

TUGAS AKHIR - KI141502

DETEKSI EKSPRESI WAJAH MENGGUNAKAN FITUR GABOR DAN HAAR WAVELET

Oleh:

CLAUDIA PRIMASIWI

NRP 5112100140

Dosen Pembimbing

Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.

Dini Adni Navastara, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2018

[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - KI141502

DETEKSI EKSPRESI WAJAH MENGGUNAKAN FITUR GABOR DAN HAAR WAVELET

Oleh:

CLAUDIA PRIMASIWI

NRP 5112100140

Dosen Pembimbing

Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.

Dini Adni Navastara, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2018

[Halaman ini sengaja dikosongkan]

DETEKSI EKSPRESI WAJAH MENGGUNAKAN FITUR GABOR DAN HAAR WAVELET

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada

Bidang Studi Komputasi Cerdas dan Visi
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh :

CLAUDIA PRIMASIWI

Nrp. 5112100140

Disetujui oleh Pembimbing Tugas Akhir

1. Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D. (Pembimbing I)
2. Dini Adni Navastara, S.Kom., M.Sc. (Pembimbing II)



**SURABAYA
JANUARI, 2018**

[Halaman ini sengaja dikosongkan]

DETEKSI EKSPRESI WAJAH MENGGUNAKAN FITUR GABOR DAN HAAR WAVELET

Nama Mahasiswa : Claudia Primasiwi
NRP : 5112100140
Jurusan : Departemen Informatika FTIK-ITS
Dosen Pembimbing : Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.
Dini Adni Navastara, S.Kom., M.Sc.

ABSTRAK

Manusia bersosialisasi secara verbal dan non-verbal. Salah satu bahasa non-verbal yang sering digunakan manusia untuk berinteraksi adalah ekspresi wajah. Sistem pengenalan ekspresi wajah manusia dilakukan untuk mengembangkan interaksi yang natural antara manusia dan komputer.

Penelitian ini dibangun dalam tiga tahap: deteksi area wajah dan pre-processing, ekstraksi fitur, dan klasifikasi. Pendeteksian area wajah dilakukan dengan algoritma Viola-Jones. Pre-processing dilakukan menggunakan histogram equalization. Selanjutnya dilakukan ekstraksi fitur Gabor dan fitur Haar. Untuk melakukan klasifikasi, digunakan SVM One-vs-All. Fitur Gabor digunakan karena invarian terhadap rotasi, penskalaan, dan translasi. Sedangkan fitur Haar digunakan karena komputasinya yang efisien dan efektif dalam merepresentasikan sinyal dalam dimensi rendah dan tetap mempertahankan energinya.

Pada penelitian ini, digunakan kombinasi fitur Gabor dan Haar untuk dibandingkan dan digabungkan dengan fitur Landmark yang telah disediakan di database. Berdasarkan hasil uji coba, fitur Gabor dan Haar memiliki akurasi sebesar 92% dengan kelas terbaik berupa ekspresi senang.

Kata kunci: Ekspresi Wajah, Gabor, Haar Wavelet, Viola-Jones, SVM.

[Halaman ini sengaja dikosongkan]

FACIAL EXPRESSION DETECTION WITH GABOR AND HAAR WAVELET FEATURES

Student Name : Claudia Primasiwi
Registration Number : 5112100140
Department : Departemen Informatika FTIK-ITS
Supervisors : Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.
Dini Adni Navastara, S.Kom., M.Sc.

ABSTRACT

Humans socialize verbally and non-verbally. One of the non-verbal interaction that frequently used is facial expression. Facial expression recognition is developed to create a natural interaction between human and computer.

This research is built in three phases: face detection and pre-processing, feature extraction, and classification. Facial area is detected using Viola-Jones algorithm and pre-processed by histogram equalization. Next, features are extracted using Gabor and Haar wavelet. Finally, the classification is done using One-vs-All SVM. Gabor is selected due to its invariance to rotation, scaling, and translation while Haar feature is selected because of its efficiency and effectivity to compute in low dimensional signal and preserve its energy.

In this research, Gabor and Haar features are used to be compared and to be combined with Landmark features which already enclosed to the database. The best features are Gabor and Landmark features resulting in 92% accuracy with happy as the best class.

Keywords: *Facial Expression, Gabor, Haar Wavelet, Viola-Jones, SVM.*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Segala puji syukur bagi Allah SWT atas rahmat dan karunia-Nya kepada penulis sehingga penulis dapat menyelesaikan tugas akhir ini sebagai syarat untuk menyelesaikan Program Sarjana (S1) pada Departemen Informatika Fakultas Teknologi Informasi dan Komunikasi Institut Teknologi Sepuluh Nopember.

Penulis menyadari bahwa penulisan ini tidak dapat terselesaikan tanpa dukungan dari berbagai pihak, baik dukungan moril ataupun materil. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada semua pihak yang telah membantu dalam penyusunan tugas akhir ini, terutama kepada:

1. Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D. selaku Dosen Pembimbing I dan dosen wali yang telah meluangkan waktu untuk memberikan ilmu, arahan, dan solusi terhadap permasalahan pada tugas akhir ini.
2. Dini Adni Navastara, S.Kom., M.Sc. selaku Dosen Pembimbing II yang telah meluangkan waktu untuk memberikan ilmu, arahan, dan solusi terhadap permasalahan pada tugas akhir ini.
3. Kedua orang tua penulis beserta adik yang telah memberikan doa serta dukungan pada penulis.
4. Seluruh Bapak/Ibu dosen Departemen Informatika yang telah memberikan ilmu pada saat perkuliahan.
5. Seluruh staf dan karyawan Departemen Informatika yang telah memberikan bantuan kepada penulis.
6. Segenap teman-teman penulis yang telah menyemangati dan membantu penulis, terutama: Prisca, Hilmia, Fauziah, Vivin, Nurul, Nadiah, dan Ronintya.
7. Kepada para hewan peliharaan saya Cansu, Oreo, dan Gembul yang menemani saya saat sedang melakukan penelitian ini.
8. Teman-teman seangkatan TC 2012.

9. Semua pihak yang telah memberikan bantuan tetapi tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa tugas akhir ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki penulis. Oleh karena itu, penulis mengharapkan segala bentuk saran dan kritik membangun dari berbagai pihak. Semoga tugas akhir ini dapat memberikan manfaat untuk penelitian-penelitian selanjutnya.

Surabaya, Januari 2018

Claudia Primasiwi

DAFTAR ISI

Abstrak	vii
Abstract	ix
KATA PENGANTAR.....	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR KODE SUMBER	xix
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Permasalahan.....	2
1.3. Batasan Permasalahan	2
1.4. Tujuan.....	2
1.5. Manfaat.....	2
1.6. Metodologi Penelitian	2
1.7. Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA	5
2.1. Ekspresi Wajah.....	5
2.2. Histogram Equalization	7
2.3. Algoritma Viola-Jones	8
2.4. Fitur Landmark.....	14
2.5. Fitur Gabor	14
2.6. Fitur Haar	16
2.7. Principal Component Analysis (PCA).....	17
2.8. Klasifikasi Support Vector Machine (SVM).....	18
2.9. Cross Validation	19
BAB III ANALISA DAN PERANCANGAN	21
3.1. Perancangan Data	21
3.1.1. Jenis Data	21
3.1.2. Sumber Data	21
3.1.3. Teknik Pengumpulan Data	21
3.2. Perancangan Sistem.....	22
3.2.1. Deteksi Area Wajah.....	22

3.2.2.	<i>Pre-processing</i>	23
3.2.3.	Ekstraksi Fitur	23
3.2.4.	Klasifikasi.....	24
BAB IV IMPLEMENTASI SISTEM.....		29
4.1.	Implementasi Deteksi Area Wajah.....	29
4.2.	Implementasi <i>Pre-processing</i>	29
4.3.	Implementasi Ekstraksi Fitur Landmark	30
4.4.	Implementasi Ekstraksi Fitur Gabor.....	30
4.5.	Implementasi Ekstraksi Fitur Haar.....	32
4.6.	Implementasi Reduksi dan Kombinasi Fitur	33
4.7.	Implementasi Klasifikasi	34
BAB V HASIL UJI COBA DAN PEMBAHASAN		37
5.1.	Lingkungan Kerja.....	37
5.1.1.	Perangkat Lunak (<i>Software</i>).....	37
5.1.2.	Perangkat Keras (<i>Hardware</i>).....	37
5.2.	Skenario Pengujian.....	37
5.3.	Hasil Pengujian.....	38
5.3.1.	Hasil Deteksi Area Wajah dan Pre-Processing....	38
5.3.2.	Hasil Ekstraksi Fitur	39
5.3.3.	Hasil Klasifikasi	43
5.4.	Pembahasan/Diskusi.....	47
BAB VI KESIMPULAN DAN SARAN.....		49
6.1.	Kesimpulan.....	49
6.2.	Saran.....	49
DAFTAR PUSTAKA.....		51
LAMPIRAN.....		53
Lampiran 1.	Contoh Hasil Tahap Pre-processing	53
Lampiran 2.	Contoh Hasil Ekstraksi fitur Gabor	57
Lampiran 3.	Contoh Hasil Ekstraksi Fitur Haar	61
Lampiran 4.	Contoh Hasil Prediksi Ekspresi	65
BIODATA PENULIS.....		67

DAFTAR GAMBAR

Gambar 2.1 Contoh Facial Action Coding System [10].....	5
Gambar 2.2 Citra sebelum dilakukan <i>histogram equalization</i>	8
Gambar 2.3 Citra setelah dilakukan <i>histogram equalization</i>	8
Gambar 2.4 Fitur Haar-like	9
Gambar 2.5 Contoh visualisasi fitur Haar-like.....	10
Gambar 2.6 Visualisasi <i>integral image</i>	11
Gambar 2.7 Ilustrasi deteksi karakteristik wajah AdaBoost [6].....	12
Gambar 2.8 Alur kerja <i>cascade classifier</i>	13
Gambar 2.9 Visualisasi fitur Landmark dengan AAM [7] [8]	14
Gambar 2.10 Respon filter Gabor frekuensi 5 dan orientasi 8	15
Gambar 2.11 Transformasi Haar (a) 1 level (b) 2 level.....	17
Gambar 2.12 Pembagian data <i>hyperplane</i>	19
Gambar 3.1 Contoh Ekspresi Wajah [7] [8].....	22
Gambar 3.2 <i>Flowchart</i> sistem	25
Gambar 3.3 <i>Flowchart</i> tahap deteksi area wajah	26
Gambar 3.4 <i>Flowchart</i> tahap <i>pre-processing</i>	26
Gambar 3.5 <i>Flowchart</i> tahap ekstraksi fitur.....	27
Gambar 3.6 <i>Flowchart</i> tahap klasifikasi	28
Gambar 5.1 Citra Original [7] [8]	38
Gambar 5.2 Citra ekspresi (a) merendahkan (b) marah [8].....	48

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 2.1 Intensitas AU.....	6
Tabel 2.2 Deskripsi FACS.....	6
Tabel 2.3 Deskripsi ekspresi dengan FACS	7
Tabel 4.1 Ekspresi dan notasi angka	35
Tabel 5.1 Hasil deteksi area wajah dan <i>pre-processing</i>	39
Tabel 5.2 Hasil ekstraksi fitur Gabor	40
Tabel 5.3 Hasil Ekstraksi Fitur Haar	42
Tabel 5.4 Dimensi matriks fitur	42
Tabel 5.5 <i>Confusion matrix</i> fitur Landmark.....	44
Tabel 5.6 <i>Confusion matrix</i> fitur Gabor	45
Tabel 5.7 <i>Confusion matrix</i> fitur Haar	45
Tabel 5.8 <i>Confusion matrix</i> fitur Gabor dan Landmark	46
Tabel 5.9 <i>Confusion matrix</i> fitur Haar dan Landmark	46
Tabel 5.10 <i>Confusion matrix</i> fitur Gabor dan Haar.....	47
Tabel 5.11 <i>Confusion matrix</i> fitur Landmark, Gabor, dan Haar	47

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SUMBER

Kode Sumber 4.1 Implementasi Viola-Jones	29
Kode Sumber 4.2 Implementasi <i>histogram equalization</i>	30
Kode Sumber 4.3 Implementasi ekstraksi fitur <i>landmark</i>	30
Kode Sumber 4.4 Gabor <i>filter bank</i>	31
Kode Sumber 4.5 Implementasi Gabor dengan <i>Gabor array</i>	32
Kode Sumber 4.6 <i>Downsampling</i> fitur Gabor	32
Kode Sumber 4.7 Implementasi ekstraksi fitur Haar	32
Kode Sumber 4.8 Normalisasi data <i>training</i> dan <i>testing</i>	34
Kode Sumber 4.9 Implementasi PCA data <i>training</i> dan <i>testing</i>	34
Kode Sumber 4.10 Implementasi Klasifikasi	35

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1. Latar Belakang

Manusia bersosialisasi secara verbal dan non-verbal. Salah satu bahasa non-verbal yang sering digunakan manusia untuk berinteraksi adalah ekspresi wajah. Pada studi [1] didapatkan bahwa ekspresi wajah dapat menyampaikan hingga 55% informasi dalam interaksi, dibandingkan dengan vokal dan verbal keseluruhan hanya dapat memberi informasi hingga sebesar 45%. Ekspresi wajah dapat digunakan untuk mengembangkan interaksi yang natural antara manusia dan komputer. Beberapa pengenalan ekspresi [2] [3] [4] [5] telah dikembangkan dalam beberapa tahun terakhir. Pada tugas akhir ini akan memfokuskan pada pengenalan ekspresi wajah dari sebuah citra digital yang telah diekstraksi beberapa fiturnya.

