



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - IF184802

IMPLEMENTASI PENGENALAN WAJAH MENGUNAKAN EKSTRAKSI FITUR *PATTERNS OF ORIENTED EDGE MAGNITUDES*

R.AY. NOORMALA NADYA
NRP 05111440000127

Dosen Pembimbing I
Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

Dosen Pembimbing II
Dr.Eng. Chastine Fatichah, S.Kom., M.Kom.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019

[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - IF184802

IMPLEMENTASI PENGENALAN WAJAH MENGUNAKAN EKSTRAKSI FITUR PATTERNS OF ORIENTED EDGE MAGNITUDES

R.AY. NOORMALA NADYA
NRP 05111440000127

Dosen Pembimbing I
Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

Dosen Pembimbing II
Dr.Eng. Chastine Fatichah, S.Kom., M.Kom.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019

[Halaman ini sengaja dikosongkan]



UNDERGRADUATE THESIS - IF184802

FACE RECOGNITION IMPLEMENTATION USING PATTERN OF ORIENTED EDGE MAGNITUDES FEATURE EXTRACTION

R.AY. NOORMALA NADYA
NRP 05111440000127

Supervisor I
Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

Supervisor II
Dr.Eng. Chastine Fatichah, S.Kom., M.Kom.

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
Sepuluh Nopember Institute of Technology
Surabaya 2019

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

IMPLEMENTASI PENGENALAN WAJAH MENGUNAKAN EKSTRAKSI FITUR PATTERNS OF ORIENTED EDGE MAGNITUDES

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Rumpun Mata Kuliah Komputasi Cerdas dan Visi
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

R.Ay. Noormala Nadya
NRP : 05111440000127

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dr.Eng. Nanik Suciati, S.Kom., M.Kom.
NIP. 19710428 199412 2 001

.....
(pembimbing 1)

Dr.Eng. Chastine Fatichah, S.Kom., M.Kom.
NIP. 19751220 200112 2 002

.....
(pembimbing 2)

SURABAYA
JANUARI 2019

[Halaman ini sengaja dikosongkan]

IMPLEMENTASI PENGENALAN WAJAH MENGUNAKAN EKSTRAKSI FITUR PATTERNS OF ORIENTED EDGE MAGNITUDES

Nama Mahasiswa : R.Ay. Noormala Nadya
NRP : 05111440000127
Departemen : Informatika FTIK-ITS
Dosen Pembimbing 1 : Dr.Eng. Nanik Suciati, S.Kom.,
M.Kom.
Dosen Pembimbing 2 : Dr.Eng. Chastine Fatichah, S.Kom.,
M.Kom.

ABSTRAK

Demi meningkatkan keamanan di daerah umum, pengenalan wajah menjadi salah satu yang dipertimbangkan. Pengenalan wajah merupakan klasifikasi pola yang sulit karena adanya inter-cluster variation (variasi citra wajah pada orang yang sama). Variasi pencahayaan merupakan salah satu faktor utama dalam inter-cluster variation. Variasi pencahayaan membuat perubahan drastis dalam tampilan sebuah wajah.

Dalam Tugas Akhir ini diimplementasikan pengenalan wajah menggunakan metode ekstraksi fitur Patterns of Oriented Edge Magnitudes (POEM). Metode klasifikasi yang dipakai adalah Support Vector Machine (SVM). Sebelum dilakukan ekstraksi fitur, terlebih dahulu dilakukan frontalization dan Difference of Gaussians (DoG) untuk memperbaiki kualitas citra.

Uji coba yang dilakukan menggunakan dataset FERET. Hasil uji coba yang menggunakan metode ekstraksi fitur POEM dengan klasifikasi SVM mendapatkan akurasi terbaik pada kelas Fb sebesar 79.1%, Fc sebesar 89.1%, Dup1 68.8%, dan Dup2 61.8%.

Kata kunci: Frontalization, Patterns of Oriented Edge Magnitudes, Difference of Gaussians, Support Vector Machine.

[Halaman ini sengaja dikosongkan]

FACE RECOGNITION IMPLEMENTATION USING PATTERN OF ORIENTED EDGE MAGNITUDES FEATURE EXTRACTION

Student Name : R.Ay. Noormala Nadya
Student ID : 05111440000127
Major : Informatics Department FTIf-ITS
1st Supervisor : Dr.Eng. Nanik Suciati, S.Kom., M.Kom.
2nd Supervisor : Dr.Eng. Chastine Fatichah, S.Kom., M.Kom.

ABSTRACT

In order to improve security in the general area, face recognition is one of the ones that supports them. Variations between clusters (variations in facial image on the same person). Lighting variation is one of the main factors in variations between clusters. Variations make drastic changes in facial appearance.

In this Final Project, will be applied face recognition using the feature extraction method Pattern Oriented Edge Magnitudes (POEM). The classification method used is Machine Vector Support (SVM). Before feature extraction, frontalization and Difference of Gaussians (DoG) were carried out to improve image quality.

Trials carried out using the FERET dataset. The results of the trial using the POEM feature extraction method with SVM classification obtained the best accuracy of Fb class gives 79.1%, Fc gives 89.1%, Dup1 68.8%, and Dup2 61.8%.

Keywords: Frontalization, Patterns of Oriented Edge Magnitudes, Difference of Gaussians, Support Vector Machine.

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji syukur bagi Allah SWT yang telah melimpahkan rahmat dan anugerah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul:

“IMPLEMENTASI PENGENALAN WAJAH MENGGUNAKAN EKSTRAKSI FITUR PATTERNS OF ORIENTED EDGE MAGNITUDES”.

Pengerjaan tugas akhir ini menjadi suatu pengalaman yang baik bagi penulis. Penulis dapat memperoleh banyak pengalaman yang berharga dalam memperdalam dan meningkatkan keilmuan dalam bidang informatika selama perkuliahan di Teknik Informatika ITS.

Tugas akhir ini selesai karena tidak lepas dari bantuan dan dukungan dari berbagai pihak. Pada kesempatan kali ini, penulis ingin mengucapkan terima kasih kepada:

1. Dr.Eng. Nanik Suciati, S.Kom., M.Kom. dan Dr.Eng. Chastine Fatichah, S.Kom., M.Kom. selaku dosen pembimbing penulis yang telah memberi ide, nasihat dan arahan sehingga penulis dapat menyelesaikan tugas akhir ini.
2. Kedua orang tua penulis, Mama dan Papa serta Nenek penulis yang telah memberikan dukungan moral, spiritual dan material serta senantiasa memberikan doa demi kelancaran dan kemudahan penulis dalam mengerjakan tugas akhir.
3. Adik dan Om penulis (Bryan dan Om Ainul) serta seluruh keluarga yang telah memberikan dukungan, baik secara langsung maupun tidak langsung.
4. Sahabat penulis, Afiiif dan Habib, yang telah banyak membantu kesulitan penulis dalam penyelesaian TA.

5. Teman-teman terdekat penulis selama berkuliah di TC: Dini, Tionia, Datin, Kukuh, Luqman, Galang, Upik, Tepe, Kevin, Lucha, Ade, dan Sita yang selalu menemani dan menjadi *support system* penulis dalam pengerjaan TA.
6. Teman-teman Admin KCV 2015 dan 2016: Chasni, Ocid, Nuzul, Prana, Upil, Dandy, Titut, Randi, Yoshi, dan Aldi yang seringkali menghibur dan membantu masalah-masalah penulis saat mengerjakan TA di Lab.
7. Teman-teman kelas PAA: Dyo, Bagus, dan Paul yang telah membantu penulis melewati masa sulit dalam perkuliahan.
8. Teman-teman TC 2014: Ghazian dan Faishal yang telah memberikan banyak pencerahan dan nasihat kepada penulis.
9. Teman-teman penulis: Erin, Dhillia, Imah, Mida, Rosyita, dan Athiyah yang telah banyak memotivasi dan menjadi tempat berkeluh kesah penulis selama ini.
10. Pihak-pihak lain yang tidak bisa penulis sebutkan satu-persatu.

Penulis menyadari bahwa Tugas Akhir ini masih memiliki banyak kekurangan. Sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depan.

