

Identifikasi Dan Pengujian keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma *Eigenface* dan *Fisherface*



TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar
Magister Ilmu Komputer (M.Kom)

AHMAD FAUZI

14001117

**PROGRAM PASCASARJANA MAGISTER ILMU KOMPUTER
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER NUSA
MANDIRI
JAKARTA
2016**

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : Ahmad Fazui
NIM : 14001117
Program Studi : Magister Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *e-Business*

Dengan ini menyatakan bahwa tesis yang telah saya buat dengan judul: “Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma Eigenface Dan Fisherface” adalah hasil karya sendiri, dan semua sumber baik yang kutip maupun yang dirujuk telah saya nyatakan dengan benar dan tesis belum pernah diterbitkan atau dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa tesis yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri dicabut/dibatalkan.

Jakarta, 9 Februari 2016
Yang menyatakan,



Ahmad Fauzi

HALAMAN PENGESAHAN

Tesis ini diajukan oleh :

Nama : Ahmad Fauzi

NIM : 14001117

Program Studi : Magister Ilmu Komputer

Jenjang : Strata Dua (S2)

Konsentrasi : *e-Business*

Judul Tesis : Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma Eigenface Dan Fisherface

Telah berhasil dipertahankan dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Ilmu Komputer (M.Kom) pada Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (STMIK Nusa Mandiri).

Jakarta, 9 Februari 2016

Pascasarjana Magister Ilmu Komputer

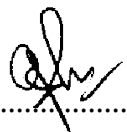
STMIK Nusa Mandiri

Direktur

Prof. Dr. Ir. R. Eko Indrajit, M.Sc, MBA

DEWAN PENGUJI

Penguji I : Dr.Sularso Budilaksono, M.Kom



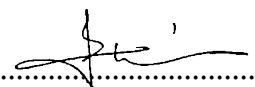
.....

Penguji II : Dr. Windu Gata, M.Kom



.....

Penguji III /
Pembimbing : Dr. Sfenrianto, S.kom, M.kom



.....

KATA PENGANTAR

Puji syukur alhamdulillah, penulis panjatkan kehadiran Allah, SWT, yang telah melimpahkan rahmat dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan tesis ini tepat pada waktunya. Dimana tesis ini penulis sajikan dalam bentuk buku yang sederhana. Adapun judul tesis, yang penulis ambil sebagai berikut **“Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma Eigenface Dan Fisherface”**.

Tujuan penulisan tesis ini dibuat sebagai salah satu untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (PPs MIK STMIK Nusa Mandiri).

Tesis ini diambil berdasarkan hasil penelitian mengenai identifikasi dan pengenalan wajah manusia melalui media digital foto. Penulis juga lakukan mencari dan menganalisa berbagai macam sumber referensi, baik dalam bentuk jurnal ilmiah, buku-buku literatur, *internet*, dll yang terkait dengan pembahasan pada tesis ini.

Penulis menyadari bahwa tanpa bimbingan dan dukungan dari semua pihak dalam pembuatan tesis ini, maka penulis tidak dapat menyelesaikan tesis ini tepat pada waktunya. Untuk itu izinkanlah penulis kesempatan ini untuk mengucapkan ucapan terima kasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Ir. R. Eko Indrjit. M.Se. MBA selaku Direktur Pascasarjana Magister Ilmu komputer STMIK Nusa Mandiri Jakarta.
2. Bapak Dr. Sfenrianto, S.kom, M.kom selaku pembimbing tesis yang telah menyediakan waktu, pikiran dan tenaga dalam membimbing penulis dalam menyelesaikan tesis ini.
2. Bapak dan Ibu instruktur yang bersedia dijadikan bahan penelitian.
3. Orang tua tercinta yang telah memberikan dukungan material dan moral kepada penulis.
4. Seluruh staf pengajar (dosen) PPs MIK STMIK Nusa Mandiri yang telah memberikan pelajaran yang berarti bagi penulis selama menempuh studi.

5. Seluruh staf dan karyawan PPs MIK STMIK Nusa Mandiri yang telah melayani penulis dengan baik selama kuliah.
6. Calon istriku yang tercinta Dessi Intan P yang telah banyak membantu dan mendukung atas pengerjaan tesis ini.
7. Sahabat-sahabat yang telah memberi kesempatan waktu untuk menyelesaikan tesis.

Serta semua pihak yang terlalu banyak untuk penulis sebutkan satu persatu sehingga terwujudnya penulisan tesis ini. Penulis menyadari bahwa penulisan tesis ini masih jauh sekali dari sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan karya ilmiah yang penulis hasilkan untuk yang akan datang.

Akhir kata semoga tesis ini dapat bermanfaat bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Jakarta, 9 Februari 2016



Ahmad Fauzi

Penulis

**SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH
UNTUK KEPENTINGAN AKADEMIS**

Yang bertanda tangan di bawah ini, saya :

Nama : Ahmad Fauzi
NIM : 14001117
Program Studi : Magsiter Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *e-Business*
Jenis Karya : Tesis

Demi pengembangan ilmu pengetahuan, dengan ini menyetujui untuk memberikan ijin kepada pihak Program Pascasarjana Magister Ilmu Komputer Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri (STMIK Nusa Mandiri) **Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalti-Free Right*)** atas karya ilmiah kami yang berjudul : “Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma Eigenface Dan Fisherface:

Dengan **Hak Bebas Royalti Non-Eksklusif** ini pihak STMIK Nusa Mandiri berhak menyimpan, mengalih-media atau *bentuk*-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya dan menampilkan atau mempublikasikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari kami selama tetap mencantumkan nama kami sebagai penulis/pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak STMIK Nusa Mandiri, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 9 Februari 2016
Yang menyatakan,



Ahmad Fauzi

ABSTRAK

Nama : Ahmad Fauzi
NIM : 14001117
Program Studi : Magsiter Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *e-Business*
Judul : Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah
Manusia Melalui Foto Dengan Menggunakan Algoritma
Eigenface Dan Fisherface

Informasi merupakan suatu hal yang dapat kita temukan kapan dan dimana saja, untuk hal apa sebuah informasi kita manfaatkan, tentunya setiap orang memiliki kepentingan yang berbeda dalam sebuah mencari atau mendapatkan sebuah informasi, dapat bermanfaat ataupun informasi tersebut tidak ada manfaatnya sedikitpun dan semuanya tergantung dengan kepentingan seorang terhadap sebuah informasi.

Salah satu yang dapat dijadikan informasi adalah wajah manusia, wajah manusia memiliki pola bentuk yang tidak sama antara individu bisa dibilang sebuah wajah adalah identitas diri manusia yang dapat kita kenali, dengan melihat wajah tentunya kita dapat memanggil bahkan mengetahui identitas diri dari seorang tersebut.

Karena wajah merupakan identitas diri yang sangat kuat bahkan sangat melekat maka dapat dijadikan sebagai sumber informasi untuk identitas diri seorang dengan pengujian terhadap wajah manusia dengan menggunakan media foto kita dapat mencari dan mengetahui identitas dari wajah yang terdapat pada foto tersebut.

Kata kunci:

Informasi, identifikasi wajah manusia

ABSTRACT

Name : Ahmad Fauzi
NIM : 14001117
Study Program : Master of Computer Science
Level : Strata Dua (S2)
Concentration : e-Business
Title : Identification and Testing Accuracy Human Face Recognition Algorithms Through Photos Using Eigenface And Fisherface

Information is a matter that can be found everywhere and anywhere, for what piece of information we use, of course, everyone has different interests in a seek or obtain an information, can be beneficial or the information is no benefit at all and it all depends on the interest one against an information.

One that can be used as information is a human face, the human face has a pattern shape is not the same between individuals arguably a human face is a self-identity that we can recognize, by looking at the faces of course, we can call even know the identity of a.

Because the face is a very strong self-identity even very attached to it can be used as a source of information for an identity by testing against human faces using photo media we can find and know the identity of the face contained in the image.

Keywords:
Information, identification of human faces

DAFTAR ISI

	Halaman
HALAMAN SAMPUL.....	i
HALAMAN JUDUL.....	ii
HALAMAN PERNYATAAN ORISINALITAS.....	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS.....	vii
ABSTRAK.....	viii
ABSTRCT.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xiii
BAB 1. PENDAHULUAN.....	1
1.1. Latar Belakang Penulisan.....	1
1.2. Identifikasi Masalah.....	2
1.3. Tujuan Penelitian.....	2
1.4. Ruang Lingkup Penelitian.....	3
1.5. Hipotesis.....	3
1.6. Sistematika Penulisan.....	4
BAB 2. LANDASAN TEORI.....	5
2.1. Definisi Citra Digital.....	5
2.1.1. Sistem Perekam Citra.....	6
2.1.2. Element fungsi dasar sistem pengolahan citra.....	6
2.2. Dasar Citra Digital.....	7
2.2.1. Image sampling & quantization.....	7
2.2.2. Representasi citra digital.....	7
2.2.3. Resolusi spasial & Gray level.....	8
2.2.4. Zooming & shrinking image.....	9
2.2.5. Beberapa dasar hubungan antara-piksel.....	10
2.3. Pengenalan Pola.....	12
2.3.1. Komponen sistem pengenalan pola.....	12
2.3.2. Fitur.....	13
2.3.3. Pola.....	13
2.3.4. Pemilah.....	13
2.3.5. Pendekatan pengenalan pola.....	14
2.4. Kompresi Citra.....	16
2.4.1. Dasar Kompresi citra.....	17
2.4.2. Coding Redudancy.....	17
2.4.3. Psychovisual Redundancy.....	18
2.4.4. Segmentasi citra.....	19
2.4.5. Deteksi tepi dengan oprator sobel.....	20
2.5. Deteksi tepi dengan canny oprator.....	21
2.5.1. Local Prossesing.....	22
2.5.2. Global Processing dengan hough transform.....	22

2.5.3. Pemrosesan global melalui teori graph.....	23
2.6. Pengertian dasar thresholding.....	25
2.6.1. Basic global thresholding.....	25
2.6.2. Basic adapter thresholding.....	26
2.7. Algoritma Eigenface dan Fisherface.....	27
BAB 3. METODE PENELITIAN.....	32
3.1. Pengenalan dan identifikasi wajah.....	32
3.1.1. Pengambilan foto wajah.....	32
3.1.2. Proses identifikasi wajah dengan algoritma Eigenface.....	33
3.1.3. Proses identifikasi pengenalan wajah dengan algoritma Fisherface.....	35
BAB 4. HASIL PENELITIAN DAN PEMBAHASAN.....	37
4.1. Data foto identifikasi wajah manusia.....	37
4.2. Metode algoritma Eigenface dan Fisherface untuk identifikasi Wajah.....	47
4.2.1. Algoritma Eigenface untuk pengenalan wajah.....	47
4.2.2. Algoritma Fisherface.....	51
4.3. Percobaan pengenalan wajah dalam bentuk tabel.....	57
4.4. Akurasi pengenalan wajah individu.....	69
4.5. Akurasi pengenalan wajah secara global.....	71
BAB 5. PENUTUP.....	75
5.1. Kesimpulan.....	75
5.2. Saran.....	76
DAFTAR REFERENSI.....	77
LAMPIRAN.....	78

DAFTAR TABEL

	Halaman
Tabel 2.1	Tabel vertikal dan horizaontal 9
Tabel 2.2	Tabel Diagonal 9
Tabel 4.1	Tabel dari 7 orang dengan 2 pose 52
Tabel 4.2	Tabel Beda Kuadrat dari masing-masing pelatihan terhadap Citra Uji 53
Tabel 4.3	Hasil Penjumlahan Beda Kuadrat Dari Masing-masing Pelatihan terhadap Citra Uji 54
Tabel 4.4	Hasil Akhir Perhitungan Tingkat Kemiripan Menggunakan Euclidean 55
Tabel 4.5	Uji coba foto formal Andika 56
Tabel 4.6	Uji coba foto aksesoris topi Andika 56
Tabel 4.7	Uji coba foto aksesoris kacamata Andika 56
Tabel 4.8	Uji coba foto aksesoris masker Foto Andik 57
Tabel 4.9	Uji coba foto formal Andry 57
Tabel 4.10	Uji coba foto aksesoris topi foto Andry 57
Tabel 4.11	Uji coba foto aksesoris kacamata foto Andry 58
Tabel 4.12	Uji coba foto aksesoris Masker foto Andry 58
Tabel 4.13	Uji coba foto formal Candra 58
Tabel 4.14	Uji coba foto aksesoris topi foto Candra 59
Tabel 4.15	Uji coba foto aksesoris kacamata foto Candra 59
Tabel 4.16	Uji coba foto aksesoris Masker foto Candra 59
Tabel 4.17	Uji coba foto formal Dessi 60
Tabel 4.18	Uji coba foto aksesoris topi foto Dessi 60
Tabel 4.19	Uji coba foto aksesoris kacamata foto Dessi 60
Tabel 4.20	Uji coba foto aksesoris Masker foto Dessi 61
Tabel 4.21	Uji coba foto formal Fauzi 61
Tabel 4.22	Uji coba foto aksesoris topi foto Fauzi 61
Tabel 4.23	Uji coba foto aksesoris kacamata foto Fauzi 62
Tabel 4.24	Uji coba foto aksesoris Masker foto Fauzi 62
Tabel 4.25	Uji coba foto formal Nawawi 62
Tabel 4.26	Uji coba foto aksesoris topi foto Nawawi..... 63

Tabel 4.27	Uji coba foto aksesoris kacamata foto Nawawi.....	63
Tabel 4.28	Uji coba foto aksesoris Masker foto Nawawi.....	63
Tabel 4.29	Uji coba foto formal Rizal	64
Tabel 4.30	Uji coba foto aksesoris topi foto Rizal	64
Tabel 4.31	Uji coba foto aksesoris kacamata foto Rizal	64
Tabel 4.32	Uji coba foto aksesoris Masker foto Rizal	65
Tabel 4.33	Uji coba foto aksesoris topi foto Uswatun	65
Tabel 4.34	Uji coba foto aksesoris kacamata foto Uswatun.....	65
Tabel 4.35	Uji coba foto aksesoris Masker foto Uswatun.....	66
Tabel 4.36	Uji coba foto formal Yunita	66
Tabel 4.37	Uji coba foto aksesoris topi foto Yunita	66
Tabel 4.38	Uji coba foto aksesoris kacamata foto Yunita	67
Tabel 4.39	Uji coba foto aksesoris Masker foto Yunita	67
Tabel 4.40	Keakuratan identifikasi wajah dengan menggunakan aksesoris	68

DAFTAR GAMBAR

Gambar 3.1	Gambar Flowchar Eigenface	32
Gambar 3.2	Gambar Flowchart Fisherface	34
Gambar 4.1	Gambar Formal menghadap kedepan tanpa senyum	36
Gambar 4.2	Gambar Pose menghadap kesamping tanpa senyum	37
Gambar 4.3	Gambar Pose menghadap kedepan tdengan senyum	37
Gambar 4.4	Gambar pose formal menghadap kedepan dengan tertawa	38
Gambar 4.5	Gambar Memakai topi manghadap kedepan tanpa senyum	38
Gambar 4.6	Gambar Memakai topi menghadap kesamping tanpa senyum ...	39
Gambar 4.7	Gambar memakai topi menghadap kedepan dengan senyum	39
Gambar 4.8	Gambar memakai topi menghadap kedepan dengan tertawa	40
Gambar 4.9	Gambar Memakai kacamata manghadap kedepan tanpa senyum	40
Gambar 4.10	Gambar Memakai kacamata menghadap kesamping tanpa senyum	41
Gambar 4.11	Gambar memakai kacamata menghadap kedepan dengan senyum	41
Gambar 4.12	Gambar memakai kacamata menghadap kedepan dengan tertawa	42
Gambar 4.13	memakai masker menghadap kedepan	42
Gambar 4.14	Foto master sebagai Andika	43
Gambar 4.15	Foto master sebagai Andry	43
Gambar 4.16	Foto master sebagai Candra	43
Gambar 4.17	Foto master sebagai Dessi	44
Gambar 4.18	Foto master sebagai Fauzi	44
Gambar 4.19	Foto master sebagai Nawawi	44
Gambar 4.20	Foto master sebagai Rizal	45
Gambar 4.21	Foto master sebagai Uswatun	45
Gambar 4.22	Foto master sebagai Yunita	45
Gambar 4.23	Mengubah proses citra RGB, Grayscale dan threshold	46
Gambar 4.24	Penyusunan Flatvector	47
Gambar 4.25	Nilai rata-rata flatvector	48
Gambar 4.26	Perhitungan Eigenface	48
Gambar 4.27	Perhitungan eigenface test image	49
Gambar 4.28	Proses identifikasi dengan input image test	50
Gambar 4.29	Mengubah proses citra RGB, Grayscale dan threshold	51
Gambar 4.24	Penyusunan Flatvector	52
Gambar 4.25	kurva pengenalan formal	70
Gambar 4.26	kurva pengenalan topi	71
Gambar 4.27	kurva pengenalan kacamata.....	71
Gambar 4.28	kurva pengenalan masker	72

Gambar 4.29	Akurasi kebenaran eigenface	73
Gambar 4.30	akurasi kebenaran fisherface	73

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Isi lampiran 1.....	66
Lampiran 2. Isi lampiran 2.....	68
Lampiran 3. Isi lampiran 3.....	70
Lampiran 4. Isi lampiran 4.....	72
Lampiran 5. Isi lampiran 5.....	80
Lampiran 6. Isi lampiran 6.....	82
Lampiran 7. Isi lampiran 7.....	83
Lampiran 8. Isi lampiran 8.....	85
Lampiran 9. Isi lampiran 9.....	90
Lampiran 10. Isi lampiran 10.....	100

BAB I

PENDAHULUAN

1.1 Latar Belakang Penulisan

Salah satu ciri-ciri sebuah negara maju adalah disaat negara tersebut dapat mengembangkan dan menerapkan sebuah teknologi didalam kehidupan bermasyarakat, banyak negara-negara yang terbilang negara maju sudah menerapkan teknologi dalam kehidupan keseharian mereka mulai dari bangun tidur hingga mereka beraktifitas dan tertidur kembali semuanya tidak luput dari penggunaan teknologi, disamping itu perlunya suatu teknologi diterapkan dalam sebuah negara dikarenakan dengan teknologi dapat mengurangi pekerjaan manusia yang memiliki kekurangan seperti perlu istirahat, makan, minum dan lain-lain, akan tetapi teknologi dapat mengisi yang menjadikan kekurangan oleh manusia.

