122	122	135
14.6	148	148	122	122	135
14.7	148	148	122	122	135
14.8	148	148	122	122	135
14.9	148	148	122	122	135
15.0	148	148	122	122	135
15.1	148	148	122	122	135
15.2	148	148	122	122	135
15.3	148	148	122	122	135
15.4	148	148	122	122	135
15.5	148	148	122	122	135
15.6	148	148	122	122	135
15.7	148	148	122	122	135
15.8	148	148	122	122	135
15.9	148	148	122	122	135
16.0	148	148	122	122	135
16.1	148	148	122	122	135
16.2	148	148	122	122	135
16.3	148	148	122	122	135
16.4	148	148	122	122	135
16.5	148	148	122	122	135
16.6	148	148	122	122	135
16.7	148	148	122	122	135
16.8	148	148	122	122	135
16.9	148	148	122	122	135
17.0	148	148	122	122	135
17.1	148	148	122	122	135
17.2	148	148	122	122	135
17.3	148	148	122	122	135
17.4	148	148	122	122	135
17.5	148	148	122	122	135
17.6	148	148	122	122	135
17.7	148	148	122	122	135
17.8	148	148	122	122	135
17.9	148	148	122	122	135
18.0	148	148	122	122	135
18.1	148	148	122	122	135
18.2	148	148	122	122	135
18.3	148	148	122	122	135
18.4	148	148	122	122	135
18.5	148	148	122	122	135
18.6	148	148	122	122	135
18.7	148	148	122	122	135
18.8	148	148	122	122	135
18.9	148	148	122	122	135
19.0	148	148	122	122	135
19.1	148	148	122	122	135
19.2	148	148	122	122	135
19.3	148	148	122	122	135
19.4	148	148	122	122	135
19.5	148	148	122	122	135
19.6	148	148	122	122	135
19.7	148	148	122	122	135
19.8	148	148	122	122	135
19.9	148	148	122	122	135
20.0	148	148	122	122	135
20.1	148	148	122	122	135
20.2	148	148</			



**Gambar 13.** Citra *input grayscale* identifikasi

Citra grayscale pada gambar 4.31 di atas, memiliki nilai piksel sebagai berikut :

**Gambar 12.** Nilai citra Grayscale

5. Dari *citra grayscale* pada 5 di atas, dengan nilai pada gambar 4.32 maka, dilakukan proses konversi *citra grayscale* ke hitam putih (*bipolar*) seperti terlihat berikut ini :



**Gambar 13.** *Citra bipolar Uji*

*Citra bipolar di atas memiliki nilai sebagai berikut :*

**Gambar 14.** Nilai Citra Bipolar

6. Dari hasil nilai *citra bipolar* pada gambar 8di atas, dilakukan perkalian pada nilai *citra bipolar* target sehingga mendapatkan hasil sebagai berikut :

**Gambar 15.** Nilai hasil perkalian

Dari hasil yang yang di dapatkan maka pada pengujian metode BAM dapat dilihat pada tabel berikut ini :

**Tabel 6.** hasil uji metode BAM

No	Citra Input (Training)	Citra input (Target)	Output
1			Tidak Dikenal
2			Tidak dikenal

#### 4. KESIMPULAN

Setelah melakukan implementasi dari Analisa Perbandingan Pengenalan Wajah Dengan Principal Component Analysis Dan Bidirectional Associative Memory, maka diperoleh kesimpulan sebagai berikut: 1. Pada hasil implementasi pada penelitian dengan wajah berhasil dilakukan pengenalan untuk menentukan perbandingan pengenalan wajah. 2. Hasil pencarian antara wajah sampel dengan wajah uji, di dapatkan hasil perbandingan pengenalan dimana metode PCA lebih efektif di gunakan di bandingkan dengan metode BAM. 3. Hasil perbandingan pengenalan wajah dengan metode Principal Component Analysis dengan Bidirectional Associative Memory pada pengenalan wajah dapat ditentukan

