

## **PERBANDINGAN ALGORITMA EIGENFACE DENGAN LOCAL BINARY PATTERN (LBP) PADA PENGENALAN WAJAH**

**Isnawati Muslihah<sup>1)</sup>, Helmi Imaduddin<sup>2)</sup>**

<sup>1</sup>Institut Teknologi Bisnis AAS Indonesia

<sup>2</sup>Universitas Muhammadiyah Surakarta

Email: [isnawati@itbaas.ac.id](mailto:isnawati@itbaas.ac.id)

**Abstrak :** Algoritma untuk klasifikasi citra dalam hal ini adalah pengenalan wajah telah banyak ditemukan, salah satunya adalah dengan menghitung jarak antar fitur dalam wajah seperti mata, hidung, telinga, dan bibir. Akan tetapi input citra yang digunakan sangat menentukan pada proses pengenalan, terutama pada tahap prapemrosesan khususnya tahap ekstraksi ciri, karena pengenalan wajah sangat bergantung pada ekstraksi ciri yang dilakukan. Input citra berwarna pada citra asli yang digunakan untuk input pengenalan di beberapa penelitian kurang optimal karena ciri yang didapatkan masih sangat kompleks. Maka dari itu pada penelitian ini digunakan metode ekstraksi dengan eigenface dan LBP yang mengubah citra input asli menjadi citra abu-abu kemudian dilakukan konversi ke dalam bentuk matriks yang selanjutnya akan dikenali dengan algoritma Euclidean distance. Hasil yang diperoleh bahwa algoritma dari metode LBP lebih optimal dengan hasil 91% wajah dikenali, dibandingkan dengan metode eigenface dengan 84% wajah dikenali pada ukuran yang sama yaitu  $500 \times 500$  piksel.

**Kata kunci:** *eigenface, euclidean distance, local binary pattern, pengenalan wajah*

**Abstract :** Algorithms for image classification, in this case facial recognition, have been found, one of which is by calculating the distance between facial features such as eyes, nose, ears and lips. However, the image input used is very decisive in the recognition process, especially at the pre-processing stage, especially the feature extraction stage, because facial recognition is very dependent on the feature extraction that is carried out. Color image input in the original image used for recognition input in some studies is not optimal because the features obtained are still very complex. Therefore, in this study, the extraction method with eigenface and LBP is used which converts the original input image into a gray image and then converts it into a matrix form which will then be recognized by the Euclidean distance algorithm. The results obtained were that the algorithm of the LBP method was more optimal with 91% of faces recognized, compared to the eigenface method with 84% of faces recognized at the same size, namely  $500 \times 500$  pixels.

**Keywords:** *eigenface, euclidean distance, local binary pattern, face recognition*

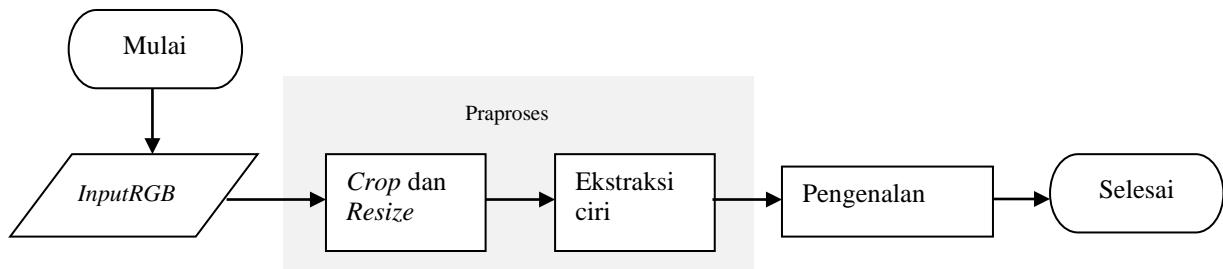
### **PENDAHULUAN**

Pengenalan wajah adalah salah satu teknik dari sistem identifikasi berbasis *biometric* yang dikembangkan berdasarkan perbedaan ciri wajah seseorang sebagai parameter utamanya, karena wajah adalah identitas utama dan komponen paling penting yang digunakan untuk mengenali seseorang. Terdapat banyak algoritma untuk melakukan klasifikasi citra, dalam hal ini adalah pengenalan wajah salah satunya adalah dengan menerapkan perhitungan

