

# IMPLEMENTASI ALGORITMA PCA PADA PENGENALAN POLA TANDA TANGAN DENGAN MENGUNAKAN BAHASA PEMROGRAMAN PHYTON

*Dadan Nugraha<sup>1</sup>, Panji Novantara<sup>2</sup>, Alman Muhamad<sup>3</sup>*

*<sup>1,2</sup>Universitas Kuningan*

*Jl. Cut Nyak Dhien no.36A Kuningan*

*[dadan.nugraha@uniku.ac.id](mailto:dadan.nugraha@uniku.ac.id)<sup>1</sup>, [panji@uniku.ac.id](mailto:panji@uniku.ac.id)<sup>2</sup>, [alman.muhamad@hotmail.com](mailto:alman.muhamad@hotmail.com)<sup>3</sup>*

## ABSTRAK

Tanda tangan adalah hasil dari proses menulis seseorang yang bersifat khusus sebagai substansi simbolik. Tanda tangan merupakan bentuk yang paling banyak digunakan untuk identifikasi seseorang. Fungsi tanda tangan sendiri adalah untuk pembuktian. Dalam kehidupan sehari-hari, tanda tangan digunakan sebagai identifikasi dari pemilik tanda tangan. Keberadaan tanda tangan dalam sebuah dokumen menyatakan bahwa pihak yang menandatangani, mengetahui, dan menyetujui seluruh isi dari suatu dokumen.

Dengan Metode Principal Component Analysis (PCA) atau dikenal juga dengan nama Karhunen-Loeve transformation (KLT), yang telah dikenal sejak 30 tahun dalam dunia pengenalan pola. PCA memberikan transformasi orthogonal yang disebut dengan "eigenimage" yang mana sebuah image direpresentasikan kedalam bentuk proyeksi linier searah dengan eigenimage. Secara praktis eigenimage ini dibangun dari sekumpulan image training yang diambil dari berbagai obyek/kelas dan dibantu dengan metode Euclidean distance yang menyatakan jarak antara dua titik dalam suatu ruang. Dimana jarak antara dua titik adalah garis terpendek diantara semua garis yang menghubungkan kedua titik tersebut, kita bisa mengidentifikasi suatu tanda tangan berdasarkan data yang dimiliki. Dengan menggunakan bahasa pemrograman python dalam pengolahan matematisnya akan lebih mudah dalam penerapannya.

Kata Kunci : Tanda Tangan, PCA, Eigenimage, Euclidean Distance dan python.

## 1. Pendahuluan

### 1.1 Latar Belakang Masalah.

Seiring dengan perkembangan zaman, secara tidak disadari diikuti oleh perkembangan teknologi dan informasi, yang mendorong manusia untuk mengatasi berbagai masalah yang timbul disekitarnya dan mengejar perkembangan tersebut dengan terus berusaha menciptakan pemikiran-pemikiran dan karya-karya yang inovatif dan kreatif. Salah satu dari perkembangan teknologi informasi yaitu adanya sistem yang dapat mengidentifikasi suatu tanda tangan seseorang.

Tanda tangan adalah hasil dari proses menulis seseorang yang bersifat khusus sebagai substansi simbolik. Tanda tangan merupakan bentuk yang paling banyak digunakan untuk identifikasi seseorang. Fungsi tanda tangan sendiri adalah untuk pembuktian.

Dalam kehidupan sehari-hari, tanda tangan sering digunakan dalam kegiatan belajar mengajar, proyek, dokumen, perbankan, pengambilan struk gaji ataupun pengambilan bonus dan lain-lain. Keberadaan tanda tangan dalam sebuah dokumen menyatakan bahwa pihak yang

menandatangani, mengetahui, dan menyetujui seluruh isi dari suatu dokumen. Pada kegiatan keuangan dalam suatu instansi atau perusahaan yaitu pada pengambilan gaji maupun bonus di suatu perusahaan sering dibutuhkan membubuhkan tanda tangan sebagai bukti pengambilan. Kadang dalam kegiatan tersebut sering menemui tanda tangan yang bukan pemiliknya sehingga akan menjadi masalah kalau yang memiliki tanda tangan tersebut tidak merasa menandatangani dan tidak mendapatkan haknya.