Pada tugas akhir ini akan digunakan metode Viola-Jones untuk mendeteksi area wajah, Gabor dan Haar Wavelet serta Landmark sebagai fitur yang diekstraksi, dan serta SVM untuk mengklasifikasi ekspresi wajah. Viola-Jones menggunakan integral image yang dapat memproses citra lebih cepat dibandingkan dengan menggunakan pixel. AdaBoost dapat digunakan dalam permasalahan dasar dan kompleks. AdaBoost juga banyak digunakan dalam bidang biologi, komputasi visi, dan *speech recognition* sehingga pada tugas akhir ini dapat digunakan untuk mendeteksi area wajah dan mengekstraksi fitur wajah.

Oleh karena itu, dengan tugas akhir ini diharapkan dapat membantu pengenalan emosi dengan pendekatan komputasi. Pada tugas akhir ini akan dilakukan *pre-processing* citra dengan *histogram equalization* terlebih dahulu untuk mendapatkan detil-detil pada wajah yang tidak terlihat pada citra. Setelah dilakukan *pre-processing*, dilakukan deteksi area wajah menggunakan metode Viola-Jones. Deteksi area wajah dengan metode yang diusulkan oleh Viola-Jones terdapat empat langkah, yaitu ekstraksi fitur Haar-like, integral image, AdaBoost, dan *cascade classifier* [6]. Selanjutnya dilakukan ekstraksi fitur pada wajah menggunakan Gabor dan Haar

wavelet. Landmark pada wajah yang dilampirkan pada database juga akan digunakan sebagai fitur. Kemudian dilakukan klasifikasi ekspresi menggunakan SVM Multiclass Linear.

1.2. Rumusan Permasalahan

Berdasarkan latar belakang tersebut, permasalahan dalam penelitian ini adalah:

1. Bagaimana mendeteksi area wajah dari sebuah foto.
2. Bagaimana mengekstraksi fitur dari wajah tersebut.
3. Bagaimana mengklasifikasi ekspresi wajah.
4. Bagaimana evaluasi kinerja sistem deteksi ekspresi wajah.

1.3. Batasan Permasalahan

Batasan permasalahan dalam tugas akhir ini meliputi:

- Dataset untuk klasifikasi ekspresi wajah akan digunakan foto dari Extended Cohn-Kanade AU-Coded Facial Expression Database [7] [8].

- Ekspresi wajah meliputi marah, merendahkan, jijik, takut, senang, sedih, dan terkejut.

- Implementasi dilakukan dengan bahasa pemrograman dan IDE Matlab.

1.4. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk membuat sistem yang dapat mengenali ekspresi manusia dengan menggunakan fitur Gabor Haar.

1.5. Manfaat

Manfaat dari tugas akhir ini adalah untuk mengembangkan interaksi yang natural antara manusia dan komputer melalui deteksi ekspresi wajah.

1.6. Metodologi Penelitian

Metodologi penyelesaian permasalahan pada tugas akhir adalah sebagai berikut:

1. Perumusan Masalah

Untuk mendeteksi ekspresi dari sebuah citra, ada beberapa tahapan yang harus dilakukan. Tahapan pertama adalah sistem harus

mampu mendeteksi area wajah yang terdapat pada citra. Dari citra wajah yang didapatkan tersebut, sistem juga harus dapat mengekstraksi fitur-fitur wajah yang selanjutnya akan dikomputasikan untuk klasifikasi ekspresi. Pada tugas akhir ini, fitur-fitur yang akan digunakan adalah fitur Gabor dan Haar yang akan dibandingkan dan digabungkan dengan fitur Landmark yang telah didapatkan pada database [8]. Tahap selanjutnya adalah melakukan klasifikasi ekspresi wajah dengan *output* berupa prediksi ekspresi.

2. Studi Literatur

Tahap ini merupakan tahap pengumpulan informasi dan mempelajarinya untuk menyelesaikan tugas akhir yang meliputi:

1. Studi literatur tentang ekspresi wajah.
2. Studi literatur tentang bahasa pemrograman Matlab.
3. Studi literatur tentang transformasi Gabor.
4. Studi literatur tentang Haar Wavelet.
5. Studi literatur tentang Principal Component Analysis.
6. Studi literatur tentang SVM.

3. Analisa dan Perancangan Sistem

Tahap ini meliputi perancangan sistem dengan mengacu pada studi literatur sebelumnya. Dalam tahap ini dilakukan perancangan dan desain alur kerja sistem menggunakan *flowchart*.

4. Implementasi Sistem

Pada tahap ini, sistem sudah mulai dibuat secara menyeluruh. Implementasi berdasarkan dari desain yang telah dibuat pada tahap sebelumnya. Pada tahap ini juga diadakan revisi dari desain yang kurang tepat dari desain tahap sebelumnya untuk dikondisikan ke kondisi riil.

5. Hasil Uji Coba dan Pembahasan

Implementasi yang sudah selesai akan dilakukan evaluasi pada tahap ini dengan alur skenario yang telah dibuat. Jika terdapat kekurangan ataupun kesalahan dapat diperbaiki di tahap ini.

6. Kesimpulan dan Saran

Dengan dikembangkannya sistem ini, komputer dapat mengetahui ekspresi manusia sehingga diharapkan dapat mengembangkan interaksi yang natural antara manusia dan komputer.

1.7. Sistematika Penulisan

Sistematika penulisan dari tiap bab yang dibahas pada tugas akhir ini antara lain :

Bab I PENDAHULUAN, berisi latar belakang penulisan laporan ini.

Bab II TINJAUAN PUSTAKA, berisi tentang tinjauan pustaka yang digunakan untuk menyelesaikan sistem yang akan dibuat.

Bab III ANALISA DAN PERANCANGAN SISTEM, berisi penjelasan dari hasil pembelajaran atau analisa hal-hal apa saja yang diperlukan untuk mengembangkan sistem yang akan dibuat.

Bab IV IMPLEMENTASI SISTEM, berisi uraian tahap-tahap yang dilakukan untuk proses implementasi sistem.

Bab V HASIL UJI COBA DAN PEMBAHASAN, berisi tentang penjelasan mengenai hasil uji coba dan pembahasan dari sistem yang telah dikembangkan.

Bab VI KESIMPULAN DAN SARAN, berisi kesimpulan dan saran yang didapat selama proses pengembangan sistem.

BAB II TINJAUAN PUSTAKA

2.1. Ekspresi Wajah

Ekspresi wajah dapat dinotasikan dengan menggunakan *Facial Action Coding System* (FACS) yang terdiri dari 46 *Action Unit* (AU) [9]. Pada database Cohn-Kanade Extension (CK+), intensitas AU dari sebuah citra diterjemahkan menjadi bahasa sehari-hari secara manual dengan paduan yang telah disediakan, seperti pada Gambar 2.1.

Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
					
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU 41	*AU 42	*AU 43	AU 44	AU 45	AU 46
					
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
					
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU 15	AU 16	AU 17	AU 18	AU 20	AU 22
					
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU 23	AU 24	*AU 25	*AU 26	*AU 27	AU 28
					
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

Gambar 2.1 Contoh Facial Action Coding System [10]

Berdasarkan penelitian Cohn-Kanade [7] [8], ekspresi wajah dapat dideskripsikan dengan *Action Unit* dengan paduan mengacu pada Tabel 2.1 yang menjelaskan intensitas *Action Unit* pada ekspresi tersebut dan Tabel 2.2 yang menjelaskan penomoran *Action Unit* tersebut. Pada penelitian ini, *Action Unit* sebuah ekspresi akan dideskripsikan seperti pada Tabel 2.3.

Tabel 2.1 Intensitas AU

Anotasi	Intensitas
A	<i>Trace/Sangat sedikit</i>
B	<i>Slight/Sedikit</i>
C	<i>Marked or pronounced/Terlihat jelas</i>
D	<i>Severe or extreme/Sangat jelas</i>
E	<i>Maximum/Maksimal</i>

Tabel 2.2 Dekripsi FACS

AU	Nama	AU	Nama
1	<i>Inner brow raiser</i>	18	<i>Lip puckerer</i>
2	<i>Outer brow raiser</i>	20	<i>Lip stretcher</i>
4	<i>Brow lowerer</i>	21	<i>Neck tightener</i>
5	<i>Upper lip raiser</i>	23	<i>Lip tightener</i>
6	<i>Cheek raiser</i>	24	<i>Lip pressor</i>
7	<i>Lid tightene</i>	25	<i>Lips pan</i>
9	<i>Nose wrinkler</i>	26	<i>Jaw drop</i>
10	<i>Upper lip raiser</i>	27	<i>Mouth stretch</i>
11	<i>Nasolabial deepener</i>	28	<i>Lip suck</i>
12	<i>Lip corner puller</i>	29	<i>Jaw thrust</i>
13	<i>Cheek puller</i>	31	<i>Jaw clencher</i>
14	<i>Dimpler</i>	34	<i>Cheek puff</i>
15	<i>Lip corner depressor</i>	38	<i>Nostril dilator</i>
16	<i>Lower lip depressor</i>	39	<i>Nostril compressor</i>
17	<i>Chin raiser</i>	43	<i>Eyes closed</i>

Tabel 2.3 Deskripsi ekspresi dengan FACS

Ekspresi	Kriteria
Marah	Terdapat AU23 dan AU 24
Merendahkan	Terdapat AU14
Jijik	Terdapat AU9 atau AU10
Takut	Terdapat kombinasi AU1+2+4, jika intensitas AU5 E maka AU4 dapat dihilangkan
Senang	Harus terdapat AU12
Sedih	Terdapat salah satu AU1+4+15 atau 11, kecuali AU6+15
Terkejut	Terdapat AU1+2 atau 5 dan intensitas AU5 tidak lebih dari B

2.2. Histogram Equalization

Citra yang akan digunakan dalam tugas akhir ini tidak diketahui *exposure* yang dimiliki, apakah *overexposure* atau *underexposure* yang disebabkan oleh pencahayaan lingkungan saat pengambilan gambar. *Exposure* yang tidak tepat dapat menyebabkan hilangnya detail-detail pada gambar (lihat Gambar 2.2 dan Gambar 2.3) yang kemudian dapat menjadikan hasil uji menjadi tidak valid. Oleh karena itu pada tugas akhir ini akan dilakukan *pre-processing* pada citra dengan cara *histogram equalization*.

Histogram equalization digunakan untuk mendistribusikan histogram agar derajat keabuan dari yang paling rendah (0) hingga paling tinggi (255) seimbang. Dengan *histogram equalization* hasil citra yang memiliki *histogram* yang tidak merata akan menjadi citra yang lebih jelas karena derajat keabuannya tidak dominan gelap atau dominan terang. *Histogram equalization* berfungsi untuk memperbaiki citra dan sangat efektif untuk mendapatkan detail-detail dari citra [11].

Nilai histogram didapatkan dari perhitungan pada (2.1)

$$w = \frac{c_w th}{n_x n_y} \quad (2.1)$$

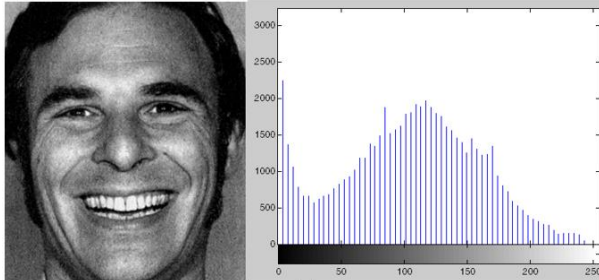
dengan

w = nilai keabuan hasil *histogram equalization*

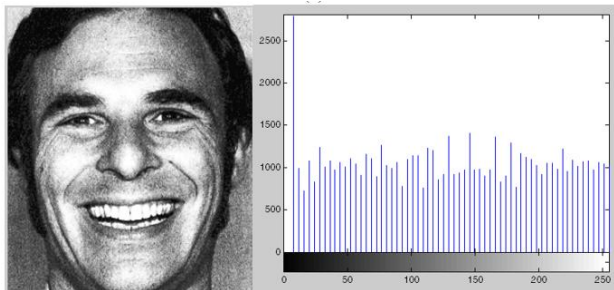
c_w = histogram kumulatif dari w

th = *threshold* derajat keabuan (256)

n_x, n_y = ukuran citra



Gambar 2.2 Citra sebelum dilakukan *histogram equalization*



Gambar 2.3 Citra setelah dilakukan *histogram equalization*

2.3. Algoritma Viola-Jones

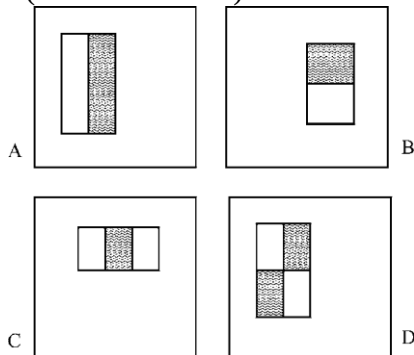
Untuk mendeteksi area wajah, sudah ada penelitian dari *Robust Real-Time Face Detection* [6] dari Viola dan Jones pada tahun 2004. Penelitian ini membahas tentang bagaimana caranya untuk mendeteksi wajah dan senyum dari suatu video secara *real time* menggunakan metode yang dirancang oleh Viola-Jones yang berisikan empat tahap, yaitu fitur Haar-like, *integral image*, AdaBoost, serta *cascade classification*.