jarak fitur. Pada penelitian sebelumnya dilakukan penelitian dengan membangun sebuah sistem pengenalan wajah dengan membandingkan metode penelitian *Eucledian Distance*, *Manhattan*, dan *Chi square* (Yushar et al., 2019). Hasil yang diperoleh adalah performa terbaik didapatkan menggunakan Eucledian Distance. Dari beberapa penelitian sebelumnya masih terdapat permasalahan pada variasi ciri tekstur yang diujikan pada *Eucledian Distance*, yaitu citra asli yang digunakan untuk *input* kurang optimal karena pengenalan wajah sangat bergantung pada hasil ekstraksi ciri yang dilakukan.

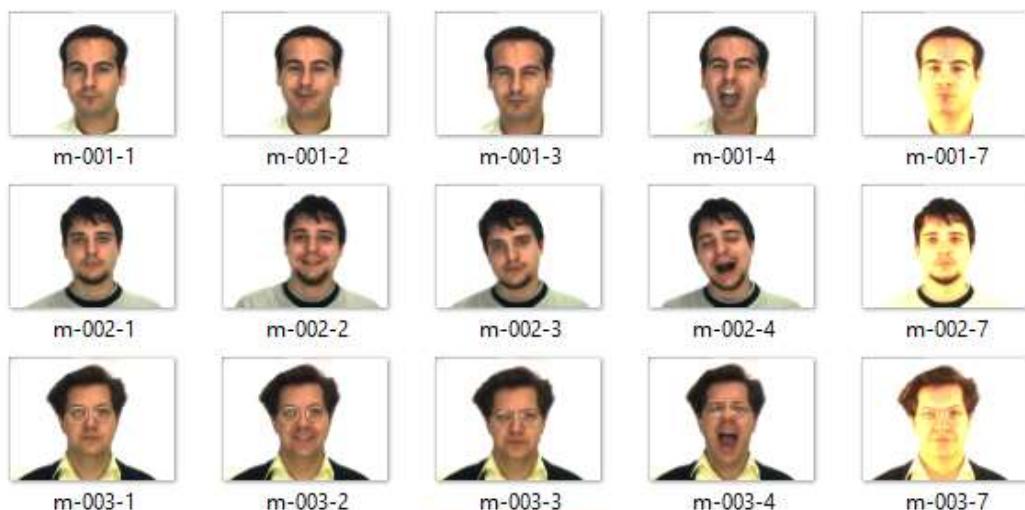
## METODE

Pengenalan wajah secara garis besar memiliki komposisi proses yang sama dengan sistem pengenalan pada umumnya, yaitu akuisisi citra, prapemrosesan, dan klasifikasi atau pengenalan. Alur keseluruhan dari proses sistem pada penelitian ini dapat dilihat pada Gambar 2.1. Pada gambar nampak bagaimana alur sistem mulai dari *input* gambar hingga mendapatkan pengenalan.