Peranan teknologi sangatlah penting dalam kehidupan bermasyarakat banyak peneliti-peneliti yang berlomba-lomba dalam menemukan dan mengembangkan teknologi dalam berbagai macam aspek bidang, salah satu adalah pencitraan.

Pencitraan atau gambar merupakan suatu *reprentasi* , kemiripan atau imitasi dari sebuah objek atau benda, pada sebuah gambar terdapat objek yang dapat kita kenali dengan indra pengelihat entah itu sebuah benda atau seorang yang dapat kita kenal melalui wajah, akan tetapi dengan adanya teknologi sebuah komputer dapat mengenali sebuah benda atau wajah seseorang dari sebuah file gambar yang dapat diidentifikasi dari bentuk atau pola tertentu yang dapat dimikinya, dari proses identifikasi sebuah gambar dapat diproses hingga dapat dikenali dari objek tersebut dalam mengenali wajah itu maka diperlukan ketepatan, dalam kasus ini penulis mencoba membandingkan antara algoritma *Eigenfaces* dan *Fisherface* untuk menguji keakuratan dalam mengenal wajah melalui foto, maka penulis tertarik untuk dijadikannya sebagai penelitian dengan judul ***“Identifikasi Dan Pengujian Keakuratan Pengenalan Wajah Manusia Melalui Foto Dengan Menggunakan Algoritma Eigenfaces dan Fisherface”***.

1.2 Identifikasi Masalah

Adapun identifikasi masalah yang terkait adalah banyaknya penggunaan kamera CCTV untuk merekam segala aktivitas kegiatan sehari-hari guna untuk mengurangi tingkat kejahatan maupun membantu mencari titik terang dalam menyelesaikan suatu tindak kejahatan yang termonitoring oleh kamera CCTV, tentu saja dapat terlihat wajah yang menjadi tersangka dan apakah dapat mudah kita mengenalinya, karena wajah dapat dijadikan sebagai informasi identitas diri, lalu bilamana ingin mencari data dengan melalui foto akan lebih mudah mencari dari data-data yang sudah terkumpul sebelumnya sedangkan komputer hanya dapat mengenal jenis data biner bilangan 0 dan 1, bagaimana kita dapat mengidentifikasi sebuah foto yang memiliki nilai informasi, bagaimana proses sebuah komputer dapat mengenali sebuah gambar mulai dari memasukkan sebuah gambar lalu tahap identifikasi hingga output yang dikeluarkan dari hasil identifikasi tersebut adalah berupa data pengenalan dari wajah manusia dan banyak algoritma yang dapat digunakan untuk identifikasi dan pengenalan wajah dan algoritma mana yang lebih akurat dalam melakukan identifikasi.

1.3 Tujuan Penelitian

Tujuan Penelitian yang penulis lakukan diantaranya adalah:

1. Mengidentifikasi pengenalan wajah melalui foto.
2. Membandingkan keakuratan algoritma *Eigenfaces* dan *Fisherface* dalam mengenal pola wajah manusia.
3. Menguji foto-foto dengan menggunakan beberapa aksesoris untuk keakuratan algoritma *Eigenfaces* dan *Fisherface* dalam mengenal identitas foto.
4. Memahami perbedaan metode algoritma *Eigenfaces* dan *Fisherface* dalam pengenalan wajah manusia.

1.4 Ruang lingkup Penelitian

Dari anggota bagian tubuh manusia peneliti menggunakan wajah yang menghadap kedepan dan samping yang tidak terhalang oleh benda apapun didepan kamera serta dengan berbagai macam ekspresi seperti diam, senyum dan tertawa sekaligus dalam pengujiannya menggunakan aksesoris seperti topi, kacamata dan masker sebagai pola yang akan dikenali oleh komputer dan metode pengenalan wajah tersebut menggunakan metode *Eigenfaces* dan *Fisherface* untuk melihat keakuratan mengidentifikasi guna memperoleh sebuah data dari gambar yang akan dijadikan sebagai bahan informasi menggunakan GUI Matlab, batas usia pria dan wanita minimal 23 tahun serta pencahayaan yang terang dan tidak gelap dengan menggunakan ukuran pass foto 3x3.

1.5 Hipotesis

Dalam menganalisa sebuah foto menjadi informasi diperlukan metode untuk mengenali foto dengan melihat kemiripan pada pola yang terdapat pada sebuah foto, bila diambil contoh foto wajah manusia yang memiliki pola yang berbeda-beda antara setiap individunya, sehingga untuk membedakannya diperlukan perbandingan antara data yang sudah tersimpan dalam sebuah komputer, dengan adanya data perbandingan tentu dapat memudahkan komputer dalam mengenali wajah manusia tersebut hingga dapat memberikan sebuah informasi yang akurat dan setiap metode algoritma *Eigenfaces* dan *Fisherface* memiliki tingkat keakuratan yang berbeda pula.

1.6 Sistematika Penulisan

Dalam penulisan penelitian ini terdapat lima bab yang meliputi beberapa tahapan diantaranya adalah:

a. BAB I PENDAHULUAN

Merupakan pembahasan awal dimana lebih dititik beratkan pada maksud diadakan penelitian tersebut.

b. BAB II LANDASAN TEORI

Merupakan pembahasan yang berisikan teori-teori pembelajaran untuk mendukung pengetahuan tentang permasalahan yang akan dibahas.

c. BAB III METODOLOGI PENELITIAN

Merupakan pembahasan yang berisikan penggunaan metode dalam menyelesaikan permasalahan tersebut.

d. BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Merupakan pembahasan yang berisikan hasil akhir dari sebuah permasalahan yang sudah dapat diselesaikan.

e. BAB V PENUTUP

Merupakan pembahasan yang berisikan kesimpulan dari apa yang sudah diteliti dan saran pendukung agar dapat dikembangkan kembali.

BAB II

LANDASAN TEORI

2.1 Definisi Citra Digital

Menurut Hermawati (2013:3) menjelaskan bahwa “Citra digital atau gambar dapat didefinisikan sebagai sebuah fungsi dua dimensi, $f(x,y)$, dimana x dan y adalah koordinat bidang datar dan harga fungsi f disetiap pasangan koordinat (x,y) disebut intensitas atau level keabuan (*grey level*) dari gambar dititik itu”.

Jika x , y dan f semuanya berhingga dan nilainya diskrit maka gambarnya disebut citra digital. Sebuah citra digital terdiri dari sejumlah elemet yang berhingga, dimana masing-masing lokasi dan nilai tertentu, elemet-element ini disebut sebagai *picture element*, *image element*, *pels* atau *pixels*. Bidang digital *image processing* meliputi pengolahan *digital image* dari suatu komputer digital. Gambar dihasilkan dari seluruh spektrum elektromagnetik dari suatu komputer digital. Gambar dihasilkan dari seluruh spektrum elektromagnetik mulai dari gamma sampai gelombang radio, terdapat tiga tipe pengolahandiataranya:

1. *Low-level process*
2. *Mid-level process*
3. *High-lavel process*

Low-level process meliputi oprasi dasar seperti iamagebpreprocessing:

1. *Reduce noise*
2. *Contrast enhancement*
3. *Image sharpening*

Pada level ini baik input maupun output adalah berupa gambar.

Mid-level process meliputi segmentasi (membagi sebuah gambar dalam region atau objek), mendeskripsikan objek tersebut untuk direduksi dalam bentuk yang diinginkan dan klasifikasi dari objek tersebut, input dan proses ini berupa gambar dan outputnya berupa artibute dari objek tertentu.

Hight-level proces meliputi pemberian arti dari suatu rangkaian objek-objek yang dikenali dan akhirnya menampilkan fungsi-fungsi kognitif secara normal sehubungan dengan pengelihatn.

2.1.1 Sistem Perekam Citra

Citra yang diperoleh tergantung pada letak dan bagiannya yang terdapat dalam 2 karakteristik diantaranya adalah :

1. Karakteristik dari objek yang direkam.
2. Kondisi variabel dari sistem perekam.

Citra merupakan gambaran tentang karakteristik suatu objek menurut kondisi variabel tertentu, sensor pasif adalah sistem sensor yang merekam data objek tanpa mengirimkan enersi, sumber enersi bisa dalam bentuk sinar matahari, sinar lampu dan lain-lain sebagainya. Sensor aktif adalah sistem sensor yang merekam data objek mengirimkan dan menerima pantulan dari enersi yang dikirim kearah objek, enersi yang dikirim bisa berupa gelombang pendek, sinar x dan lainsebagainya.

2.1.2 Elemet fungsi dasar sistem pengolahan citra

Pembentukan citra, sensor yang sensitif terhadap gelombang EM menghasilkan sinyal listrik sesuai dengan enersi yang diterima. *Analog to digital converter/digitalizer* akan mengubah sinyal listrik tersebut menjadi bentuk digital.

Penyimpanan citra, penyimpanan jangka pendek atau sedang diproses membutuhkan *memory*, penyimpanan *on-line* atau siap dipakai dengan menggunakan disk magnetik, sedangkan penyimpanan arsip menttunakan pita atau CD.

Pemrosesan citra dan komunikasi, dalam citra berukuran besar yang tidak membutuhkan tempat penyimpanan yang besar serta waktu proses yang lama, issue penting pada komunikasi seperti kompresi citra dan issue penting pada pemrosesan citra adalah peroses paralel.

Peragaan citra, dalam bentuk *softcopy* dengan menggunakan layar atau monitor, sedangkan dalam bentuk *hardcopy* dengan menggunakan printer, film writer atau plotter.

2.2 Dasar Citra Digital

2.2.1 Image Sampling & Quantization

Output dari kebanyakan sensor berupa gambar kontinyu. Untuk membuat sebuah gambar digital kita perlu merubah data kontinyu kedalam bentuk digital prosesnya terdiri dari:

1. *Sampling* : mendigitalkan nilai-nilai koordinatnya
2. *Quantization* : mendigitalkan nilai amplitudonya

Proses sampling dilakukan dengan membagi garis horizontal dalam beberapa kotak, semakin rapat pembagiannya maka citra yang dihasilkan akan semakin halus. Garis vertikal menggambarkan tingkat kecerahan piksel disuatu titik atau kotak. Untuk mendapatkan suatu nilai diskrit maka perlu membagi garis vertikal dalam beberapa kotak yang menggambarkan tingkat kecerahan mulai dari yang paling gelap atau paling bawah ke yang paling terang atau paling atas.

2.2.2 Representasi Citra Digital

Hasil dari sampling dan dijittisari adalah sebuah matriks dari bilangan-bilangan riil. Misalkan sebuah citra $f(x,y)$ disampling sehingga menghasilkan gambar digital dengan M baris dan N kolom. Nilai dari koordinat (x,y) sekarang menjadi kuantitas diskrit, yang biasanya dinyatakan dengan bilangan bulat. Nilai koordinat diskrit yang biasanya dinyatakan dengan bilangan bulat nilai koordinat titik asal adalah $(x,y)=(0,1)$.

Element-element dari matriks disebut image element, picture element, pixel atau pel. Sebuah piksel mempunyai dua properti yaitu koordinat atau posisi dari piksel tersebut dan nilai dari piksel tersebut. Sehingga sebuah piksel dapat dinyatakan sebagai fungsi dua dimensi $f(x,y)$, bilamana sebuah pixel $f(0,2)=7$, berarti piksel tersebut berada pada garis 0 dan kolom 2 dengan nilai intensitas kecerahan=7.

Dalam proses dijittisasi perlu ditetapkan nilai M, N dan L (level keabuan yang diperbolehkan untuk tiap pixel). Nilai M dan N bebas tetapi berupa bilangan bulat positif. Dengan mempertimbangkan penggunaan perangkat keras untuk pengolahannya, penyimpanan dan sampling, jumlah gray level merupakan 2 pangkat

bilangan bulat ($L=2^k$). Ini berarti nilai level dari elemen citra yang mempunyai range antara $[0, L-1]$, seperti sebuah citra yang mempunyai level keabuan $L=8$, berarti nilai intensitas keabuan dari setiap pikselnya mempunyai range antara 0-7.

Jumlah bit yang dibutuhkan untuk menyimpan gambar digital adalah $b=M \times N \times k$. Dimana M = jumlah baris citra, N = jumlah kolom citra dan k = jumlah bit yang dibutuhkan untuk menyatakan sebuah nilai keabuan.

Sebuah citra mempunyai jumlah baris $M=10$ dan jumlah kolom $N=10$ dengan level keabuan $L=8$, tentukan jumlah bit yang dibutuhkan untuk menyimpan citra data tersebut, maka didapatkan level keabuan citra $L=8$, maka nilai keabuan pada sebuah piksel mempunyai range antara 0-7. Jumlah bit yang dibutuhkan untuk menyatakan nilai 0-7 dalam bentuk biner adalah 3 ($k=3$), yaitu 000-111. Dimana $L=8=2^3$. Maka jumlah bit yang dibutuhkan untuk menyatakan citra tersebut $=M \times N \times k=10 \times 10 \times 3=300$ bit.

2.2.3 Resolusi Spasial & Grey Level (Brightness)

Resolusi spasial dan resolusi kecermelangan berpengaruh pada besarnya informasi citra yang hilang. Resolusi spasial halus atau kasar pembagian kisi-kisi baris dan kolom. Transformasi citra kontinyu ke citra digital disebut digitasi atau sampling. Sebuah gambar digital L -level dengan ukuran $M \times N$ mempunyai resolusi spasial $=M \times N$ pixel dan resolusi level keabuan $=L$ level. Hasil digitasi dengan jumlah baris 256 dan jumlah kolom 256 mempunyai resolusi spasial 256×256 . Akibat dari bervariasi jumlah sampel dalam resolusi kecermelangan (intensitas/brightness) halus atau kasar pembagian tingkat kecermelangan. Intensitas disebut juga dengan grey-level transformasi data analog yang bersifat kontinyu ke daerah intensitas diskrit disebut kwantisasi. Bila intensitas piksel berkisar antara 0 dan 255 resolusi kecermelangan citra adalah 256. Sedangkan akibat bervariasinya jumlah gray level pada gambar digital.

2.2.4 Zooming & Shrinking Digital Images

Zooming adalah proses memperbesar gambar dan sebaliknya shrinking adalah memperkecil gambar:

Zooming membutuhkan dua langkah untuk membentuk dan menggantikan pixel yang semula menjadi pixel pembesar, diantaranya:

1. Membuat lokasi pixel yang baru
2. Memberikan nilai grey-level kelokasi-lokasi baru tersebut dengan:
 - a. Nearest neighbor interpolation
 - b. Pixel replication
 - c. Biliner interpolation

Misalkan kita mempunyai citra berukuran 500x500 piksel dan kita ingin memperbesar 1.5 kali menjadi 750x750. Salah satu cara yang paling mudah secara visualisasi adalah dengan membuat 750x750 grid pada citra aslinya. Akibatnya ukuran grid menjadi lebih kecil. Kemudian kita memberikan nilai kepixel baru dalam grid tersebut dengan melihat nilai piksel terdekat dari citra aslinya. Metode assignment ini disebut dengan interpolaris tetangga terdekat. Metode assignment pixel replication adalah kasus khusus dari metode nearest neighbor interpolation. Pixel replication digunakan jika akan memperbesar ukuran citra sebesar sebuah bilangan bulat. Untuk memperbesar 2 kali lipat dapat diduplikatkan masing-masing kolom dimana hal itu akan memperbesar citra ke horizontal kemudian dapat menduplikasikannya masing-masing sehingga akan memperbesar kearah vertikal. Meskipun metode nearest neighbour cepat, tetapi metode ini mengakibatkan efek checkerboard atau kotak-kotak pada citra hasil zooming atau shrinking.

Metode bilinear interpolation menggunakan empat tetangga terdekat sebuah titik misalkan (x', y') adalah koordinat titik pada citra yang diperbesar dan $v(x', y')$ adalah nilai yang akan diberikan ketitik tersebut yang dinyatakan sebagai berikut:

$$v(x', y') = ax' + by' + cx'y' + d$$

dimana keempat koefisien ditentukan dari empat persamaan dalam empat variabel yang tidak diketahui yang dapat ditulis menggunakan empat tetangga terdekat dari titik (x', y')

2.2.5 Beberapa Dasar Hubungan Antara – Piksel

Tetangga sebuah piksel, sebuah piksel P pada koordinat (x,y) mempunyai empat tetangga horisontal dan vertikal dengan koordinat dengan $(x+1,y)$, $(x-1,y)$, $(x,y+1)$, $(x,y-1)$. Himpunan piksel-piksel ini disebut 4 – tetangga dari 4, dinotasikan dengan $N_4(P)$.

Tabel II.1

Tabel Vertikal dan Horizontal

	$(x-1,y)$	
$(x,y-1)$	P	$(x,y+1)$
	$(x+1,y)$	

Keempat tetangga diagonal dari p , $N_D(P)$ mempunyai koordinat $(x+1,y+1)$, $(x+1,y-1)$, $(x-1,y+1)$, $(x-1,y-1)$.

Tabel II.2

Tabel Diagonal

$(x-1,y-1)$		$(x-1,y+1)$
	P	
$(x+1,y-1)$		$(x+1,y+1)$

Adhacency/konektivitas, 4-tetangga atau 8-tetangga dengan kriteria gray level yang sama, misal: sama-sama 0 atau sama-sama 1 atau sama-sama bedanya tidak lebih dari 5 tingkat keabuan, dan lain sebagainya. Misalkan V adalah himpunan nilai grey-level yang digunakan untuk mendefinisikan adjacency. Maka ada 3 tipe adjacency :

1. *4-adjacency*, dua pixel p dan q dengan nilai dari V dikatakan 4-adjacent jika Q ada di dalam himpunan $N_4(p)$.
2. *8-adjacency*, Dua pixel p dan q dengan nilai dari V dikatakan 8-adjacent jika Q ada di dalam himpunan $N_8(p)$.
3. *M-adjacency* (mixed adjacency), Dua pixel p dan q dengan nilai dari V dikatakan m -adjacent jika Q ada di $N_4(p)$, atau Q ada di $N_4(p)$ dan $N_4(p)$ o $N_4(q)$ tidak mengandung nilai.