Surabaya, Januari 2019

R.Ay. Noormala Nadya

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
DAFTAR KODE SUMBER	xxi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Tugas Akhir	3
1.5. Manfaat Tugas Akhir.....	3
1.6. Metodologi.....	3
1.7. Sistematika Penulisan.....	4
BAB II DASAR TEORI.....	7
2.1. <i>Difference of Gaussian</i>	7
2.2. <i>Patterns of Oriented Edge Magnitudes (POEM)</i>	8
2.3. <i>Local Binary Pattern (LBP)</i>	10
2.4. <i>Support Vector Machine (SVM)</i>	11
BAB III PERANCANGAN.....	13
3.1. Data	13
3.2. Perancangan Sistem.....	13
3.3. Perancangan Proses	15
3.3.1. <i>Pre-processing</i>	15
3.3.2. Ekstraksi Fitur POEM.....	17
3.3.3. Klasifikasi SVM	19
BAB IV IMPLEMENTASI.....	21
4.1. Lingkungan Implementasi	21
4.1.1. Perangkat Keras.....	21
4.1.2. Perangkat Lunak.....	21

4.2. Implementasi Frontalisai	22
4.3. Implementasi Metode <i>Pre-Processing</i>	23
4.3.1. Implementasi <i>Difference of Gaussian</i>	23
4.3.2. Implementasi Normalisasi	23
4.4. Implementasi Ekstraksi Fitur <i>Patterns of Oriented Edge Magnitudes</i> (POEM)	24
4.5. Implementasi Klasifikasi SVM	24
BAB V PENGUJIAN DAN EVALUASI	27
5.1. Lingkungan Uji Coba	27
5.2. Data Uji Coba	27
5.3. Skenario Uji Coba	28
5.3.1. Uji Coba Penentuan Jumlah Bin Pada Ekstraksi Fitur POEM	28
5.3.2. Uji Coba Penentuan Ukuran Sel Pada Ekstraksi Fitur POEM	29
5.3.3. Uji Coba Penentuan Ukuran Blok Pada Ekstraksi Fitur POEM	29
BAB VI KESIMPULAN DAN SARAN	31
6.1. Kesimpulan	31
6.2. Saran	31
DAFTAR PUSTAKA	33
LAMPIRAN	35
BIODATA PENULIS	39

DAFTAR GAMBAR

Gambar 2.1 Langkah Utama dalam Penghitungan Ekstraksi Fitur POEM	8
Gambar 2.2 Sel yang menyusun sebuah blok.....	9
Gambar 2.3 Ilustrasi Dasar Operator LBP	10
Gambar 2.4 Tahapan Proses SVM [7]	12
Gambar 3.1 Contoh dataset FERET: (a) Kelas Fa, (b) Fb, (c) Fc, (d) Dup1, dan (e) Dup2	14
Gambar 3.2 Diagram alur dari metode yang diusulkan	15
Gambar 3.3 Diagram Alir <i>Pre-processing</i>	16
Gambar 3.4 Diagram Alir Ekstraksi Fitur POEM	17
Gambar 3.6 Diagram Alur SVM	20
Gambar 5.1 Citra Wajah FERET.....	28

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 4.1 Lingkungan Implementasi Perangkat Lunak	21
Tabel 5.1 Hasil Uji Coba.....	29

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SUMBER

Kode Sumber 4.1 Implementasi Pengambilan Komponen Citra Masukan.....	23
Kode Sumber 4.2 Implementasi Difference of Gaussian	23
Kode Sumber 4.3 Implementasi Normalisasi	24
Kode Sumber 4.4 Implementasi Ekstraksi Fitur POEM	25
Kode Sumber 4.5 Implementasi Klasifikasi SVM.....	25

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini akan dibahas hal-hal yang mendasari tugas akhir. Bahasan meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi, dan sistematika laporan tugas akhir.

1.1. Latar Belakang

Pengenalan wajah telah menjadi area studi intensif sejak 1960-an. Aplikasi yang inovatif menyebabkan teknologi ini terus dikembangkan dengan cepat. Aplikasi pengenalan wajah kontemporer bisa dibagi menjadi tiga bidang yang bergantung pada tujuan dari tugas pengenalan wajah: (1) verifikasi wajah, dimana tujuannya adalah untuk mengotentikasi identitas dari gambar wajah dengan template yang sesuai; (2) identifikasi wajah, dimana tujuannya adalah untuk menemukan kecocokan dalam database gambar wajah; (3) penandaan wajah (variasi yang relatif baru dari identifikasi wajah), dimana tujuannya adalah untuk memberi label gambar wajah berdasarkan identifikasi saat dicocokkan. Kini, pengenalan wajah merupakan komponen penting dalam keamanan biometrik, manajemen akses, identifikasi kriminal, serta penyortiran dan pengambilan gambar.

Tujuan utama pengenalan wajah adalah membandingkan dua gambar wajah dan memecahkan masalah dalam menentukan apakah kedua gambar tersebut merupakan orang yang sama ataukah dua orang berbeda. Masalah ini sulit dipecahkan, karena dua gambar orang yang sama dapat sangat bervariasi dalam segi waktu pengambilan gambar, pose, ekspresi wajah, kondisi pencahayaan, dan kualitas gambar yang diambil. Sebagian besar teknik pengenalan wajah mutakhir berkinerja baik saat gambar wajah ditangkap dalam kondisi optimal, dimana pencahayaan terkontrol dan percobaan memperlihatkan pandangan frontal yang penuh. Namun, bila gambar wajah ditangkap di lingkungan luar - di mana pose, usia, dan ekspresi wajah berubah dan dimana kondisi

lingkungan, seperti pencahayaan yang tidak ideal - kinerjanya memburuk.

Vu N.S.[1,2] membuktikan bahwa *Patterns of Oriented Edge Magnitudes* (POEM) mempunyai kemampuan diskriminatif dan invariant yang besar. Metode ini mampu menangani perbedaan ekspresi, pencahayaan, dan pose serta penuaan pada dataset wajah dengan baik.

Pada machine learning, untuk mengklasifikasikan data berlabel digunakan teknik *supervised learning*. Untuk mengenali dua wajah, dapat menggunakan *binary classifier*. Tetapi, jika ingin mengenali lebih dari dua wajah, menggunakan *multi class classifier*. Fiturnya merupakan kumpulan informasi dari gambar yang digunakan pada saat training.

Dalam metode yang diusulkan dalam tugas akhir ini, POEM digunakan untuk mengekstraksi fitur, kemudian hasil dari ekstraksi fitur tersebut diklasifikasikan menggunakan *Support Vector Machine* (SVM). Fokus utama tugas akhir ini yaitu untuk mengetahui kinerja metode *Patterns of Oriented Edge Magnitudes* (POEM) pada sistem pengenalan wajah. Diharapkan metode ini dapat menghasilkan performa yang baik pada sistem pengenalan wajah.

1.2. Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara mengimplementasi *Patterns of Oriented Edge Magnitudes* untuk ekstraksi fitur citra wajah?
2. Bagaimana cara mengimplementasi pengenalan wajah dengan metode pengenalan POEM?
3. Bagaimana mengevaluasi kinerja aplikasi pengenalan wajah yang telah dibuat?

1.3. Batasan Masalah

Beberapa hal yang menjadi batasan masalah dalam pengerjaan tugas akhir:

1. Data yang digunakan adalah dataset FERET yang tersedia secara terbuka di internet.
2. Implementasi program dilakukan pada lingkungan komputer desktop dan bahasa pemrograman Matlab dan Python.

1.4. Tujuan Tugas Akhir

Tujuan dari tugas akhir ini adalah untuk membangun aplikasi pengenalan wajah dengan metode ekstraksi fitur *Patterns of Oriented Edge Magnitudes*.

1.5. Manfaat Tugas Akhir

Tugas akhir ini diharapkan mampu membangun sebuah model ekstraksi fitur *Patterns of Oriented Edge Magnitudes* yang dapat melakukan klasifikasi wajah manusia dengan lebih rinci dengan waktu yang lebih cepat.

1.6. Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Studi literatur

Pada tahap ini dilakukan pencarian literatur berupa jurnal atau paper yang digunakan sebagai referensi untuk pengerjaan tugas akhir ini. Literatur yang dipelajari pada pengerjaan Tugas Akhir ini berasal dari jurnal ilmiah yang diambil dari berbagai sumber di internet, beserta berbagai literatur online tambahan terkait *Patterns of Oriented Edge Magnitudes*, Makalah yang digunakan sebagai acuan adalah “*Ensemble of texture descriptors and classifiers for face recognition*” [3].