Dengan Metode Principal Component Analysis (PCA) atau dikenal juga dengan nama Karhunen-Loeve transformation (KLT), yang telah dikenal sejak 30 tahun dalam dunia pengenalan pola. PCA memberikan transformasi orthogonal yang disebut dengan "eigenimage" yang mana sebuah image direpresentasikan kedalam bentuk proyeksi linier searah dengan eigenimage. Secara praktis eigenimage ini dibangun dari sekumpulan image training yang diambil dari berbagai obyek/kelas dan dibantu dengan metode Euclidean distance yang menyatakan jarak antara dua titik dalam suatu ruang. Dimana jarak antara dua titik adalah garis terpendek diantara semua garis yang

menghubungkan kedua titik tersebut, kita bisa mengidentifikasi suatu tanda tangan berdasarkan data yang dimiliki.

Dengan adanya permasalahan-permasalahan diatas, penulis mengusulkan untuk membuat sebuah penelitian yaitu "implementasi algoritma PCA pada pengenalan pola tanda tangan dengan menggunakan bahasa pemrograman python".

## 2. Landasan Teori

### 2.1 Pengenalan Pola.

Pengenalan pola adalah salah satu cabang dari bidang kecerdasan buatan. Ada beberapa definisi yang berbeda tentang pengenalan pola, diantaranya :

1. Penentuan suatu objek fisik atau kejadian ke dalam salah satu atau beberapa kategori. (Duda dan Hart)
2. Ilmu pengetahuan yang menitik-beratkan pada deskripsi dan klasifikasi (pengenalan) dari suatu pengukuran. (Schalkoff)
3. Proses untuk memberikan nama  $\omega$  untuk pengamatan  $x$ . (Schurmann)

Berdasarkan beberapa definisi di atas, pengenalan pola bisa didefinisikan sebagai cabang kecerdasan buatan yang menitik-beratkan pada metode pengklasifikasian objek ke dalam klas-klas tertentu untuk menyelesaikan masalah tertentu. (Hanif al Fatta, 2009 :5)

Sistem pengenalan pola dasar terdiri dari :

1. Sensor  
Sensor digunakan untuk menangkap objek yang ciri atau fiturnya akan diekstraksi.
2. Pre-processing (Pra-pengolahan)  
Mekanisme pengolahan objek yang ditangkap oleh sensor biasanya digunakan untuk mengurangi kompleksitas ciri yang akan dipakai untuk proses klasifikasi. Adapun beberapa teknik pre-processing diantaranya adalah sistem ruang warna, grayscale (keabuan), thresholding (pengembangan), normalisasi intensitas, histogram equalization (equalisasi histogram), contrast stretching (peregangan kontras), tapis gaussian dan normalisasi kemiringan.
3. Pencari Fitur  
Bagian ini digunakan untuk mengekstraksi ciri yang telah melalui tahapan preprocessing untuk memisahkan dari fitur-fitur pada objek yang tidak diperlukan dalam proses klasifikasi. Beberapa teknik pemisahan ciri diantaranya adalah deteksi garis, deteksi tepi, blok, moment, histogram, co-occurrence, ruang eigen, spektrum fourier, alihragam wavelet, dimensi fraktal, dan metode simpul proyeksi.
4. Metode Pemilah / pencocokan  
Pada tahapan ini proses klasifikasi dilakukan dengan menggunakan klasifikasi tertentu. Hasil dari tahapan ini adalah klasifikasi dari ojek ke dalam kriteria-kriteria yang telah ditentukan. Adapun beberapa metode pemilah/pencocokan diantaranya adalah : Euclidean Distance, City

Block Distance, Chebyshev Distance, Minkowski Distance, Canberra Distance, Bray Curtis Distance, AngularSeparation, Correlation Coefficient, Hamming Distance dan Dynamic Time Warping(DTW).