Sebuah path digital atau kurva dari pixel p dengan koordinat $=(x,y)$ pixel q dengan koordinat (s,t) merupakan barisan dari piksel-piksel yang berbeda dengan koordinat. Dalam hal ini n adalah panjang dari path tersebut. Misalkan S merupakan sub-set dari piksel-piksel dalam sebuah citra. Dua piksel p dan q dikatakan terhubung (connected) dalam S jika ada sebuah path antara piksel-piksel tersebut yang berisi seluruh piksel dalam S . Misal p adalah suatu iksel dalam S , Himpunan piksel-piksel yang terhubung kepadanya dalam S disebut komponen terhubung (connected component) dari S , Jika hanya mempunyai satu connected component, maka S disebut connected set.

2.3 Pengenalan Pola

Menurut Fatta (2009:5) pengenalan pola adalah "salah satu cabang dari bidang kecerdasan buatan", adapun beberapa definisi yang berbeda tentang pengenalan pola diantaranya:

1. Penentuan suatu objek fisik atau kejadian kedalam salah satu atau beberapa katagori. (Duda dan Hart).
2. Ilmu pengetahuan yang menitik beratkan pada deskripsi dan klasifikasi dari suatu pengukuran. (Schafkoff)
3. Proses untuk memberikan nama u untuk pengamatan x . (Schurmann)

2.3.1 Komponen Sistem Pengenalan Pola

Sistem pengenalan pola terdiri dari beberapa bagian dan sensor tersebut digunakan untuk mendapatkan sebuah pencitraan, diantaranya:

1. Sensor

Sensor digunakan untuk menangkap objek yang ciri atau fiturnya akan dieksekusi.

2. Mekanisme pre-processing

Mekanisme pengenalan objek yang ditangkap oleh sensor biasanya digunakan untuk mengurangi kompleksitas ciri yang akan dipakai untuk proses klasifikasi.

3. Mekanisme pencarian fitur

Bagian ini digunakan untuk mengekstraksi yang telah melalui tahapan preprocessing untuk memisalkan dari fitur-fitur pada objek yang tidak dapat diperlukan dalam proses klasifikasi.

4. Algoritma Pemilah

Pada tahapan ini proses klasifikasi dilakukan dengan menggunakan algoritma klasifikasi tertentu. Hasil dari tahapan ini adalah klasifikasi dari objek yang ditangkap kedalam kriteria-kriteria yang telah ditentukan.

2.3.2 Fitur

Menurut Fatta (2009;6) menjelaskan bahwa fitur adalah “segala jenis aspek pembeda, kualitas atau karakteristik”. Fitur bisa berwujud simbolik misalnya warna atau numerik misalnya tinggi. Kombinasi dari d-buah fitur dinyatakan sebagai vektor kolom dimensi $-d$ dan disebut vektor fitur. Objek dinyatakan sebagai sebuah titik dalam fitur. Penggambaran tersebut sebagai diagram hambur. Kualitas dari vektor fitur dilihat dari kemampuannya membedakan objek yang berasal dari kelas yang berbeda-beda dimana berlaku ketentuan sebagai berikut:

1. Objek dalam kelas yang sama harus memiliki nilai vektor fitur yang sama.
2. Objek dalam kelas yang berbeda harus memiliki nilai vektor fitur yang berlainan.

2.3.3 Pola

Menurut Fatta (2008:7) pola adalah “komposit/gabungan dari fitur yang merupakan sifat dari objek. Dalam klasifikasi, pola berupa sepasang variabel (x, w) dimana:

1. x adalah sekumpulan pengamatan atau fitur (vektor fitur).
2. w adalah konsep dibalik pengamatan (label).

2.3.4 Pemilah

Tugas dari pemilah adalah menyekat ruang fitur kedalam daerah-daerah yang dilabeli dengan kelas. Garis batas antar daerah keputusan disebut sebagai pembatas keputusan, pemilihan vektor fitur x meliputi penentuan daerah keputusan yang sesuai dan pengelompokan x ke dalam kelas ini.

2.3.5 Pendekatan Pengenalan Pola

Aplikasi pengenalan pola dapat dibuat dengan beberapa pendekatan. Ada pendekatan yang menggunakan basis statistikal untuk menghasilkan pola. Pendekatan lainnya menggunakan struktur dari pola yang menyediakan informasi fundamental untuk pengenalan pola. Pendekatan lainnya adalah dengan membangun dan melatih suatu arsitektur yang secara akurat mengasosiasikan input pada pola tertentu dengan response yang diharapkan, tinggal dipertimbangkan pendekatan mana yang lebih tepat untuk masalah yang akan dipecah.

1. Pendekatan pengenalan pola statistikal (StartPR)

Pengenalan pola statistikal memiliki asumsi suatu basis statistik untuk algoritma klasifikasi. Sekelompok karakteristik pengukuran yang melanjutkan fitur diekstraksi dari data input dan digunakan untuk menemukan setiap vektor fitur kedalam salah satu kelas c . Fitur diasumsikan dihasilkan secara natural sehingga model yang bersangkutan adalah suatu state of nature atau klas-klas probability atau fungsi kepadatan probabilitas yang telah dikondisikan. Dengan demikian kesimpulannya adalah sebagai berikut:

- a. Pola dipilah berdasarkan model statistik dari fitur.

- b. Model statistik didefinisikan sebagai sebuah keluarga dari fungsi kerapatan peluang bersyarat kelas $\Pr(x|c_i)$ – peluang vektor fitur x jika diberikan kelas c_i .

2. Pendekatan pengenalan pola sintaktik (syntPR)

Untuk pendekatan sintatik dapat dijelaskan sebagai berikut:

- a. Pola dipilah berdasarkan keserupaan ukuran struktural.
- b. ‘Pengetahuan’ direpresentasikan secara formal grammer
- c. Deskripsi relasional
- d. SyntPR dipakai tidak hanya untuk pemilahan, tetapi juga untuk deskripsi
- e. Biasanya SyntPR memformulasikan deskripsi hierarkis dari pola kompleks yang tersusun dari pola bagian yang lebih sederhana

3. Pendekatan pengenalan pola neural (NeuroPR)

Pendekatan ini menggunakan metode jaringan syaraf tiruan untuk mengidentifikasi pola. Pendekatan ini dapat dijelaskan sebagai berikut:

- a. Pemilahan dilakukan berdasarkan tanggapan suatu neuron jaringan pengolah sinyal terhadap stimulasi masukan pola.
- b. ‘Pengetahuan’ disimpan dalam sambungan antarneuron dan kekuatan pembobot sinpatik.
- c. NeuronPR dapat dilatih, non-algoritmatis, strategi black box.
- d. NeuronPR sangat menarik karena dengan sejumlah lapisan dan neuron secukupnya, JST dapat membentuk semua jenis daerah keputusan yang rumit sekalipun.

Jangkauan dari aplikasi-aplikasi sistem pengenalan pola cukup beragam, diantaranya:

1. *Voice recognition*

Beberapa sistem rahasia menggunakan pengenalan-pengenalan suara sebagai kunci bagi pengguna sistemnya.

2. *Fingerprint identification*

Sistem pengenalan sidik jari telah dipakai secara luas sebagai pengganti password atau pin untuk mengakses sistem komputer tertentu.

3. *Face identification*

Badan penegak hukum sedang mengembangkan sistem untuk mengidentifikasi para buronon dengan melakukan scanning pada sejumlah besar data wajah para pelaku yang sudah didatabsekan berdasarkan foto pelaku kejahatan tersebut.

4. *Handwriting identifiaction*

Aplikasi perbankan menggunakan pengenalan tulisan untuk membuktikan pelaku transsaksi adalah orang yang bener-benar berhak.

5. *Optical Character Recognition (OCR)*

OCR sekarang digunakan secara luas pada took retail untuk meningkatkan produktivitas pada konter pengecekan barang.

6. *Robot vision*

Banyak aplikasi robotik menggunakan pengenalan pola untuk mengenali objek tertentu pada lingkungan yang unik.

2.4 Kompresi Citra

Kompresi data merupakan proses reduksi jumlah data yang diperlukan untuk menyatakan suatu jumlah informasi yang diberikan. Perlunya mekukan kompresi dan reduksi data karena:

1. Data citra umunya berukuran besar
2. Tidak peraktis dalam aspek penyimpanan, proses dan transmisi
3. Perlu reduksi atau pemampatan data dengan mengurangi redudancy atau duplikat data

Teknik kompresidibedakan menjadi dua yaitu:

1. *Information preserving*

Teknik yang memproses data asli menjadi bentuk yang lebih ringkas tanpa hilangnya informasi, contoh: aplikasi biomedis.

2. *Lossy compression*

Teknik mendapatkan data yang lebih ringkas dengan melalui suatu proses penghampiran dari data asli dengan tingkat error yang dapat diterima, contoh: TV broadcast.

Untuk lebih memahami kebutuhan akan representasi citra yang kompak. Sejumlah data disajikan dalam film televisi SD (standar definition) dengan panjang 2 jam menggunakan array piksel ukuran 720x480x24 bit. Sebuah film digital atau video adalah suatu rangkaian frame video yang tiap framenya merupakan sebuah citra berwarna karena video player harus menjalankan frame secara berurutan pada rate mendekati 30 fps, SD digital video data harus diakses pada 30 frame/second x (720x480) piksel/frame x 3 byte/piksel = 31.104.000 bytes/sec dan film berdurasi 2 jam berisi 31.104.000 bytes/sec x (60x60) sec/hour x 2 hour = 2.24×10^{11} bytes (224 GB). Aplikasi yang membutuhkan image compresion dimana perkembangannya ditentukan oleh efisiensi pada manipulasi data, penyimpanan dan transmisi citra biner/monokrom/berwarna:

1. Remote sensing
2. Telemedical / medical imaging
3. Facsimile transmision

2.4.1 Dasar Kompresi Citra

Perlunya mengetahui perbedaan antara data dan informasi, berita ‘pembinaan’ mahasiswa STPDN dimuat dikoran, kata-kata yang membentuk berita tersebut disebut data, informasi diturunkan dari data. Isi berita tentang ‘pembinaan’ tersebut merupakan sebuah informasi. Informasi yang sama dapat direpresentasikan dalam bentuk data yang berbeda. Bandingkan antara informasi ‘pembinaan’ tersebut yang ditulis dikoran atau ditayangkan di TV.

Data redudancy adalah bagian data yang tidak mengandung informasi terkait atau merupakan pengulangan dari informasi yang sudah dinyatakan sebelumnya atau sudah diketahui. Data redudency merupakan pusat masalah dari kompresi citra digital. Data redudency bersifat tidak abstrack dan dapat dinyatakan dalam entititas besaran matematis. Jika n_1 dan n_2 merupakan jumlah unit pembawa informasi dalam dua himpunan data yang menyatakan informasi yang sama maka relative data redudancy , R_D dari himpunan pertama adalah:

$$R_D = 1 - \frac{1}{CR}$$

Dimana CR disebut compression ratio dan didefinisikan sebagai berikut:

$$C_R = \frac{n_1}{n_2}$$

Jika $n_1=n_2$ maka $C_R=1$ dan $R_D = 0$ (tidak ada redundancy). Jika $n_2 \ll n_1$ Maka C_R mendekati tak terhingga dan R_D mendekati 1 (terjadi redundancy yang tinggi).

2.4.2 Coding Redundancy

Terjadi bila suatu kode simbol yang digunakan terdiri dari sejumlah bit yang melebihi jumlah bit yang diperlukan untuk representasi setiap simbol dalam hal ini : tingkat keabuan piksel citra. Misalkan diasumsikan bahwa sebuah variabel acak diskrit r_k dengan interfal (0,1) menyatakan tingkat keabuan dari sebuah citra dan tiap r_k terjadi dengan probabilitas $p_r(r_k)$, maka :

$$P_r(r_k) = \frac{n_k}{n} \quad K= 0,1,2,\dots,L - 1$$

Dimana L adalah jumlah tingkat keabuan , n_k adalah jumlah kemunculan gray-level yang ke-k dalam citra, dan n adalah jumlah piksel keseluruhan dalam sebuah citra. Jika jumlah bit yang digunakan untuk menyatakan tiap nilai dari r_k adalah $l(r_k)$, maka rata-rata jumlah bit yang diperlukan untuk menyatakan tiap piksel adalah :

$$L_{avg} = \sum_{k=0}^{L-1} l(r_k) P_r(r_k)$$

Jumlah bit keseluruhan yang dibutuhkan untuk kode sebuah citra berukuran MxN adalah MNL_{AVG} . 30 frame/second x (720 x 480)piksel/frame x 3 byte/piksel = 31.104.000 bytes/sec dan film berdurasi 2 jam berisi : 31.104.000 bytes/sec x (60 x 60)sec / hour x 2 hour =224 x 10¹¹ bytes (224 GB). Jika kita menggunakan DVD berukuran 8.5 GB maka kita membutuhkan 12 buah DVD. Untuk dapat disimpan pada satu buah DVD, maka tiap frame perlu dikompres dengan facktor 26.3(=224/8.5). Aplikasi yang membutuhkan image compression yaitu dimana perkembangannya ditentukan oleh efisiensi pada manipulasi data, penyimpanan dan transmisi citra biner/monokrom/berwarna :televideo-conferencing yaitu remote sensing. Intrepixel Redudancy sering disebut juga sebagai spatial redudancy. Geometric redudancy, atau interframe redudancy dimana N adalah jumlah piksel

pada baris x dan Δn dan $\Delta n < N$. Nilai y tinggi (berkorelasi tinggi pada $\Delta n = 45$. Dimana nilai y adalah 0.9922 dan 0.9928.

2.4.3 Psychovisual Redundancy

Suatu fenomena dimana intensitas keabuan yang bervariasi dilihat dari intensitas konstan, artinya mata tidak mempunyai sensitivitas yang sama terhadap semua informasi yang dianggap penting dan tidak penting, disebut dengan match band. Hal tersebut dapat dikatakan sebagai psychovisual redundancy, yang bila dihilangkan tidak mengganggu persepsi kualitas citra. Eliminasi psychovisual redundancy mengakibatkan kerugian informasi bersifat kualitas. Teknik ini disebut quantization yaitu pemetaan dari daerah intensitas yang lebar menjadi daerah intensitas terbatas. Proses kwantisasi dapat disesuaikan untuk kepentingan mempertahankan informasi edge seperti yang dilakukan pada improved gray scale (IGS) quantization. Fidelity criteria terdiri dari beberapa kriteria diantaranya yaitu : excellent, fine, passable, marginal, inferior, unusable. Kesalahan akibat hilangnya informasi dapat dihitung dengan menggunakan MSE, misal $f(x, y)$ adalah citra asli dan $f(x, y)$ adalah citra aproksimasi. Pengukuran informasi yaitu informasi yang dikandung oleh suatu pesan dapat dinyatakan dengan suatu model yang bersifat *probabilistic* dalam bentuk yang dapat diterima secara intuitif. Informasi channel yaitu suatu sumber informasi terdiri dari simbol $\{a_1, a_2, \dots, a_j\}$ dengan $P[a_j]$ adalah probabilitas sumber tersebut menghasilkan simbol a_j dimana $\sum_{j=1}^J P(a_j) = 1$. Sumber informasi berbentuk (A, z) dengan $A = \{a_1, a_2, \dots, a_j\}$ dan $z = \{P\{a_1\}\}$.

2.4.4 Segmentasi Citra

Segmentasi merupakan proses membagi suatu citra ke dalam komponen – komponen region atau objek. Algoritma secara umum didasarkan pada salah satu dari sifat dasar nilai intensitas yaitu : Discontinuity dan similarity yaitu pendekatan dengan membagi citra berdasarkan perubahan besar pada nilai intensitas nya. Ada 3 tipe diskontinuitas gray – level yaitu titik, garis, tepi. Salah satu cara yang paling banyak digunakan untuk melihat diskontinuitas adalah menjalankan sebuah mask / filter. Operasi yang digunakan adalah sum of product dari koefisien dalam mask dengan grey-level yang ada di daerah dimana mask itu berada. Deteksi dari titik yang terisolasi dalam sebuah citra cukup mudah, dengan menggunakan mask di dapatkan bahwa suatu titik dapat di deteksi pada pusat lokasi dimana mask tersebut

berbeda. Isolated point adalah sebuah titik yang gray-level nya sangat berbeda dari background nya dan ada di daerah yang homogen atau hampir homogen. Deteksi garis digerakkan keseluruh citra maka menghasilkan hasil yang lebih kuat pada garis. Dengan background yang tetap, repon maksimum akan dihasilkan jika garis tersebut di lewati baris tengah dari mask. Deteksi tepi adalah sebuah himpunan dari piksel – piksel yang terhubung yang berada pada batas (boundary) diantara dua region. Definisi tepi membutuhkan kemampuan untuk mengukur transisi gray-level dengan cara yang tepat. Ketebalan dari tepi di tentukan oleh panjang dari ramp yang merupakan transisi dari gray level awal sampai gray level akhir. Turunan pertama dalam pengolahan citra diimplementasikan menggunakan besaran dari gradient. Metode yang paling umum dari diferensiensi dalam operasi pengolahan citra adalah gradient. Untuk suatu fungsi $f(x,y)$, gradient dari f pada koordinat (x, y) didefinisikan sebagai vector.

2.4.5 Deteksi Tepi Dengan Operator Sobel

Deteksi tepi dengan operator sobel adalah gambaran proses deteksi tepi dengan operator sobel, misalkan citra input I , maka langkah pertama dilakukan operasi konvolusi antara citra input I dengan operator sobel arah horisontal $\frac{d}{dx} I$, dan operator sobel vertikal $\frac{d}{dy} I$, kemudian dihitung magnitude besaran) dengan menggunakan rumus : $M = (\frac{d}{dx} I) + (\frac{d}{dy} I)$ selanjutnya dilakukan proses thresholding, dengan menggunakan sebuah nilai ambang(threshold) tertentu. Mart Hildert Operator merupakan salah satu bentuk operator yang menggunakan turunan kedua. Tanda turunan kedua digunakan untuk menentukan apakah tepi tersebut ada di sisi gelap atau di sisi terang dari suatu tepi, jika negatif berarti ada di sisi terang dan jika positif maka titik tersebut ada di sisi gelap. Catatan tentang sisi tambahan dari turunan kedua di sekitar tepi terbagi menjadi 2 yaitu :

1. Turunan ke dua menghasilkan dua nilai untuk tiap tepi dari sebuah citra.
2. Dapat dibayangkan bahwa sebuah garis lurus menggabungkan nilai turunan, kedua ekstrem positif dan negatif yang akan memotong nol di titik tengah tepinya. Sifat zero-crossing dari turunan kedua berguna untuk menentukan lokasi pusat dari tepi yang tebal. Setelah dilakukan proses dengan Log dilanjutkan dengan proses thresholding untuk mendapatkan gambar hitam dan putih saja.