**Gambar 2.1 Gambaran umum alur sistem**

Pengenalan wajah pada penelitian ini tidak melalui akuisisi citra, akan tetapi mengambil dataset dari *AR Face Database* dengan menggunakan 680 citra wajah, yang terdiri dari 76 objek laki-laki dan 60 objek perempuan, dengan 5 ekspresi pada masing-masing individu. Ekspresi wajah yang digunakan dari *AR Face* meliputi wajah netral, tersenyum, marah, teriak, dan terdapat cahaya. Contoh citra asli dari *AR Face* yang digunakan dapat dilihat pada Gambar 2.2



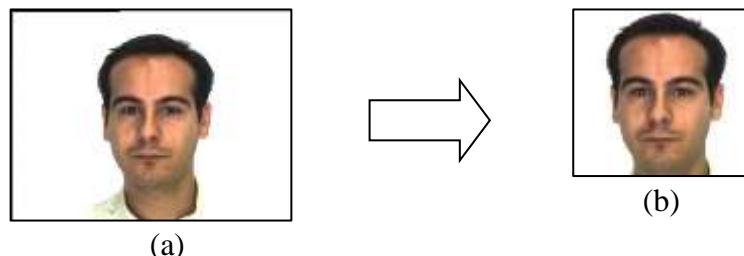
**Gambar 2.2 Ekspresi dari beberapa Objek pada AR Face Database**

### **1.1 Preprocessing (prapemrosesan)**

Sebelum citra gambar diolah untuk proses pengenalan, harus dilakukan prapemrosesan terlebih dahulu untuk mendapatkan ciri yang kita inginkan sesuai dengan *input* yang dibutuhkan (Sugiarta et al., 2016). Tahapan prapemrosesan tidak semuanya dilakukan pada penelitian ini, misalnya proses akuisisi citra karena pada kasus penelitian ini data yang diperoleh sudah dalam bentuk database citra hasil (tidak diambil sendiri oleh penulis). Tahapan prapemrosesan pada penelitian ini meliputi cropping image, dan ekstraksi ciri dengan *eigenface* dan LBP.

#### **Cropping Image (Pemotongan Ukuran Gambar)**

Proses yang pertama dilakukan pada citra gambar sebelum masuk ke sistem adalah melakukan *cropping image*. Proses pemotongan gambar dilakukan dengan menentukan titik piksel pada gambar dan memotong bagian wajah dengan ukuran  $500 \times 500$ . Pemotongan gambar area wajah dimaksutkan untuk mengurangi penghitungan matriks pada area *background* gambar. Proses *cropping image* dapat dilihat pada Gambar 2.3



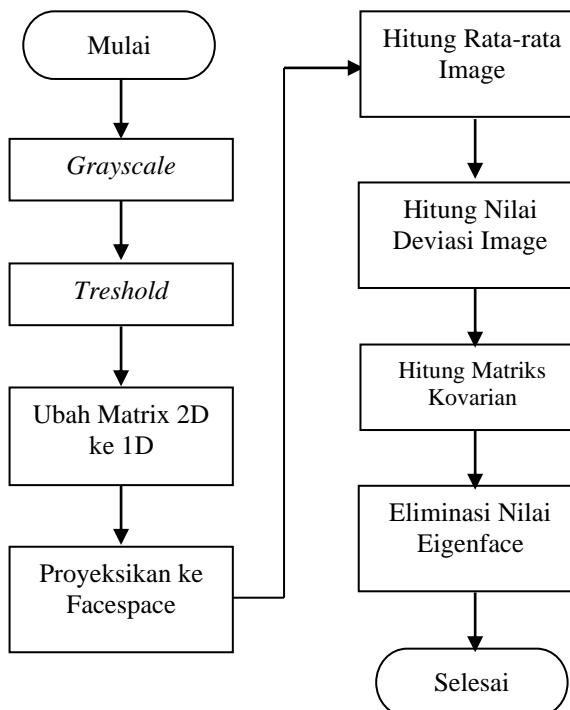
**Gambar 2.3** *Cropping image* dari gambar (a) citra asli ukuran  $768 \times 576$  piksel, ke gambar (b) citra hasil cropping ukuran  $500 \times 500$  piksel

#### **Feature Extraction (Ekstraksi Ciri)**

Ekstraksi ciri dilakukan untuk mendapatkan fitur-fitur dari citra gambar yang digunakan setelah melalui prapemrosesan. Penelitian ini ekstraksi yang digunakan adalah dengan dua metode yaitu *eigenface* dan LBP. Dengan melakukan ekstraksi ini akan didapatkan nilai matriks baru dan menghasilkan *feature vector* dari setiap citra, sehingga citra gambar yang digunakan untuk *input* sistem pada proses pengenalan memiliki kualitas yang lebih baik.