Mengacu pada metode Viola-Jones [6], selanjutnya akan dilakukan beberapa tahap, yaitu:

1. Fitur Haar-like

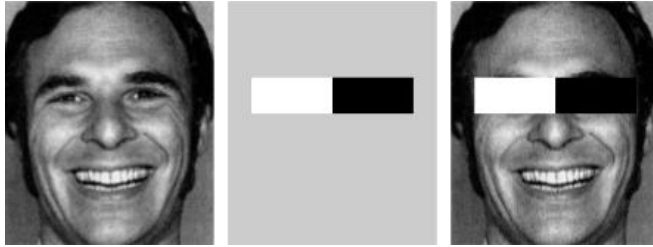
Viola-Jones menggunakan fitur Haar-like yang merupakan produk skalar dari citra dan fitur Haar-like. Pada algoritma Viola-Jones digunakan fitur Haar-like karena waktu pemrosesan citra dapat lebih cepat dibandingkan menggunakan pixel.

Dalam metode ini, digunakan tiga jenis fitur. Jumlah dari dua fitur persegi adalah selisih dari jumlah pixel dalam dua region. Fitur pertama merupakan dua-fitur-persegi yang memiliki bentuk dan ukuran yang sama secara horizontal maupun vertikal yang berdekatan. Fitur kedua adalah sebuah fitur yang terbentuk dari tiga-fitur-persegi yang didapatkan dari pengurangan dua persegi di bagian luar sehingga membentuk suatu persegi pusat. Dan fitur ketiga merupakan empat-fitur-persegi yang menghitung selisih di antara dua persegi diagonal (lihat Gambar 2.4).



Gambar 2.4 Fitur Haar-like

Dari perhitungan Haar-like tersebut diasumsikan bahwa fitur tersebut memiliki informasi yang dapat digunakan mendeteksi area wajah seperti terlihat pada **Gambar 2.5**.



Gambar 2.5 Contoh visualisasi fitur Haar-like

2. Integral Image

Integral image menghitung nilai di setiap pixel (x,y) yang merupakan jumlah dari nilai pixel yang berada di atas dan kirinya (lihat Gambar 2.6) yang dirumuskan pada persamaan (2.2).

$$II(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y') \quad (2.2)$$

dengan

$II(x, y) = \text{integral image}$

$I(x, y)$ adalah citra original.

Integral image dapat dihitung sekali dari gambar asli seperti pada Gambar 2.6 menggunakan persamaan (2.3) dan persamaan (2.4).

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2.3)$$

$$II(x, y) = II(-1, y) + s(x, y) \quad (2.4)$$

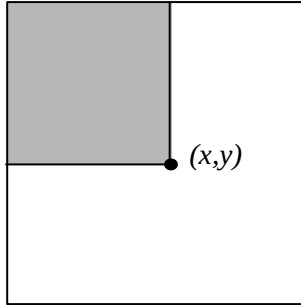
dengan

$s(x,y)$ = penjumlahan kumulatif baris

$s(x, -1) = 0$

$II(-1, y) = 0$

Dengan menggunakan *integral image*, perhitungan komputasional cenderung lebih rendah dibandingkan dengan menjumlahkan seluruh *pixel* dalam satu *window*.



Gambar 2.6 Visualisasi *integral image*

3. AdaBoost

Viola-Jones menggunakan AdaBoost untuk memilih fitur dan untuk men-*training classifier*. AdaBoost sebenarnya adalah algoritma pembelajaran yang digunakan untuk meningkatkan performa klasifikasi pada algoritma pembelajaran yang sederhana.

AdaBoost merupakan algoritma *boosting* pertama yang dapat beradaptasi dengan *weak learner* yaitu *classifier* yang memiliki sedikit korelasi dengan klasifikasi yang sebenarnya yang memenuhi syarat pada persamaan (2.5)

$$h(x, f, p, \theta) = \begin{cases} 1, & pf(x) < p\theta \\ 0, & otherwise \end{cases} \quad (2.5)$$

dengan

f = fitur

θ = threshold

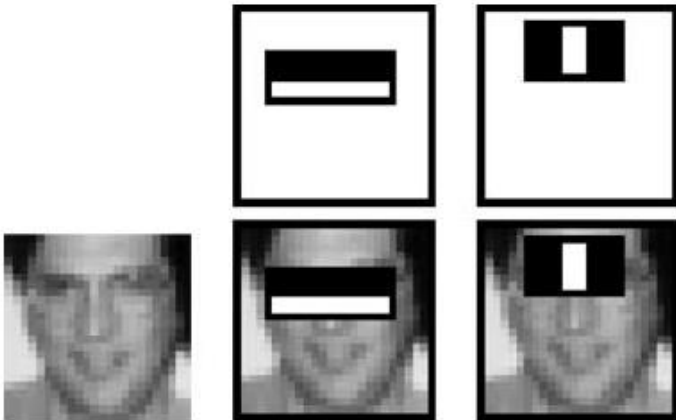
p = polaritas

AdaBoost menggabungkan beberapa *weak learner* tersebut agar menjadi *strong classifier*. Untuk meningkatkan performa *weak classifier*, setelah dilakukan pembelajaran pertama, contoh-contoh diberikan bobot ulang dengan tujuan untuk menegaskan mana contoh yang salah dalam proses klasifikasi oleh *weak learner* sebelumnya. *Strong classifier* mengambil bentuk dari *perceptron*, yaitu gabungan dari beberapa *weak classifier* yang telah diberi bobot dan *threshold*.

Dalam metode Viola-Jones, AdaBoost digunakan baik untuk memilih fitur-fitur dalam jumlah banyak maupun untuk training

classifier. Viola-Jones membatasi *weak learner* ke suatu set fungsi klasifikasi yang masing-masing bergantung pada satu fitur. Untuk itu, *weak learner* didesain untuk memilih satu fitur persegi yang dapat memisahkan contoh positif dan negatif secara baik. Sedangkan untuk setiap fitur, *weak learner* menentukan fungsi klasifikasi threshold optimal sehingga dapat meminimalisir kesalahan dalam klasifikasi.

Sebagai contoh, pada Gambar 2.7 terdapat dua fitur yang dipilih oleh AdaBoost. Fitur pertama (tengah) menghitung perbedaan intensitas terhadap area mata dan area pipi. Dari fitur pertama diketahui bahwa area mata biasanya lebih gelap dibandingkan area pipi. Sedangkan pada fitur kedua (kanan) menghitung perbedaan intensitas area mata dan area batang hidung.



Gambar 2.7 Ilustrasi deteksi karakteristik wajah AdaBoost [6]

4. Cascade Classifier

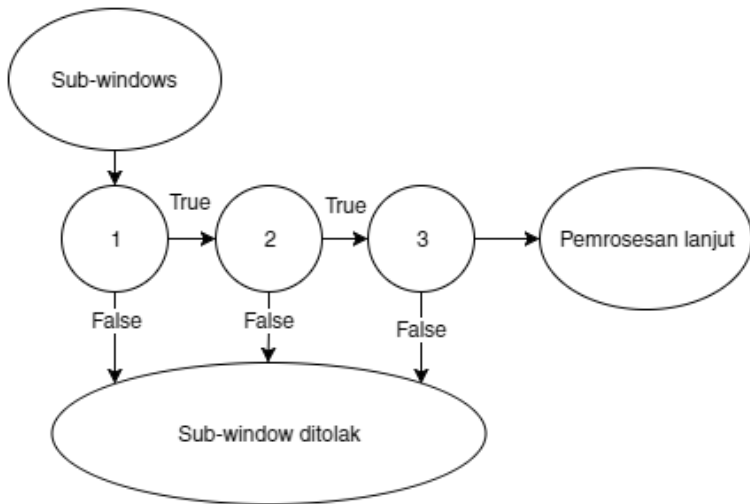
Tahapan dalam cascade classifier dibangun melalui training classifier menggunakan AdaBoost. Agar dapat mendeteksi area wajah diperlukan beberapa operasi:

1. Evaluasi fitur persegi. Diperlukan 6 hingga 9 referensi array per fitur.
2. Hitung *weak classifier* untuk setiap fitur. Diperlukan satu operasi *threshold* per fitur.

3. Gabungkan beberapa *weak classifier*.

Setiap tahap akan mengurangi tingkat false positive dan akan meningkatkan pendeteksian. Sedangkan jumlah tahapan akan terus bertambah hingga target deteksi dan false positive tertentu.

Struktur dari rancang bangun cascade (lihat Gambar 2.8) menyatakan bahwa dalam satu citra didominasi oleh sub-citra negatif. Dengan demikian *cascade* akan melakukan penolakan terhadap sub-window negatif (bukan objek yang dicari) sebanyak mungkin pada tahap sedini mungkin. Sedangkan sub-window positif (objek yang dicari) akan menyebabkan evaluasi dari setiap classifier dalam cascade, yang jarang terjadi.



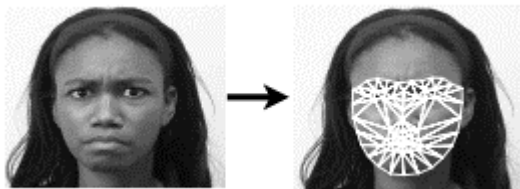
Gambar 2.8 Alur kerja *cascade classifier*

2.4. Fitur Landmark

Titik Landmark adalah titik korespondensi pada setiap obyek yang sesuai antara dan di dalam populasi [12]. Titik Landmark banyak digunakan pada bidang biometrik untuk memecahkan masalah deteksi dan pengenalan.

Pada setiap wajah terdapat berbagai macam Landmark. Beberapa Landmark (pupil, ujung hidung, dan kontur wajah) tidak mempengaruhi ekspresi seseorang, fitur seperti alis, mulut, dan bentuk mata dapat digunakan sebagai fitur untuk menentukan ekspresi seseorang. Pada penelitian ini digunakan titik landmark yang telah diekstraksi pada penelitian sebelumnya [7] [8] menggunakan *Active Appearance Model* (AAM). Dari penelitian [7] [8] tersebut, dihasilkan Landmark sebanyak 68 titik. Titik-titik tersebut disimpan ke dalam matriks berukuran 68x2 yang berisi koordinat x dan y dari titik-titik Landmark tersebut.

Titik landmark yang telah diekstraksi dari AAM disertakan dalam bentuk *wire-frame mesh* seperti terlihat pada Gambar 2.9. Titik landmark yang diekstraksi dari citra wajah akan digunakan sebagai fitur wajah.



Gambar 2.9 Visualisasi fitur Landmark dengan AAM [7] [8]

2.5. Fitur Gabor

Fitur Gabor digunakan pada penelitian *Analysis of facial expression using Gabor and SVM* [2] serta di penelitian *CloudID: Trustworthy cloud-based and cross-enterprise biometric identification* [3]. Fitur Gabor didapatkan pada citra yang diproses menggunakan *Gabor Filter Bank*. Keunggulan fitur Gabor adalah invarian terhadap rotasi, skalasi, dan translasi [3]. Pada domain

spasial, 2D Gabor Filter dapat dinotasikan seperti pada persamaan (2.6).

$$G(x, y) = \frac{f^2}{\pi\gamma\eta} \exp\left(-\frac{x^2 + \gamma^2 y^2}{2\sigma^3}\right) \exp(j2\pi f x' + \phi) \quad (2.6)$$

untuk x' dan y' masing-masing dirumuskan pada persamaan (2.7) dan persamaan (2.8).

$$x' = x \cos \phi + y \sin \phi \quad (2.7)$$

$$y' = -x \sin \phi + y \cos \phi \quad (2.8)$$

dengan

f = frekuensi sinusoida

θ = orientasi dari Gabor filter terhadap *parallel stripes*

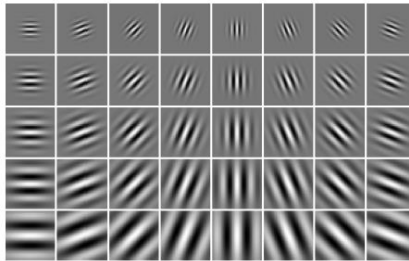
ϕ = *phase offset*

σ = standar deviasi dari *Gaussian envelope*

γ = rasio yang menentukan kelonggaran dari Gabor

Jika semua *Gabor filter* dengan variasi frekuensi (f) dan orientasi (θ) diterapkan pada satu titik tertentu (x, y), didapatkan beberapa respon *filter* untuk titik tersebut, misal: digunakan lima frekuensi ($f = 1, 2, 3, 4, 5$) dan delapan orientasi (θ), maka akan dihasilkan 40 respon *filter* untuk tiap titik citra yang dikonvolusikan dengan *filter* tersebut.

Dengan menggunakan frekuensi 5 dan orientasi 8, filter Gabor dapat divisualisasikan seperti pada Gambar 2.10.



Gambar 2.10 Respon filter Gabor frekuensi 5 dan orientasi 8

2.6. Fitur Haar

Pada penelitian yang berjudul *Face recognition based on Haar wavelet transform and principal component analysis via Levenberg-Marquardt backpropagation neural network* [4] dan *Multi-Level Haar Wavelet based Facial Expression Recognition using Logistic Regression* [5], fitur Haar diperoleh dari matriks *coefficient*-nya. Pada penelitian tersebut fitur Haar digunakan karena kemampuannya melakukan komputasi secara efisien dan efektif pada dimensi rendah dengan tetap mempertahankan energi sinyalnya.

Haar wavelet merupakan sebuah rangkaian sinyal berbentuk kotak yang diubah frekuensinya sehingga membentuk sebuah wavelet basis [5]. Wavelet induk dirumuskan pada persamaan (2.9).