Tepi yang dihasilkan dengan zero-crossing lebih tipis dibandingkan dengan gradient-operator. Tepi yang diperoleh dengan zero-crossing membentuk banyak loop tertutup, yang sering disebut dengan spaghetti effect yang merupakan satu dari banyak kekurangan yang cukup serius karena itu metode dengan operator gradient lebih banyak digunakan untuk segmentasi dibandingkan dengan zero-crossing.

2.5 Deteksi Tepi Dengan Canny Operator

Canny operator merupakan salah satu bentuk operator, operator ini diformulasikan dengan 3 tujuan utama :

1. *Optimal detection*, dengan tidak adanya respon yang salah (palsu).
2. *Good localization*, dengan jarak minimal antara posisi tepi sebenarnya dengan posisi yang terdeteksi.
3. *Single respon*, dengan mengeliminasi multiple response menjadi satu garis tepi saja.

Algoritma dari deteksi tepi dengan canny operator adalah sebagai berikut :

4. *Smoothing* dengan *gaussian filter*, tujuannya adalah untuk mengurangi respon terhadap noise.
5. Cari turunan pertama arah horizontal, S_x dengan vertical, S_y dengan menggunakan operator sobel, kemudian hitung besaran atau magnitudenya.
6. *Non maximum suppression*, sesuai dengan canny operator adalah untuk mendeteksi lokasi tepi yang tepat, maka non maximum suppression bertujuan untuk mempertahankan puncak. Non maximum suppression pada dasarnya adalah mencari titik tertinggi pada data besaran tepi. Ini diperoleh dengan menggunakan informasi arah tepi untuk memeriksa apakah titik tersebut berada pada puncak punggung bukit (ridge).
7. Hysteresis Thresholding yaitu jika gradien pada sebuah piksel diatas nilai “High” piksel tersebut merupakan “edge pixel”
8. Jika gradien pada sebuah piksel dibawah nilai “Low” maka pixel tersebut bukan “edge pixel”
9. Jika gradien pada sebuah pixel antara nilai “Low” dan “High” maka pixel tersebut termasuk “edge pixel” dan hanya jika pixel tersebut terhubung dengan

sebuah “edge pixel” secara langsung atau melalui pixel – pixel antara “Low” dan “High”.

2.5.1 *Local Processing*

Salah satu pendekatan sederhana dari edge linking adalah menganalisa karakteristik dari piksel – piksel dalam sebuah ketetanggaan yang kecil misalnya 3 x 3 atau 5 x 5 apakah setiap titik (x, y) dalam sebuah citra berada pada tepi dengan salah satu teknik, semua titik yang mempunyai kesamaan menurut kriteria yang ditentukan sebelumnya dihubungkan. Ada 2 sifat utama yang digunakan untuk menetapkan similitas pixel – pixel tepi, yaitu :

1. Kekuatan respon dari operator gradien yang digunakan untuk menghasilkan piksel tepi.
2. Arah dari vektor gradien.

Dengan demikian sebuah pixel tepi dari operator gradien (x_0, y_0) ada dalam suatu ketetanggaan dari (x, y) sama dengan besaran piksel tersebut ke (x, y).

Sebuah pixel tepi (x_0, y_0) yang berada pada ketetanggaan (x, y) mempunyai sudut yang sama dengan pixel (x, y).

2.5.2 **Global Processing Dengan Hough Transform**

Yaitu titik – titik yang dihubungkan dengan menentukan pertama kali apakah titik – titik tersebut ada pada satu kurva bentuk tertentu. Jika terdapat n titik pada sebuah citra., misalkan kita ingin menemukan subset dari titik – titik ini yang terletak pada garis lurus. Salah satu solusi adalah pertama kali menemukan semua garis yang ditentukan oleh setiap pasang titik dan kemudian menemukan semua subset dari titik – titik yang dekat dengan garis tertentu. Yang menjadi permasalahan adalah untuk menemukan semua garis dari semua pasang titik yang ada membutuhkan waktu komputasi yang tinggi, Hough mengajukan satu pendekatan yang disebut dengan **Transformasi Hough**. Transformasi Hough muncul dari subdivision-subdivision yang disebut dengan accumulator cell, dimana (a_{max}, a_{min}) , (b_{max}, b_{min}) merupakan range nilai kemiringan dari nilai intercept yang diinginkan. Mula – mula cell – cell tersebut diset nol, kemudian untuk setiap titik (x_k, y_k) dalam plane citra diberikan parameter a yang sama untuk tiap subdivisi pada sumbu-a dan

memberikan nilai pada b sesuai dengan persamaan $b = -x_k.a + y_k$ hasil b kemudian di bulatkan mendekati nilai yang diperbolehkan dalam sumbu- b . Jika nilai a_p menghasilkan b_q diberikan $A(p, q) + 1$. Pada akhir prosedur nilai M dalam $A(i, f)$ berkorelasi dengan M titik dalam bidang $-xy$ berada pada garis $y = a, x + b$, jadi jumlah subdivisi pada bidang ab menentukan keakuratan dari kolinieritas dari titik – titik ini. Subdivisi sumbu- menjadi K buah menjadikan setiap titik (x_k, y_k) terdapat K buah nilai b yang berkaitan dengan K nilai yang mungkin untuk a . Dengan jumlah n titik dalam citra, maka metode ini memerlukan komputasi Nk kali. Permasalahan saat menggunakan persamaan $y = ax + b$ untuk menyatakan sebuah garis adalah bahwa kedua kemiringan dan intercept mendekati tak terhingga saat garis mendekati vertical. Salah satu cara mengatasi kesulitan ini adalah dengan menggunakan representasi garis normal. Penggunaan representasi ini dalam membangun tabel akumulator sama seperti metode representasi slope – intercept . Penyelesaian permasalahan edge – linking dengan menggunakan pendekatan transformasi hough adalah sebagai berikut :

1. Hitung gradien dari sebuah citra
2. Tentukan subdivision dalam plane
3. Periksa jumlah accumulator cell untuk konsentrasi piksel tinggi
4. Periksa relasi antara piksel – piksel cell dalam sel yang di pilih.

2.5.3 Pemrosesan Global Melalui Teori Graph

Deteksi dan penggabungan edge dilakukan dengan mempresentasikan segmen – segmen edge dalam bentuk graph dan mencari graph yang memiliki path dengan cost rendah. Representasi graph memiliki performance yang cukup baik dengan keberadaan noise. Tetapi, prosedurnya lebih kompleks dan memerlukan waktu proses yang lebih banyak dibanding transformasi hough. Graph G adalah himpunan simpul (node) N yang tidak kosong dan berhingga. Tiap pasang (n_i, n_j) dari U disebut rusuk atau arkus (arc). Graph yang rusuknya berarah disebut graph berarah. Proses pengidentifikasian successor dari sebuah simpul disebut ekspansi simpul. Level 0 dari graph terdiri dari satu buah simpul yang disebut dengan simpul akar dan simpul pada level terakhir disebut simpul tujuan. Cost $c(n_i, n_j)$ bisa diasosiasikan dengan setiap rusuk (n_i, n_j) . Rangkaian simpul $n_i, n_j, \dots, n_k$ dengan

setiap simpul n_i menjadi sucsessor dari simpul n_{i-1} disebut jalur (path) dari n_i ke n_k . Cost dari keseluruhan path adalah $c = \sum_{i=2}^k c(n_{i-1}, n_i)$. Elemen edge di definisikan sebagai boundary di antara dua piksel p dan q, sedemikian sehingga p dan q adalah tetangga – 4. Elemen edge diidentifikasi dengan koordinat xy dari titik p dan q. Dengan kata lain elemen rusuk edge adalah serangkaian elemen edge yang terhubung. Tiap elemen edge didefinisikan dengan piksel p dan q, memiliki coast $c(p, q) = H - [f(p) - f(q)]$. Pendekatan *heuristic* untuk mencari graph dengan cost path minimum diantaranya :

1. Tandai start simpul OPEN dan beri nilai $g(s) = 0$.
2. Jika tidak ada simpul yang OPEN keluar dengan kondisi gagal (tidak di temukan tepi).
3. Tandai dngan CLOSED, simpul n yang OPEN dan memiliki estimasi $r(n)$ terkecil.
4. Jika n adalah simpul tujuan (goal), keluar dengan path solusi yang di dapat dengan melakukan penelusuran kembali pointer.
5. Ekspen simpul n, untuk menghasilkan semua sucsessor
6. Jika successor n_i belum ditandai , maka hitung $r(n_i) = g(n) + c(n, n_i)$ kemudian tandai simpul dengan OPEN, dan arahkan pointer kembali ke n.
7. Jika successor n_i di tandai dengan CLOSE atau OPEN, ubah nilai nya.
8. Tandai OPEN semua successor yang CLOSED dan memiliki nilai g lebih rendah dan arahkan kembali ke n pointer dari semua simpul yang nilai g nya lebih rendah.

2.6 Pengertian Dasar *Thresholding*

Histogram Sebelah kiri mewakili citra $f(x, y)$ yang tersusun atas objek terang di atas background gelap. Piksel piksel objek dan background dikelompokkan menjadi dua mode yang dominan. Cara untuk mengekstrak dari background adalah dengan memilih threshold T yang memisahkan dua metode tersebut. Thresholding multilevel digunakan untuk mengklasifikasikan suatu titik (x, y) sebagai bagian dari class objek, $f(x, y)$ adalah tingkat keabuan pada titik (x, y) dan $p(x, y)$ menyatakan properti lokal dari titik tersebut. Aturan Iluminasi yaitu fungsi reflektan yang dibangkitkan komputer. Histogram pada fungsi ini ditunjukkan pada bimodal dan dapat di partisi dengan menempatkan single global threshold T pada lembah histogram. Perkalian fungsi reflektan oleh fungsi iluminasi ditunjukkan pada yang menghasilkan image yang menunjukkan histogram dari image. Ketika mengakses sumber iluminasi yang tersedia, sebuah solusi yang digunakan untuk mengkompensasi ketidakseragaman adalah memproyeksikan pola iluminasi ke dalam sebuah konstanta, permukaan putih yang memantulkan cahaya. Ini menghasilkan sebuah image $g(x, y) = k \cdot f(x, y)$ dimana k adalah sebuah konstanta yang tergantung pada permukaan dan $f(x, y)$ adalah pola iluminasi. $f(x, y)$ dibagi $g(x, y)$ menghasilkan fungsi normal $h(x, y) = f(x, y) / g(x, y) = f(x, y) / k$ jika $f(x, y)$ dapat disegmen menggunakan single threshold T , kemudian $h(x, y)$ dapat disegmen menggunakan single threshold dari T/k .

2.6.1 *Basic Global Thresholding*

Teknik thresholding paling sederhana adalah mempartisi histogram dengan menggunakan sebuah threshold global T . Segmentasi dilakukan dengan memeriksa piksel demi piksel dari citra dengan melabeli setiap piksel dengan objek atau sebagai background tergantung pada tingkat keabuan dari piksel tersebut, apakah lebih besar atau lebih kecil dari T . Thresholding global diharapkan bisa sukses pada lingkungan yang sangat terkontrol, misalnya pada pengawasan industri biasanya kontrol cahaya sangat bagus. Algoritma tersebut bisa digunakan untuk mendapatkan nilai T secara otomatis diantaranya :

1. Pilih estimasi awal untuk T .
2. Lakukan segmentasi terhadap citra menggunakan T . Proses ini akan menghasilkan kelompok piksel.
3. Hitung rata – rata tingkat keabuan u_1 dan u_2 untuk piksel – piksel dalam region G_1 dan G_2 .
4. Hitung nilai threshold yang baru

Jika area background dan area objek perbandingannya hampir sama, nilai awal yang baik untuk T adalah tingkat keabuan rata – rata dari citra. Jika area objek dibandingkan area background atau sebaliknya. Satu kelompok piksel akan mendominasi histogram sehingga nilai tengah tingkat keabuan maksimum dan minimum merupakan pilihan awal yang baik untuk T dengan algoritma iteratif dihasilkan $T = 125.4$ setelah tiga iterasi. T diinisialisasi dengan tingkat keabuan rata-rata dan $T_0 = 0$. Untuk mensegmentasi citra, digunakan nilai $T = 125$ dan optimal thresholding adalah threshold yang menghasilkan rata- rata eror segmentasi minimum. Salah satu metodenya adalah metode otsu, teknik otsu memaksimalkan kemiripan yang akan dapat memisahkan antara obyek dengan backgroundnya.

2.6.2 Basic Adapter Thresholding

Faktor imaging seperti pencahayaan yang tidak merata, menyebabkan histogram tidak bisa dipartisi dengan baik sehingga citra tidak bisa disegmentasi dengan menggunakan sebuah threshold global. Untuk mengatasi hal ini citra asal dibagi menjadi beberapa sub citra. Pada setiap sub citra segmentasi dilakukan dengan menggunakan threshold yang berbeda. Yang menjadi isu penting adalah bagaimana membagi citra dan bagaimana mengestimasi threshold untuk setiap sub citra karena threshold yang digunakan untuk setiap piksel tergantung pada lokasi piksel, maka threshold ini disebut adaptif. Semua sub citra yang tidak berisi boundary antara objek dan background memiliki variance kurang dari 75. Semua sub citra yang berisi boundary memiliki variance melebihi 100. Tiap citra dengan variance lebih besar dari 100 disegmentasi dengan threshold yang di hitung pada sub citra tersebut dengan algoritma iteratif, Nilai awal untuk T adalah titik tengah antara tingkat keabuan maksimum dan minimum dalam sub citra. Semua sub citra dengan variance kurang dari 100 diperlukan sebagai citra komposit yang disegmentasi menggunakan threshold tunggal dan diestimasi menggunakan algoritma yang sama.

Threshold optimal adalah threshold yang menghasilkan rata – rata error segmentasi minimum. Sebelum segmentasi semua citra di pra-proses ada berbagai langkah yaitu sebagai berikut :

1. Tiap piksel dipetakan dengan fungsi log untuk menghitung efek esponensial yang disebabkan penyerapan radioaktif.
2. Citra yang diambil sebelum pemberian medium kontras dikurangkan dengan citra yang diambil sesudah pemberian medium kontras, untuk memindahkan kolom spinal yang ada pada kedua citra.
3. Beberapa angiogram dijumlahkan untuk mengurangi noise random.

Untuk menghitung threshold optimal, citra dibagi menjadi 49 region dengan menempatkan grid berukuran 7 x 7 dengan overlap masing – masing citra sebesar 50 %. Masing – masing dari region overlap berisi 64 x 64 piksel, setelah 49 histogram di hitung maka dilakukan tes bimodality untuk menghilangkan histogram yang unimodal.

2.7 Algoritma Eigenface dan Fisherface

1. Algoritma Eigenface Untuk Pengenalan Wajah

Prinsip dasar pengenalan wajah adalah dengan mengutip informasi unik wajah tersebut kemudian di encode dan dibandingkan dengan hasil decode yang sebelumnya dilakukan, dalam metode eigenfaces decoding dilakukan dengan menghitung eigenvector kemudian direpresentasikan dalam sebuah matriks yang berukuran besar. Eigenvektor juga dinyatakan sebagai karakteristik wajah oleh karena itu mengapa metode ini disebut dengan eigenfaces, setiap wajah digambarkan dalam kombinasi linear eigenface, mengembangkan sebuah teknik yang menunjukkan wajah menggunakan analisa komponen, teknik tersebut menghitung koordinat sistem wajah disebut dengan eigenpicture, dalam perhitungan eigenvectro menjadi mean.

Algoritma Eigenfaces secara keseluruhan cukup sederhana Image Matriks (Γ) direpresentasikan kedalam sebuah himpunan matriks ($\Gamma_1, \Gamma_2, \dots, \Gamma_m$), cari nilai rata-rata (Ψ) yang digunakan untuk mengekstraksi eigenvector (v) dan eigenvalue (λ)

dari himpunan matriks, guna nilai eigenvektor untuk mendapatkan nilai eigenfaces dari image. Apabila ada sebuah pmage baru atau test faces (r_{new}) yang ingin dikenali, proses yang sama juga diberlakukan untuk images (r_{new}) untuk mengekstraksi eigenvektor (v) dan eigenvalue (λ), kemudian cari nilai eigenfaces image test face (r_{new}) barulah setelah itu iamge baru (r_{new}) memasuki tahapan pengenalan dengan menggunakan metode euclidean distance.

1. Buat *MakeFlatVectors (ImageList, N, M)*: Image List adalah kumpulan dari N training image, dimana setiap image adalah $W \times H$ 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 $N \times WH$ berisi semua gambar yang digabung.
3. Jumlahkan semua baris pada image matriks dan bagi dengan N untuk mendapatkan rata-rata vektor gabungan. Namakan vektor elemen WH dengan R .
4. Kurangi image matriks dengan average image R . Namakan matriks baru ukuran $N \times WH$ sebagai R' .
5. Jika pada elemen-elemen dari matriks R' ditemukan nilai negative, ganti nilainya dengan nilai 0.

Kemudian identifikasi dilakukan dengan proyeksi dengan algoritma sebagai berikut:

1. Buat *ProjectToFaceSpace(test_image)*: Image berukuran $W \times H$ Piksel.
2. Gabungkan elemen vektor WH dan namakan *img*.
3. Load nilai R dari database.
4. Kurangi *img* dengan R hingga didapat *img'*.
5. Jika pada *img'* ditemukan elemen dengan nilai negatif, ganti dengan nilai untuk mendapatkan vektor ukuran *img''*.

Proses terakhir adalah identifikasi yaitu dengan memproyeksikan *test image* ke *face space* dan menghitung *distance*:

1. Load semua wajah yang sudah diproyeksikan ke database
 2. $Proj = \text{projectToFaceSpace}(\text{test_image})$.
 3. Lakukan operasi pengurangan, $proj$ dengan semua wajah yang telah diproyeksikan. Ambil nilai absolutnya dan jumlahkan, hasil adalah *distance*.
 4. Ambil *distance* terkecil sebagai hasil dari wajah yang telah diproyeksikan. Wajah ini menjadi hasil identifikasi.
2. Algoritma Fisherface

Dasar dari metode *Fisherface* ini adalah *Fisher's Linear Discriminant* (FLD). FLD merupakan salah satu contoh metode *class specific*, karena metode ini berusaha untuk membentuk jarak (*scatter*) antar kelas dan intra kelas sehingga dapat menghasilkan klasifikasi yang lebih baik. Jika terdapat 20 sampel ($N = 20$) yang terbagi atas 2 kelas sampel. Dimensi sampel, $n = 2$ dan dimensi reduksi, $m = 1$. Jadi, sampel akan ditransformasikan dari ruang sampel dimensi-2 ke ruang ciri dimensi-1. Transformasi dilakukan dengan Metode PCA dan FLD. Hasil proyeksi vektor ciri sampel pada ruang ciri dimensi-1 dengan metode PCA dan FLD.