#### **Eigenface**

*Eigenface* adalah metode ekstraksi ciri yang mempresentasikan gambar sebagai vector untuk menentukan komponen utama dari wajah, atau *eigenvector* dari matriks kovarian dari kumpulan gambar wajah (Mutiara & Prasetyo, 2019). Alur proses ekstraksi ciri dengan *eigenface* dapat dilihat pada Gambar 2.4

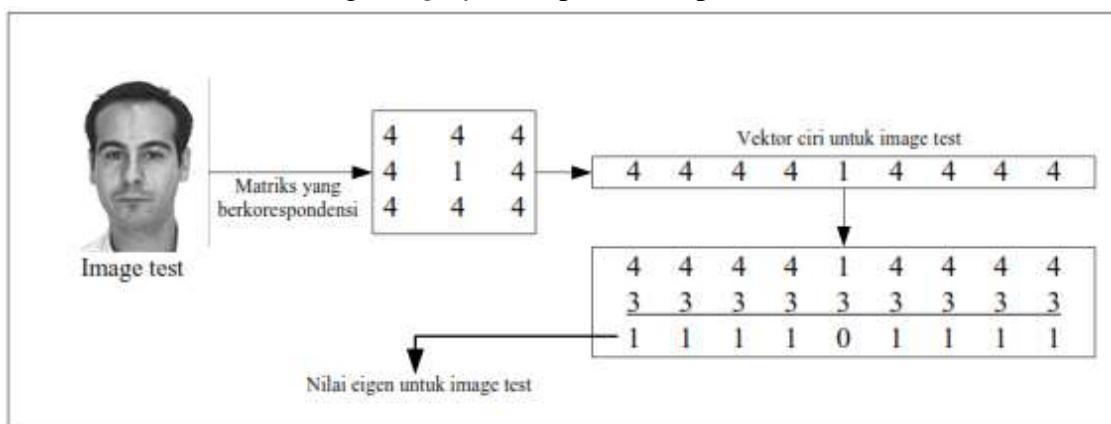


Gambar 2.4 Alur proses *eigenface*

Adapun algoritma selengkapnya adalah:

1. Buat *MakeFlatVectors* (*ImageList*, *N*, *M*): *Image List* adalah kumpulan dari *N training image*, dimana setiap *image* adalah *W x H* piksel. *M* adalah ukuran dari *vector flat* yang harus dibuat.
2. Gabungkan setiap *image* dalam *WH* elemen *vector* dengan menggabungkan semua baris. Buat *image* matriks sebagai matriks *N x WH* berisi semua gambar yang digabung.
3. Jumlahkan semua baris pada *image* matriks dan bagi dengan *N* untuk mendapatkan rataan vektor gabungan. Namakan vektor elemen *WH* dengan *R*.
4. Kurangi *image* matriks dengan *average image R*. Namakan matriks baru ukuran *N x WH* sebagai *R'*.
5. Jika pada elemen-elemen dari matriks *R'* ditemukan nilai negatif, ganti nilainya dengan nilai 0.