#### REFERENSI

- [1] R. F. e. all, "Implementasi Principal Component Analysis (PCA) Untuk Pengenalan Wajah Manusia," Nusantara of Engineering, vol. 2, no. 2355-6684, p. 69, 215.
- [2] S. M. Rizqi Elmuna Hidayah, "Implementasi Metode Principal Component Analysis pada pengenalan wajah berbasis Eigenface," Technologia, vol. 7, no. 2656-8047, pp. -, 2016.
- [3] Z. Arufun, "Jaringan Syaraf Tiruan Bidirectional Associative Memory (BAM) Sebagai Identifikasi Pola Sidik jari Manusia," Jurnal Informatika Mulawarman, vol. 4, p. 26, 2009.
- [4] A. Giawa, "Implementasi Metode Bidirectional associative memory pada absensi berbasis identifikasi wajah," Jurnal Pelita Informatika, vol. 8, no. 2301-9425, p. 111, 2019.
- [5] A. Sudarsono, "Jaringan Syaraf Tiruan Untuk Memprediksi Laju Pertumbuhan Penduduk Menggunakan Metode Bacpropagation (Studi Kasus Di Kota Bengkulu)," Jurnal Media Infotama, vol. 12, no. 1858-2680, pp. 62-63, 2016.
- [6] A. B. S. Sumainna and H. Maulana, "Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode Principal Component Analysis (PCA)," JURNAL TEKNIK INFORMATIKA, vol. 9, no. 1979-9160, pp. 166-175, 2016.
- [7] Salamun and F. Wazir, "Rancangan Bangun Sistem Pengenalan Wajah Dengan Metode Principal Component Analysis," RABIT: JurnalTeknologi dan Sistem Informasi Univrab, vol. 1, no. 2502-891X, pp. 62-65, 2016.
- [8] A. R. Syakhala, D. Puspitaningrum and E. P. P. , "Perbandingan Metode Principal Component Analysis (Pca) Dengan Metode Hidden Markov Model (Hmm) Dalam Pengenalan Identitas Seseorang Melalui Wajah," Jurnal Rekursif, vol. 3, no. 2303-0755, pp. 70-71, 2015.
- [9] M. Fikry, F. Fadlisyah and D. Putri, "Sistem Pendekripsi Pola Citra Tajwid Alquran Mad Lazim Mutsaqal Kilmi Menggunakan Metode Algoritma BAM & FAM," TECHSI-Jurnal Teknik Informatika, vol. 10, no. DOI: <https://doi.org/10.29103/techsi.v10i2.885>, pp. 101-102, 2018.
- [10] Santoso and R. Nurmaliha, "Perencanaan dan Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas (Studi Kasus Politeknik Negeri Tanah Laut)," Jurnal Integrasi, vol. 9, no. 2548-9828, pp. 86-87, 2017.
- [11] F. Ayu and N. Permatasari, "Perancangan Sistem Informasi Perancangan Data Praktek Kerja Lapangan (PKL) Pada Devisi Humas PT Pegadaian," Jurnal Intra-Tech, vol. 2, no. 2549-0222, p. 13, 2018.
- [12] R. Irviani and R. Oktaviana, "Aplikasi Perpustakaan Pada SMA N1 Kelumbayan Barat Menggunakan Visual Basic," Jurnal TAM (Technology Acceptance Model), vol. 8, no. 2579-4221, pp. 63-69, 2017.
- [13] Muhammad Saed Novendri, Ade Saputra and Chandra Eri Firman, "Aplikasi Inventaris Barang Pada MTS Nurul Islam Dumai Menggunakan PHP dan MYSQL," LENTERA DUMAI, vol. 10, no. 2528-1062, pp. 48-49, 2019.
- [14] Y. Heriyanto, "Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car," Jurnal Intra-Tech, vol. 2, no. 2549-0222, pp. 66-69, 2018.