Konstruksi *fisherface* adalah pembuatan suatu set *fisherface* dari suatu set gambar *training* dengan menggunakan perhitungan PCA dan FLD. Perhitungan PCA dilakukan dengan langkah sebagai berikut :

1. Buat *MakeFlatVectors (ImageList, N, M)*: Image List adalah kumpulan dari N *training image*, dimana setiap image adalah $W \times H$ 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 $N \times WH$ berisi semua gambar yang digabung.
3. Jumlahkan semua baris pada *image* matriks dan bagi dengan N untuk mendapatkan rata-rata vektor gabungan. Namakan vektor elemen WH dengan R .

4. Kurangi *image* matriks dengan *average image R*. Namakan matriks baru ukuran $N \times WH$ sebagai R' .

Kemudian identifikasi dilakukan dengan menggunakan perhitungan FLD sebagai berikut:

1. Buat `ProjectToFaceSpace(test_image)`: Image berukuran $W \times H$ Piksel.
2. Transformasikan *training set* ke dalam vektor kolom.
3. Bentuk *average face* dari *facespace*
4. Hitung nilai deviasi *image*.
5. Eliminasi nilai terkecil *eigenvalue*.
6. Hitung *eigenvector* dan matriks kovarian.
7. Hitung matriks proyeksi *fisher* dengan mengurutkan vektor eigen berdasarkan besarnya nilai *eigen* masing-masing vektor *eigen* dan mengambil komponen vektor *eigen* yang memiliki nilai *eigen* tidak nol.
8. Hitung bobot tiap *fisherface* terhadap masing-masing gambar wajah pada *training set* dengan memproyeksikan nilai deviasi *facespace* terhadap *average face* kedalam matriks proyeksi optimal.

Proses terakhir adalah identifikasi yaitu dengan memproyeksikan *test image* ke *face space* dan menghitung *score*.

1. Load semua wajah yang sudah diproyeksikan ke database
2. `Proj=projectToFaceSpace(test_image)`.
3. Lakukan operasi pengurangan, `proj` dengan semua wajah yang telah diproyeksikan. Ambil nilai absolutnya dan jumlahkan, hasil adalah *distance*.
4. Hitung jarak *distance* antara bobot *image test* dan bobot *training set*.
5. Ambil *distance* terkecil sebagai hasil dari wajah yang telah diproyeksikan. Wajah ini menjadi hasil identifikasi.

BAB III

METODELOGI

3.1 Pengenalan dan Identifikasi Wajah

3.1.1 Pengambilan Foto Wajah

Data dalam penelitian ini, penulis menggunakan beberapa sample diantaranya dengan kriteria 3 orang wanita dan 6 orang pria dengan umur rata-rata 23 tahun.

Adapun beberapa pose dan aksesoris yang digunakan, untuk pose diantaranya:

1. Formal menghadap kedepan tanpa senyum.
2. Formal menghadap kesamping tanpa senyum.
3. Formal menghadap kedepan dengan senyum
4. Formal menghadap kedepan dengan tertawa

Dengan menggunakan aksesoris topi:

1. Memakai topi menghadap kedepan tanpa senyum.
2. Memakai topi menghadap samping tanpa senyum.
3. Memakai topi menghadap kedepan dengan senyum.
4. Memakai topi menghadap kedepan dengan tertawa.

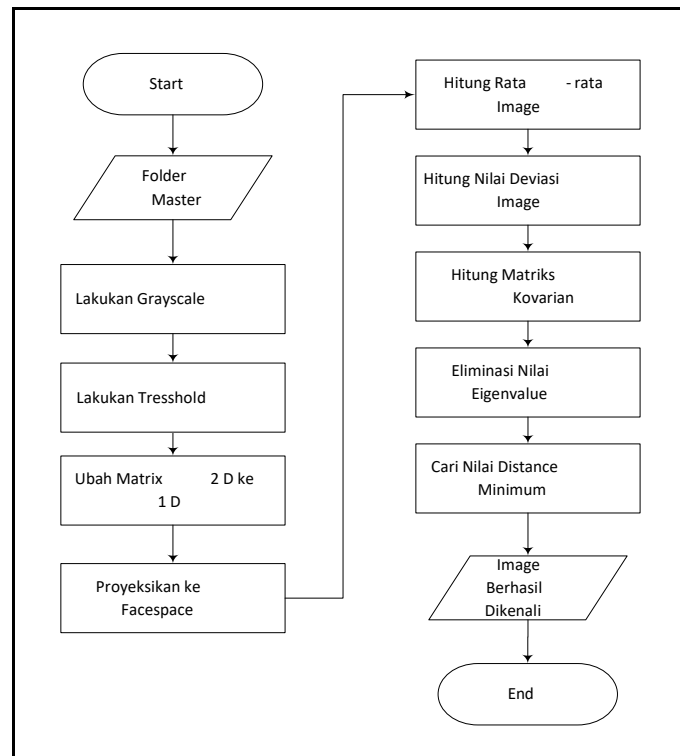
Dengan menggunakan aksesoris kacamata:

1. Memakai kacamata menghadap kedepan tanpa senyum.
2. Memakai kacamata menghadap samping tanpa senyum.
3. Memakai kacamata menghadap kedepan dengan senyum.
4. Memakai kacamata menghadap kedepan dengan tertawa.

Dengan menggunakan masker:

1. Memakai masker menghadap kedepan.

3.1.2 Proses Identifikasi Pengenalan Wajah Dengan Algoritma Eigenface



Gambar III.1

Flowchart Eigenface

Untuk identifikasi pengenalan wajah dengan menggunakan algoritma eigenface melalui tahapan-tahapan sebagai berikut:

1. Folder Master

Folder master merupakan folder yang berisikan tentang data-data foto yang dijadikan sebagai acuan atau pembanding atas data foto yang akan diuji identitasnya dan folder master terdiri dari empat pose tanpa menggunakan aksesoris menghadap kedepan tanpa senyum, menghadap samping, menghadap depan tersenyum dan menghadap depan dengan tertawa.

2. Lakukan grayscale

Dari keseluruhan foto yang berada pada folder foto dilakukan konversi dari foto yang mengandung warna RGB awalnya menjadi foto yang tampak keabuan yang disebut dengan Grayscale.

3. Lakukan Treshold

Dari keseluruhan foto yang berada di folder master yang sudah mengalami perubahan menjadi Grayscale maka diubah kembali dalam pencitraan Treshold yang menghasilkan gambar hitam dan putih.

4. Ubah matriks 2D ke 1 D

Dalam pencitraan hitam putih atau Treshold maka foto yang berada folder master dapat dikonversi sebagai bilangan matriks dengan gambar berukuran Piksel dikalikan dengan jumlah foto yang berada dalam folder master tersebut, sehingga menghasilkan matriks tunggal untuk mendapatkan nilai rata-rata flat factor.

5. Proyeksikan kefacespace

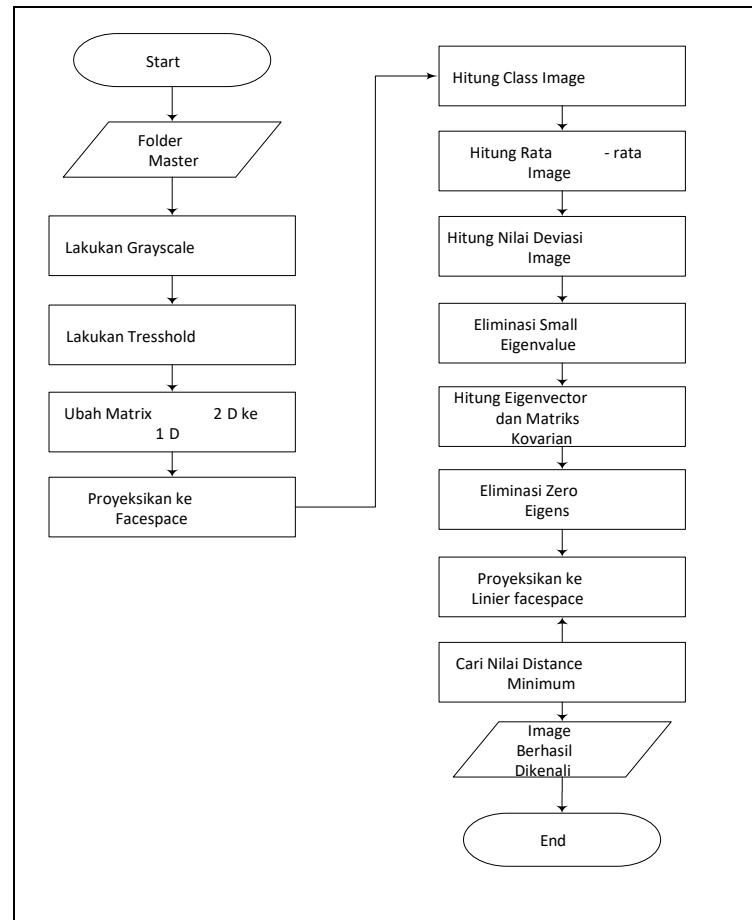
Setelah mengalami perhitungan untuk mendapatkan nilai rata-rata flatfactor maka dapat dicari nilai suatu Eigenface dengan mengurangi nilai 1D matriks foto treshold dengan nilai rata-rata flatfactor dan menghasilkan bilangan 1 dan 0. Yang tersusun membentuk eigenvalue.

6. Cari nilai distance minimum

Proses ini adalah proses dimana perbandingan dari foto yang akan diuji dengan mendapatkan nilai eigenface yang menghasilkan matriks bilangan 0 dan 1 sehingga dapat diperoleh jarak terpendek.

Setelah mendapatkan nilai jarak terpendek dan dapat dihasilkan kemiripan antara foto yang diuji dengan foto yang berada pada master foto.

3.1.3 Proses Identifikasi Pengenalan Wajah Dengan Algoritma Fisherface



Gambar III.2

Flowchart Fisherface

Untuk identifikasi pengenalan wajah dengan menggunakan algoritma fisherface melalui tahapan-tahapan sebagai berikut:

1. Folder Master

Folder master merupakan folder yang berisikan tentang data-data foto yang dijadikan sebagai acuan atau pembanding atas data foto yang akan diuji identitasnya dan folder master terdiri dari empat pose tanpa menggunakan aksesoris menghadap kedepan tanpa senyum, menghadap samping, menghadap depan tersenyum dan menghadap depan dengan tertawa.

2. Lakukan grayscale

Dari keseluruhan foto yang berada pada folder foto dilakukan konversi dari foto yang mengandung warna RGB awalnya menjadi foto yang tampak keabuan yang disebut dengan Grayscale.

3. Lakukan Treshold

Dari keseluruhan foto yang berada di folder master yang sudah mengalami perubahan menjadi Grayscale maka diubah kembali dalam pencitraan Treshold yang menghasilkan gambar hitam dan putih.

4. Ubah matriks 2D ke 1 D

Dalam pencitraan hitam putih atau Treshold maka foto yang berada folder master dapat dikonversi sebagai bilangan matriks dengan gambar berukuran Piksel dikalikan dengan jumlah foto yang berada dalam folder master tersebut, sehingga menghasilkan matriks tunggal untuk mendapatkan nilai rata-rata flat factor.

5. Proyeksikan kefacespace

Dalam proyeksi faceface pada algoritma fisherface maka melalui berbagai macam tahapan, diantaranya:

- a. Hitung class image merupakan tahapan perhitungan image berdasarkan class yang lebih dari satu image.
- b. Hitung rata-rata image merupakan rata-rata dari class image yang telah mengalami gabungan dari class foto yang berada pada master foto.
- c. Hitung nilai deviasi image merupakan nilai yang didapat dari pengurangan nilai matriks baru dari rata-rata dengan matriks.
- d. Eliminasi small eigenvalue merupakan pencarian nilai terkecil dari suatu nilai matriks baru.

6. Cari nilai distance umum

Membandingkan nilai euclidean distance minimal dari image yang dijadikan sebagai pengujian foto dengan menentukan jarak minimum dari eigenface.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Data Foto Identifikasi wajah manusia

1. Foto Master

Dalam penelitian identifikasi wajah data foto digolongkan dalam berbagai macam kategori yang dijadikan sebagai foto master yang akan ditempatkan pada folder Master foto diantaranya:



Gambar IV.1

Pose Formal Menghadap Kedepan Tanpa Senyum

foto diatas merupakan foto yang dijadikan foto master memiliki fungsi sebagai identitas awal seorang dengan pose formal tanpa senyum.



Gambar IV.2

Pose Formal Menghadap Kesamping Tanpa Senyum

Foto diatas merupakan foto master yang dijadikan sebagai bahan acuan dengan pose formal menghadap kesamping tanda senyum.



Gambar IV.3

Pose Formal Menghadap Kedepan Dengan Senyum

Foto diatas merupaka foto master yang dijadikan sebagai bahan acuan dengan pose formal menghadap kedepan dengan senyum.



Gambar IV.4

Pose Formal Menghadap Kedepan Dengan Tertawa

2. Foto-Foto yang Akan Diujicobakan

Setelah menggolongkan folder master yang sudah memiliki identitas, maka akan dilakukan pengujian keakurasian dalam pengenalan wajah dengan menggunakan beberapa aksesoris diantaranya dengan menggunakan topi, kacamata dan masker, dengan menggunakan objek yang sama dan pose yang sama pula akan dilakukan pengujiannya berikut adalah foto-foto yang akan diujikan dengan berbagai macam aksesoris diantaranya:

a. Dengan menggunakan aksesoris topi



Gambar IV.5

Memakai Topi Menghadap Kedepan Tanpa Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa topi dengan pose menghadap kedepan dengan formal tanpa senyum yang akan dibandingkan dengan foto master.



Gambar IV.6

Memakai Topi Menghadap Kesamping Tanpa Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa topi dengan pose menghadap kesamping dengan formal tanpa senyum yang akan dibandingkan dengan foto master.



Gambar IV.7

Memakai Topi Menghadap Kedepan Dengan Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa topi dengan pose menghadap kedepan dengan formal dengan senyum yang akan dibandingkan dengan foto master.



Gambar IV.8

Memakai Topi Menghadap Kedepan Dengan Tertawa

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa topi dengan pose menghadap kedepan dengan tertawayang yang akan dibandingkan dengan foto master.

b. Dengan aksesoris kaca mata



Gambar IV.9

Memakai Kacamata Menghadap Kedepan Tanpa Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa kaca mata dengan pose menghadap kedepan tanpa senyum yang akan dibandingkan dengan foto master.



Gambar IV.10

Memakai Kacamata Menghadap Samping Tanpa Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa kaca mata dengan pose menghadap kesamping tanpa senyum yang akan dibandingkan dengan foto master.



Gambar IV.11

Memakai Kacamata Menghadap Kedepan Dengan Senyum

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa kaca mata dengan pose menghadap kedepan dengan senyum yang akan dibandingkan dengan foto master



Gambar IV.12

Memakai Kacamata Menghadap Kedepan Dengan Tertawa

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa kaca mata dengan pose menghadap kedepan dengan tertawa yang akan dibandingkan dengan foto master

c. Dengan menggunakan aksesoris masker

Selain dengan menggunakan kacamata, maka pengujian foto diujikan dengan menggunakan aksesoris masker, diantaranya sebagai berikut:



Gambar IV.13

Memakai Masker Menghadap Kedepan

Foto diatas merupakan pengujian objek dengan menggunakan aksesoris berupa masker dengan pose menghadap kedepan tanpa senyum yang akan dibandingkan dengan foto master.

Dalam penelitian ini dibutuhkan satu foto yang masing-masing berukuran 3x3 yang dijadikan sebagai acuan atau sebagai pembanding dari foto yang akan dikenali, berada pada folder Master foto yang berisikan data sebagai berikut:

1. Master Foto dikenali sebagai Andika



Gambar IV.14

Foto Master Sebagai Andika

2. Master Foto dikenali sebagai Andry



Gambar IV.15

Foto Master Sebagai Andry

3. Master Foto dikenali sebagai Candra



Gambar IV.16

Foto Master Sebagai Candra

4. Master Foto dikenal sebagai Dessi



Gambar IV.17
Foto Master Sebagai Dessi

5. Master Foto dikenal sebagai Fauzi



Gambar IV.18
Foto Master Sebagai Fauzi

6. Master Foto dikenal sebagai Nawawi



Gambar IV.19
Foto Master Sebagai Nawawi

7. Master Foto dikenal sebagai Rizal



Gambar IV.20

Foto Master Sebagai Rizal

8. Master Foto dikenal sebagai Uswatun



Gambar IV.21

Foto Master Sebagai Uswatun

9. Master Foto dikenal sebagai Yunita



Gambar IV.22

Foto Master Sebagai Yunita

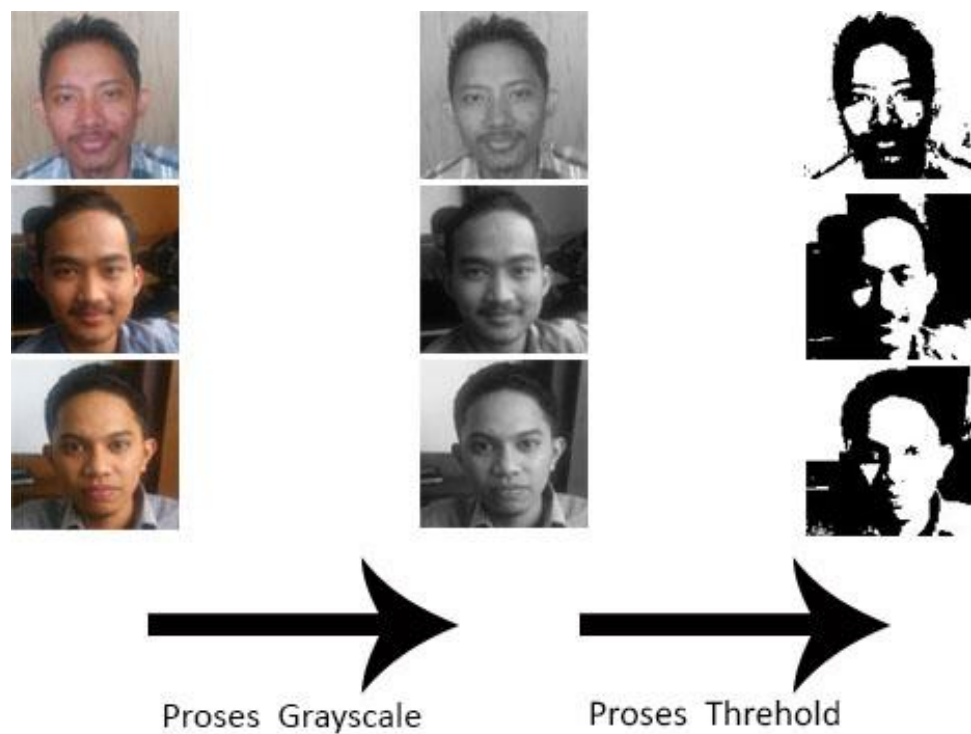
4.2 Metode Algoritma Eigenface dan Fisherface Untuk Identifikasi wajah

4.2.1 Algoritma Eigenface Untuk Pengenalan Wajah

Untuk lebih jelasnya algoritma *eigenface* dapat diuraikan sebagai berikut :

1. Proses *Image Test*

Langkah pertama setelah didapatkan citra wajah hasil *capture*, citra wajah dirubah ke dari bentuk RGB ke dalam bentuk *grayscale*, setelah didapatkan citra keabuan ubah menjadi citra hitam putih dengan melakukan *treshold* agar kompleksitas citra lebih sederhana.