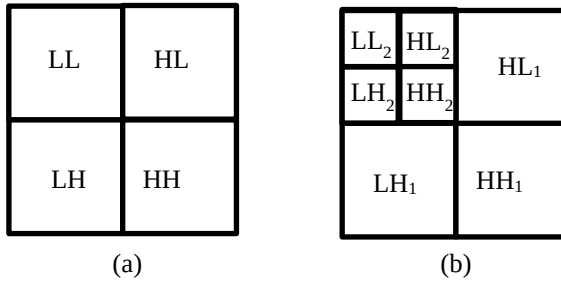
$$\psi(t) = \begin{cases} 1, & 0 \leq t < 0.5 \\ -1, & 0.5 \leq t < 1 \\ 0, & \text{otherwise} \end{cases} \quad (2.9)$$

Untuk fungsi skala wavelet, dirumuskan pada persamaan (2.10).

$$\phi(t) = \begin{cases} 1, & 0 \leq t < 1 \\ 0, & \text{otherwise} \end{cases} \quad (2.10)$$

Dalam transformasi Wavelet Haar, terdapat dua proses yang harus dilakukan yaitu transformasi *forward* (dekomposisi) dan transformasi *inverse* (rekonstruksi). Transformasi *forward* berguna untuk memecah gambar. Sedangkan transformasi *inverse* adalah kebalikannya, yaitu membentuk kembali pecahan-pecahan gambar dari proses *forward* menjadi sebuah citra seperti semula (proses rekonstruksi).

Dalam transformasi dekomposisi, sebuah citra akan ditransformasikan dengan sebuah *filter* wavelet. Hasil *filter* tersebut adalah 4 sub-bidang citra dari citra asal, yaitu *low pass-low pass* (LL), *low pass-high pass* (LH), *high pass-low pass* (HL), dan *high pass-high pass* (HH) (lihat Gambar 2.11). Tahap ini dapat dilanjutkan dengan *low pass-low pass* (LL) untuk memasuki tahap dekomposisi atau level selanjutnya.



Gambar 2.11 Transformasi Haar (a) 1 level (b) 2 level

Vektor fitur yang akan digunakan didapatkan dari *approximation coefficients* dari dekomposisi Haar level 2 pada citra wajah.

2.7. Principal Component Analysis (PCA)

PCA adalah sebuah teknik untuk membangun variabel-variabel baru yang merupakan kombinasi linear dari variabel-variabel asli. Jumlah maksimum dari variabel baru ini akan sama dengan jumlah dari variabel lama, dan variabel-variabel baru ini tidak saling berkorelasi satu sama lain [13].

Pada tugas akhir ini, fitur yang akan direduksi adalah fitur Gabor dan fitur Haar. Keduanya memiliki dimensi matriks yang cukup besar sehingga diperlukan reduksi untuk membuat komputasi lebih efektif, sedangkan fitur Landmark tidak dilakukan reduksi fitur dikarenakan dimensinya yang kecil.

Langkah-langkah PCA [14] adalah:

1. Standardisasi data.
2. Menghitung *eigenvector* (*principal component*) dan *eigenvalue*. *Eigenvector* digunakan untuk menentukan arah dari fitur baru, sedangkan *eigenvalue* menentukan magnitude atau varian data dari fitur baru.
3. Memilih *principal component* atau komponen utama dengan cara mengurutkan *eigenpairs* dari nilai *eigenvalue* terbesar.
4. Menentukan banyaknya komponen utama dengan mempertahankan 99% nilai varians.

5. Proyeksi matriks ke *space* fitur baru.

2.8. Klasifikasi Support Vector Machine (SVM)

Untuk mengklasifikasi ekspresi pada tugas akhir ini akan digunakan SVM. SVM merupakan salah satu *classifier* yang banyak digunakan. SVM dapat digunakan dalam ruang dimensi tinggi dan mudah untuk di-*training*. SVM telah terbukti memiliki kerja yang lebih baik saat digunakan dalam ruang yang padat atau pada ruang yang memiliki relevansi tinggi antar fitur-fiturnya [15]. SVM dapat digunakan untuk klasifikasi data dalam bentuk linear dan non-linear.

Untuk model klasifikasi ke- i , diasumsikan data positif merupakan seluruh data yang berada di kelas i , sedangkan data negatif adalah seluruh data yang tidak berada di kelas i .

Pada Gambar 2.12, setiap (x_i, y_i) , dengan $x_i \{ x_1, x_2, x_3, \dots, x_n \}^T$ merupakan *data training* dan $y_i \{-1, +1\}$ merupakan label kelas.

Hyperplane adalah garis yang memisahkan kelas -1 dan +1. Data x_i ditemukan di *hyperplane* apabila memenuhi persamaan (2.11).

$$x_i w + b = 0 \quad (2.11)$$

Apabila terletak di kelas -1, berarti x_i memenuhi persamaan (2.12), sedangkan jika terletak di kelas +1, persamaan yang terpenuhi adalah persamaan (2.13).

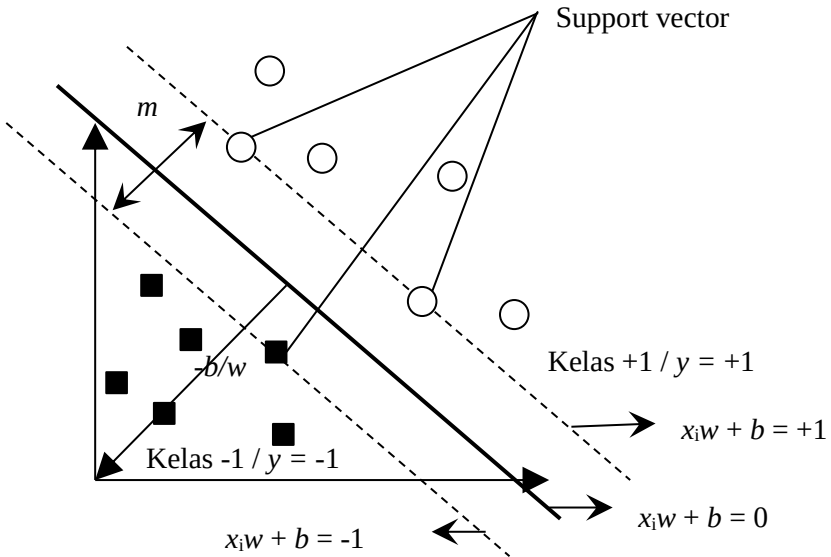
$$x_i w + b \leq -1 \quad (2.12)$$

$$x_i w + b \geq +1 \quad (2.13)$$

Hyperplane yang baik memiliki margin atau jarak yang besar di antara *hyperplane* ke data terdekat di setiap kelas.

Metode SVM hanya dapat melakukan klasifikasi untuk dua kelas. *Multiclass* mereferensikan klasifikasi yang terdiri dari lebih dari dua kelas. Pada tugas akhir ini diperlukan tujuh kelas (kelas ekspresi marah, merendahkan, jijik, takut, senang, sedih, dan terkejut). Tipe klasifikasi *multiclass* yang akan diimplementasi adalah *One-vs-All* menggunakan fungsi kernel *linear*. Pada penelitian

ini, pendekatan yang digunakan adalah *Error Correcting Output Code* (ECOC). Cara kerja ECOC yaitu dengan memberikan *string bit* yang disebut *codeword* dengan panjang n sesuai jumlah kelas ke setiap kelas. Setelah itu dibuat sejumlah N -*binary classifier* untuk memprediksi setiap bit *codeword*.



Gambar 2.12 Pembagian data hyperplane

2.9. Cross Validation

Cross-validation merupakan metode statistik yang mengevaluasi dan membandingkan algoritma pembelajaran dengan membagi data menjadi dua yaitu *data training* dan *data testing*. Bentuk dari cross validation adalah *k-fold cross validation*.

Pada tugas akhir ini akan digunakan *10-cross validation*. Penggunaan 10 fold ini dianjurkan karena merupakan jumlah fold terbaik untuk uji validitas [16]. *10-fold* tersebut akan mengulang pengujian sebanyak 10 kali dan hasil pengukuran adalah nilai rata-rata dari 10 kali pengujian tersebut. Validasi model dapat diukur

dengan menghitung akurasi data. Akurasi dapat dirumuskan pada persamaan (2.14)

$$akurasi = \frac{jumlah\ klasifikasi\ benar}{jumlah\ data} \times 100\% \quad (2.14)$$

BAB III

ANALISA DAN PERANCANGAN

3.1. Perancangan Data

Pada sub-bab ini akan dijelaskan perancangan data yang meliputi jenis data, sumber data, dan teknik pengumpulan data.

3.1.1. Jenis Data

Dalam penelitian ini, data yang digunakan merupakan data kualitatif, yaitu data yang berupa angka. Data tersebut berupa citra wajah yang menampilkan tujuh ekspresi dasar manusia.

3.1.2. Sumber Data

Data yang digunakan dalam penelitian ini merupakan data sekunder. Data yang diperoleh tidak didapatkan secara langsung, tetapi didapatkan dari penelitian yang telah dilakukan sebelumnya.

3.1.3. Teknik Pengumpulan Data

a. Studi Literatur

Studi literatur didapatkan dari berbagai sumber tentang ekspresi wajah, ekstraksi fitur Gabor dan Haar, serta klasifikasi SVM yang didapatkan dari jurnal, laporan penelitian, buku, serta beberapa artikel di internet.

b. CK+ Database

Extended Cohn-Kanade AU-Coded Facial Expression Database atau CK+ Database [8] merupakan database yang berisi rangkaian ekspresi wajah manusia yang terdiri dari 327 citra. Database tersebut mewakili masing-masing ekspresi dasar manusia, diantaranya ekspresi marah sebanyak 45 citra, ekspresi merendahkan sebanyak 18 citra, ekspresi jijik sebanyak 59 citra, ekspresi takut sebanyak 25 citra, ekspresi senang sebanyak 69 citra, ekspresi sedih sebanyak 28 citra, dan ekspresi terkejut sebanyak 83 citra. Contoh ekspresi wajah dapat dilihat pada Gambar 3.1.

Selain berisi citra, database tersebut juga dilengkapi dengan Landmark dari hasil penelitian sebelumnya [7] [8] yang digunakan untuk mengetahui posisi fitur wajah dari citra tersebut.



Gambar 3.1 Contoh Ekspresi Wajah [7] [8]

3.2. Perancangan Sistem

Penelitian ini akan diimplementasi menggunakan Matlab. Sistem ini berisi beberapa skrip yang dapat dipanggil melalui *command window* dan fungsi yang dipanggil dari fungsi.

Input dari sistem ini berupa citra dengan dimensi 640x490 pixel. *Output* yang dihasilkan berupa prediksi kelas serta kelas aktual dari citra yang diprediksi yang dinotasikan dengan angka.

Secara keseluruhan, rancangan alur program dibagi menjadi tiga proses, yaitu *pre-processing*, ekstraksi fitur, dan klasifikasi. Proses tersebut dapat dilihat secara lengkap pada Gambar 3.2.

3.2.1. Deteksi Area Wajah

Pada tahap ini, hal pertama yang akan dilakukan adalah menginputkan citra original yang akan diproses. Citra original tersebut lalu dideteksi area wajahnya menggunakan algoritma Viola-Jones yang akan menghasilkan citra hasil deteksi area wajah.

Algoritma Viola-Jones terdiri dari empat proses, yaitu ekstraksi fitur Haar-like, perhitungan fitur menggunakan *integral image*, pemilihan fitur menggunakan AdaBoost, dilanjutkan dengan

cascade classifier untuk mencari area wajah. Apabila terdapat *false detection*, yang akan diambil adalah area terdeteksi yang paling besar. Dari tahap ini, dihasilkan citra hasil deteksi area wajah yang berukuran 360x360. Alur kerja pada tahap ini dapat dilihat pada Gambar 3.3.

3.2.2. *Pre-processing*

Setelah didapatkan citra wajah, dilakukanlah *histogram equalization* sehingga menghasilkan citra hasil *pre-processing* yang siap untuk diproses. Input yang akan digunakan adalah citra hasil deteksi area wajah yang telah didapatkan pada tahapan sebelumnya. Pada tahap *pre-processing* akan didapatkan citra hasil *pre-processing* yang telah diseimbangkan derajat keabuannya. Secara visual, alur kerja tahap *pre-processing* ini dapat dilihat pada Gambar 3.4.

3.2.3. Ekstraksi Fitur

Setelah citra hasil *pre-processing* didapatkan, selanjutnya dilakukan tahap ekstraksi fitur. Pada penelitian ini menggunakan tiga fitur, yaitu fitur Landmark, fitur Gabor, dan fitur Haar. Untuk mengekstraksi fitur Landmark, dilakukan dengan membaca data yang sudah disediakan di dalam database.

Untuk mengekstraksi fitur Gabor, terdapat dua proses utama yang akan dilakukan yaitu pembuatan *Gabor array* dan ekstraksi fitur Gabor. Proses pertama yaitu pembuatan *Gabor array* menggunakan frekuensi bernilai 5 dan orientasi bernilai 8, yang menghasilkan *array* sebesar 5x8, dengan masing-masing elemen sebesar 39x39. Setelah didapatkan *Gabor array*, dilanjutkan dengan ekstraksi *vector feature* dari citra hasil *pre-processing* menggunakan filter dari *Gabor array* tersebut.

Sedangkan untuk mengekstraksi fitur Haar, didapatkan dari matriks *approximation coefficient* dari transformasi dekomposisi Haar level ke-2 terhadap citra hasil *pre-processing*.