### 2.2 Pengolahan Citra

Pengolahan Citra merupakan suatu sistem dimana proses dilakukan dengan masukan berupa citra dan hasilnya juga berupa citra. Proses ini merupakan proses yang dilakukan terhadap citra sebelum proses ekstrasi ciri. Terdapat dua hal mendasar dalam memahami pembentukan citra yaitu :

Geometri formasi citra yang menentukan lokasi suatu titik dalam pemandangan yang diproyeksikan pada bidang citra.

Fisik cahaya yang menentukan kecerahan suatu titik pada bidang citra sebagai fungsi pencahayaan pemandangan dan sifat-sifat permukaan.

Haruslah ada penghubung antara notasi matematika untuk mengembangkan algoritma pengolahan citra dan notasi algoritma yang digunakan untuk pembuatan program komputer, dan dalam komputer sisipan dalam sistem penyimpanan memori dua dimensi yang disebut larik (array). (Achmad Basuki dkk, 2005:1)

Tujuan utama dari pengolahan citra ini adalah untuk memperbaiki data citra yang menekan distorsi yang tidak diinginkan pada citra, atau meningkatkan beberapa fitur citra yang relevan untuk diproses dan dianalisis lebih lanjut. Pada proses ini citra akan dinormalisasikan agar citra lebih siap diolah pada jaringan. Kualitas hasil ciri yang dihasilkan tergantung pada pengolahan citra ini. Proses ini memiliki tahapan-tahapan tertentu dengan berbagai metode dan perhitungan sesuai dengan kebutuhan penelitian. Tahapan-tahapan pengolahan citra yang dilakukan dalam penelitian sebagai berikut :

#### RGB to Grayscale

RGB to Grayscale merupakan proses konversi dari citra berwarna menjadi citra keabuan atau sering disebut dengan citra hitam putih. Proses perubahan warna ini dilakukan karena citra dalam bentuk grayscale lebih mudah dianalisa apabila dibandingkan dengan citra berwarna. Rumus untuk mengkonversi nilai RGB ke grauscale pada umumnya adalah sebagai berikut :

$$v=(R+G+B)/3 \quad \dots\dots\dots (2.1)$$

v=nilai grayscale

G=nilai warna hijau

R=nialai warna merah

B=nilai warna biru

#### Feature extraction

Feature extraction merupakan pengambilan ciri dari suatu bentuk yang nantinya nilai yang didapatkan akan dianalisis untuk proses selanjutnya. Dalam penelitian ini penulis menggunakan metode eignface PCA (Principal Component Analysis).

### 2.3 Algoritma Principal Component Analysis (PCA)

Analisis Komponen Utama (Principal Component Analysis) atau PCA adalah suatu metode yang melibatkan prosedur matematika yang mengubah dan mentransformasikan sejumlah variabel yang berkorelasi menjadi sejumlah kecil variabel yang tidak berkorelasi, tanpa menghilangkan informasi penting di dalamnya.

Sejumlah citra dari dua dimensi dari setiap objek tiga dimensi yang akan dikenali, dikumpulkan untuk mewakili objek tersebut sebagai citra acuan. Dari sekumpulan citra acuan tersebut, kemudian akan dilakukan ekstraksi ciri untuk memperoleh informasi karakteristik (ciri) dari objek tersebut. Hasil ekstraksi ciri digunakan untuk proses pengenalan objek.