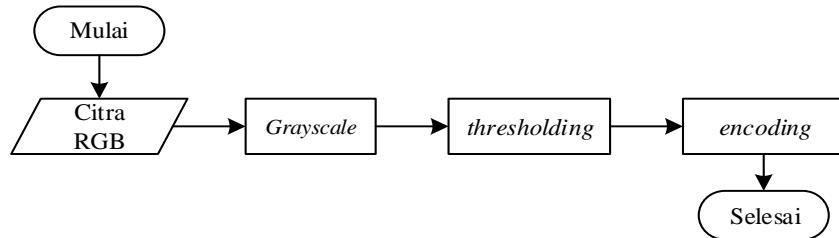
Proses ekstraksi ciri dengan *eigenface* dapat dilihat pada Gambar 2.5



Gambar 2.5 Proses ekstraksi ciri dengan LBP

### Local Binary Pattern (LBP)

LBP merupakan metode ekstraksi ciri untuk menggambarkan tekstur dan struktur gambar yang bekerja pada *8-neighbors pixels* (Simaremare, H. dan Kurniawan, 2016). Alur proses pada metode LBP dapat diliha pada diagram Gambar 2.6

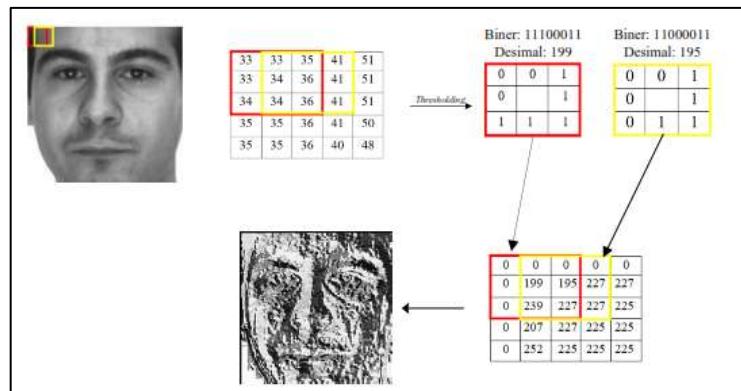


**Gambar 2.6 Alur proses LBP**

Adapun tahap-tahap dari proses LBP yaitu:

1. Sebelumnya, citra RGB yang diperoleh diubah menjadi citra *grayscale* terlebih dahulu.
2. Langkah pertama adalah melakukan inisialisasi terhadap nilai tengah ( $x_c, y_c$ ) dan variabel *nilai* yang berguna untuk menampung nilai yang nantinya menggantikan nilai piksel tengah.
3. Menggunakan kondisi  $I_n \geq (x_c, y_c) = 1$  dan  $I_n < (x_c, y_c) = 0$ .
4. Apabila kondisi tersebut terpenuhi, maka langkah selanjutnya adalah mengambil nilai piksel tengah ( $x_c, y_c$ ) dan piksel tetangga dari  $I_7$  sampai  $I_0$ .
5. Langkah selanjutnya adalah melakukan perbandingan nilai piksel tengah ( $x_c, y_c$ ) dengan piksel tetangga  $I_n$ , dan dilakukan penjumlahan pada variabel *nilai* sesuai dengan bobot masing-masing piksel tetangga.
6. Mengubah semua nilai warna pada piksel tengah ( $x_c, y_c$ ) dengan value pada variabel *nilai*, sehingga mendapat nilai baru  $LBP(x_c, y_c)$ .
7. Selanjutnya dilakukan penjumlahan nilai dan memproses piksel selanjutnya. Setelah semua piksel diproses maka akan terbentuk citra hasil LBP