2. Analisis dan Desain Perangkat Lunak

Pada tahap ini disusun rancang bangun dari perangkat lunak yang akan dibuat. Pada tahap ini akan dilakukan analisis dan desain perancangan model sesuai dengan tujuan yang

dijabarkan. Selain itu, pada tahap ini akan dilakukan eksplorasi terkait cara kerja *Patterns of Oriented Edge Magnitudes*.

3. Implementasi Perangkat Lunak

Pada tahap ini akan dilaksanakan implementasi metode dan algoritma yang telah direncanakan. Implementasi perangkat lunak dilakukan di dalam platform desktop dengan menggunakan bahasa pemrograman Matlab.

4. Pengujian dan Evaluasi

Pada tahap ini dilakukan uji coba dengan menggunakan dataset FERET. Hasil dari uji coba dilakukan evaluasi dengan menghitung nilai akurasi dan presisi.

5. Penyusunan buku Tugas Akhir

Pada tahap ini dilakukan proses dokumentasi dan pembuatan laporan dari seluruh konsep, dasar teori, implementasi, proses yang telah dilakukan, dan hasil-hasil yang telah didapatkan.

1.7. Sistematika Penulisan

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab yang berisi mengenai latar belakang, tujuan dan manfaat dari pembuatan tugas akhir. Selain itu permasalahan, batasan masalah, metodologi yang digunakan dan sistematika penulisan juga merupakan bagian dari bab ini.

Bab II Dasar Teori

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

Bab III Perancangan

Bab ini berisi tentang penjelasan dataset yang digunakan, serta perancangan desain sistem pengenalan citra wajah manusia menggunakan metode yang diajukan.

Bab IV Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode yang digunakan untuk proses implementasi.

Bab V Uji Coba dan Evaluasi

Bab ini membahas tahap-tahap uji coba. Kemudian hasil uji coba dievaluasi untuk kinerja dari aplikasi yang dibangun.

Bab VI Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang berisi kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan aplikasi ke depannya.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan tugas akhir.

Lampiran

Merupakan bab tambahan yang berisi daftar istilah atau kode-kode sumber yang penting pada sistem.

[Halaman ini sengaja dikosongkan]

BAB II DASAR TEORI

Pada bab ini diuraikan mengenai dasar-dasar teori yang digunakan dalam pengerjaan tugas akhir dengan tujuan untuk memberikan gambaran secara umum terhadap penelitian yang dikerjakan.

2.1. *Difference of Gaussian*

Difference of Gaussian (DoG) merupakan salah satu metode yang digunakan dalam *preprocessing* citra. DoG dihasilkan dari konvolusi citra menggunakan filter DoG dua dimensi [2]. Filter DoG didapatkan dengan mencari selisih dari dua fungsi Gaussian dengan standar deviasi yang berbeda. Gaussian merupakan low-pass filter. Sedangkan, DoG merupakan band-pass filter yang dapat menghilangkan komponen dengan frekuensi tinggi (*noise*) [6] dan frekuensi rendah (area homogen pada citra). DoG seringkali digunakan untuk mempertegas tepian pada citra yang mengandung banyak *noise*.

$$G_{\sigma}(x, y) = \frac{1}{\sqrt{2\pi}\sigma^2} e\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (2.1)$$

Kernel Gaussian (2.1) dengan standar deviasi σ_1 digunakan untuk mengkonvolusi citra $f(x, y)$ untuk mendapatkan $g_1(x, y)$ seperti pada persamaan (2.2).

$$g_1(x, y) = G_{\sigma_1}(x, y) * f(x, y) \quad (2.2)$$

Kemudian citra $f(x, y)$ dikonvolusi dengan fungsi Gaussian yang memiliki standar deviasi σ_2 , sehingga didapatkan $g_2(x, y)$ seperti pada persamaan (2.3).

$$g_2(x, y) = G_{\sigma_2}(x, y) * f(x, y) \quad (2.3)$$

Selisih kedua hasil konvolusi tersebut merupakan hasil konvolusi dengan kernel DoG. Sehingga, kernel DoG dapat didefinisikan dengan persamaan (2.4) dan (2.5).

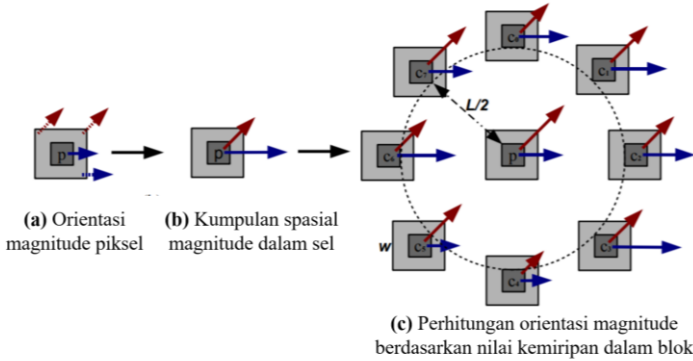
$$DoG = g_1(x, y) - g_2(x, y) \quad (2.4)$$

$$DoG = \frac{1}{\sqrt{2\pi}} \left(\frac{1}{\sigma_1} e^{-\frac{x^2+y^2}{2\sigma_1^2}} - \frac{1}{\sigma_2} e^{-\frac{x^2+y^2}{2\sigma_2^2}} \right) \quad (2.5)$$

2.2. Patterns of Oriented Edge Magnitudes (POEM)

Tujuan dari penelitian ini adalah untuk menemukan ekstraksi fitur yang dapat mewarisi berbagai properti baik dari ekstraksi fitur yang telah ada, namun dengan biaya komputasi yang rendah. Ekstraksi fitur ini menerapkan struktur berbasis LBP pada orientasi magnitude untuk membangun deskriptor baru: *Patterns of Oriented Edge Magnitudes* (POEM) [2].

Singkatnya, untuk menghitung POEM untuk satu piksel, nilai intensitas dalam perhitungan LBP konvensional digantikan oleh besaran magnitude gradien, yang dihitung dengan mengakumulasi histogram arah gradien dalam sel.

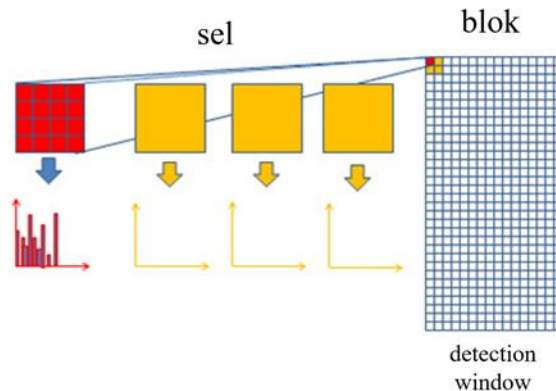


Gambar 2.1 Langkah Utama dalam Penghitungan Ekstraksi Fitur POEM

Sel (kotak besar pada Gambar 2.1(a) merujuk ke wilayah spasial di sekitar piksel saat ini (p), di mana histogram orientasi diakumulasi dan ditetapkan ke piksel pusat sel. Blok

(melingkar pada Gambar 2.1(c)) merujuk ke wilayah spasial yang lebih luas, di mana operator LBP diterapkan.

Berikut adalah contoh dari visualisasi sel dan blok. Sel memiliki ukuran 4x4 piksel pada sebuah citra. Sedangkan blok memiliki ukuran 2x2 sel. Ilustrasi ditunjukkan pada Gambar 2.2



Gambar 2.2 Sel yang menyusun sebuah blok

Dalam POEM, untuk setiap piksel, histogram gradien lokal atas semua piksel dalam sel, yang berpusat pada piksel yang dipertimbangkan, digunakan sebagai representasi piksel tersebut. Demikian pula, pola istilah dalam POEM tidak lokal seperti dalam metode berbasis LBP konvensional. Metode LBP sering menghitung kemiripan diri dalam lingkungan kecil sementara blok yang digunakan dalam POEM agak diperluas.