Sebuah citra tanda tangan dapat dilihat sebagai sebuah vektor ruang yang jika panjangnya  $w$  dan  $h$  pixel maka jumlah komponen dari vektor ini adalah  $w \cdot h$  (setiap pixel dikodekan oleh satu komponen vektor). Vektor eigen tersebut berada dalam ruang eigen yang merupakan ruang dari semua gambar. PCA digunakan untuk mereduksi dimensi dari sekumpulan atau ruang gambar, sehingga bisa mendapatkan ruang eigen yang lebih baik. (Sutojo, T., dkk., 2010:290)

Berikut langkah-langkah pencarian eigenface dan fitur PCA-nya:

1. Sebagai contoh terdapat dua citra  $4 \times 4$  piksel yang terdapat dalam database yang diberikan simbol  $a$ . Dari kumpulan citra database  $a = [a_1 \ a_2]$ , lalu cari nilai rata-rata dari tiap baris yang ada dalam matrik  $a$  setelah itu kurangi tiap nilai yang ada dalam matrik  $a$  dengan nilai rata-rata yang sesuai dengan barisnya hasil pengurangannya tersebut dimasukkan kedalam matrik  $A$ .
2. Setelah mendapatkan matrik  $A$  lalu cari matrik kovarian  $L$  dengan cara mengalikan matrik  $A$  dengan transposenya dengan rumus :  
$$L = A^T \cdot A \dots\dots\dots (2.2)$$
3. Selanjutnya mencari nilai eigen (NE) dan vektor eigen (VE) dari matrik kovarian  $L$ .
4. Setelah itu hilangkan nilai eigen yang nilainya lebih kecil dari 1 lalu akan dicari nilai eigenfaces dengan rumus :  
$$\text{Eigenfaces} = A \cdot \text{NE} \dots\dots\dots (2.3)$$
5. Lalu cari nilai PCA nya dengan rumus :  
$$\text{PCA} = \text{Eigenfaces} \cdot A \dots\dots\dots (2.4)$$

### 2.4 Metode Euclidean Distance

Pengenalan pola (pattern recognition) adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran

kuantitatif fitur (ciri) atau sifat utama dari suatu objek. (Darma Putra, 2010:303)

Setelah didapatkan vektor ciri dari citra karakter melalui proses ekstraksi ciri, maka proses selanjutnya adalah membandingkan vektor ciri dari citra uji dengan vektor ciri dari citra latih. Perbandingan tersebut dapat dilakukan dengan cara menghitung jarak euclidean (euclidean distance), yang merupakan selisih antara 2 vektor tersebut. Jarak euclidean adalah akar dari jumlah selisih kuadrat antara 2 vektor.

Adapun rumus mencari jarak euclidean adalah sebagai berikut :

$$\text{dist}(i, k) = \sqrt{\sum_{i=j}^D (i_j - k_j)^2} \quad (2.5)$$

Dengan  $\text{dist}(i, k)$  adalah jarak euclidean antara vektor  $i$  dan vektor  $k$  ;

$i_j$  adalah komponen ke  $j$  dari vektor  $i$  ;

$k_j$  adalah komponen ke  $j$  dari vektor  $k$  ;

$D$  adalah jumlah komponen dari vektor  $i$  dan vektor  $k$ .

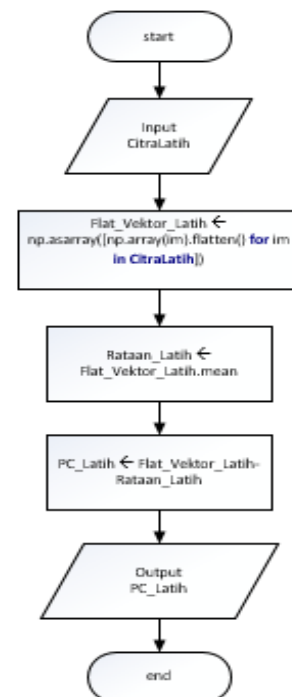
Dari hasil perhitungan jarak euclidean tersebut dapat ditentukan suatu citra tandatangan adalah mirip bila memiliki jarak yang paling dekat. (Muhammad Jatra, 2007).