Setelah didapatkan fitur Gabor dan fitur Haar pada tahap sebelumnya, dilakukan pembagian data secara acak untuk menjadi data *training* dan data *testing*. Normalisasi data hanya dilakukan pada data *training* dan *testing* fitur Gabor dan Haar. Data normalisasi

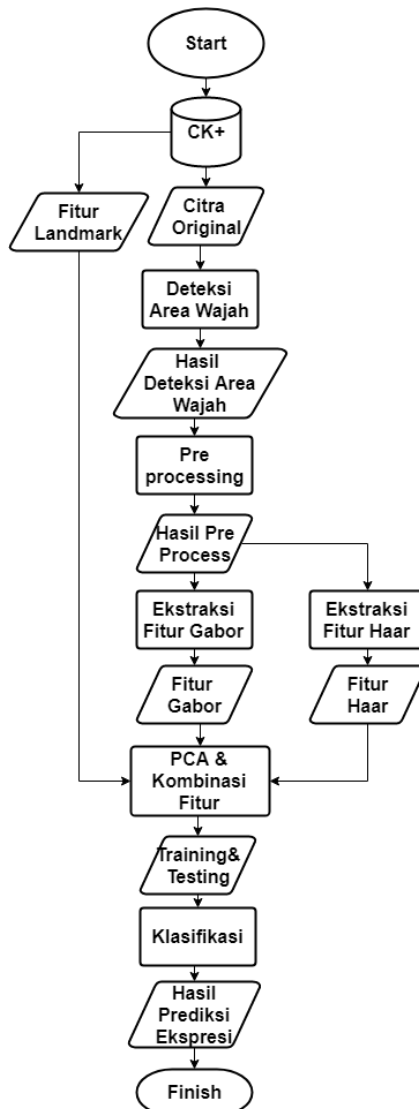
tersebut akan digunakan untuk reduksi fitur menggunakan PCA pada data *training* dan *testing* fitur Gabor dan fitur Haar. Selanjutnya dihitung *principal component* dari fitur ternormalisasi tersebut. Hasil dari proses ini merupakan kombinasi fitur Gabor, Haar, dan Landmark yang dilakukan pada tiap data *training* dan data *testing* untuk digunakan pada tahap klasifikasi.

Untuk mengekstraksi fitur Gabor dan Haar, citra hasil *pre-processing* yang telah di-*resize* menjadi 120x120 akan digunakan sebagai input. Secara lengkap, alur kerja tahap ekstraksi fitur dapat dilihat pada Gambar 3.5.

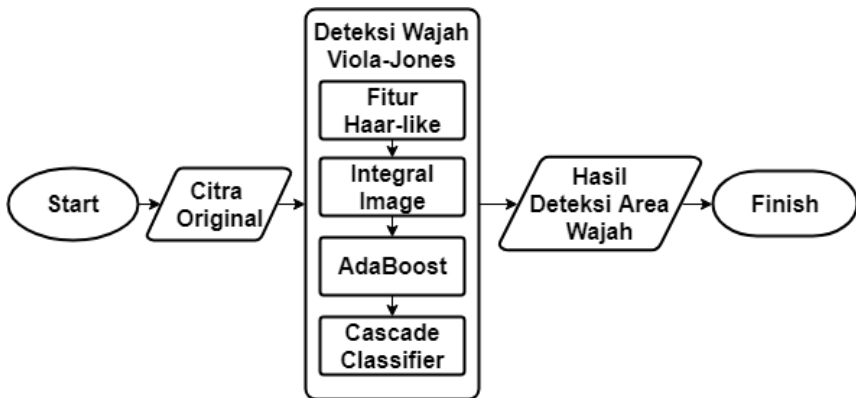
3.2.4. Klasifikasi

Setelah didapatkan fitur yang diperlukan, dilakukan tahap klasifikasi. Dari 327 data fitur-fitur kombinasi, data yang digunakan untuk klasifikasi akan dibagi secara random menjadi 10 bagian dengan menggunakan *10-cross validation*, dengan 9 bagian *data training* dan 10 bagian *data testing*. Proses tersebut dilakukan sebanyak 10 kali dengan kombinasi kelompok data berbeda-beda.

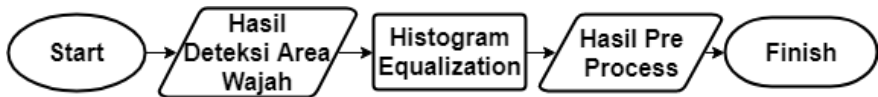
Selanjutnya dibuat model kasifikasi SVM sebelum melakukan prediksi data *testing* dengan SVM. Hasil dari proses klasifikasi disimpan dalam sebuah tabel *confusion matrix*. Dari tabel *confusion matrix* dapat diketahui jumlah data yang teridentifikasi secara benar untuk selanjutnya dihitung akurasi hasil prediksinya. Secara lengkap, alur kerja tahap klasifikasi dapat dilihat pada Gambar 3.6.



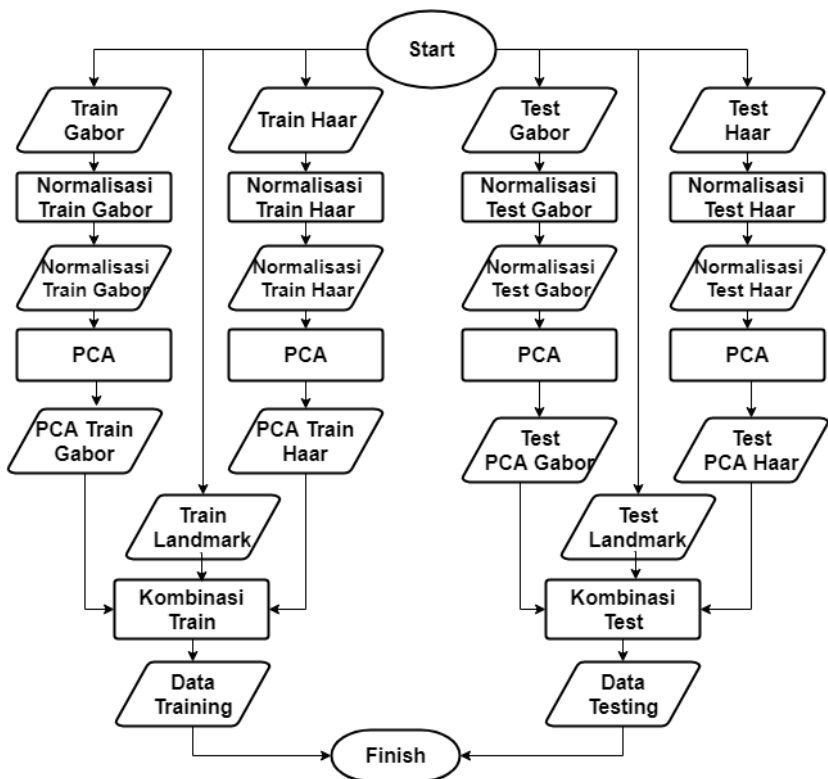
Gambar 3.2 *Flowchart* sistem



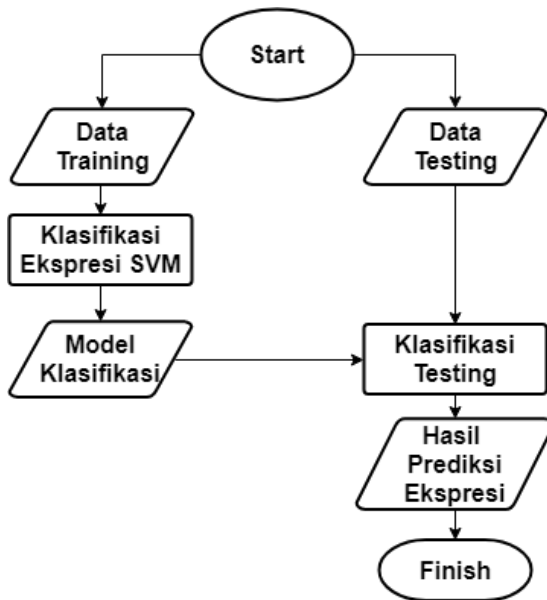
Gambar 3.3 *Flowchart* tahap deteksi area wajah



Gambar 3.4 *Flowchart* tahap *pre-processing*



Gambar 3.5 Flowchart tahap ekstraksi fitur



Gambar 3.6 *Flowchart* tahap klasifikasi

BAB IV IMPLEMENTASI SISTEM

4.1. Implementasi Deteksi Area Wajah

Pada tahap ini, langkah pertama yang akan dilakukan adalah mendeteksi area wajah dari citra original dengan algoritma Viola-Jones. Langkah ini dilakukan untuk menghilangkan *noise* berupa latar belakang yang terdapat pada gambar. Untuk mengimplementasikan Viola-Jones, digunakan fungsi yang terdapat pada Matlab seperti pada Kode Sumber 4.1. Pada baris ke-2, parameter 'FrontalFaceCART' digunakan untuk mendeteksi area wajah dengan tampak depan. Untuk menghindari *false detection*, jika terdapat beberapa deteksi, hasil yang akan digunakan adalah deteksi dengan area terbesar (Kode Sumber 4.1 baris ke-4 s.d 12). *Output* dari Viola-Jones berupa matriks berisi koordinat x, koordinat y, lebar, dan tinggi citra.

```
1 faceDetect = vision.CascadeObjectDetector
2 ('FrontalFaceCART');
3 BBox = step(faceDetect,Img);
4 if size(BBox,1)>1
5     for k=1:size(BBox,1)
6         areaFace(k)= BBox(k,3)*BBox(k,4);
7     end
8     [val,idx] = max(areaFace);
9     faceBBox= BBox(k,:);
10 else
11     faceBBox= BBox;
12 end
```

Kode Sumber 4.1 Implementasi Viola-Jones

4.2. Implementasi *Pre-processing*

Setelah citra wajah didapatkan, dilanjutkan dengan *histogram equalization* pada citra wajah yang telah dilakukan sebelumnya. Proses ini diimplementasikan menggunakan fungsi Matlab seperti pada Kode Sumber 4.2. *Input* pada tahap ini adalah citra hasil deteksi area wajah dengan Viola-Jones. *Histogram equalization* menghasilkan *output* yang disebut citra hasil *pre-processing*. Citra

hasil *pre-processing* tersebut kemudian di-*resize* menjadi 120x120 untuk mempercepat komputasi pada tahap selanjutnya.

```
1 faceImg = histeq(faces);
```

Kode Sumber 4.2 Implementasi *histogram equalization*

4.3. Implementasi Ekstraksi Fitur Landmark

Fitur Landmark sudah didapatkan pada penelitian sebelumnya dengan metode *Active Appearance Model* (AAM) [8]. Fitur tersebut berupa matriks dengan dimensi 136x327.

Untuk menyamakan titik Landmark dengan citra hasil *pre-processing* pada penelitian ini, dilakukan beberapa penyesuaian dengan tahapan pada Kode Sumber 4.3. Pada tahap ini, masing-masing titik x dan y Landmark akan dikurangi dengan titik x dan y dari citra hasil deteksi area wajah pada tahap *pre-processing*.

```
1 landmarkX = ptsLand(:,1)-faceBBox(1,1);  
2 landmarkY = ptsLand(:,2)-faceBBox(1,2);  
3 landmarkFeature = [landmarkX(:);landmarkY(:)];
```

Kode Sumber 4.3 Implementasi ekstraksi fitur *landmark*

4.4. Implementasi Ekstraksi Fitur Gabor

Untuk mengekstraksi fitur Gabor, diperlukan sebuah *Gabor array* yang dibuat dengan perhitungan *Gabor filter bank* yang diimplementasikan pada Kode Sumber 4.4. Untuk melakukan perhitungan tersebut, diperlukan empat parameter, yaitu *u* berupa frekuensi yang bernilai 5, *v* berupa jumlah orientasi yang bernilai 8, *m* dan *n* berupa jumlah matriks Gabor 2-dimensi yang nilainya harus integer ganjil yaitu 39 secara *default* untuk masing-masing variabel. Berdasarkan nilai tersebut, didapatkan *Gabor array* sebesar 5x8, dengan masing-masing elemen berukuran 39x39.

Gabor array tersebut kemudian digunakan sebagai *filter* untuk mengekstraksi fitur Gabor pada Kode Sumber 4.5 pada baris ke-5 hingga ke-6. *Input* dari proses ini adalah citra hasil *pre-processing* yang telah di-*resize*.