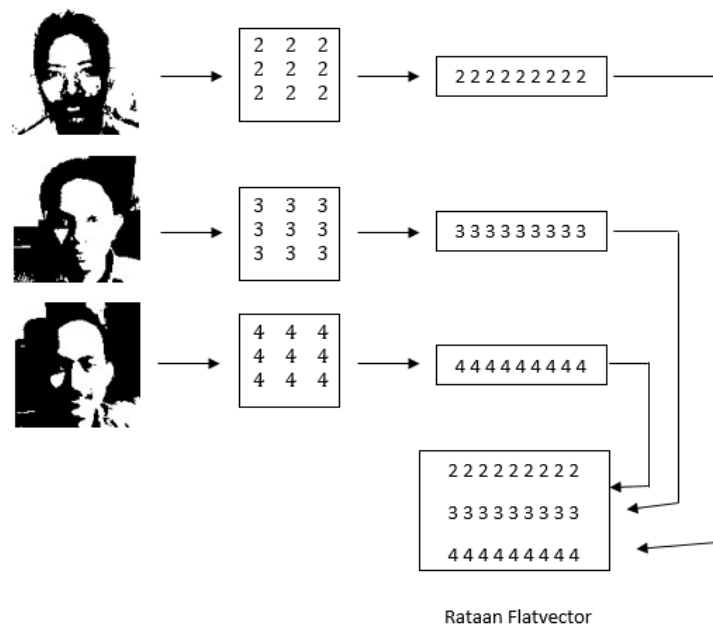


Gambar IV.23

Mengubah Proses Citra RGB, Grayscale dan threhold

2. Penyusunan *flatvector*

Selanjutnya adalah menyusun seluruh *training image* menjadi suatu matriks tunggal. Misalnya *image* yang disimpan berukuran $H \times W$ piksel dan jumlahnya N buah, maka akan dimiliki *vector* ciri dengan dimensi $N \times (W \times H)$. Misalnya dalam *training image* terdapat 3 *image* dengan ukuran 3×4 piksel maka kita akan mempunyai *eigenvector* ukuran 3×9 . Ilustrasinya sebagai berikut:

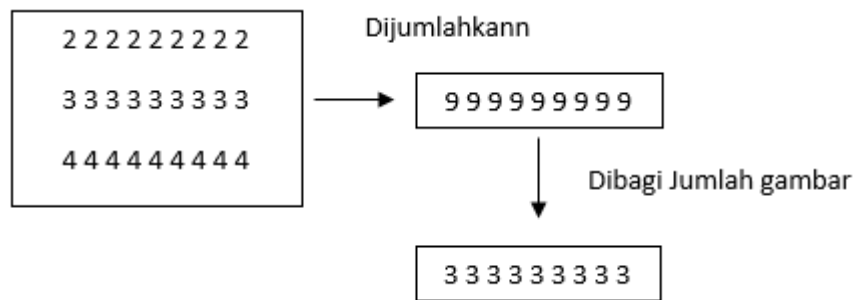


Gambar IV.24

Penyusunan Flatvector

3. Perhitungan Rataan Flatvector

Dari vektor ciri yang diperoleh, jumlah seluruh barisannya sehingga diperoleh matriks berukuran $1 \times (W \times H)$. Setelah itu bagi matriks tersebut dengan jumlah image N untuk mendapatkan nilai rata-rata vektor ciri.

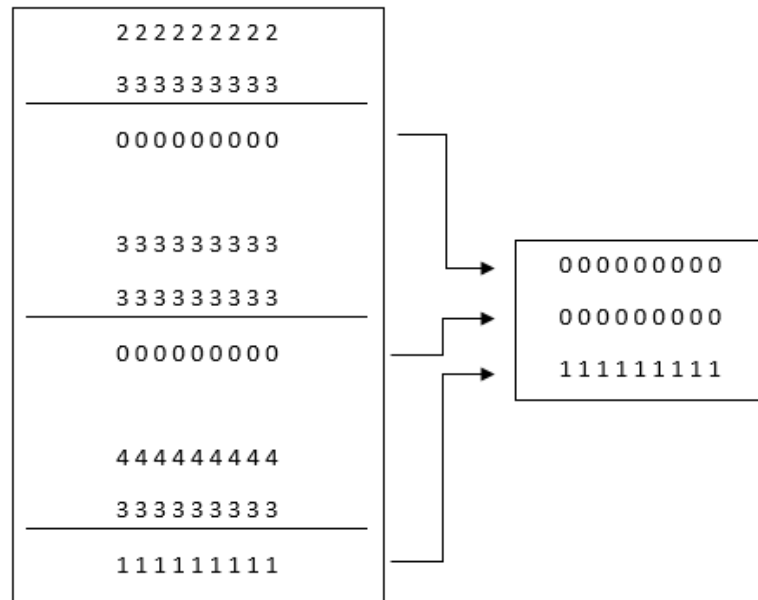


Gambar IV.25

Nilai Rata-rata Flatfactor

4. Proyeksi image ke facespace

Image diproyeksi ke facespace dengan mengkalikan dibasis eigenface proyeksi vektor wajah akan dibandingkan dengan vektor yang sesuai dengan menggunakan nilai rata-rata vektor ciri, akan dihitung eigenface untuk matriks vektor ciri dengan nilai rata-rata vektor ciri, jika didapatkan nilai dibawah nol, ganti nilainya dengan nol.

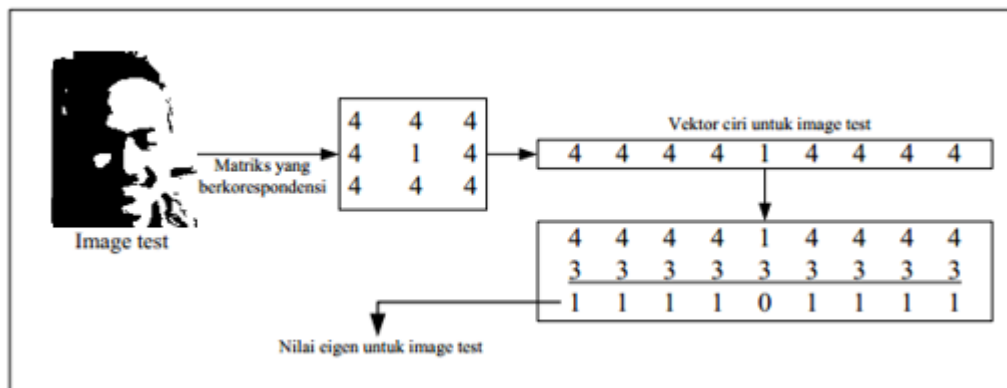


Gambar IV.26

Perhitungan Eigenface

5. Ekstraksi PCA image test

Hasil proyeksi tersebut diekstraksi dengan perhitungan PCA untuk mendapatkan feature dari image, fitur adalah komponen-komponen penting dari image-image training yang didapatkan dari proses training. Fitur inilah yang nanti akan digunakan untuk mengidentifikasi image yang akan dikenali. Kalkulasi nilai eigenface untuk matriks testface, dengan cara yang sama dengan penentuan eigenface untuk vektor ciri.

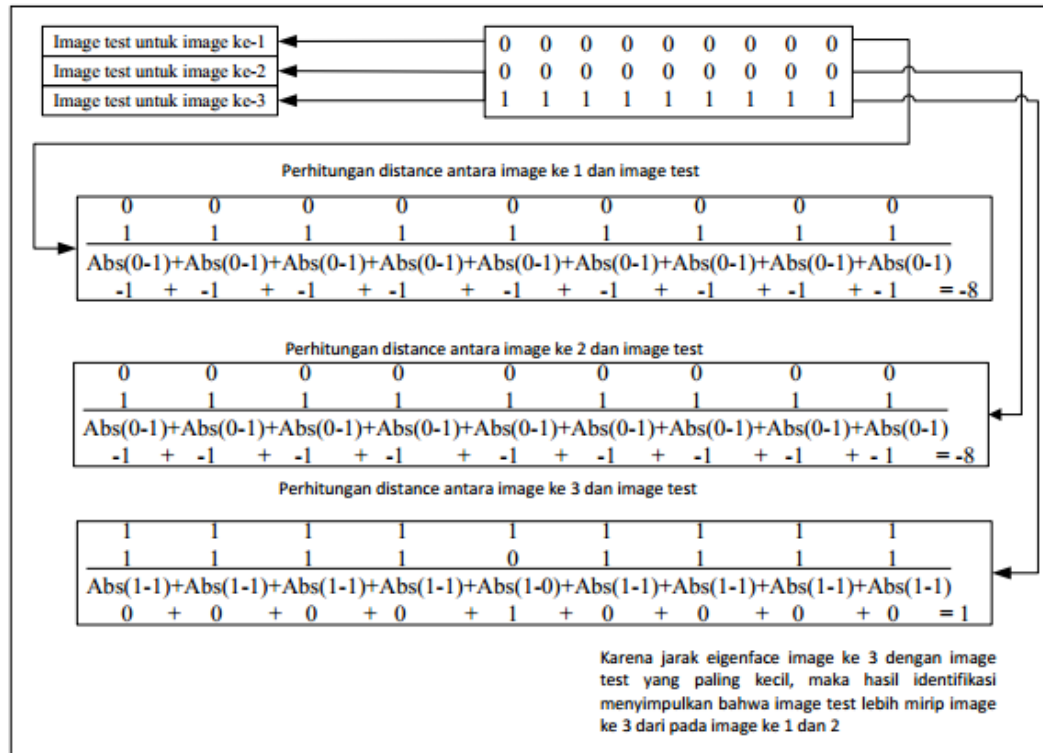


Gambar IV.27

Perhitungan eigenface untuk test image

6. Cari distance minimum antara image test dan image hasil image

Bandingkan nilai euclidean distance minimum dari image yang di capture dengan image yang sudah ada didatabase. Setelah nilai eigenface untuk test image diperoleh maka kita akan melakukan pengenalan dengan menentukan jarak (distance) terpendek dengan dengan eigenface dari eigenvektro training image, caranya dengan menentukan nilai absolute dari pengurangan baris i pada matriks eigen training image dengan eigenface dari testface, kemudian jumlahkan elemen-elemen penyusun vektor yang dihasilkan dari pengurangan tadi dan ditentukan jarak d indeks i. Lakukan untuk semua baris , cari nilai d yang paling kecil.



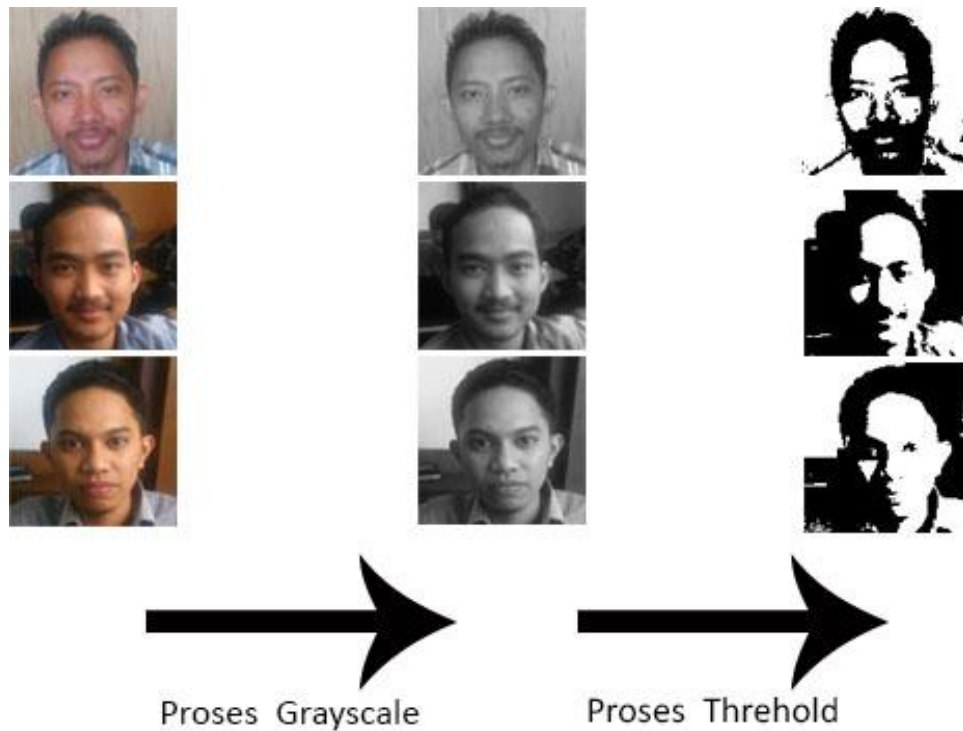
Gambar IV.28

Proses Identifikasi dengan input image test

4.2.2 Algoritma Fisherface

1. Proses Image Test

Langkah pertama setelah didapatkan citra wajah hasil *capture*, citra wajah dirubah ke dari bentuk RGB ke dalam bentuk *grayscale*, setelah didapatkan citra keabuan ubah menjadi citra hitam putih dengan melakukan *treshold* agar kompleksitas citra lebih sederhana.

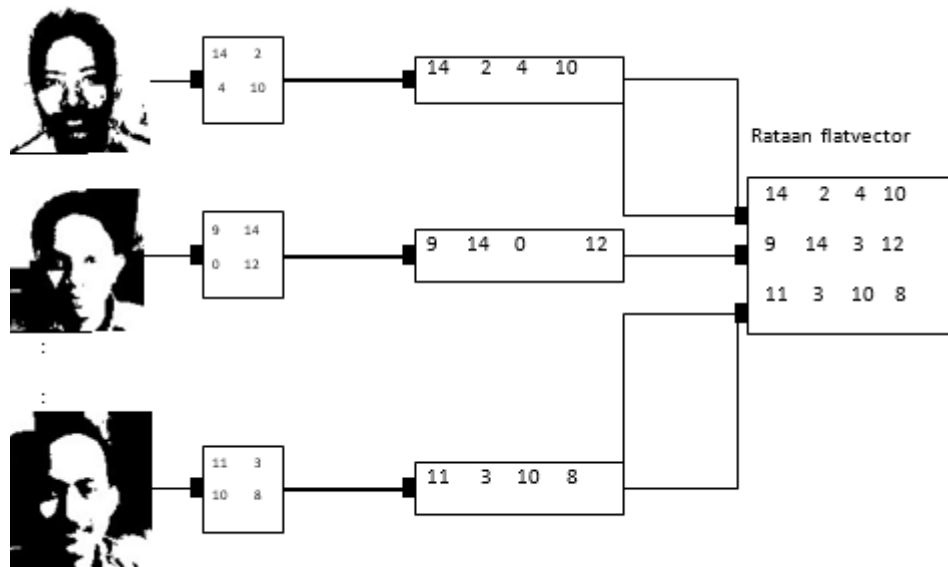


Gambar IV.29

Proses Mengubah Citra RGB, Grayscale dan Threhold

2. Penyusunan *flatvector*

Langkah pertama adalah menyusun seluruh *training image* menjadi suatu matriks tunggal. Misalnya *image* yang disimpan berukuran $H \times W$ piksel dan jumlahnya N buah, maka akan dimiliki *vector* ciri dengan dimensi $N \times (W \times H)$. Misalnya dalam *training image* terdapat 3 *image* dengan ukuran 3 x 4 piksel maka kita akan mempunyai *eigenvector* ukuran 3 x 9. sebagai berikut :



Gambar IV.30

Penyusunan Flatfactor

3. Perhitungan Tingkat Kemiripan

Dari *vector* ciri yang telah diperoleh, jumlahkan seluruh barisnya sehingga diperoleh matriks berukuran $1 \times (W \times H)$. Setelah itu bagi matriks tersebut dengan jumlah *image* N untuk mendapatkan nilai rata-rata *vector* ciri, ilustrasinya sebagai berikut :

Misalkan kita memiliki data sample random sejumlah 7 orang dengan 2 pose.