## 3. Perancangan Sistem

Analisis perancangan sistem menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem. Analisis yang dilakukan dimodelkan dengan menggunakan UML (Unified Modeling Language).

### 3.1 Flowchart Pelatihan Citra dan Pengujian citra

#### 3.1.1 Flowchart Pelatihan Citra

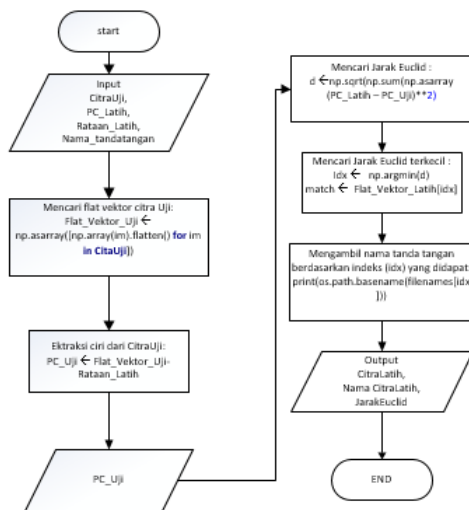


Gambar Flowchart Pelatihan Citra

Berikut langkah langkah dari flowchart Pelatihan Citra :

1. Masukan beberapa citra latih
2. Set setiap citra latih buat menjadi vektor satu kolom dengan menggunakan rumus  $Flat\_Vektor\_Latih = np.asarray([np.array(CitraLatih).flatten() \text{ for } im \text{ in } CitraLatih])$ .
3. Set nilai rata-rata dari citra latih dengan rumus  $Rataan\_Latih = Flat\_Vektor\_Latih.mean$
4. Set ekstraksi ciri dari citra latih dengan menggunakan rumus  $PC\_Latih = Flat\_Vektor\_Latih - Rataan\_Latih$

3.1.2 Flowchart Pengenalan Citra Uji



Gambar Flowchart Pengenalan Citra Uji berikut langkah - langkah dari flowchart Pengenalan Citra Uji :

1. Masukan beberapa citra Uji, PC\_Latih, Rataan\_Latih, Nama Tandatangan(Nama CitraLatih)
2. Set citra uji buat menjadi vektor satu kolom dengan menggunakan rumus  $Flat\_Vektor\_Latih = np.array(CitraLatih).flatten()$ .
3. Set ekstraksi ciri dari citra latih dengan menggunakan rumus  $PC\_Uji = Flat\_Vektor\_Uji - Rataan\_Latih$ .
4. Set Jarak euclid dengan menggunakan rumus  $d = np.sqrt(np.sum(np.asarray(PC\_Latih - PC\_Uji)**2))$ .
5. Set jarak euclid terkecil dengan menggunakan rumus  $Idx = np.argmin(d)$  dan  $match = Flat\_Vektor\_Latih[idx]$
6. Set nama tandatangan berdasarkan indeks (idx) yang didapat.
7. Tampilkan CitraLatih beserta nama CitraLatih berdasarkan nilai indeks(idx)

yang didapat dan tampilkan nilai Euclidnya .

8. Selesai.

4. Pembahasan

4.1 Implementasi Aplikasi

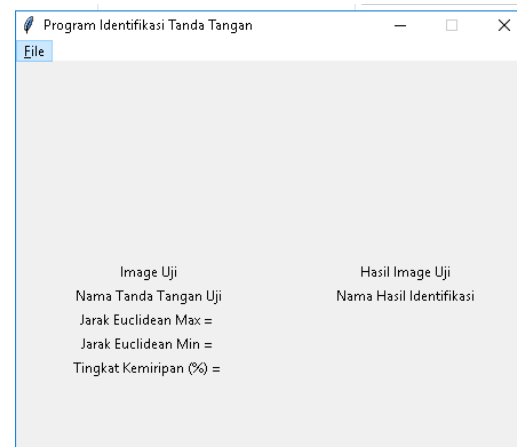
Implementasi yang dilakukan antara lain adalah pembuatan kode program untuk membangun komponen-komponen pokok sebuah sistem berdasarkan desain yang sudah dibuat, menerapkan hasil perancangan *interface*, dan sebagainya.