Langkah selanjutnya yaitu *downsampling*. *Downsampling* diperlukan untuk mengecilkan ukuran matriks dengan mengambil poin-poin penting dari matriks tersebut. Keuntungan dari

downsampling adalah komputasi yang lebih cepat. Nilai *downsampling* yang diperlukan harus merupakan faktor bilangan bulat dari baris atau kolom. Pada Kode Sumber 4.6 baris ke-4 dan ke-5 diberikan nilai *d1* dan *d2* masing-masing 4.

```

1 fmax=0.25;
2 gama=sqrt(2);
3 eta=sqrt(2);
4 for i=1:u
5   fu=fmax/((sqrt(2))^(i-1));
6   alpha=fu/gama;
7   beta=fu/eta;
8   for j=1:v
9     tetav=((j-1)/v)*pi;
10    gFilter=zeros(m,n);
11    for x=1:m
12      for y=1:n
13        xprime = (x-((m+1)/2))*cos(tetav)+
14        (y-((n+1)/2))*sin(tetav);
15        yprime = -(x-((m+1)/2))*sin(tetav)+
16        (y-((n+1)/2))*cos(tetav);
17        gFilter(x,y) = (fu^2/(pi*gama*eta))*
18        exp(-((alpha^2)*(xprime^2)+(beta^2)*
19        (yprime^2))) * exp(1i*2*pi*fu*xprime);
20      end
21    end
22    gaborArray{i,j} = gFilter;
23  end
24 end

```

Kode Sumber 4.4 Gabor *filter bank*

```

1 [u,v]=size(gaborArray);
2 gaborResult=cell(u,v);
3 for i=1:u
4   for j=1:v
5     gaborResult{i,j}=imfilter(img,
6       gaborArray{i,j});
7   end
8 end

```

Kode Sumber 4.5 Implementasi Gabor dengan *Gabor array*

```

1 for i=1:u
2   for j=1:v
3     gaborAbs=abs(gaborResult{i,j});
4     gaborAbs=downsample(gaborAbs,d1);
5     gaborAbs=downsample(gaborAbs.',d2);
6     gaborAbs=gaborAbs(:);
7     featureVector=[featureVector;gaborAbs];
8   end
9 end

```

Kode Sumber 4.6 *Downsampling* fitur Gabor

4.5. Implementasi Ekstraksi Fitur Haar

Fungsi dari haarFeatures adalah mencari hasil dari matriks *approximation coefficient* dari operasi dekomposisi level 2 Haar-Wavelet untuk dijadikan fitur Haar. Pada Kode Sumber 4.7, *input* fungsi ini berupa citra hasil *pre-processing* yang telah di-*resize*, level dekomposisi Haar yang bernilai 2, dan metode transformasi yaitu 'Haar'. *Output* dihasilkan berupa matriks 'C' dan 'S', masing-masing adalah vektor dekomposisi dan *bookkeeping matrix* dari vektor dekomposisi 'C'.

```

1 [C,S] = wavedec2(img,level,'haar');

```

Kode Sumber 4.7 Implementasi ekstraksi fitur Haar

4.6. Implementasi Reduksi dan Kombinasi Fitur

Proses ini bertujuan menggabungkan fitur-fitur yang telah didapatkan dari operasi sebelumnya untuk dijadikan satu matriks. Dikarenakan ukuran fitur yang cukup besar, dilakukan pengurangan dimensi matriks dengan Principal Component Analysis (PCA). Reduksi dan kombinasi fitur dilakukan pada masing-masing data *training* dan *testing* pada fitur Gabor dan Haar, sehingga pembagian data dilakukan secara acak dengan proporsi 90% data *training* dan 10% data *testing*. Reduksi fitur tidak dilakukan pada fitur Landmark dikarenakan ukuran fitur yang relatif kecil.

Sebelum dilakukan reduksi fitur, dilakukan normalisasi terhadap data *training* pada Kode Sumber 4.8. Dari normalisasi data *training*, didapatkan nilai normalisasi data *training*, rata-rata dari data *training*, dan deviasi standar dari data *training*. Untuk melakukan normalisasi pada data *testing*, digunakan nilai rata-rata dan deviasi standar pada data *training* (lihat Kode Sumber 4.8 baris ke-4).

Selanjutnya, dicari koefisien PCA pada fitur yang telah dinormalisasi (lihat Kode Sumber 4.9 baris ke-4). *Output* dari tahap ini adalah matriks koefisien C , matriks *score* dari PCA S , serta nilai *latent* pada L . Sebelum dilakukan proyeksi data, dicari nilai K dengan cara mencari *eigenvalue* yang bernilai kurang dari 1. *Eigenvalue* memiliki nilai yang sama dengan matriks *latent*, tetapi pada matriks *latent* nilai *eigenvalue* sudah diurutkan dari nilai terbesar hingga terkecil.

Koefisien dan nilai K tersebut digunakan untuk memproyeksikan data pada *space* fitur baru (lihat Kode Sumber 4.9 baris ke-7). Tahap ini menghasilkan matriks fitur baru yang telah dinormalisasi dan direduksi. Untuk melakukan PCA pada data *testing*, dilakukan proyeksi pada normalisasi data *testing* terhadap nilai koefisien data *training* (lihat Kode Sumber 4.8 baris ke-8). Langkah yang sama juga diaplikasikan pada fitur Haar.

Selanjutnya dilakukan penggabungan fitur dalam satu vektor. Dari proses tersebut dihasilkan tujuh kombinasi fitur vektor, dengan mempertahankan 99% varians untuk fitur Gabor dan fitur Haar.

```

1 [normTrain, meanTrain, stdTrain] = zscore(Train);
2 [x,y] = size(Test);
3 for ii = 1:x
4   normTest(ii,:)=(Test(ii,:)-meanTrain)./stdTrain;
5 end

```

Kode Sumber 4.8 Normalisasi data *training* dan *testing*

```

1 meanNorm = mean(normTrain);
2 normTrain = bsxfun(@minus, normTrain, meanNorm);
3 normTest = bsxfun(@minus, normTest, meanNorm);
4 [CTrain, STrain, LTrain] = pca(normTrain);
5 covTrain = cumsum(LTrain) ./ sum(LTrain);
6 K = max(find(covTrain < 1));
7 FeatTrain = normTrain * CTrain(:,1:K);
8 FeatTest = normTest * CTrain(:,1:K);

```

Kode Sumber 4.9 Implementasi PCA data *training* dan *testing*

4.7. Implementasi Klasifikasi

Sebelum dilakukan klasifikasi, dibuat model klasifikasi SVM dengan pendekatan *multiclass*. *Multiclass* digunakan karena kelas yang akan di-*training* terdiri dari tujuh kelas, masing-masing diwakilkan dengan angka 1-7 (lihat Tabel 4.1). Dengan menggunakan ECOC seperti pada Kode Sumber 4.10 baris ke-1. Pada model tersebut, parameter *Coding* yang digunakan adalah *onevsall*, yang berarti pada setiap *binary learner*, satu kelas adalah positif dan sisanya merupakan kelas negatif serta menggunakan fungsi kernel *linear*. Dari proses tersebut, dihasilkan sebuah model klasifikasi ECOC bernama *cval_mdl*.

Dari model klasifikasi tersebut dilakukan validasi menggunakan *k-fold validation* (Kode Sumber 4.10 baris ke-4). Secara *default*, *cross validation* yang digunakan adalah *10-cross validation*. Proses tersebut menghasilkan model klasifikasi ECOC yang telah di-*training* dengan *cross validated-folds* bernama *cval*. Selanjutnya hasil prediksi validasi tersebut disimpan dalam *confusion matrix* untuk dihitung akurasi. *Error rate* model klasifikasi *cval* juga didapatkan pada Kode Sumber 4.10 baris ke-8.

Tabel 4.1 Ekspresi dan notasi angka

Angka	Ekspresi	Angka	Ekspresi
1	Marah	5	Senang
2	Merendahkan	6	Sedih
3	Jijik	7	Terkejut
4	Takut		

```

1 SVM = templateSVM('KernelFunction','linear');
2 cval_md1 = fitcecoc(cval_feat,...
3     cval_class,'Coding','onevsall',...
4     'Learners',SVM);
5 cval = crossval(cval_md1,'KFold',10);
6 cval_prdct = kfoldPredict(cval);
7 cval_conf = confusionmat(cval.Y,cval_prdct);
8 cval_error = kfoldLoss(cval);

```

Kode Sumber 4.10 Implementasi Klasifikasi

[Halaman ini sengaja dikosongkan]

BAB V

HASIL UJI COBA DAN PEMBAHASAN

5.1. Lingkungan Kerja

Lingkungan kerja pada penelitian ini meliputi lingkungan kerja perangkat lunak atau *software* dan lingkungan kerja perangkat keras atau *hardware*.

5.1.1. Perangkat Lunak (*Software*)

Kebutuhan perangkat lunak yang digunakan dalam penelitian ini antara lain:

a. Sistem Operasi

Sistem operasi yang digunakan dalam penelitian ini adalah Microsoft Windows 10 Home 64-bit.

b. Matlab

Untuk implementasi penelitian dari tahap *preprocessing* hingga perhitungan akurasi digunakan Matlab 2016a.

5.1.2. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan untuk mengimplementasi penelitian ini adalah laptop merek ASUS tipe N46V, prosesor Intel® Core™ i7-3610QM CPU @ 2.30GHz.

5.2. Skenario Pengujian

Skenario pengujian yang dilakukan adalah dengan menggunakan kombinasi fitur yang berbeda untuk melihat fitur manakah yang memiliki akurasi paling baik. Kombinasi fitur yang akan digunakan adalah:

1. Fitur Landmark
2. Fitur Gabor
3. Fitur Haar
4. Fitur Gabor dan Landmark
5. Fitur Haar dan Landmark
6. Fitur Gabor dan Haar
7. Fitur Gabor, Haar, dan Landmark

5.3. Hasil Pengujian

Pada subbab ini akan dibahas hal-hal yang berkaitan dengan hasil yang didapatkan dari pengujian-pengujian yang dilakukan, serta analisa dari hasil pengujian.

5.3.1. Hasil Deteksi Area Wajah dan Pre-Processing

Pada proses ini, citra original (lihat Gambar 5.1) akan dideteksi area wajahnya menggunakan algoritma Viola-Jones. Deteksi area wajah dengan algoritma Viola-Jones dapat mendeteksi seluruh wajah pada yang terdapat pada *database* secara tepat dengan ukuran 360x360. Selanjutnya, dilakukan operasi *histogram equalization*. Seperti terlihat pada Tabel 5.1, kontur wajah terlihat lebih jelas dibandingkan dengan sebelum dilakukan *histogram equalization*. Nilai histogram juga dapat dilihat pada Tabel 5.1 kolom ke-3. Untuk hasil *pre-processing* citra lainnya dapat dilihat pada Lampiran 1.



Subyek 52-001



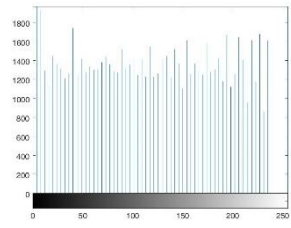
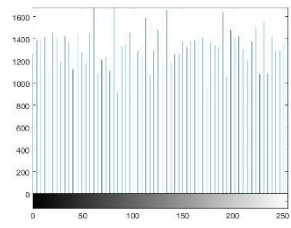
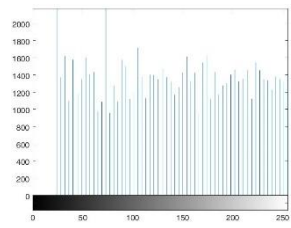
Subyek 55-006



Subyek 130-009

Gambar 5.1 Citra Original [7] [8]

Tabel 5.1 Hasil deteksi area wajah dan *pre-processing*

	Hasil Deteksi Wajah	Hasil Histeq	Histogram
Subyek 74-001			
Subyek 106-002			
Subyek 111-001			

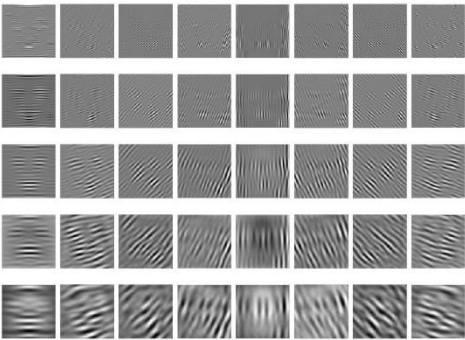
5.3.2. Hasil Ekstraksi Fitur

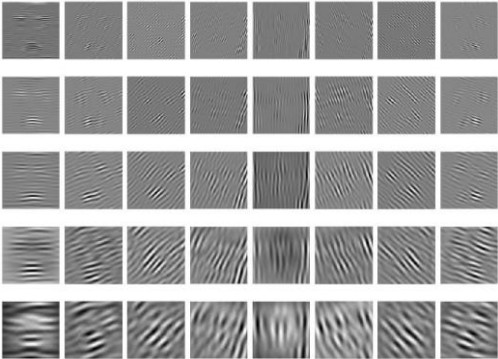
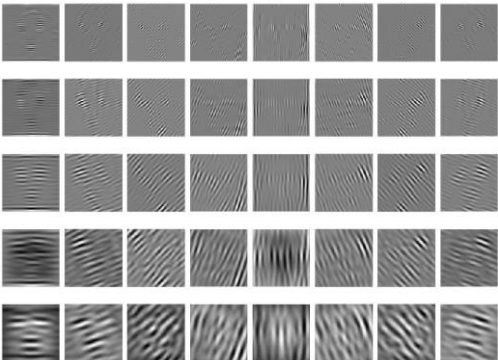
Pada Tabel 5.2 merupakan aplikasi filter Gabor dari citra wajah yang telah dilakukan histogram equalization. Untuk setiap citra, didapatkan fitur Gabor sebanyak 40 fitur dari aplikasi filter Gabor dengan frekuensi 1 sampai dengan 5 serta 8 orientasi berbeda. Visualisasi dari aplikasi filter Gabor terhadap citra dapat dilihat pada Tabel 5.2 kolom ke-3. Pada setiap baris adalah aplikasi filter Gabor dengan orientasi dari 1 sampai dengan 8 dan setiap kolom merupakan aplikasi filter Gabor dengan frekuensi berbeda dari 1

sampai dengan 5. Proses tersebut menghasilkan matriks fitur Gabor sebesar 36.000×327 . Untuk hasil ekstraksi fitur Gabor pada citra lainnya dapat dilihat pada Lampiran 2. Sedangkan pada Tabel 5.3 dapat dilihat hasil transformasi Haar pada citra hasil *pre-processing*. Dari hasil transformasi tersebut, fitur yang akan digunakan adalah *approximation coefficient* dari transformasi Haar dengan level 2. Matriks fitur Haar yang dihasilkan adalah sebesar 14.000×327 . Untuk hasil ekstraksi fitur Haar pada citra lainnya dapat dilihat pada 0.