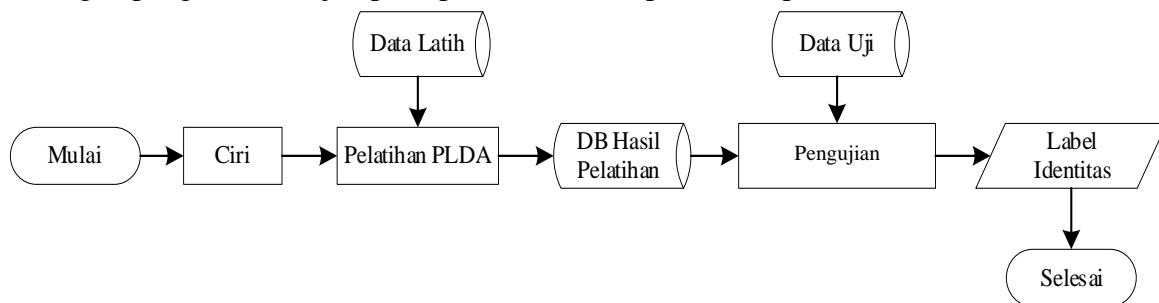
Secara garis besar, algoritma ekstraksi LBP terdiri dari dua langkah, yaitu thresholding dan encoding. Pada langkah thresholding, semua piksel tetangga dalam setiap pola dibandingkan dengan nilai piksel pusat dari pola untuk mengubah nilainya ke nilai biner (0 atau 1). Kemudian pada langkah encoding, angka biner yang diperoleh dari langkah thresholding dikodekan dan diubah menjadi angka desimal yang baru. Contoh perhitungan ekstraksi ciri dengan LBP dapat dilihat pada Gambar 2.7



**Gambar 2.7 Contoh perhitungan ekstraksi ciri dengan LBP**

### 1.2 Recognition (Pengenalan)

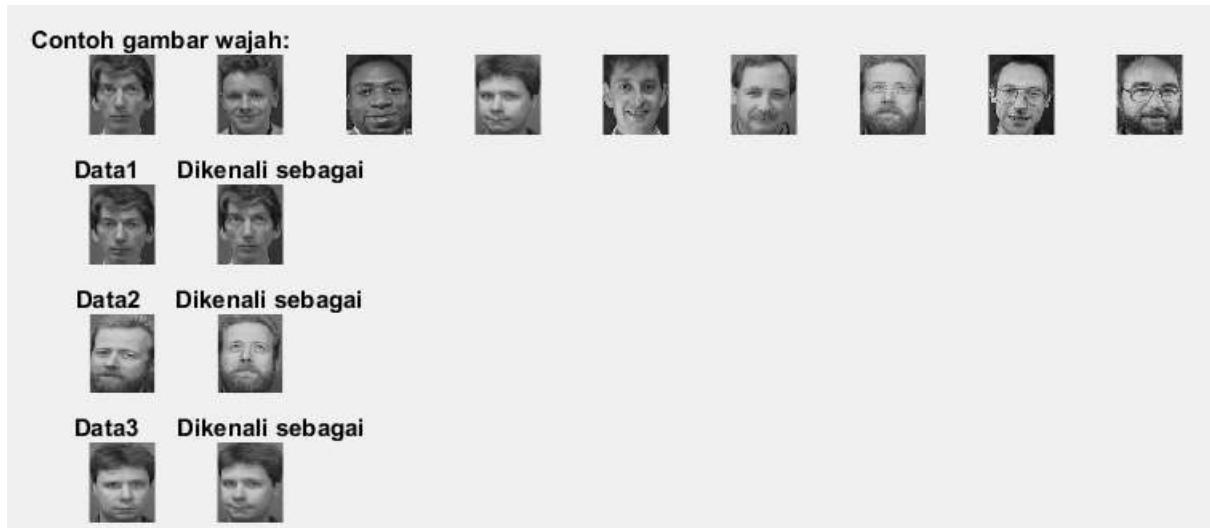
Pengenalan pada penelitian ini menggunakan metode *Euclidean Distance*, untuk mengklasifikasikan pola yang diperoleh dari perhitungan *eigenface* dan LBP (Prakasa, 2016). Rancangan pengenalan wajah pada penelitian ini dapat dilihat pada Gambar 2.8