4.1.1 Implementasi Desain Interface

Dalam mengimplementasikan sistem dengan menggunakan editor JetBrains PyCharm Community Edition 2016.1.2. Berdasarkan rancangan *interface* yang telah ada pada bab III sebelumnya maka dihasilkan tampilan sebagai berikut:

1. Halaman Utama

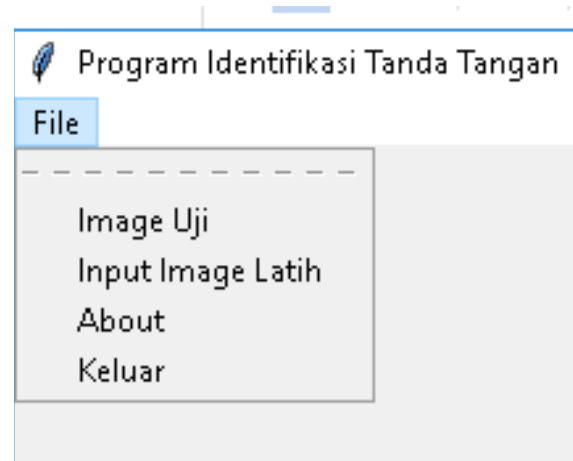
Tampilan menu seperti pada gambar 4.1 dibawah ini :



Gambar 4.1 Halaman Utama

2. Menu File

Gambar 4.2 merupakan tampilan ketika kita menekan tombol file dan terdapat beberapa menu.

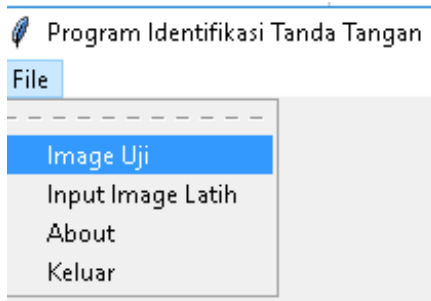


Gambar 4.2 Tampilan Menu File

Gambar 4.2 merupakan halaman utama ketika kita menekan tombol file, terlihat jelas ada 4 buah tombol yang masing masing tombol mengarah pada halaman tertentu sesuai dengan rancangan pada bab III.

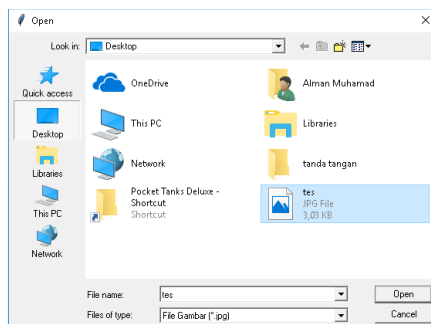
3. Menu Image Uji

Setelah menekan tombol file, user memilih menu Image Uji untuk menguji suatu gambar tanda tangan:



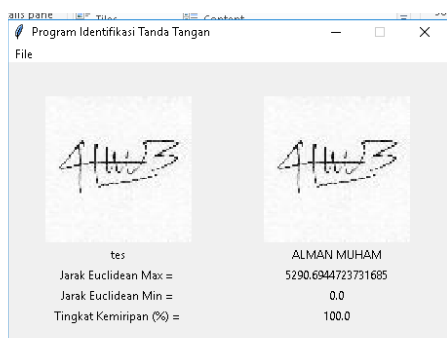
Gambar 4.3 Tampilan Memilih Image Uji

Setelah menekan tombol Image Uji sistem akan menampilkan form open file, dimana user diminta untuk memilih file gambar yang berektensi .JPEG yang akan dibandingkan (dicocokkan) dengan Image latih yang telah ada.