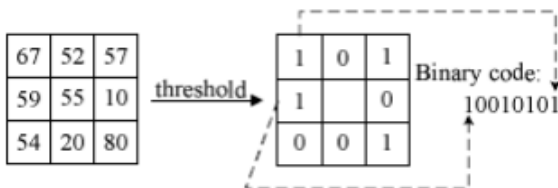
Untuk setiap piksel, POEM mencirikan tidak hanya penampilan dan bentuk objek lokal, tetapi juga hubungan antara informasi ini di wilayah tetangga. POEM memiliki sifat-sifat berikut:

POEM adalah fitur yang berorientasi. Karena jumlah arah diskrit dapat bervariasi, POEM memiliki kemampuan untuk menangkap informasi gambar di segala arah dan dapat disesuaikan untuk representasi objek dengan tingkat akurasi orientasi yang berbeda.

- Dihitung pada skala sel dan blok yang berbeda, POEM juga merupakan fitur multi-resolusi spasial. Ini memungkinkannya untuk menangkap informasi lokal dan struktur yang lebih global.

2.3. Local Binary Pattern (LBP)

LBP telah terbukti sebagai deskriptor tekstur yang tangguh. Operator LBP melabeli piksel-piksel dari sebuah citra dengan melakukan proses thresholding ketetanggaan 3x3 dari masing-masing piksel sebagai nilai tengah dan mengubah hasilnya menjadi nilai biner, dan 256-bin LBP histogram digunakan sebagai *texture descriptor*. Bilangan biner yang dihasilkan (disebut *Local Binary Pattern* atau LBP codes) mengkodekan lokal primitif termasuk variasi dari lengkungan sisi, bintik, area datar, dsb. Batasan dari operator dasar LBP yaitu 3x3 piksel tetangga, tidak dapat menangkap ciri dominan yang memiliki struktur dengan skala besar. Karena itu kemudian operator dikembangkan untuk menggunakan tingkat ketetanggaan dengan ukuran yang berbeda [6]. Menggunakan circular neighborhood dan bilinear interpolation nilai piksel memungkinkan beragam ukuran radius dan jumlah pengambilan titik sampel. Gambar 2.3 merupakan contoh LBP operator, dimana (P, R) menunjukkan jumlah pengambilan titik sampel P dari radius R dari sebuah *circular symmetric neighbor set*.



Gambar 2.3 Ilustrasi Dasar Operator LBP

Operator LBP P, R menghasilkan 2^P nilai output berbeda, berdasarkan dari 2^P binary pattern yang berbeda yang dapat dibentuk oleh P piksel dalam neighbor set. Telah terlihat bahwa bin tertentu mengandung informasi yang lebih dari yang lainnya. Oleh

karenanya dimungkinkan untuk menggunakan hanya subset dari 2^P Local Binary Pattern untuk mendeskripsikan tekstur citra. Ojala et al [6], menyebut *fundamental pattern* ini sebagai *uniform pattern*. Sebuah *local binary pattern* disebut *uniform* apabila mengandung paling banyak dua *bitwise* transition dari 0 ke 1 dan sebaliknya pada sirkular binary string. Sebagai contoh, 00000000, 001110000, dan 1110001 adalah uniform pattern. Telah diteliti bahwa uniform pattern terhitung hampir 90% dari jumlah pattern keseluruhan pada (8,1) neighborhood dan sekitar 70% pada (16,2) neighborhood di texture image. Mengumpulkan patterns yang memiliki lebih dari 2 transisi ke dalam single bin menghasilkan sebuah LBP operator, diindikasikan dengan dengan jumlah bin kurang dari 2^P . Sebagai contoh jumlah label pada neighborhood 8 piksel dihasilkan 256 label sedangkan untuk LBP_{u2} dihasilkan 59.

Setelah melabeli citra dengan local binary pattern. Histogram dari citra $f(x, y)$ dapat didefinisikan sebagai:

$$H_i = \sum_{x,y} I(f_i(x,y) = i), \quad i = 0, \dots, n-1 \quad (2.6)$$

Dimana n adalah jumlah label biner yang dihasilkan oleh LBP operator dan

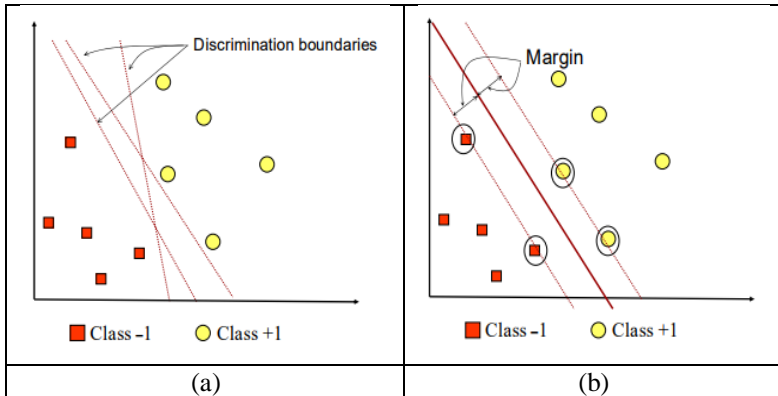
$$I(A) = \begin{cases} 1 & A \text{ is true} \\ 0 & A \text{ is false} \end{cases} \quad (2.7)$$

LBP histogram ini mengandung informasi tentang distribusi local micro-pattern, seperti sisi-sisi, titik-titik, dan area datar, di seluruh permukaan citra, sehingga secara statistik dapat mendeskripsikan karakteristik yang terdapat pada citra.

2.4. Support Vector Machine (SVM)

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vapnik pada tahun 1992. *SVM* adalah metode *learning machine* yang bekerja atas prinsip *Structural Risk Minimization*

(SRM) dengan tujuan menemukan *hyperlane* terbaik yang memisahkan dua buah *class* pada *input space* [7].



Gambar 2.4 Tahapan Proses SVM [7]

Gambar 2.4 menunjukkan sekumpulan data yang terdiri dari dua *class* yaitu *class*: -1 dan +1. *Class* -1 direpresentasikan dengan kotak berwarna merah, sedangkan *class* +1 direpresentasikan dengan lingkaran berwarna kuning. Problem klasifikasi yang harus dilakukan adalah dengan menemukan *hyperplane* yang memisahkan antara dua *class* tersebut.

Hyperplane terbaik yang memisahkan kedua kelas dihitung dengan mengukur *margin hyperplane*, dan mencari titik maksimal. *Margin* adalah jarak terdekat *hyperplane* tersebut dengan *pattern* terdekat dari masing-masing *class*. Garis solid yang terdapat pada gambar 2b adalah *hyperplane* terbaik, yaitu yang terletak tepat ditengah-tengah kedua *class*. Sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah *support vector*.

BAB III

PERANCANGAN

Pada bab ini diuraikan mengenai perancangan aplikasi agar dapat mencapai tujuan dari tugas akhir. Perancangan yang dibuat meliputi data, perancangan sistem, dan proses.

3.1. Data

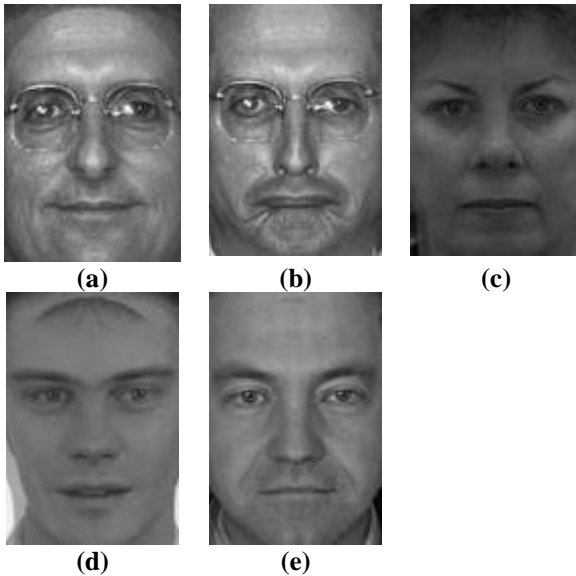
Terdapat banyak dataset wajah manusia yang dipublikasikan secara umum, yang akan digunakan dalam tugas akhir ini adalah dataset FERET.