**Gambar 2.8 Rancangan pengenalan wajah**

### HASIL DAN PEMBAHASAN

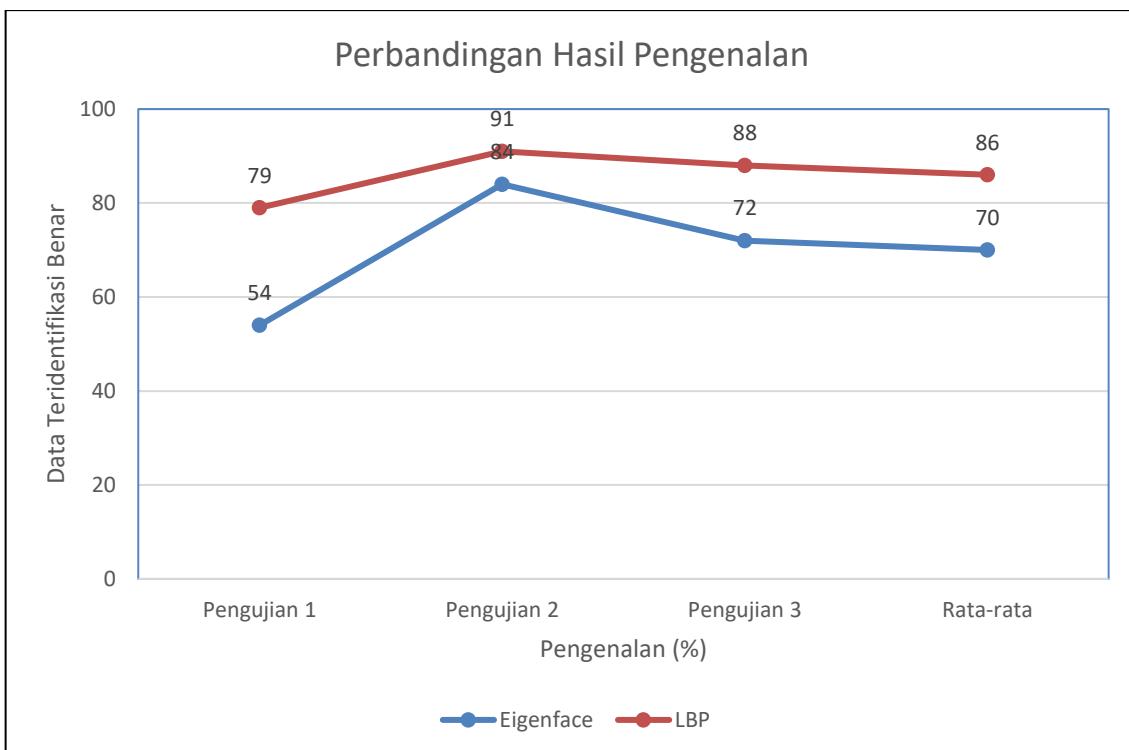
Pada sistem terdapat dua proses utama yaitu prapemrosesan dan pengenalan. Pada tahap prapemrosesan terdiri dari dua proses yaitu *cropping image* dan ekstraksi ciri. Pada *cropping image* dilakukan pemotongan gambar menjadi dua ukuran yaitu ukuran 250x250 dan 500x500 piksel untuk perbandingan pengujian. Lalu pada tahap ekstraksi ciri dilakukan dengan dua metode yaitu *eigenface* dan LBP. Hasil pengenalan dapat dilihat pada Gambar 3.1



**Gambar 3.1 Hasil Pengenalan**

Pada tahap pengenalan diperoleh hasil seperti pada Gambar 2.5, diperoleh citra data latih dengan label contoh gambar wajah, kemudian data uji dengan label Data1, Data2, Data3, dan seterusnya. Lalu hasilnya akan dikenali sebagai wajah yang ada pada database citra latih.