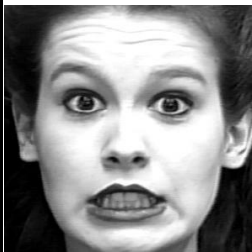
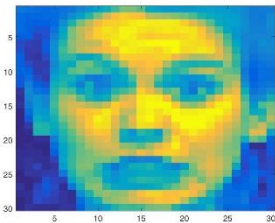
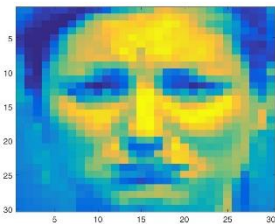
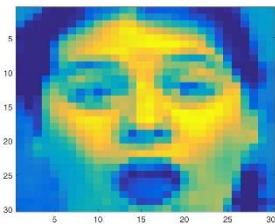
Pada proses reduksi dan kombinasi fitur, didapatkan matriks fitur Gabor yang semula berukuran 327×36.000 , direduksi menjadi 327×292 . Sedangkan untuk fitur Haar, fitur matriks awal sebesar 327×14.000 direduksi menjadi 327×292 . Dimensi fitur-fitur lainnya dapat dilihat pada Tabel 5.4.

Tabel 5.2 Hasil ekstraksi fitur Gabor

	Hasil Histeq	Fitur Gabor
Subyek 74-001		

	Hasil Histeq	Fitur Gabor
Subyek 106-002		
Subyek 111-001		

Tabel 5.3 Hasil Ekstraksi Fitur Haar

	Hasil Histeq	Hasil Ekstraksi Haar
Subyek 74-001		
Subyek 106-002		
Subyek 111-001		

Tabel 5.4 Dimensi matriks fitur

Data Training	Dimensi
Landmark	294x136
Gabor	294x292
Haar	294x292
Gabor dan Landmark	294x428
Haar dan Landmark	294x428
Gabor dan Haar	294x584
Gabor, Haar, dan Landmark	294x720

5.3.3. Hasil Klasifikasi

Evaluasi sistem dilakukan dengan 10-cross validation. Output dari hasil tersebut disimpan di dalam sebuah confusion matrix.

1. Hasil Klasifikasi Fitur Landmark

Pada Tabel 5.5, hasil klasifikasi dengan fitur Landmark menghasilkan akurasi sebesar 91%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi 100%. Sedangkan akurasi terburuk ada pada ekspresi merendahkan dengan akurasi sebesar 66%, dengan misklasifikasi terbanyak terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0,0826.

2. Hasil Klasifikasi Fitur Gabor

Pada Tabel 5.6, hasil klasifikasi dengan fitur Gabor menghasilkan akurasi sebesar 93,5%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi 100%. Sedangkan akurasi terburuk ada pada ekspresi sedih dengan akurasi sebesar 72%, dengan misklasifikasi terbanyak terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0,0642.

3. Hasil Klasifikasi Fitur Haar

Pada Tabel 5.7, hasil klasifikasi dengan fitur Haar menghasilkan akurasi sebesar 70%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi sebesar 84,3%. Sedangkan akurasi terburuk ada pada ekspresi merendahkan dengan akurasi sebesar 5%, dengan misklasifikasi terbanyak terhadap ekspresi takut dan terkejut. Pada fitur ini, error rate yang dihasilkan sebesar 0,2997.

4. Hasil Klasifikasi Fitur Gabor dan Landmark

Pada Tabel 5.8, hasil klasifikasi dengan fitur Gabor dan Landmark menghasilkan akurasi sebesar 94,8%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi sebesar 100%. Sedangkan akurasi terburuk ada pada ekspresi merendahkan dengan akurasi sebesar 77%, dengan misklasifikasi terbanyak terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0.052.

5. Hasil Klasifikasi Fitur Haar dan Landmark

Pada Tabel 5.9, hasil klasifikasi dengan fitur Haar dan Landmark menghasilkan akurasi sebesar 89%, dengan kelas terbaik

yaitu ekspresi senang dengan akurasi sebesar 100%. Sedangkan akurasi terburuk ada pada ekspresi merendahkan dengan akurasi sebesar 50%. Pada ekspresi merendahkan, misklasifikasi terbanyak terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0,1009.

6. Hasil Klasifikasi Fitur Gabor dan Haar

Pada Tabel Tabel 5.10, hasil klasifikasi dengan fitur Gabor dan Haar menghasilkan akurasi sebesar 92%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi sebesar 100%. Sedangkan akurasi terburuk ada pada ekspresi sedih dengan akurasi sebesar 67%, dengan misklasifikasi terbanyak terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0,0795.

7. Hasil Klasifikasi Fitur Gabor, Haar, dan Landmark

Pada Tabel Tabel 5.11, hasil klasifikasi dengan fitur Gabor, Haar, dan Landmark menghasilkan akurasi sebesar 93,5%, dengan kelas terbaik yaitu ekspresi senang dengan akurasi sebesar 100%. Sedangkan akurasi terburuk ada pada ekspresi sedih dengan akurasi sebesar 67%, dengan misklasifikasi terjadi terhadap ekspresi marah. Pada fitur ini, error rate yang dihasilkan sebesar 0.0642.

Tabel 5.5 *Confusion matrix* fitur Landmark

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	41	1	2	0	0	0	1
	2	3	12	0	0	0	2	1
	3	3	2	54	0	0	0	0
	4	0	1	1	19	0	3	1
	5	0	0	0	0	69	0	0
	6	2	1	1	0	0	23	1
	7	0	1	0	0	0	0	82

Tabel 5.6 *Confusion matrix* fitur Gabor

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	42	1	0	0	0	2	0
	2	3	13	0	0	0	1	1
	3	0	0	58	0	1	0	0
	4	0	0	0	23	0	1	1
	5	0	0	0	0	69	0	0
	6	4	0	0	2	0	21	1
	7	1	1	0	1	0	0	80

Tabel 5.7 *Confusion matrix* fitur Haar

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	30	0	5	1	3	6	0
	2	3	1	3	4	2	1	4
	3	6	0	49	1	1	2	0
	4	1	1	1	16	1	1	4
	5	1	1	4	7	56	0	0
	6	10	0	3	2	1	7	5
	7	2	0	1	6	0	4	70

Tabel 5.8 *Confusion matrix* fitur Gabor dan Landmark

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	41	1	0	0	0	3	0
	2	3	14	1	0	0	0	0
	3	0	0	58	0	1	0	0
	4	0	0	0	22	0	1	2
	5	0	0	0	0	69	0	0
	6	2	0	0	0	0	25	1
	7	0	1	0	1	0	0	81

Tabel 5.9 *Confusion matrix* fitur Haar dan Landmark

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	41	0	4	0	0	0	0
	2	5	9	0	2	1	1	0
	3	3	0	55	0	0	1	0
	4	0	0	0	21	1	1	2
	5	0	0	0	1	68	0	0
	6	6	0	1	2	0	19	0
	7	0	1	0	1	0	0	81

Tabel 5.10 *Confusion matrix* fitur Gabor dan Haar

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	39	0	1	1	0	4	0
	2	3	13	0	0	0	1	1
	3	0	0	58	1	0	0	0
	4	1	0	2	21	0	0	1
	5	0	0	0	0	69	0	0
	6	5	0	0	3	0	19	1
	7	0	1	0	0	0	0	82

Tabel 5.11 *Confusion matrix* fitur Landmark, Gabor, dan Haar

		Kelas Prediksi						
		1	2	3	4	5	6	7
Kelas Aktual	1	42	0	0	0	0	3	0
	2	3	14	0	0	0	1	0
	3	1	0	57	0	0	1	0
	4	0	0	0	23	0	0	2
	5	0	0	0	0	69	0	0
	6	6	0	0	2	0	19	1
	7	0	1	0	0	0	0	82

5.4. Pembahasan/Diskusi

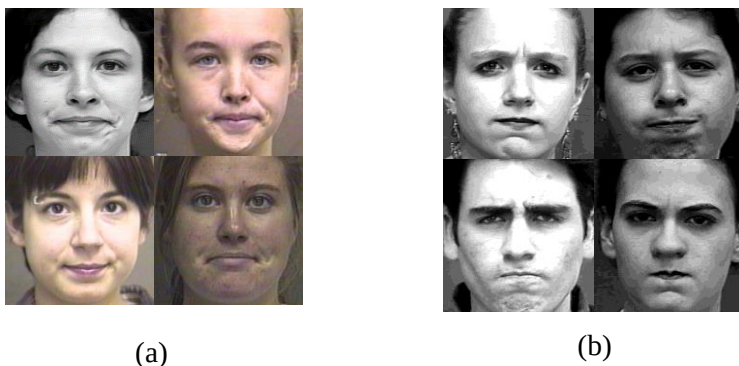
Secara umum, ekspresi yang paling banyak diprediksi benar adalah ekspresi senang dengan tingkat akurasi pada setiap mayoritas fitur sebanyak 100%. Sedangkan untuk ekspresi yang banyak diprediksi salah adalah ekspresi merendahkan yang paling banyak diprediksi sebagai ekspresi marah.

Dilihat dari jumlah data, ekspresi senang memiliki data yang cukup banyak, 69 citra atau 21% dari keseluruhan data. Sedangkan

untuk ekspresi merendahkan memiliki jumlah data yang relatif sedikit dibandingkan dengan ekspresi lainnya. Data ekspresi merendahkan hanya 5,8% dari total keseluruhan data yaitu 18 citra.

Dilihat dari *Facial Action Coding System* pada Tabel 2.1 dan Tabel 2.2, ekspresi merendahkan dideskripsikan memiliki AU14. Sedangkan untuk ekspresi marah, memiliki AU23 dan AU 24. Jika dilihat dari Gambar 5.2, kedua citra memiliki persamaan pada bagian bibir yang terkatup dan bagian alis dalam yang naik tetapi keduanya memiliki intensitas yang berbeda.

Untuk fitur yang memiliki akurasi terbaik adalah fitur Gabor dan Landmark dengan akurasi sebesar 94,8%. Sedangkan untuk fitur yang memiliki akurasi terendah adalah fitur Haar dengan akurasi sebesar 70%.



Gambar 5.2 Citra ekspresi (a) merendahkan (b) marah [8]

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Setelah melakukan perancangan, pengamatan dan pengujian, kesimpulan yang didapatkan pada penelitian ini adalah:

1. Sistem dapat mendeteksi area wajah dengan sempurna pada seluruh citra.
2. Akurasi fitur Gabor dan Haar Wavelet sebesar 92%.
3. Akurasi tertinggi didapatkan dengan menggunakan fitur Landmark dan Gabor sebesar 94,8% sedangkan akurasi terendah didapatkan dengan menggunakan fitur Haar yaitu 84,3%.
4. Kelas ekspresi yang paling banyak jumlah prediksi benarnya adalah ekspresi senang, sedangkan yang paling banyak terjadi misklasifikasi adalah kelas merendahkan dengan prediksi marah.

6.2. Saran

Beberapa saran yang dapat digunakan untuk pengembangan pada penelitian selanjutnya antara lain:

1. Memperbanyak jumlah dan variasi ekspresi data *training* yang digunakan untuk menjangkau keberagaman model tiap-tiap ekspresi.
2. Mencoba alternatif metode ekstraksi fitur lain yang lebih optimal dalam memproses citra digital, khususnya citra wajah.

[Halaman ini sengaja dikosongkan]

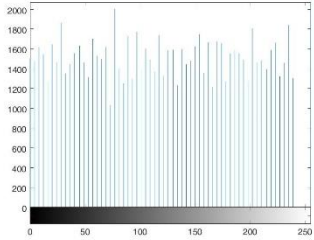
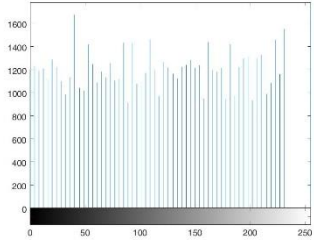
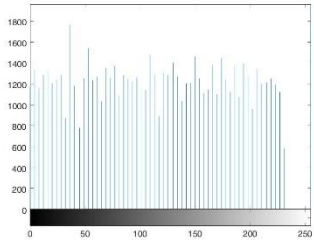
DAFTAR PUSTAKA

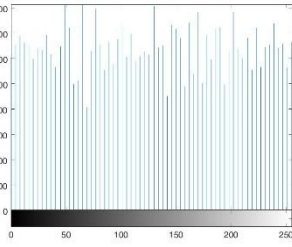
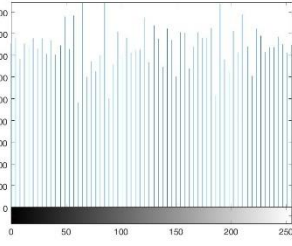
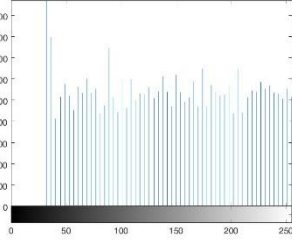
- [1] A. Mehrabian, "Communication without words," *Psychology Today*, vol. 2, pp. 52-55, 1968.
- [2] V. Lekshmi and M. Sasikumar, "Analysis of facial expression using Gabor and SVM," *International Journal of Recent Trends in Engineering*, vol. 1, no. 2, pp. 47-50, 2009.
- [3] M. Haghighat, S. Zonouz and M. Abdel-Mottaleb, "CloudID: Trustworthy cloud-based and cross-enterprise biometric identification," *Expert Systems with Applications*, vol. 42, no. 21, pp. 7905-7916, 2015.
- [4] M. Alwakeel and Z. Shaaban, "Face recognition based on Haar wavelet transform and principal component analysis via Levenberg-Marquardt backpropagation neural network," *European Journal of Scientific Research*, vol. 42, no. 1, pp. 25-31, 2010.
- [5] M. Goyani and N. Patel, "Multi-Level Haar Wavelet based Facial Expression Recognition using Logistic Regression," *Indian Journal of Science and Technology*, vol. 10, no. 9, 2017.
- [6] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [7] T. Kanade, J. F. Cohn and Y. Tian, "Comprehensive database for facial expression analysis," in *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00)*, Grenoble, France, 2000.
- [8] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *IEEE Computer Society Conference*, San Francisco, CA, USA, 2010.
- [9] P. Ekman and E. (. Rosenberg, What the face reveals: Basic and