Dataset FERET berisi lima set data: Fa (1196 citra), Fb (1195 citra), Fc (194 citra), Dup1 (722 citra), dan Dup2 (234 citra). Fa adalah set yang berisi citra utama, sementara set lainnya digunakan untuk pengujian. Fb berisi citra yang dibuat pada hari yang sama dengan Fa, menggunakan kamera yang sama, dengan pencahayaan yang sama. Fc adalah kumpulan citra yang diambil pada hari yang sama dengan Fa, namun dengan kamera yang berbeda, dan diambil di bawah pencahayaan yang berbeda. Dup1 berisi citra yang dibuat pada tahun yang sama dengan Fa. Sementara, Dup2 adalah citra yang dibuat lebih dari 1 tahun setelah Fa dibuat.

Protokol evaluasi standar FERET membandingkan gambar di set Fa dengan masing-masing gambar di set pengujian. Semua citra grayscale FERET disejajarkan (frontalisasi) dengan posisi mata, lalu dipotong dengan ukuran 86 x 126 piksel.

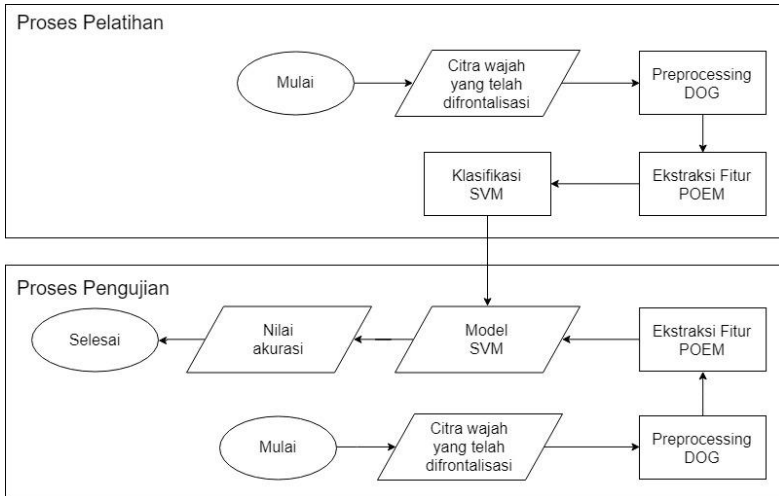
3.2. Perancangan Sistem

Pada sub bab ini akan dijelaskan secara runtut tahapan demi tahapan proses yang dilakukan pada sistem pengenalan wajah di tugas akhir ini. Adapun proses yang dijalankan pada tugas akhir ini terdiri dari 3 tahapan utama, yakni *preprocessing*, ekstraksi fitur, dan pengenalan wajah atau klasifikasi. Diagram alir sistem dapat dilihat pada Gambar 3.2.



Gambar 3.1 Contoh dataset FERET: (a) Kelas Fa, (b) Fb, (c) Fc, (d) Dup1, dan (e) Dup2

Pertama, adalah deteksi wajah. Posisi tepat dari gambar wajah terdeteksi sebagai wajah manusia, dan wajah yang dihasilkan dipotong dan diselaraskan sesuai dengan posisi mata. Kemudian frontalization, untuk mensintesisakan pandangan frontal wajah dari wajah yang terdeteksi (langkah ini berguna dalam membuat representasi fitur mandiri dari perubahan pose). Selanjutnya adalah metode preprocessing. Diterapkan metode enhancing yang telah diuji dalam paper acuan, untuk membuat ekstraksi fitur lebih kuat terhadap perubahan pencahayaan, kebisingan citra, dan sebagainya. Masukan dari langkah ini adalah gambar yang telah melalui proses frontalization, dan keluarannya berupa gambar yang telah diolah.



Gambar 3.2 Diagram alur dari metode yang diusulkan

Lalu, dilakukan ekstraksi fitur secara terpisah pada setiap gambar yang dihasilkan dari metode preprocessing sebelumnya, untuk mendapatkan deskriptor POEM.

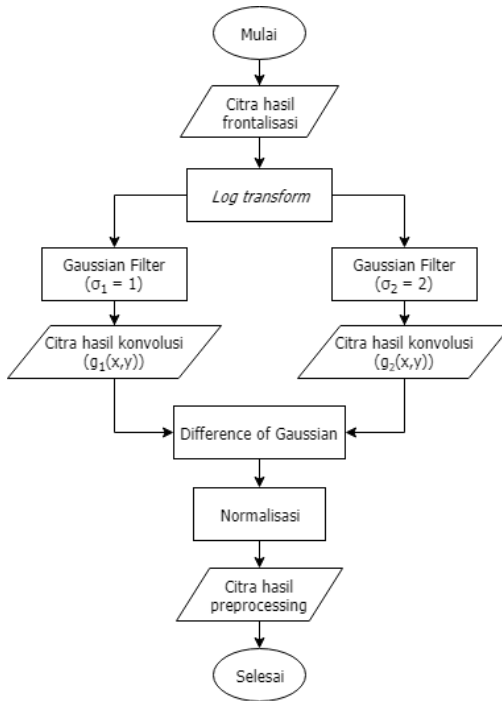
Tahap terakhir adalah klasifikasi, SVM digunakan dalam dataset FERET, dimana tujuannya adalah identifikasi/pengenalannya.

3.3. Perancangan Proses

Perancangan proses dilakukan untuk memberikan gambaran mengenai setiap proses yang terdapat pada Bagian dari setiap proses utama aplikasi dapat dilihat pada **Gambar 3.2**.

3.3.1. Pre-processing

Citra yang telah melalui proses frontalisasi selanjutnya akan dipreprocess menggunakan Difference of Gaussian (DoG). Seperti dijelaskan pada Gambar 3.3 Diagram Alir *Pre-processing*, sebelum melakukan *filtering*, terlebih dahulu dilakukan *log transform* pada citra hasil frontalisasi. Setelah itu, dilakukan konvolusi citra dengan Gaussian yang memiliki dua standar deviasi yang berbeda.



Gambar 3.3 Diagram Alir *Pre-processing*

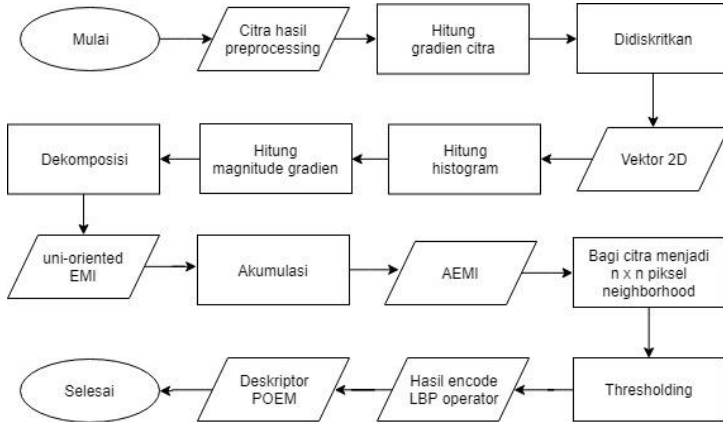
Filter pertama menggunakan $\sigma = 1$ sehingga menghasilkan $g_1(x, y)$ dan filter lainnya menggunakan $\sigma = 2$ sehingga menghasilkan $g_2(x, y)$.

Setelah didapatkan dua citra hasil konvolusi, maka perhitungan DoG dilakukan dengan mencari selisih antara kedua citra hasil konvolusi menggunakan persamaan (2.5).

Selanjutnya, dilakukan normalisasi pada hasil dari DoG. Pertama-tama dilakukan pemotongan dari persentase batas bawah dan batas atas yang sudah ditentukan dari histogram citra. Persentase yang digunakan sebagai batas atas dan bawah pada tugas akhir ini adalah 0.2. Tujuan dari pemotongan histogram ini adalah untuk menghindari adanya nilai yang terlalu terang pada citra yang menyebabkan bagian lain pada citra menjadi lebih gelap

jika dilakukan *rescaling*. Kemudian, dilakukan pemetaan kembali nilai intensitas pada citra ke dalam interval 0 sampai 255.