Tabel IV.1

Ciri dari 7 orang dengan 2 pose

Orang ke	j	$f_{j,1}$	$f_{j,2}$	$f_{j,3}$	$f_{j,4}$
1	1	14	4	4	10
	2	3	2	2	5
2	3	9	0	0	12
	4	7	11	11	7
3	5	13	6	6	10
	6	11	13	13	6
4	7	6	6	6	4

Orang ke	j	$f_{j,1}$	$f_{j,2}$	$f_{j,3}$	$f_{j,4}$
	8	0	6	6	2
5	9	12	12	12	2
	10	6	7	7	10
6	11	9	3	3	4
	12	11	10	10	8
7	13	13	12	12	2
	14	11	0	0	10

Misalkan citra yang diuji mempunyai ciri ke 1, 2, 3 dan 4 dan masing masing adalah 7, 6, 10, 9. Maka beda kuadrat dari masing-masing ciri pelatihan terhadap citra uji dapat dilihat pada tabel III.1, dengan persamaan:

$$(f_{j,i} - f_{uji,i})^2$$

Maka didapatkan nilai beda kuadrat dari masing-masing ciri pelatihan terhadap citra uji:

Tabel IV.2

Beda Kuadrat dari Masing-masing ciri Pelatihan terhadap Citra Uji

No	Ciri ke-1	Ciri ke-2	Ciri ke-3	Ciri ke-4
1	$(14-7)^2$	$(2-6)^2$	$(4-10)^2$	$(10-9)^2$
2	$(3-7)^2$	$(6-6)^2$	$(2-10)^2$	$(5-9)^2$
3	$(9-7)^2$	$(14-6)^2$	$(0-10)^2$	$(12-9)^2$
4	$(7-7)^2$	$(13-6)^2$	$(11-10)^2$	$(7-9)^2$
5	$(13-7)^2$	$(6-6)^2$	$(6-10)^2$	$(10-9)^2$
6	$(11-7)^2$	$(13-6)^2$	$(13-10)^2$	$(6-9)^2$
7	$(6-7)^2$	$(0-6)^2$	$(6-10)^2$	$(4-9)^2$
8	$(0-7)^2$	$(5-6)^2$	$(6-10)^2$	$(2-9)^2$
9	$(12-7)^2$	$(12-6)^2$	$(12-10)^2$	$(2-9)^2$
10	$(6-7)^2$	$(0-6)^2$	$(7-10)^2$	$(10-9)^2$
11	$(9-7)^2$	$(2-6)^2$	$(3-10)^2$	$(4-9)^2$
12	$(11-7)^2$	$(3-6)^2$	$(10-10)^2$	$(8-9)^2$
13	$(13-7)^2$	$(2-6)^2$	$(12-10)^2$	$(2-9)^2$
14	$(11-7)^2$	$(9-6)^2$	$(0-10)^2$	$(10-9)^2$

Setelah didapatkan nilai kuadrat dari masing-masing ciri pelatihan terhadap citra uji. Jumlahkan secara keseluruhan pada tiap-tiap ciri pada masing-masing citra dengan persamaan.

$$(f_{j,1} - f_{uji,1})^2 + (f_{j,2} - f_{uji,2})^2 + (f_{j,3} - f_{uji,3})^2 + (f_{j,4} - f_{uji,4})^2$$

Sehingga diperoleh nilai seperti dibawah ini:

Tabel IV.3

Hasil Penjumlahan Beda Kuadrat Dari Masing-masing Pelatihan terhadap Citra Uji

No	Ciri ke-1	Ciri ke-2	Ciri ke-3	Ciri ke-4	Jumlah
1	49	16	36	1	102
2	16	0	64	16	96
3	4	64	100	9	177
4	0	49	1	4	54
5	36	0	16	1	53
6	16	49	9	9	83
7	1	36	16	25	78
8	49	1	16	49	115
9	25	36	4	49	114
10	1	36	9	1	47
11	4	16	49	25	94
12	16	9	0	1	26
13	36	16	4	49	105
14	16	9	100	1	126

Hasil perhitungan selanjutnya akan diakarkan, maka akan menghasilkan nilai seperti tabel III.4 , pada tabel III.4 tampak bahwa nilai minimal terletak pada data ke-12 data ke-12 merupakan orang ke 6 pose ke 2. Sehingga data yang diuji tersebut dikenali sebagai orang yang ke-6 pose ke-2.

4. Cari distance minimal antara image test dan image hasil training

Bandingkan nilai euclidean distance minimal dari image yang di capture dengan image yang sudah ada di data base, setelah nilai eigenface untuk image test diperoleh maka kita bisa melakukan identifikasi dengan menentukan jarak minimum dengan eigenface dari eigenvector training image, dengan persamaan:

$$d_v = \sum_{j=1}^m \|F_{trainke\ j,w} - F_{uji,w}\|$$

Maka akan didapatkan nilai distance minimal dari setiap image. Seperti terlihat tabel dibawah ini:

Tabel IV.4
Hasil Akhir Perhitungan Tingkat Kemiripan Menggunakan Euclidean

No	Jumlah Ciri	Nilai <i>Euclidean Distance</i>	Hasil
1	102	10.1	
2	96	9.798	
3	177	13.304	
4	54	7.3485	
5	53	9.1104	
6	83	8.8318	
7	78	10.724	
8	115	6.8557	
9	114	9.9654	
10	47	5.099	Minimal
11	94	10.247	
12	26	11.225	

4.3. Percobaan Pengenalan Wajah Dalam Bentuk Tabel

Dengan menggunakan Matlab maka didapatkan hasil kebenaran dari data foto yang diujikan yang mengacu pada foto master, adapun hasil pengujian kebenarannya adalah sebagai berikut:

Percobaan untuk Foto yang terdeteksi sebagai Andika:

1. Uji Foto tanpa aksesoris atau formal Andika.

Tabel IV.5

Uji coba foto formal Andika

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Andika	Andika	Benar Benar
Samping tanpa senyum	Andika	Andika	Benar Benar
Depan dengan senyum	Andika	Andika	Benar Benar
Depan dengan tertawa	Andika	Andika	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andika perbandingan tersebut dilakukan antara foto master dan foto master

2. Uji Foto aksesoris Topi Andika.

Tabel IV.6

Uji coba foto aksesoris topi foto Andika

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Andika	Yunita	Benar Salah
Samping tanpa senyum	Andry	Nawawi	Salah Salah
Depan dengan senyum	Andry	Yunita	Salah Salah
Depan dengan tertawa	Andika	Yunita	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andika perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Andika:

Tabel IV.7
Uji coba foto aksesoris kacamata foto Andika

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Nawawi	Andika	Salah Benar
Samping tanpa senyum	Nawawi	Andika	Salah Benar
Depan dengan senyum	Nawawi	Andika	Salah Benar
Depan dengan tertawa	Andika	Andika	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andika perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Andika:

Tabel IV.8
Uji coba foto aksesoris masker foto Andika

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan Formal	Nawawi	Andry	Salah Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andika perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Andry:

1. Uji Foto tanpa aksesoris atau formal Andry.

Tabel IV.9

Uji coba foto formal Andry

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Andry	Andry	Benar Benar
Samping tanpa senyum	Andry	Andry	Benar Benar
Depan dengan senyum	Andry	Andry	Benar Benar
Depan dengan tertawa	Andry	Andry	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andry perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Andry.

Tabel IV.10

Uji coba foto aksesoris topi foto Andry

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Andry	Andry	Benar Benar
Samping tanpa senyum	Andry	Rizal	Benar Salah
Depan dengan senyum	Andry	Yunita	Benar Salah
Depan dengan tertawa	Candra	Andry	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andry perbandingan tersebut dilakukan antara foto master dan foto dengan menggunakan aksesoris topi.

3. Uji foto aksesoris kacamata Andry:

Tabel IV.11
Uji coba foto aksesoris kacamata foto Andry

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Nawawi	Andry	Salah Benar
Samping tanpa senyum	Andry	Andika	Benar Salah
Depan dengan senyum	Nawawi	Andry	Salah Benar
Depan dengan tertawa	Rizal	Andry	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andry perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Andry:

Tabel IV.12
Uji coba foto aksesoris masker foto Andry

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Rizal	Rizal	Salah Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Andry perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Candra:

1. Uji Foto tanpa aksesoris atau formal Candra.

Tabel IV.13

Uji coba foto formal Candra

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Candra	Benar Benar
Samping tanpa senyum	Candra	Candra	Benar Benar
Depan dengan senyum	Candra	Candra	Benar Benar
Depan dengan tertawa	Candra	Candra	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Candra perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Candra.

Tabel IV.14

Uji coba foto aksesoris topi foto Candra

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Andry	Benar Salah
Samping tanpa senyum	Candra	Rizal	Benar Salah
Depan dengan senyum	Candra	Rizal	Benar Salah
Depan dengan tertawa	Candra	Rizal	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Candra perbandingan tersebut dilakukan antara foto master dan foto aksesoris foto.

3. Uji foto aksesoris kacamata Candra:

Tabel IV.15
Uji coba foto aksesoris kacamata foto Candra

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Candra	Benar Benar
Samping tanpa senyum	Candra	Rizal	Benar Salah
Depan dengan senyum	Candra	Rizal	Benar Salah
Depan dengan tertawa	Candra	Candra	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Candra perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Candra:

Tabel IV.16
Uji coba foto aksesoris masker foto Candra

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Rizal	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Candra perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Dessi:

1. Uji Foto tanpa aksesoris atau formal Dessi.

Tabel IV.17

Uji coba foto formal Dessi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Dessi	Dessi	Benar Benar
Samping tanpa senyum	Dessi	Dessi	Benar Benar
Depan dengan senyum	Dessi	Dessi	Benar Benar
Depan dengan tertawa	Dessi	Dessi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Dessi perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Dessi.

Tabel IV.18

Uji coba foto aksesoris topi foto Dessi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Dessi	Dessi	Benar Benar
Samping tanpa senyum	Nawawi	Dessi	Salah Benar
Depan dengan senyum	Dessi	Dessi	Salah Benar
Depan dengan tertawa	Dessi	Dessi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Dessi perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Dessi:

Tabel IV.19
Uji coba foto aksesoris kacamata foto Dessi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Dessi	Dessi	Benar Benar
Samping tanpa senyum	Nawawi	Dessi	Salah Benar
Depan dengan senyum	Dessi	Dessi	Benar Benar
Depan dengan tertawa	Fauzi	Dessi	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Dessi perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Dessi:

Tabel IV.20
Uji coba foto aksesoris masker foto Dessi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Fauzi	Dessi	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Dessi perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Fauzi:

1. Uji Foto tanpa aksesoris atau formal Fauzi.

Tabel IV.21

Uji coba foto formal Fauzi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Fauzi	Fauzi	Benar Benar
Samping tanpa senyum	Fauzi	Fauzi	Benar Benar
Depan dengan senyum	Fauzi	Fauzi	Benar Benar
Depan dengan tertawa	Fauzi	Fauzi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Fauzi perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Fauzi.

Tabel IV.22

Uji coba foto aksesoris topi foto Fauzi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Fauzi	Fauzi	Benar Benar
Samping tanpa senyum	Fauzi	Dessi	Benar Salah
Depan dengan senyum	Fauzi	Fauzi	Benar Benar
Depan dengan tertawa	Fauzi	Fauzi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Fauzi perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Fauzi:

Tabel IV.23
Uji coba foto aksesoris kacamata foto Fauzi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Fauzi	Fauzi	Benar Benar
Samping tanpa senyum	Fauzi	Fauzi	Benar Benar
Depan dengan senyum	Fauzi	Fauzi	Benar Benar
Depan dengan tertawa	Fauzi	Fauzi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Fauzi perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Fauzi:

Tabel IV.24
Uji coba foto aksesoris masker foto Fauzi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Fauzi	Fauzi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Fauzi perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Nawawi:

1. Uji Foto tanpa aksesoris atau formal Nawawi.

Tabel IV.25

Uji coba foto formal Nawawi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Nawawi	Nawawi	Benar Benar
Samping tanpa senyum	Nawawi	Nawawi	Benar Benar
Depan dengan senyum	Nawawi	Nawawi	Benar Benar
Depan dengan tertawa	Nawawi	Nawawi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Nawawi perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Nawawi.

Tabel IV.26

Uji coba foto aksesoris topi foto Nawawi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Nawawi	Andika	Benar Salah
Samping tanpa senyum	Andry	Andika	Salah Salah
Depan dengan senyum	Nawawi	Nawawi	Benar Benar
Depan dengan tertawa	Nawawi	Nawawi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Nawawi perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Nawawi:

Tabel IV.27
Uji coba foto aksesoris kacamata foto Nawawi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Nawawi	Nawawi	Benar Benar
Samping tanpa senyum	Andry	Nawawi	Salah Benar
Depan dengan senyum	Nawawi	Nawawi	Benar Benar
Depan dengan tertawa	Nawawi	Nawawi	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Nawawi perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Nawawi:

Tabel IV.28
Uji coba foto aksesoris masker foto Nawawi

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Rizal	Salah Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Nawawi perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Rizal:

1. Uji Foto tanpa aksesoris atau formal Rizal.

Tabel IV.29

Uji coba foto formal Rizal

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Rizal	Rizal	Benar Benar
Samping tanpa senyum	Rizal	Rizal	Benar Benar
Depan dengan senyum	Rizal	Rizal	Benar Benar
Depan dengan tertawa	Rizal	Rizal	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Rizal perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Rizal.

Tabel IV.30

Uji coba foto aksesoris topi foto Rizal

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Rizal	Rizal	Benar Benar
Samping tanpa senyum	Rizal	Rizal	Benar Benar
Depan dengan senyum	Candra	Rizal	Salah Benar
Depan dengan tertawa	Rizal	Candra	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Rizal perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Rizal:

Tabel IV.31
Uji coba foto aksesoris topi kacamata Rizal

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Rizal	Candra	Benar Salah
Samping tanpa senyum	Candra	Rizal	Salah Benar
Depan dengan senyum	Candra	Rizal	Salah Benar
Depan dengan tertawa	Rizal	Rizal	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Rizal perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Rizal:

Tabel IV.32
Uji coba foto aksesoris masker foto Rizal

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Candra	Rizal	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Rizal perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

Percobaan untuk Foto yang terdeteksi sebagai Uswatun:

1. Uji Foto tanpa aksesoris atau formal Uswatun.

Tabel IV.33

Uji coba foto formal Uswatun

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Uswatun	Uswatun	Benar Benar
Samping tanpa senyum	Uswatun	Uswatun	Benar Benar
Depan dengan senyum	Uswatun	Uswatun	Benar Benar
Depan dengan tertawa	Uswatun	Uswatun	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Uswatun perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji foto aksesoris kacamata Uswatun:

Tabel IV.34

Uji coba foto aksesoris topi kacamata Uswatun

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Uswatun	Uswatun	Benar Benar
Samping tanpa senyum	Uswatun	Nawawi	Benar Salah
Depan dengan senyum	Uswatun	Uswatun	Benar Benar
Depan dengan tertawa	Yunita	Uswatun	Salah Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Uswatun perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris masker Uswatun:

Tabel IV.35
Uji coba foto aksesoris masker foto Uswatun

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Uswatun	Rizal	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Uswatun perbandingan tersebut dilakukan antara foto master dan foto master.

Percobaan untuk Foto yang terdeteksi sebagai Yunita:

1. Uji Foto tanpa aksesoris atau formal Yunita.

Tabel IV.36
Uji coba foto formal Yunita

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Yunita	Yunita	Benar Benar
Samping tanpa senyum	Yunita	Yunita	Benar Benar
Depan dengan senyum	Yunita	Yunita	Benar Benar
Depan dengan tertawa	Yunita	Yunita	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Yunita perbandingan tersebut dilakukan antara foto master dan foto master.

2. Uji Foto aksesoris Topi Yunita.

Tabel IV.37
Uji coba foto aksesoris topi foto Yunita

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Yunita	Yunita	Benar Benar
Samping tanpa senyum	Yunita	Yunita	Benar Benar
Depan dengan senyum	Yunita	Yunita	Benar Benar
Depan dengan tertawa	Yunita	Yunita	Benar Benar

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Yunita perbandingan tersebut dilakukan antara foto master dan foto aksesoris topi.

3. Uji foto aksesoris kacamata Yunita:

Tabel IV.38
Uji coba foto aksesoris topi kacamata Yunita

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Yunita	Yunita	Benar Benar
Samping tanpa senyum	Yunita	Andry	Benar Salah
Depan dengan senyum	Yunita	Yunita	Benar Benar
Depan dengan tertawa	Yunita	Andika	Benar Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Yunita perbandingan tersebut dilakukan antara foto master dan foto aksesoris kacamata.

4. Uji foto aksesoris masker Yunita:

Tabel IV.39
Uji coba foto aksesoris masker foto Yunitas

POSE	Eigenface Terdeteksi sebagai	Fisherface Terdeteksi sebagai	Keterangan
Depan tanpa senyum	Uswatun	Andry	Salah Salah

Dari table diatas menjelaskan perbandingan keakurasian algoritma eigenface dan fisherface dalam mengenali wajah sebagai Yunita perbandingan tersebut dilakukan antara foto master dan foto aksesoris masker.

4.4. Akurasi Pengenalan Wajah Secara Gelobal dan Pengujian T-Test

Dari semua wajah sample inputan berjumlah 113 foto yang terdiri dari foto formal dan foto dengan berbagai macam aksesoris seperti topi, kacamata dan masker didapatkan bahwa total kebenaran metode Eigenface adalah 89 dan kesalahan 24, sedangkan untuk metode Fisherface total kebenaran 86 dan kesalahan 27. Maka didapatkan hasil keakurasian antara algoritma Eigenface dan fisherface sebagai berikut:

Tabel IV.41
Tabel kebenaran Keseluruhan

No	Pose	aksesoris	Kebenaran Fisherface	kebenaran eigenfaces
1	Depan formal	non aksesoris	6	9
2	Samping formal		5	9
3	Depan Senyum		5	9
4	Depan Tertawa		5	9
5	Depan formal	Topi	5	8
6	Samping formal		4	7
7	Depan Senyum		6	8
8	Depan Tertawa		6	8
9	Depan formal	kacamata	6	7
10	Samping formal		5	8
11	Depan Senyum		5	8
12	Depan Tertawa		5	9
13	Depan formal	masker	0	1

Untuk menguji dan membandingkan 2 komparatif antara algoritma eigenface dan fisherface maka diuji dengan menggunakan T-Test yang merupakan uji komparatif untuk menilai antara nilai tertentu dengan rata-rata kelompok populasi, untuk membandingkan 2 sample yang saling keterkaitan maka digunakan uji T dua sample secara *dependent sample T-test*.

Independent T-test adalah jenis uji statistic yang bertujuan untuk membandingkan rata-rata dua grup yang tidak saling berpasangan, sample berpasangan dapat diartikan sebagai sebuah sample dengan subjek yang sama namun mengalami 2 perlakuan atau pengukuran yang berbeda yaitu pengukuran sebelum dan sesudah dilakukan treatment.

Untuk itu maka dibuatkanlah sebuah hipotesa untuk sementara dengan variable H_0 dan H_1 dengan hipotesa sebagai berikut:

H_0 : kebenaran fisherface = kebenaran eigenface

H_1 : kebenaran fisherface $\neq$ kebenaran eigenface

Jika nilai sig. t hitung $> 0,05$ maka H_0 diterima sedangkan bilamana sig. t hitung $< 0,05$ maka H_0 ditolak.