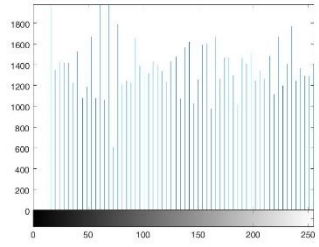
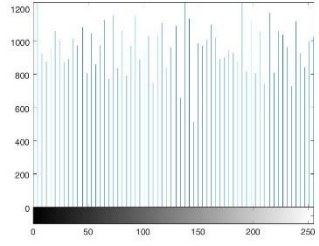
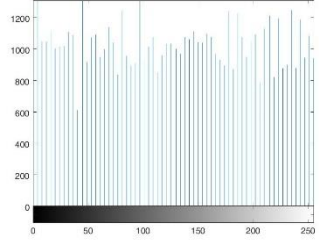
- applied studies of spontaneous expression using the Facial Action Coding System (FACS), USA: Oxford University Press, 1997.
- [10] Muscle Anatomy, "Muscles Of The Face And Their Functions Facial Expression Recognition (Face Recognition Techniques) Part 1," [Online]. Available: <https://anatomybodysystem.com/muscles-of-the-face-and-their-functions/muscles-of-the-face-and-their-functions-smile-line-muscles-of-the-facial-expression-smile-line/>. [Accessed 30 November 2017].
 - [11] How It Works, "Histogram Equalization," [Online]. Available: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/histeq.htm>. [Accessed 23 August 2017].
 - [12] M. B. Stegmann, R. Fisker, B. K. Ersbøll, H. H. Thodberg and L. Hyldstrup, "Active Appearance Models: Theory and Cases," 20 September 2000. [Online]. Available: <http://www2.imm.dtu.dk/~aam/main/node34.html>. [Accessed 24 November 2017].
 - [13] S. Halim, "Principal Component Analysis," 2012. [Online]. Available: faculty.petra.ac.id/halim/index_files/Stat2/PCA.pdf. [Accessed 23 August 2017].
 - [14] S. Raschka, "Principal Component Analysis in 3 Simple Steps," 2015. [Online]. Available: http://sebastianraschka.com/Articles/2015_pca_in_3_steps.html. [Accessed 2017 August 2017].
 - [15] D. Roth, M. Yang and N. Ahuja, "Learning to recognize three dimensional objects," *Neural Computation*, vol. 14, pp. 1071-1103, 2002.
 - [16] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," in *International Joint Conference on Artificial Intelligence (IJCAI)*, Montreal, 1995.

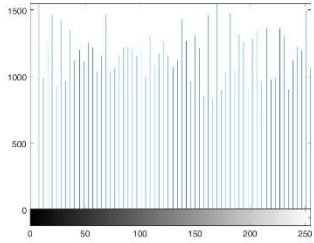
LAMPIRAN

Lampiran 1. Contoh Hasil Tahap Pre-processing

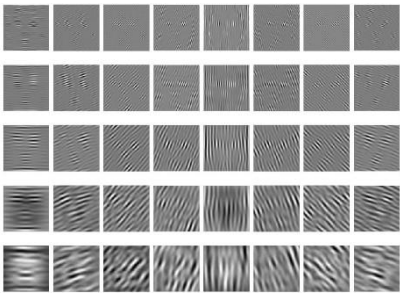
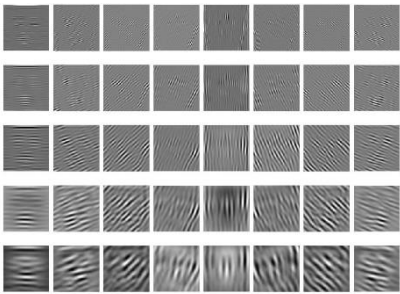
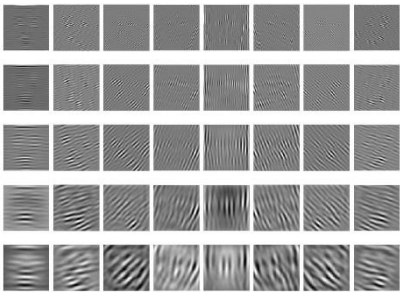
	Hasil Deteksi Area Wajah	Hasil Histeq	Histogram
Subyek 52-001			
Subyek 55-006			
S055-006			

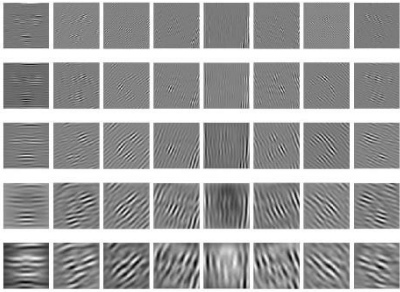
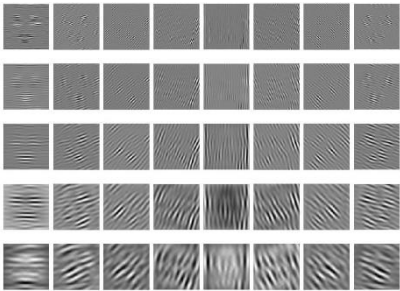
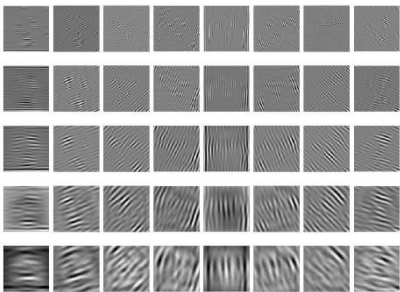
	Hasil Deteksi Area Wajah	Hasil Histeq	Histogram
S106-004			
S106-006			
S124-003			

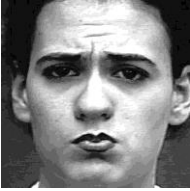
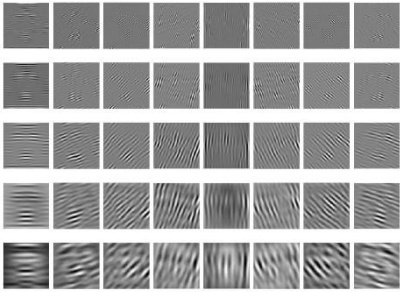
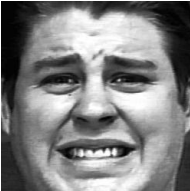
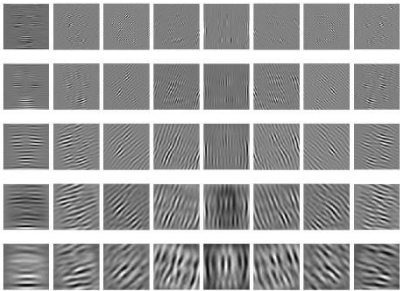
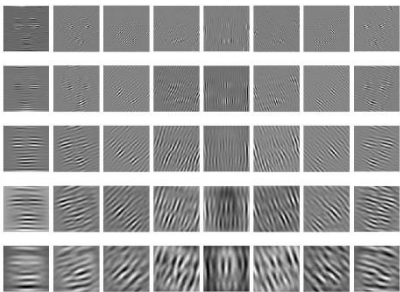
	Hasil Deteksi Area Wajah	Hasil Histeq	Histogram
Subyek 130-009			
S132-003			
S132-006			

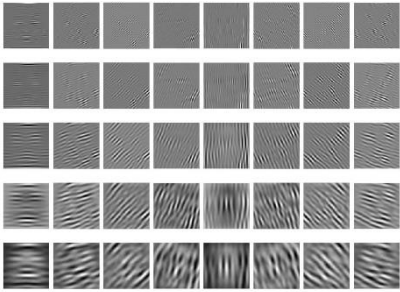
	Hasil Deteksi Area Wajah	Hasil Histeq	Histogram
S138-008			

Lampiran 2. Contoh Hasil Ekstraksi fitur Gabor

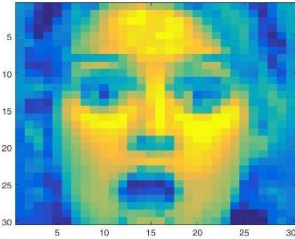
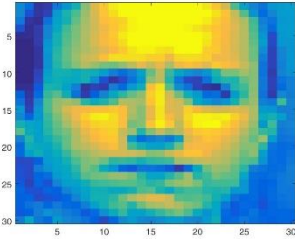
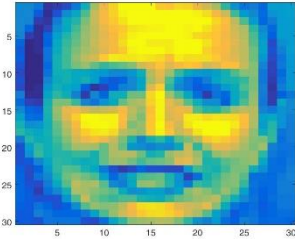
	Hasil Pre-Processing	Fitur Gabor
Subyek 52-001		
Subyek 55-006		
S055-006		

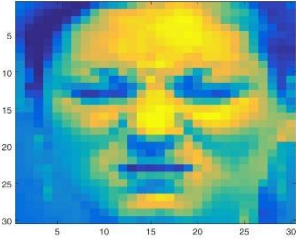
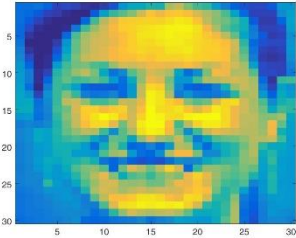
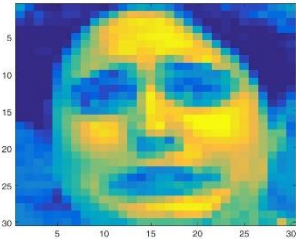
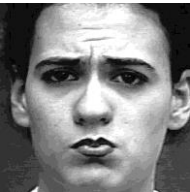
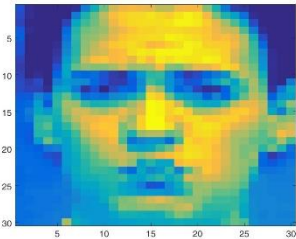
	Hasil Pre-Processing	Fitur Gabor
S106-004		
S106-006		
S124-003		

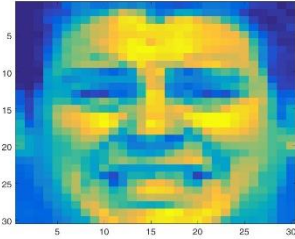
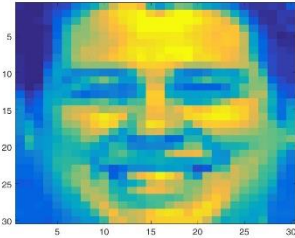
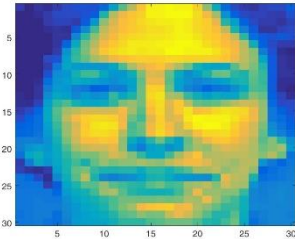
	Hasil Pre-Processing	Fitur Gabor
Subyek 130-009		
S132-003		
S132-006		

	Hasil Pre-Processing	Fitur Gabor
S138-008		

Lampiran 3. Contoh Hasil Ekstraksi Fitur Haar

	Hasil Pre-Processing	Fitur Haar
Subyek 52-001		
Subyek 55-006		
S055-006		

	Hasil Pre-Processing	Fitur Haar
S106-004		
S106-006		
S124-003		
Subyek 130-009		

	Hasil Pre-Processing	Fitur Haar
S132-003		
S132-006		
S138-008		

[Halaman ini sengaja dikosongkan]

Lampiran 4. Contoh Hasil Prediksi Ekspresi

CITRA	EX	LAND	GABOR	HAAR	GL	HL	GH	LGH
'S052\001'	7	7	7	7	7	7	7	7
'S055\004'	1	1	1	1	1	1	1	1
'S055\006'	4	4	4	4	4	4	4	4
'S106\004'	3	3	3	3	3	3	3	3
'S106\006'	5	5	5	5	5	5	5	5
'S124\003'	4	4	6	7	6	4	6	6
'S130\009'	6	6	6	3	6	6	6	6
'S132\003'	4	4	4	5	4	5	4	4
'S132\006'	5	5	5	5	5	5	5	5
'S138\008'	2	2	2	5	2	2	2	2

EX = Ekspresi asli/*ground truth*

LAND = Fitur Landmark

GABOR = Fitur Gabor

HAAR = Fitur Haar

GL = Fitur Gabor + Landmark

HL = Fitur Haar + Landmark

GH = Fitur Gabor + Haar

LGH = Fitur Gabor + Haar + Landmark

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Penulis dilahirkan di Temanggung, 23 Oktober 1994. Penulis telah menempuh pendidikan formal yaitu di TK Islam Al-Jihad Bekasi, SD Islam Al-Fajar Bekasi selama lima tahun dan SDN Kertajaya Surabaya selama setahun, SMPN 6 Surabaya, dan SMAN 2 Surabaya.

Setelah lulus dari SMAN tahun 2012, penulis melanjutkan pendidikan S1 di Departemen Informatika ITS Surabaya terdaftar dengan NRP. 5112100140.

Di Departemen Informatika, Penulis mengambil Bidang Studi Komputasi Cerdas dan Visi.