3.3.2. Ekstraksi Fitur POEM



Gambar 3.4 Diagram Alir Ekstraksi Fitur POEM

Langkah pertama dalam ekstraksi fitur ini adakah menghitung gradien citra grayscale hasil *pre-processing*. Gradien merupakan turunan pertama yang dihitung sebagai akar pangkat dari jumlah kuadrat dari dua derivatif (x dan y) dan dinotasikan secara matematis sebagai persamaan (3.1) atau (3.2). Gradien citra dapat dihitung dengan menggunakan rumus:

$$\text{Gradien} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (3.1)$$

atau

$$\text{Gradien} = \sqrt{(I_x)^2 + (I_y)^2} \quad (3.2)$$

Dimana I adalah citra graylevel. I_x merupakan matrik terhadap sumbu-x dan I_y merupakan matrik terhadap sumbu-y. I_x dan I_y dapat dihitung dengan (3.3).

$$I_x = I * D_x, I_y = I * D_y \quad (3.3)$$

D_x adalah mask $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$, sedangkan D_y adalah mask $\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$

masing-masing dihitung dengan cara konvolusi. Kemudian gradien ditransformasi ke dalam kordinat sumbu dengan sudut diantara 0 sampai 180° yang disebut orientasi gradien. Orientasi gradien (θ) dapat dihitung dengan (3.4).

$$\theta = \arctan\left(\frac{I_x}{I_y}\right) \quad (3.4)$$

Kemudian, orientasi gradien dari setiap piksel didiskritkan (0- π) tidak bertanda. Sehingga, di setiap piksel memiliki gradien berupa vektor 2D, yang berisi orientasi dan arah.

Tahap selanjutnya adalah melakukan perhitungan histogram dari orientasi gradien tiap sel. Setiap piksel dalam sebuah sel mempunyai nilai histogram sendiri-sendiri berdasarkan nilai yang dihasilkan dalam perhitungan gradien.

Langkah selanjutnya adalah menggabungkan informasi gradien dari piksel-piksel tetangga dengan menghitung histogram dari orientasi gradien atas semua piksel. Pada setiap piksel, fitur yang dimiliki sekarang adalah vector m , dimana m merupakan jumlah bin dalam histogram. Bobot setiap piksel bergantung pada magnitude gradien dari piksel tersebut.

Dalam praktiknya, gambar magnitude tepi berorientasi (uni-oriented EMI) didapatkan dari hasil dekomposisi orientasi gradien piksel. Nilai piksel dalam uni-oriented EMI adalah gradien magnitude. Untuk setiap piksel pada uni-oriented EMI, nilainya kemudian diganti dengan jumlah semua nilai dalam sel, yang piksel centralnya merupakan piksel saat ini. Perhitungan ini sangat cepat (menggunakan keunggulan integral gambar). Gambar hasil dirujuk ke akumulasi EMI (AEMI). Operator LBP diterapkan pada AEMI ini untuk mendapatkan gambar POEM. Untuk memasukkan lebih banyak informasi spasial ke deskriptor akhir, gambar POEM secara

spasial dibagi menjadi beberapa wilayah yang tidak tumpang tindih, dan histogram diekstraksi dari masing-masing daerah.

Akhirnya, pengkodean besaran akumulasi dilakukan menggunakan operator LBP di dalam blok. AEMI yang didapatkan dibagi ke dalam daerah berukuran $n \times n$. Kemudian, dilakukan *thresholding* pada piksel tetangga dengan nilai dari piksel pusat. Jika piksel tetangga bernilai kurang dari piksel pusat, maka dikodekan sebagai 0. Sebaliknya, jika piksel tetangga bernilai lebih dari piksel pusat, maka dikodekan sebagai 1. Pengkodean dimulai dari piksel di atas piksel pusat dan bergerak searah jarum jam sehingga menghasilkan 8-bit *code*. Hanya pola yang seragam, yang merupakan pola biner yang memiliki paling banyak 2 transisi dari 0 ke 1, biasanya digunakan untuk mempercepat metode ini.

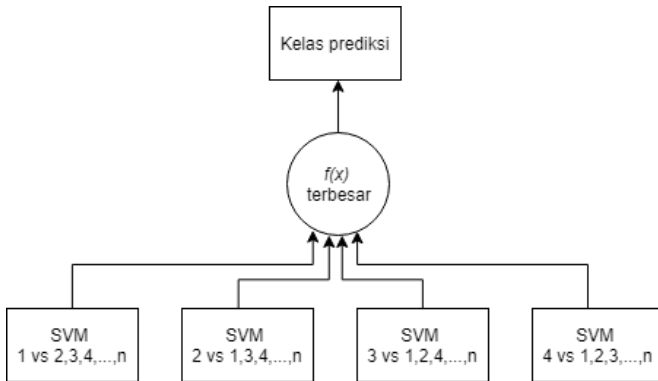
Terakhir, hasil perhitungan semua histogram yang telah dibagi dari semua wilayah POEM disatukan ke dalam histogram tunggal (POEM-HS) untuk merepresentasikan wajah yang diberikan.

3.3.3. Klasifikasi SVM

Pada Klasifikasi multi kelas, output dari set data memiliki lebih dari dua kelas atau kategori. Dengan jumlah kelas yang lebih dari dua, maka diperlukan pendekatan yang berbeda dengan kasus dua kelas. Ada dua pilihan untuk mengimplementasikan multi class SVM yaitu dengan menggabungkan semua data yang terdiri dari beberapa kelas ke dalam sebuah bentuk permasalahan optimasi. Pada tugas akhir ini, dilakukan pendekatan “*one vs rest*”.

Pengenalan dilakukan setelah mendapatkan fitur dari masing-masing wajah yang selanjutnya diproses oleh classifier support vector machine untuk mendapatkan cost penempatan pada classifier dari tiap wajah. *Classifier support vector machine* dilatih dengan cara *one vs rest*. *Classifier support vector machine* hasil latihan menggunakan *one vs rest* akan berjumlah sebanyak n kelas data latih wajah yang digunakan untuk melatih classifier support vector machine berupa classifier ke-1 sampai classifier ke- n . Tiap classifier support vector machine akan dilatih dengan hanya terdiri dari dua kelas yaitu benar dan salah. Kelas benar adalah wajah yang

merupakan kelas ke- i dari data latih, sedangkan kelas yang salah adalah semua kelas wajah yang bukan dari kelas ke- i .



Gambar 3.5 Diagram Alur SVM

Model klasifikasi dibangun sebanyak $\frac{n(n-1)}{2}$ dimana n adalah jumlah kelas. Pada tahap ini, data masukan berupa fitur hasil ekstraksi fitur POEM.

Pada metode *one vs rest*, Setiap model klasifikasi dilatih pada data dari semua kelas. Selanjutnya model klasifikasi yang telah dibangun dilakukan voting sebagaimana ditunjukkan pada Gambar 3.6.

BAB IV IMPLEMENTASI

Pada bab ini diuraikan mengenai implementasi sistem dari rancangan metode yang telah dibahas pada Bab III meliputi algoritma dan kode program dalam sistem. Selain itu, implementasi dari tiap proses, parameter masukan, keluaran dan beberapa keterangan yang berhubungan dengan program dan teori juga dijelaskan.

4.1. Lingkungan Implementasi

Objek citra yang akan diolah pada implementasi tugas akhir ini adalah sebuah citra wajah manusia. Objek citra adalah gambar *Joint Photographic Experts Group* (JPEG).

Dalam implementasi algoritma pengolahan citra tersebut, digunakan perangkat pendukung sebagai berikut:

4.1.1. Perangkat Keras

Lingkungan implementasi pada tugas akhir ini adalah sebuah *personal computer* (PC). Perangkat PC yang digunakan adalah tipe desktop *mini tower* bertipe DELL INSPIRON 3668 dengan sebuah layar monitor.

Spesifikasi dari PC yang digunakan pada tugas akhir ini adalah: memiliki prosesor Intel Core i5-7400 dengan kecepatan 3,0 GHz dan *Random Access Memory* (RAM) untuk proses menjalankan program sebesar 8,00 GB.

4.1.2. Perangkat Lunak

Pada **Tabel 4.2** menjelaskan implementasi perangkat lunak yang digunakan dalam proses pengembangan dan implementasi sistem ini.

Tabel 4.1 Lingkungan Implementasi Perangkat Lunak

Nama Aplikasi	Kegunaan
Microsoft Windows 10	Sebagai sistem operasi pada perangkat keras pengembangan aplikasi

MATLAB R2017a	Sebagai IDE untuk implementasi ekstraksi fitur POEM yang didukung dengan dua <i>toolbox</i> utama yaitu <i>statistical and machine learning toolbox</i> dan <i>image processing toolbox</i> .
Python 3.6	Sebagai IDE untuk implementasi klasifikasi SVM.
Microsoft Excel	Sebagai alat dukung untuk pengolahan angka.