Gambar 4.4 Tampilan Pemilihan File Image Uji

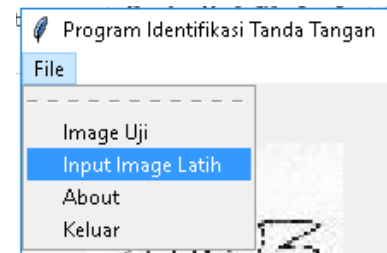
Setelah user memilih file image yang akan diuji, kemudian user menekan tombol open untuk diujikan dan sistem akan menampilkan hasil pengujian seperti berikut :



Gambar 4.5 Tampilan File Image Uji

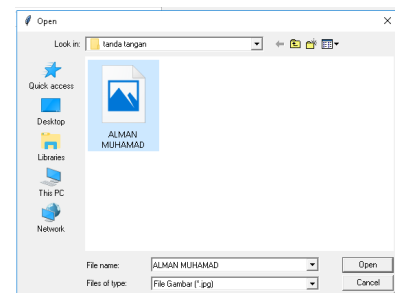
4. Menu Input Image Latih

Setelah menekan tombol file, user memilih menu Input Image Latih untuk memasukkan suatu gambar tanda tangan yang akan dijadikan master (Image Latih):



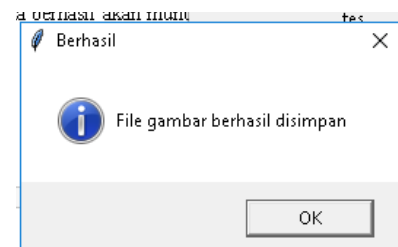
Gambar 4.6 Tampilan Memilih Input Image Latih

Setelah menekan tombol Input Image Latih sistem akan menampilkan form open file, dimana user diminta untuk memilih file gambar yang berektensi .JPEG yang akan disimpan oleh sistem pada folder yang telah ditentukan.



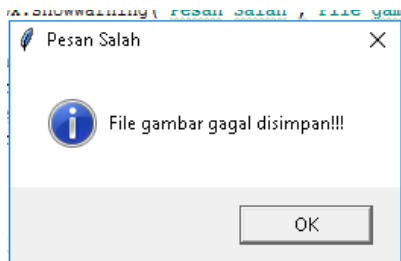
Gambar 4.7 Tampilan Pemilihan File Input Image Latih

Setelah user memilih file image yang akan diinput, kemudian user menekan tombol open untuk disimpan di folder yang telah ditentukan. Jika berhasil akan muncul pesan seperti gambar berikut :



Gambar 5.8 Tampilan Pesan Berhasil Disimpan

Jika gagal maka akan muncul pesan teks seperti gambar berikut :



Gambar 4.9 Tampilan Pesan Gagal Disimpan

## 4.2 Analisis dan Evaluasi Hasil

Setelah dilakukan penerapan algoritma *Principle Component Analysis* (PCA) untuk mengekstraksi ciri dan algoritma *Euclidean Distance* untuk menghitung kesamaan 2 vektor pada citra tanda tangan secara keseluruhan cukup sederhana. Pelatihan citra direpresentasikan dalam sebuah vektor flat (gabungan vektor) dan digabung bersama-sama menjadi sebuah matrik tunggal. Ruang eigen dari masing-masing citra kemudian diekstraksi dan disimpan dalam file temporary. Image uji yang masuk didefinisikan juga nilai eigenfacenya dan dibandingkan dengan eigenfaces dari image latih dalam file temporary.