Pengujian dilakukan sebanyak 3 kali, yaitu dengan citra *input* asli ukuran 768×576 piksel, ukuran 250×250 piksel, dan ukuran 500×500 piksel. Hasil perbandingan dari pengenalan *eigenface* dan LBP pada Euclidean distance dapat dilihat pada Gambar 2.6, dan tabel perbandingan dapat dilihat pada Tabel 3.2



**Gambar 3.2 Hasil perbandingan data teridentifikasi benar**

**Tabel 3.1 Hasil perbandingan pengenalan eigenface dan LBP**

Algoritma Ekstraksi	Pengenalan (%)			Rata-rata
	Pengujian 1	Pengujian 2	Pengujian 3	
	768×576	500×500	250×250	
Eigenface	54	84	72	70
LBP	79	91	88	86

Penghitungan akurasi pengujian dilakukan dengan menghitung jumlah hasil pengenalan atau pengujian benar dibagi dengan jumlah data uji lalu dikalikan 100 (Tran et al., 2014).

Penghitungan akurasi dapat dilihat pada persamaan 3.1

$$\text{Akurasi (\%)} = \frac{\text{Hasil pengujian benar}}{\sum \text{Data Uji}} \times 100 \quad (\text{persamaan 3.1})$$

## KESIMPULAN DAN SARAN

### *Kesimpulan*

Berdasarkan penelitian yang telah dilakukan, ciri tekstur yanghasilkan dengan Local Binary Pattern (LBP) lebih optimal dibandingkan ciri tekstur yang dihasilkan dari ekstraksi *Eigenface*. Hal ini dibuktikan dengan perbandingan prosentase akurasi dari pengenalan Euclidean distance dengan LBP 91% pada *input* ukuran 500×500. Sedangkan untuk *Eigenface* sebesar 84% dengan ukuran yg sama.

Dari hasil yang diperoleh dapat disimpulkan bahwa penggunaan ciri tekstur LBP dan *Eigenface* sebagai *input* pada Euclidean Distance dapat mengatasi permasalahan *input* citra RGB (berwarna).

**Saran**

Dalam penelitian ini masih ada beberapa kekurangan yang masih bisa diperbaiki lagi pada penelitian selanjutnya. Beberapa saran untuk penelitian selanjutnya diantaranya adalah penelitian ini masih terbatas pada jumlah data yang kecil, selanjutnya dapat dilakukan percobaan untuk kasus lain dengan data yang lebih banyak dan bervariasi, kemudian juga memperbanyak variasi percobaan dengan ekstraksi ciri lain tentunya selain *eigenface* dan LBP.

**DAFTAR PUSTAKA**

- Mutiara, Q., & Prasetyo, E. (2019). *LBPH pada Sistem Pengenalan Wajah*. 18.
- Prakasa, E. (2016). *Ekstraksi Ciri Tekstur dengan Menggunakan Local Binary Pattern Texture Feature Extraction by Using Local Binary Pattern*. 9(2), 45–48.
- Simaremare, H. dan Kurniawan, A. (2016). Perbandingan Akurasi Pengenalan Wajah Menggunakan Metode LBPH dan Eigenface dalam Mengenali Tiga Wajah Sekaligus secara Real-Time. *Jurnal Sains, Teknologi Dan Industri*, vol.14(1), 66–71.
- Sugiarkha, I. G. R. A., Sudarma, M., & Widhyantara, I. M. O. (2016). Ekstraksi Fitur Warna, Tekstur dan Bentuk untuk Clustered-Based Retrieval of Images (CLUE). *Majalah Ilmiah Teknologi Elektro*, 16(1), 85. <https://doi.org/10.24843/mite.1601.12>
- Tran, C. K., Lee, T. F., Chang, L., & Chao, P. J. (2014). Face description with local binary patterns and local ternary patterns: Improving face recognition performance using similarity feature-based selection and classification algorithm. *Proceedings - 2014 International Symposium on Computer, Consumer and Control, IS3C 2014*, 520–524. <https://doi.org/10.1109/IS3C.2014.141>
- Yushar, I., Purnama, I. P. N., Sutardi, & Aksara, L. B. (2019). Pengenalan Wajah Berbasis Perhitungan Jarak Fitur LBP Menggunakan Euclidean, Manhattan, Chi Square Distance. *Seminar Nasional APTIKOM (SEMNASTIK)*, 386–393.