4.2. Implementasi Frontalisasi

Tahap frontalisasi pada tugas akhir ini menggunakan metode pengenalan wajah SDM[xx]. Pertama, citra dimasukkan dalam variable *I_Q*. Kemudian *fitur_XY* merupakan variable penyimpanan titik – titik fitur wajah. *Facial_feature_detection* adalah fungsi yang dipanggil untuk mengidentifikasi wajah pada citra tersebut. *C_Q* adalah matriks proyeksi yang didapatkan dengan cara mengestimasi model 3D yang diciptakan, dengan *fitur_XY* yang telah didapatkan sebelumnya. *Frontal_sym* dan *frontal_raw* adalah variable hasil frontalisasi. Disini yang diambil adalah variable *frontal_sym* yang berarti hasil frontalisasi secara *soft symetry*. Kemudian dilakukan *cropping* dengan ukuran 86 x 126 piksel. Implementasi frontalisasi ditunjukkan pada Kode Sumber 4.1.

1	<code>I_Q = imread(image_path);</code>
2	<code>fitur_XY = [];</code>
3	<code>facial feature detection;</code>
4	<code>[C_Q, ~,~,~] = estimateCamera (Model3D, fitur_XY);</code>
4	<code>[frontal_sym, frontal_raw] = Frontalize(C_Q, I_Q, Model3D.refU, eyemask);</code>
5	<code>imcrop(frontal_sym, [250/4+20,250/4,250/2-40,250/2])</code>

Kode Sumber 4.1 Implementasi Pengambilan Komponen Citra Masukan

4.3. Implementasi Metode *Pre-Processing*

Tahap metode *pre-processing* citra pada tugas akhir ini menggunakan metode *Difference of Gaussian* yaitu selisih antara kedua citra hasil konvolusi menggunakan persamaan (2.5). Kemudian hasil dari DoG akan dilakukan normalisasi terlebih dahulu. Penjelasan dari metode *Difference of Gaussian* dan normalisasi terdapat pada penjelasan berikut:

4.3.1. Implementasi *Difference of Gaussian*

Tahap *Difference of Gaussian* digunakan untuk menghindari adanya nilai yang terlalu terang pada citra yang menyebabkan bagian lain pada citra menjadi lebih gelap jika dilakukan *rescaling*. Hasil dari Citra yang telah dilakukan *DoG* disimpan dalam variable *I*. Untuk mengimplementasikan *DoG* digunakan metode ke 8 yang ditunjukkan pada Kode sumber 4.2.

1	<code>I=PrePimageTool(I Q ,metode = 8);</code>
---	---

Kode Sumber 4.2 Implementasi *Difference of Gaussian*

4.3.2. Implementasi Normalisasi

Tahap normalisasi digunakan untuk menghindari adanya nilai yang terlalu terang pada citra yang menyebabkan bagian lain pada citra menjadi lebih gelap jika dilakukan *rescaling*. Hasil dari normalisasi disimpan dalam variable *normI*. Implementasi normalisasi ditunjukkan pada Kode sumber 4.3.

1	<code>normI = I - min(I(:));</code>
2	<code>normI = normI ./ max(normI(:)); % *</code>

Kode Sumber 4.3 Implementasi Normalisasi

4.4. Implementasi Ekstraksi Fitur *Patterns of Oriented Edge Magnitudes* (POEM)

Tahap ekstraksi fitur *Patterns of Oriented Edge Magnitudes* (POEM) bekerja dengan memanfaatkan karakteristik arah tepi (edge direction). Citra hasil pre-processing disimpan dalam variable *I*. untuk POEM, *metode* yang digunakan adalah *metode*=3. *Map1* adalah *struct* untuk menjalankan fitur POEM. Kemudian hasil ekstraksi fitur disimpan dalam variable *feat*. Implementasi POEM terdapat pada kode sumber 4.4

1	<code>Feat = FEtool(I,metode = 3,map1);</code>
2	<code>gradConv=[-1 0 1];</code>
3	
4	<code>radius=5;</code>
5	<code>neighbors=3;</code>
6	<code>mapping=getmapping(neighbors,'u2');</code>
7	<code>numofBlock=3;</code>
8	<code>signMode=0; % =0 unsigned</code>
9	<code>else</code>
10	<code>fx = conv2(img,gradConv,'same');</code>
11	<code>fy = conv2(img,gradConv','same');</code>
12	<code>grad = (fx.^2 + fy.^2);</code>
13	<code>grad = grad.^0.5;</code>
14	<code>%grad=max(grad,0.5);</code>
15	
16	<code>if signMode==0 %unsigned</code>
17	<code>orient = atan2(fy,fx);</code>
18	<code>orient = (orient<0)*pi + orient;</code>
19	<code>grad = (fx.^2 + fy.^2);</code>
20	<code>grad = grad.^0.5;</code>
21	<code>%grad=max(grad,0.5);</code>

Kode Sumber 4.4 Implementasi Ekstraksi Fitur POEM

4.5. Implementasi Klasifikasi SVM

1	<code>from sklearn.svm import SVC</code>
---	--

2	<code>svm = SVC()</code>
3	<code>svm.fit(data, label)</code>
4	<code>from sklearn.metrics import accuracy_score</code>

Kode Sumber 4.5 Implementasi Klasifikasi SVM

[Halaman ini sengaja dikosongkan]

BAB V

PENGUJIAN DAN EVALUASI

Pada bab ini dibahas tentang hasil uji coba dan evaluasi pada sistem yang telah dirancang dan dibuat pada tugas akhir ini. Uji coba dilakukan untuk mengetahui kinerja sistem dengan lingkungan uji coba yang telah ditentukan.

5.1. Lingkungan Uji Coba

Dalam proses pengujian sistem, dibutuhkan suatu lingkungan uji coba yang sesuai dengan standar kebutuhan. Uji coba sistem ini dilakukan dengan menggunakan *personal computer* (PC). Berikut ini spesifikasi PC yang digunakan.

- Perangkat keras
 - Intel® Core™ i5-4460S CPU @ 2.90GHz 2.90 GHz
 - *Installed Memory* (RAM) 8,00 GB
- Perangkat lunak
 - Windows 10
 - MATLAB R2017a didukung dengan *toolbox* yaitu *image processing toolbox*
 - Python 3.6
 - Microsoft Excel

5.2. Data Uji Coba

Data uji coba yang digunakan sebagai masukan adalah citra wajah manusia. Dari database FERET dilakukan pemilihan citra sebanyak 1196 buah. Untuk menguji kebenaran dari hasil deteksi, digunakan kelas Fa sebagai *ground truth* yang telah disediakan oleh FERET.

Untuk pemilihan citra dilakukan dengan mengambil citra berdasarkan kelas dari FERET yang tergolong sebagai kelas Fb, Fc, Dup1, Dup 2, sebanyak 2345 buah. Contoh citra Fa beserta

deteksinya berdasarkan *ground truth* dari FERET ditunjukkan pada **Gambar 5.1**.



Gambar 5.1 Citra Wajah FERET
(a) Kelas Fa, (b) Kelas Fb

5.3. Skenario Uji Coba

Uji coba dilakukan untuk mengetahui nilai-nilai parameter yang tepat untuk digunakan pada masing-masing proses. Penting untuk mengetahui nilai parameter yang tepat karena penggunaan parameter yang tepat akan memberikan hasil yang terbaik pada keluaran tiap proses. Data masukan adalah set Fa dari *dataset* FERET.

Skenario pengujian terdiri dari tiga macam yaitu:

1. Uji coba penentuan jumlah bin
2. Uji coba penentuan jumlah sel
3. Uji coba penentuan jumlah blok

5.3.1. Uji Coba Penentuan Jumlah Bin Pada Ekstraksi Fitur POEM

Uji coba dilakukan dengan menghitung akurasi dari klasifikasi data fitur bentuk citra. Uji coba dilakukan dengan mengubah nilai parameter jumlah bin pada fitur POEM. Hal ini penting ditentukan karena jumlah bin yang sesuai dapat meningkatkan kinerja klasifikasi. Jumlah bin yang diujikan adalah 2, 3, dan 4.