Tabel IV.41
Tabel uji coba T-Test dengan SPSS

Group Statistics				
algorithm	N	Mean	Std. Deviation	Std. Error Mean
Jumlah_benar eigenfaces	13	7.6923	2.13638	.59252
fisherface	13	4.8462	1.57301	.43627

Independent Samples Test									
		Levene's Test for Equality of Variances		t-test for Equality of Means					
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference
Jumlah_benar	Equal variances assumed	.386	.540	3.868	24	.001	2.84615	.73581	1.32751 4.36480
	Equal variances not assumed			3.868	22.056	.001	2.84615	.73581	1.32040 4.37191

Melihat dari hasil Analisa untuk nilai signifikan dengan hasil 0,001 maka dapat disimpulkan bahwa H_0 ditolak dan H_1 diterima Karena $0,001 < 0,005$ dan dapat disimpulkan bahwa dalam hal kebenaran dengan berbagai macam aksesoris

algoritma fisherface tidak sama dengan algoritma eigenface dan lebih akurat dengan menggunakan algoritma eigenface dalam membaca pola wajah dengan berbagai macam pose dan juga aksesoris.

BAB V

PENUTUP

5.1 Kesimpulan

Dalam penelitian mengenai pengenalan wajah dengan menggunakan dua metode yaitu metode Eigenface dan Fisherface dapat disimpulkan sebagai berikut:

1. Kedua metode ini sama-sama menggunakan konversi gambar menjadi bilangan matriks.
2. Metode eigenface dapat membaca atau membandingkan hanya dengan satu foto master.
3. Metode Fisherface tidak dapat membaca atau membedakan foto dengan acuan satu foto, akan tetapi diperlukan pose yang berbeda untuk dijadikan master.
4. Metode Eigenface lebih mengenal karakter dengan aksesoris topi dibandingkan dengan aksesoris kaca mata.
5. Metode Fisherface lebih mengenal karakter dengan aksesoris kaca mata dibandingkan dengan aksesoris topi.
6. Untuk pengenalan wajah yang menggunakan aksesoris masker, kedua metode ini sangat kecil kemungkinan dalam proses pengenalannya.
7. Metode Eigenface memiliki keakurasian yang lebih unggul dari Fisherface yang diambil dari 113 sample foto dengan aksesoris topi dan kaca mata, dan menggunakan empat pose yaitu menghasilkan nilai keakurasian 79% sedangkan tingkat kesalahannya 21%.
8. Metode Fisherface memiliki keakurasian yang tidak unggul dari Eigenface yang diambil dari 113 sample foto dengan aksesoris topi dan kaca mata, dan menggunakan empat pose yaitu menghasilkan nilai keakurasian 76% sedangkan tingkat kesalahannya 24%.
9. Untuk keakurasian diantara keduanya Eigenface lebih unggul daripada Fisherface.

5.2 Saran

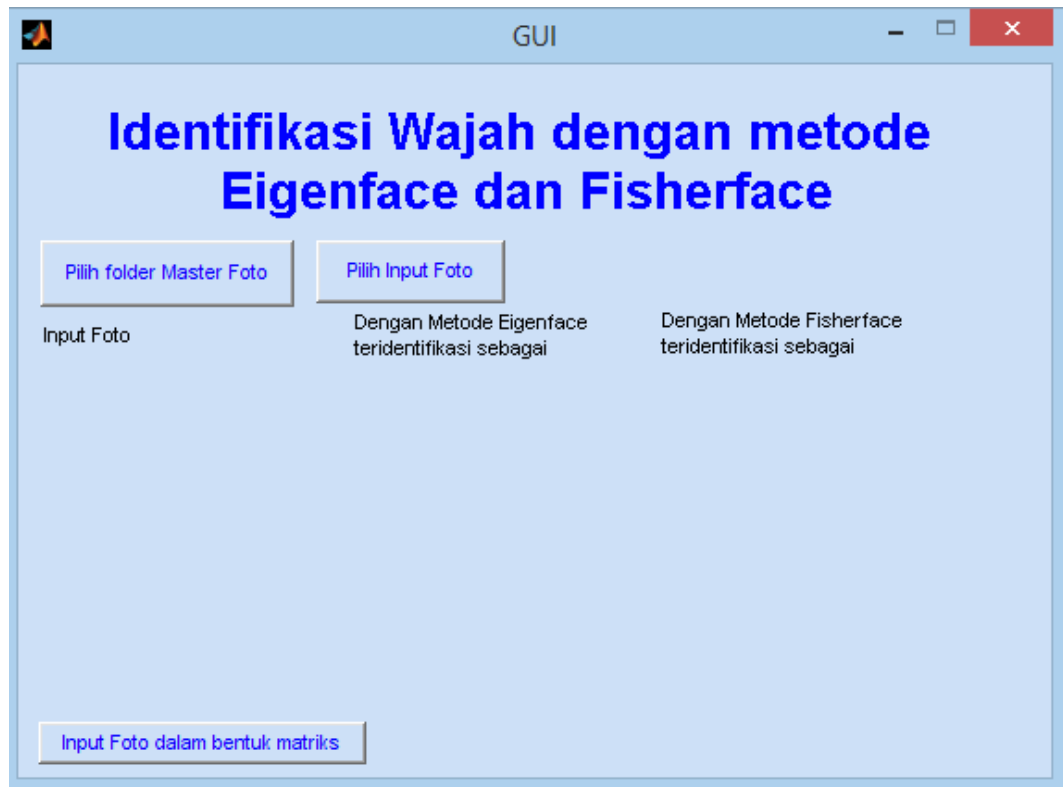
Dalam penelitian ini penulis memberikan saran agar nilai dari akurasi antar kedua metode tersebut bertambah:

1. Untuk master foto baiknya tidak hanya menggunakan satu foto sebagai acuan atau perbandingan, akan tetapi menggunakan lebih dari satu foto yang dapat dijadikan sebagai data master, bahkan perlu database dengan aksesoris seperti topi, kacamata bahkan masker sekalipun, agar lebih banyak pilihan atau perbandingan pada foto master.
2. Untuk penelitian ini hanya menggunakan sebuah gambar, dikarenakan dalam matlab dibutuhkan konfigurasi tambahan untuk mengakses kamera dan tentunya berbayar.
3. Menggunakan video jauh lebih baik untuk mengenal pola wajah seseorang bilamana terekam dan teridentifikasi secara langsung seorang yang akan hendak dikenali.
4. Dapat diaplikasikan dalam berbagai macam kebutuhan identitas diri warga negara, dengan menggunakan data foto, sehingga untuk menindak tindakan kriminal tidak terlalu sulit dan memakan banyak waktu mencari identitas warga dari sebuah negara.

DAFTAR PUSTAKA

- Fatta, H. A. (2009). *Rekayasa pengenalan wajah*. Yogyakarta: Andi Publisher.
- Hermawati, F. A. (2013). *Pengolahan Citra Digital*. Yogyakarta: Andi Publisher.
- Marti, N. W. (2010). Pemanfaatan GUI dalam pengembangan perangkat lunak pengenalan citra wajah manusia menggunakan metode eigenface. *SNATI*.
- Mittal, A., & Dubey, S. K. (2013). A Literature Review on Analysis of MRI images of rheumatoid arthritis through morphological image processing techniques. *IJCTT*, 50-62.
- Muliawan, M. R., Irawan, B., & Brianorman, Y. (2015). Implementasi pengenalan wajah dengan metode Eigenface pada sistem absensi. *Junal Coding*, 41-50.
- Putra, R. S. (2013). Perancangan aplikasi Absensi dengan deteksi wajah menggunakan metode eigenface. *Pelita Informatika Budi Dharma*, 50-62.
- Saragih, R. A. (2007). Pengenalan Wajah Menggunakan Metode Fisherface. *Jurnal Teknik Elektro*, 50-62.
- Sutoyo, T., Mulyanto, E., Suhartono, V., & Dwi, O. (2009). *Teori Pengolahan Citra Digital*. yogyakarta: Andi publisher.
- Vincent, O. R., & Folorunsu, O. (2009). A Descriptive algorithm for sobel image edge detection. *InSite*, 60-65.

1. Tampilan program GUI Matlab



2. Memproses master folder algoritma Fisherface

```

3. function [rata2ContohData, V_PCA, V_Fisher,
   ProyeksiGambar_Fisher] = Fisherface_Fisherface(contohData)
4. %Algoritma ini menggunakan 2 buah metode yaitu PCA
   (Principle Component Analysis) and FLD (Fisher Linear
   Discriminant)
5.
6. %Konsep utama fungsi ini adalah:
7. %Diketahui Ti adalah matriks contoh data, P adalah jumlah
   baris matriks contoh data, C adalah jumlah kolom matriks
   contoh data
8. %* Hitung Zi = V_PCA * (Ti - rata2ContohData)
9. % Zi adalah Ti yang telah dipetakan ke dalam subruang
   linier (P-C) menggunakan matriks transfer V_PCA
10. %* Hitung Yi = V_Fisher' * Zi = V_Fisher' * V_PCA' *
   (Ti - rata2ContohData)
11. % Zi akan dikonversi menjadi Yi dengan cara
   diproyeksikan kedalam subruang linier (C-1)
12. % sehingga gambar wajah yang sama akan berada lebih
   dekat dan gambar wajah yang berbeda akan berada lebih jauh
13.
14. %rata2ContohData = rata-rata dari contoh data
15. %V_PCA - matriks berukuran (M*Nx(P-
   C)) Eigenvector dari kovarians matriks contoh data

```

```

16.      %V_Fisher          - matriks berukuran ((P-C)x(C-
17.      1)) Eigenvector (C-1) terbesar dari matriks J = inv(Sw) *
      Sb
18.
19.      %ProyeksiGambar_Fisher - matriks berukuran ((C-1)xP)
      Gambar contoh data yang sudah diproyeksikan ke dalam ruang
      linier Fisher
20.
21.      %3a. Tentukan jumlah class dan jumlah gambar per class
      jumlahClass = ( size(contohData,2) )/2;      % Jumlah
      class / gambar / orang / wajah
22.      jumlahGambarPerClass = 2;                  % Jumlah
      gambar pada masing-masing class
23.      P = jumlahGambarPerClass * jumlahClass;      % Jumlah
      contoh data
24.
25.      %3b. Hitung rata-rata dari matriks contoh data
      rata2ContohData = mean(contohData,2);
26.
27.      %3c. Hitung deviasi dari masing-masing gambar terhadap
      rata-rata matriks contoh data
28.      A = contohData - repmat(rata2ContohData,1,P);
29.
30.      %3d. Lakukan proses dekomposisi kovarians matriks L
      untuk mendapatkan nilai eigenvector dan eigenvalue
31.      %Proses ini mirip dengan proses yang dilakukan oleh
      algoritma Eigenface
32.      L = A'*A;
33.      [V, lambda] = eig(L);
34.
35.      %3e. Lakukan eliminasi nilai eigenvalue yang cukup
      rendah
36.      eigenVectorTerbesar = [];
37.      for i = P : -1 : jumlahClass + 1
38.          eigenVectorTerbesar = [eigenVectorTerbesar
      V(:,i)];
39.      end
40.
41.      %3f. Hitung eigenvector dari matriks kovarians L
42.      V_PCA = A * eigenVectorTerbesar;
43.
44.      %3g. Lakukan proyeksi gambar vektor ke dalam ruang
      eigen
45.      %Gunakan rumus Zi = V_PCA' * (Ti - rata2ContohData)
46.      ProyeksiGambar_PCA = [];
47.      for i = 1 : P
48.          tmp = V_PCA'*A(:,i);
49.          ProyeksiGambar_PCA = [ProyeksiGambar_PCA tmp];
50.      end
51.
52.      %3h. Hitung rata-rata dari masing-masing class dalam
      ruang eigen
53.      %Proses ini akan menghasilkan matriks Within Scatter
      dan matriks Between Scatter

```

```

54.     rata2PCA = mean(ProyeksiGambar_PCA,2);
55.
56.     rata2 = zeros(P-jumlahClass,jumlahClass);
57.     Sw = zeros(P-jumlahClass,P-jumlahClass); % Matriks
    Within Scatter
58.     Sb = zeros(P-jumlahClass,P-jumlahClass); % Matriks
    Between Scatter
59.
60.     for i = 1 : jumlahClass
61.         rata2(:,i) = mean( ( ProyeksiGambar_PCA(:,((i-
    1)*jumlahGambarPerClass+1):i*jumlahGambarPerClass) ), 2 )';
62.
63.         S = zeros(P-jumlahClass,P-jumlahClass);
64.         for j = ( (i-1)*jumlahGambarPerClass+1 ) : (
    i*jumlahGambarPerClass )
65.             S = S + (ProyeksiGambar_PCA(:,j)-
    rata2(:,i))*(ProyeksiGambar_PCA(:,j)-rata2(:,i))';
66.         end
67.
68.         Sw = Sw + S;
    % Matriks Within Scatter
69.         Sb = Sb + (rata2(:,i)-rata2PCA) * (rata2(:,i)-
    rata2PCA)'; % Matriks Between Scatter
70.     end
71.
72.     %3i. Hitung nilai basis dari Fisher Discriminant
73.     %Tujuan utama yang dilakukan adalah memaksimalkan
    nilai matriks Between Scatter,
74.     %sambil meminimalkan nilai matriks Within Scatter
75.     [J_eig_vec, J_eig_val] = eig(Sb,Sw);
76.     J_eig_vec = fliplr(J_eig_vec);
77.
78.     %3j. Hapus nilai eigen yang cukup rendah,
79.     %dan lakukan pengurutan dari nilai tertinggi ke nilai
    terendah
80.     for i = 1 : jumlahClass-1
81.         V_Fisher(:,i) = J_eig_vec(:,i); % Eigenvector (C-
    1) terbesar dari matriks J
82.     end
83.
84.     %3k. Lakukan proyeksi gambar kedalam ruang linier
    Fisher
85.     %Gunakan rumus  $Y_i = V\_Fisher' * V\_PCA' * (T_i -$ 
    rata2ContohData)
86.     for i = 1 : jumlahClass*jumlahGambarPerClass
87.         ProyeksiGambar_Fisher(:,i) = V_Fisher' *
    ProyeksiGambar_PCA(:,i);
88.     end

```

3. Pengenalan wajah algoritma Fisherface

```
function hasil = Fisherface_PengenalanGambar(dataBaruTerpilih,
rata2ContohData, V_PCA, V_Fisher, ProyeksiGambar_Fisher)
jumlahContohData = size(ProyeksiGambar_Fisher,2);

%4b1. Lakukan proses perhitungan proyeksi gambar pada data yang
baru
%Sama seperti perhitungan sebelumnya,
%Gunakan rumus  $Y_i = V\_Fisher' * V\_PCA' * (T_i - rata2ContohData)$ 
dataBaru = imread(dataBaruTerpilih);
%tmp = dataBaru(:,:,1);
if length(size(dataBaru))==3
    tmp = rgb2gray(dataBaru);
else
    tmp = dataBaru;
end

[panjang, lebar] = size(tmp);
vektorGambar(:,1) = reshape(tmp,panjang*lebar,1);
deviasi = double(vektorGambar) - rata2ContohData;
ProyeksiGambar = V_Fisher' * V_PCA' * deviasi;

%4b2. Hitung jarak Euclidean antara proyeksi gambar data baru
dengan proyeksi Fisher dari semua contoh data
%Ambil jarak paling minimal sebagai jawaban gambar contoh data
yang paling sesuai dengan gambar data baru
jarakEuclidean = [];
for i = 1 : jumlahContohData
    q = ProyeksiGambar_Fisher(:,i);
    tmp = ( norm( ProyeksiGambar - q ) )^2;
    jarakEuclidean = [jarakEuclidean tmp];
end

[minJarak, idxMinimal] = min(jarakEuclidean);
% hasil = strcat(int2str(idxMinimal),'.jpg');
hasil = idxMinimal;
```

4. Memproses folder master Eigenface

```
function
[H,W,M,m,U,omega]=Eigenface_PerhitunganContohData(gmbContohData,
matriksContohData)
%1b. Tentukan tinggi (H) dan lebar (W) dari masing-masing
gambar
%Kemudian hitung jumlah gambar yang tersedia (M)
[H, W, M] = size(gmbContohData);

%1d. Hitung rata-rata (m) dari masing-masing gambar dengan
rumus:
%m = (1/M) * E(vec)
m=sum(matriksContohData,2)/M;

%1e. Hitung selisih antara gambar dengan rata-rata gambar
dengan rumus:
%A(i) = vec(i) - M
A=matriksContohData-repmat(m,1,M);
```

```

%1f. Hitung matriks kovarian (C) dengan rumus:
%C = AT * A
C=A'*A;

%1g. Hitung vektor eigen (V) dan nilai eigen (lambda) dengan
rumus:
%Fungsi eig akan melakukan dekomposisi matriks kovarian
menjadi matriks V dan lambda
[V,lambda]=eig(C);

%1h. Hitung nilai eigenface dari masing-masing contoh gambar
dengan rumus:
%U = A * V
U=A*V;

%1i. Setelah menemukan nilai eigenface,
%Hitung nilai bobot (W) untuk masing-masing gambar dengan
rumus:
%W(i) = U(i)T * A
%Kemudian masukkan semua nilai bobot pada masing-masing gambar
sebagai nilai omega
omega=U'*A;
end

```

5. Perhitungan pengujian Eigenface

```

function [idxHasil] =
Eigenface_PerhitunganDataBaru(pathGambar,H,W,M,m,U,omega)
tmpBmp=imread(pathGambar);
if length(size(tmpBmp))==3
    tmp = rgb2gray(tmpBmp);
else
    tmp = tmpBmp;
end

bmp=reshape(tmp,H*W,1);
testBmp=double(bmp);

%2e1. Hitung selisih antara gambar dengan rata-rata gambar (A)
selisihBmp=testBmp-m;

%2e2. Hitung nilai bobot (om) pada gambar yang baru dengan rumus:
%om = UT * A
om=U'*selisihBmp;

%* Tentukan gambar contoh yang paling baik dalam mendeskripsikan
data baru
%dengan cara mencari minimal dari jarak Euclidean

%2e3. Hitung nilai E(k) dengan rumus:
%E(k) = om(k) - omega
d= repmat(om,1,M)-omega;

dist=zeros(M,1);

```

```

%2e4. Hitung jarak euclidean dari nilai E(k) yang sudah ditentukan
sebelumnya
for j=1:M
    dist(j,1)=norm(d(:,j));
end

%2e5. Tentukan jawaban gambar, yaitu gambar yang memiliki jarak
Euclidean terendah dari semua gambar
idxHasil=Eigenface_MinimalIndeks(dist);
end

```