Untuk memperoleh tingkat kemiripan yang mendekati berdasarkan nilai euclidean distance dipengaruhi oleh beberapa hal diantaranya adalah alat untuk mengambil citra uji maupun citra latih misal scan ataupun menggunakan kamera. Sebaiknya agar memperoleh tingkat kemiripan yang tinggi gunakanlah satu alat saja untuk mengambil citra uji maupun citra latih dan sebaiknya menggunakan scanner, agar tingkat pencahayaannya akan sama dibandingkan dengan penggunaan kamera dimana tingkat pencahayaan dari luar sangat mempengaruhi terhadap nilai citra yang akan diambil. Selain itu tipe ekstensi dari file pun harus diperhatikan, karena tiap tipe file berbeda nilai tiap pixelnya walaupun gambar terlihat sama. Oleh karena itu pergunakanlah satu ekstensi file yaitu .JPEG dimana tipe file tersebut sama dihasilkan oleh alat yang digunakan untuk mengambil citra yaitu scanner.

## 5. Penutup

### 5.1 Kesimpulan

Berdasarkan uraian dan hasil pembahasan yang dilakukan dari bab-bab sebelumnya dan pembangunan aplikasi ini, dapat diambil kesimpulan kesimpulan sebagai berikut :

1. Algoritma *Principle Component Analysis* (PCA) diterapkan pada aplikasi identifikasi tandatangan dengan menggunakan bahasa pemrograman *python* berfungsi untuk mengekstraksi (menyederhanakan) ciri baik untuk image latih maupun image uji dimana hasilnya berupa vektor tunggal.
2. Untuk melakukan proses identifikasi atau pencocokan digunakan algoritma *Euclidean Distance* dimana hasil ekstraksi ciri atau vektor

fitur citra uji dibandingkan dengan ekstraksi ciri atau vektor fitur pelatihan citra, dimana semakin kecil nilai *Euclidean*-nya maka semakin mirip kedua vektor fitur yang dicocokkan.

### 5.2 Saran

Saran-saran yang bisa penulis berikan untuk pengembangan lebih lanjut adalah sebagai berikut :

1. Untuk memperoleh hasil pencocokan yang lebih baik, perbanyaklah image uji latih agar tingkat kemiripan antara image latih dengan image uji lebih baik.
2. Perlu adanya sebuah metode untuk pemotongan image capture secara otomatis (cropping), sehingga untuk selanjutnya sistem berjalan lebih baik dari sebelumnya.
3. Mencoba algoritma lain untuk proses pencocokan *image* sehingga didapat algoritma yang lebih efisien dari algoritma sebelumnya.
4. Mengubah tampilan *aplikasi* menjadi lebih Menarik.

### DAFTAR PUSTAKA

- [1] Hanif al Fatta. *Rekayasa Sistem Pengenalan Wajah*. 2009. Yogyakarta : Penerbit Andi Yogyakarta.
- [2] Kharisman Kholid Hudaya. *Pemrograman Desktop Database Python-MySQL dengan BOA Constructor*. 2013. Yogyakarta : Penerbit Andi Yogyakarta.
- [3] Fajar Astuti Hermawati. *Pengolahan Citra Digital Konsep & Teori*. 2013. Yogyakarta : Penerbit Andi Yogyakarta.
- [4] Tika Sartika. *Pengenalan Wajah Menggunakan Metode 2 Dimension Principal Component Analysis dua tahap*. (2013). Fakultas Ilmu Komputer Universitas Kuningan.
- [5] Riza Firdaus Ardiansyah. *Jurnal Pengenalan Pola Tanda Tangan Dengan Menggunakan Metode Principal Component Analysis (PCA)*. Fakultas Ilmu Komputer Universitas Dian Nuswantoro.
- [6] A.S Rosa, Shalahuddin M.(2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung. Penerbit: Informatika.
- [7] Suryana, Taryana, (2007), *Metode RUP*, STMIK LIKMI Bandung.
- [8] Basuki, Achmad, dkk. (2005). *Pengolahan Citra Digital menggunakan Visual Basic*. Yogyakarta : Graha Ilmu.
- [9] Ridwan Fadjat Septian. (2013). *Belajar Pemrograman Python Dasar*. Bandung : POSS – UPI.
- [10] Darma Putra. *Sistem Biometrika*. 2009. Yogyakarta : Penerbit Andi Yogyakarta.