Hasil uji coba ditunjukkan pada **Tabel 5.1**

5.3.2. Uji Coba Penentuan Ukuran Sel Pada Ekstraksi Fitur POEM

Uji coba dilakukan dengan menghitung akurasi dari klasifikasi data fitur bentuk citra. Uji coba dilakukan dengan mengubah nilai parameter jumlah blok pada fitur POEM. Hal ini penting ditentukan karena jumlah sel yang sesuai dapat meningkatkan kinerja klasifikasi. Jumlah sel yang diujikan adalah 3x3 dan 5x5. Pengujian dilakukan dengan menghitung akurasi dari masing-masing *testing set* Fb, Fc, Dup1, dan Dup2.

Hasil uji coba ditunjukkan pada **Tabel 5.1**.

5.3.3. Uji Coba Penentuan Ukuran Blok Pada Ekstraksi Fitur POEM

Uji coba dilakukan dengan menghitung akurasi dari klasifikasi data fitur bentuk citra. Uji coba dilakukan dengan mengubah nilai parameter jumlah sel pada fitur POEM. Hal ini penting ditentukan karena jumlah blok yang sesuai dapat meningkatkan kinerja klasifikasi. Jumlah blok yang diujikan adalah 3x3 dan 5x5. Pengujian dilakukan dengan menghitung akurasi dari masing-masing *testing set* Fb, Fc, Dup1, dan Dup2.

Hasil uji coba ditunjukkan pada **Tabel 5.1**.

Tabel 5.1 Hasil Uji Coba

Jumlah Bin	Ukuran Sel	Ukuran Blok	Akurasi FERET (%)			
			Fb	Fc	Dup1	Dup2
3	5x5	5x5	70.1	78.3	51.6	46.0
		10x10	68.7	76.2	53.1	50.3
	8x8	5x5	71.9	80.4	53.7	48.6
		10x10	71.9	80.4	57.3	54.1
4	5x5	5x5	70.5	76.2	56.4	48.2
		10x10	68.8	76.8	57.2	54.1
	8x8	5x5	79.1	89.1	68.8	61.8
		10x10	71.9	86.0	61.7	59.7

5.4. Evaluasi

Pada subbab ini akan dijelaskan hasil dari serangkaian uji coba yang dilakukan dan kendala yang dihadapi selama proses pengerjaan. Evaluasi pada tahap pengenalan adalah jumlah bin, ukuran sel, dan ukuran blok pada Ekstraksi fitur POEM. Hasil pengenalan menunjukkan bahwa kelas Fc memiliki tingkat akurasi yang lebih baik daripada kelas Fb, Dup1, dan Dup2. Hal ini dikarenakan Fc memiliki jumlah kelas orang yang sedikit dan gambar yang hampir mirip dengan kelas Fa.

5.5. Evaluasi Tahap Frontalisasi

Dalam tahap frontalisasi, tidak semua citra wajah pada kelas Fa dapat terdeteksi. Frontalisasi dilakukan pada semua kelas, tetapi karena kelas Fa digunakan sebagai galeri atau data latih, maka tingkat keberhasilannya harus tinggi. Dari jumlah 1196 citra, hanya terdeteksi sebanyak 1184 citra. Hal ini dikarenakan oleh input citra wajah yang kurang dikenali fitur wajahnya (dua mata, satu hidung, satu mulut). Beberapa dikarenakan adanya pantulan cahaya pada orang yang menggunakan kacamata. Contoh citra yang tidak dideteksi adalah gambar 5.2 berikut.



Gambar 5.2 Kelas Fa yang tidak terdeteksi

BAB VI

KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang diperoleh berdasarkan hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah. Selain itu juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

6.1. Kesimpulan

Dari hasil pengamatan yang diperoleh dari uji coba dan evaluasi yang dilakukan, maka dapat disimpulkan bahwa akurasi terbaik didapatkan ketika jumlah bin sebanyak 4, ukuran sel 8x8, dan ukuran blok 5x5. Akurasi yang didapatkan pada kelas Fb sebesar 79.1%, Fc sebesar 89.1%, Dup1 68.8%, dan Dup2 61.8%.

6.2. Saran

Saran-saran yang dapat diberikan dalam pengujian metode ekstraksi fitur POEM adalah sebagai berikut:

1. Peningkatan kemampuan perangkat keras untuk melakukan komputasi dapat mempercepat proses learning, serta memungkinkan penelitian pada banyak parameter.
2. Melakukan perhitungan klasifikasi atau pencocokan dengan metode yang lain.
3. Mencoba sistem ini dengan menggunakan dataset lain.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA





- [1] N.S. Vu, Exploring patterns of gradient orientations and magnitudes for face recognition, *IEEE Trans. Inf. Foren. Secur.* 8 (2) (2013) 295–304.
- [2] N.-S. Vu and A. Caplier, "Face Recognition with Patterns of Oriented Edge Magnitudes," in *European Conference on Computer Vision*, Heraklion, 2010.
- [3] A. Lumini, L. Nanni and S. Branham, "Ensemble of texture descriptors and classifiers for face recognition," *Applied Computing and Informatics*, pp. 79-91, 2017.
- [4] S. Wang, W. Li, Y. Jiang, S. Jiang and R. Zhao, "An Improved Difference of Gaussian Filter in Face Recognition," *Journal of Multimedia*, vol. 7, no. 6, pp. 429-433, 2012.
- [5] M. W. Davidson and M. Abramowitz, "Molecular Expressions Microscopy Primer: Digital Image Processing - Difference of Gaussians Edge Enhancement Algorithm," 19 February 2002. [Online]. Available: <http://micro.magnet.fsu.edu/primer/java/digitalimaging/processing/diffgaussians/index.html>. [Accessed 20 December 2018].
- [6] T Ojala, M Pietikinen, and T M Enp, "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 7, 2002, pp. 971-987.
- [7] A. S. Nugroho, A. B. Witarto, D. Handoko, "Support Vector Machine, Teori dan Aplikasinya dalam Bioinformatika," 2003. [Online]. Available: <http://asnugroho.net/papers/ikcsvm.pdf>. [Accessed 3 Januari 2019].

[Halaman ini sengaja dikosongkan]





LAMPIRAN

Contoh Uji Coba



1. Fb



	Truth	Predicted
Sesuai		
Tidak sesuai		

2. Fc





	Truth	Predicted
Sesuai		
Tidak sesuai		

3. Dupl

	Truth	Predicted
Sesuai		

Tidak sesuai		
--------------	---	---

4. Dup2

	Truth	Predicted
Sesuai		
Tidak sesuai		

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Penulis, R.Ay. Noormala Nadya, lahir di Lamongan, 3 Februari 1997. Penulis menempuh pendidikan sekolah dasar di SD Negeri Babat VII (2002-2008). Melanjutkan pendidikan sekolah menengah pertama di SMP Negeri 1 Babat (2008-2011) dan selanjutnya di SMA Negeri 1 Tuban (2011-2014). Selanjutnya penulis melanjutkan pendidikan sarjana di Departemen Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut

Teknologi Sepuluh Nopember Surabaya.

Selama kuliah, penulis aktif menjadi administrator Laboratorium Komputasi Cerdas dan Visi, dan aktif dalam organisasi tingkat departemen dan kepanitiaan yang mendukung minatnya di bidang kaderisasi. Diantaranya penulis berpartisipasi sebagai anggota Departemen Kaderisasi dan Pemetaan HMTTC ITS (2015-2016), anggota Departemen PSDM UKM Robotika ITS (2015-2016), dan *Instructing Committee* (IC) Departemen Kaderisasi dan Pemetaan pada tahun 2017.

Penulis juga mengikuti kegiatan di bidang keminatan sebagai anggota UKM Robotika ITS (2014-2015). Penulis juga mengikuti kegiatan pelatihan, diantaranya berpartisipasi sebagai peserta aktif LKMM Pra Tingkat Dasar FTIf 2014 dan peserta Diklat UKM Robotika ITS 2015. Selain itu, penulis juga berpartisipasi sebagai anggota Sie NLC Schematics 2015 dan Schematics 2016.

Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Komputasi Cerdas Visi (KCV) dan memiliki ketertarikan dalam bidang Data Mining, Komputasi Biomedik, Pengolahan Citra Digital dan Visi Komputer. Penulis dapat dihubungi melalui email: raymalanadya@gmail.com