



TUGAS AKHIR - SM141501

KLASIFIKASI CITRA KANKER KULIT MELANOMA MENGUNAKAN METODE *SUPPORT VECTOR MACHINE* (SVM)

**KUSUMA WAHANINGGAR
NRP 1211 100 124**

**Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si,MT**

**JURUSAN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2016**



FINAL PROJECT - SM141501

IMAGE CLASSIFICATION OF SKIN CANCER MELANOMA USING SUPPORT VECTOR MACHINE (SVM) METHOD

KUSUMA WAHANINGGAR
NRP 1211 100 124

Supervisor
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

DEPARTMENT OF MATHEMATICS
Faculty of Mathematics and Natural Science
Sepuluh Nopember Institute of Technology
Surabaya 2016

LEMBAR PENGESAHAN

KLASIFIKASI CITRA KANKER KULIT MELANOMA MENGUNAKAN METODE *SUPPORT VECTOR MACHINE* (SVM)

IMAGE CLASSIFICATION OF SKIN CANCER MELANOMA USING SUPPORT VECTOR MACHINE (SVM) METHOD

TUGAS AKHIR

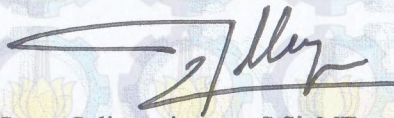
Diajukan Untuk Memenuhi Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Sains
Pada Bidang Studi Ilmu Komputer
Program Studi S-1 Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :

KUSUMA WAHANINGGAR
NRP. 1211 100 124

Menyetujui,

Dosen Pembimbing,



Dr. Dwi Ratna Sulistyaningrum, S.Si, MT
NIP. 19690405 199403 2 003

Mengetahui,

Ketua Jurusan Matematika
FMIPA ITS



Dr. Imam Mukhlas, S.Si, MT
NIP. 19700831 199403 1 003

Surabaya, Januari 2016

KATA PENGANTAR

Alhamdulillahirobbil'alamin penulis dapat menyelesaikan Tugas Akhir yang berjudul: **“Klasifikasi Citra Kanker Kulit Melanoma Menggunakan Metode *Support Vector Machine* (SVM)”** yang merupakan salah satu persyaratan dalam menyelesaikan Program Studi S-1 pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dapat terselesaikan dengan baik dan lancar berkat kerja sama, bantuan, dan dukungan dari banyak pihak. Sehubungan dengan hal itu, penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Imam Mukhlash, S.Si, MT selaku Ketua Jurusan Matematika ITS.
2. Ibu Dr. Dwi Ratna Sulistyaningrum, S.Si, MT selaku dosen pembimbing yang senantiasa meluangkan waktunya untuk memberikan dukungan kepada penulis.
3. Bapak Prof. DR. Mohammad Isa Irawan, MT, Bapak Drs. Daryono Budi Utomo, MSi, dan Bapak Drs. Nurul Hidayat, M.Kom selaku dosen penguji yang telah bersedia memberikan masukan berupa kritik dan saran yang bersifat membangun guna kesempurnaan Tugas Akhir ini.
4. Bapak Drs. Chairul Imron, MI.Komp. selaku Ketua Program Studi Jurusan Matematika ITS.
5. Ibu Dian Winda Setyawati, S.Si, M.Si selaku dosen wali yang senantiasa membimbing dan menasehati penulis.
6. Seluruh jajaran dosen dan staf jurusan Matematika ITS.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan saran dan kritik dari pembaca. Akhir kata, semoga Tugas Akhir ini bermanfaat bagi semua pihak yang berkepentingan.

Surabaya, Januari 2016

Penulis

special thanks to

1. Bapak Wahono dan Ibu Sri Setianingsih tersayang yang selalu memberikan dukungan, nasehat, dan doa pada penulis ketika pengerjaan Tugas Akhir ini.
2. Mbak Kusuma Sri Yuwono, Mas Nanang Setia K, Dek Nindy, Dek Happy, Dek Yaya, Mbak Gina yang selalu memotivasi penulis untuk menyelesaikan Tugs Akhir ini.
3. Achmad Firmansyah yang selalu setia membantu dan mendukung penulis. Terimakasih selalu memberikan semangat ketika pengerjaan Tugas Akhir ini.
4. Teman-teman “Bodo Amat” : Irvan, Raffy, Ade, Syukron, Yongky, Fiyan, Ibing, Tyara, Intan, Putri, Esti yang selalu menjadi penghibur dan penyemangat dikala suka dan duka.
5. Stephanie, Hasna, Wina, Ahmad, Mas Lutfi, Shahab, Mas Danang, Henny, Wanda, Desy, Tutut, Farah, Habib, Lisna terimakasih atas segala bentuk semangat dan dukungannya kepada penulis.
6. Segenap fungsionaris Himatika-ITS periode 2012-2013 dan teman-teman angkatan 2011 yang telah memberikan pengalaman dan kenangan berharga bagi penulis.

Tentu saja masih banyak pihak lain yang turut andil dalam penyelesaian Tugas Akhir ini yang tidak bisa penulis sebutkan satu persatu. Semoga Allah membalas dengan balasan yang lebih baik bagi semua pihak yang telah membantu penulis. *Amin ya rabbal’alamin.*

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xix
BAB I. PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Manfaat.....	3
1.6 Sistematika Penulisan Tugas Akhir.....	3
BAB II. KAJIAN TEORI	
2.1 Studi Penelitian Sebelumnya.....	5
2.2 Pengolahan Citra.....	5
2.2.1 Median <i>Filtering</i>	6
2.2.2 Metode <i>Mapping</i> Nilai Intensitas.....	7
2.2.3 Segmentasi Citra.....	8
2.2.4 Metode <i>Flood Filling</i>	9
2.3 Ekstraksi Fitur.....	9
2.3.1 Asymmetry.....	9
2.3.2 Border Irregularity.....	11
2.3.3 Color Variation.....	14
2.3.4 GLCM.....	15
2.4 Klasifikasi.....	16
BAB III. METODOLOGI PENELITIAN	
3.1 Studi Literatur.....	23
3.2 Analisa dan Penerapan Metode SVM untuk	

Klasifikasi Kanker Kulit Melanoma	23
3.3 Membangun Prototipe Perangkat Lunak	25
3.4 Uji Coba dan Evaluasi	26
3.5 Penulisan Tugas Akhir	26
BAB IV. PERANCANGAN DAN IMPLEMENTASI	
4.1 Perancangan Data	27
4.1.1 Data Masukan	27
4.1.2 Data Proses	27
4.1.3 Data Luaran	28
4.2 Perancangan Proses	28
4.2.1 Data Input	29
4.2.2 Proses Pra-pemrosesan	29
4.2.3 Proses Segmentasi	30
4.2.4 Proses Ekstraksi Fitur ABC dan GLCM	31
4.2.5 Proses Klasifikasi dengan SVM	33
4.3 Implementasi	35
4.3.1 Implementasi Antarmuka	36
4.3.2 Implementasi Tahap Input Data	42
4.3.3 Implementasi Tahap Pra-pemrosesan	42
4.3.4 Implementasi Tahap Segmentasi	43
4.3.5 Implementasi Tahap Ekstraksi Fitur dengan ABC dan GLCM	44
4.3.6 Implementasi Tahap Klasifikasi Citra dengan SVM	44
BAB V. PENGUJIAN DAN PEMBAHASAN HASIL	
5.1 Lingkungan Pengujian Sistem	47
5.2 Pengujian Tahap Pra-pemrosesan	47
5.3 Pengujian Tahap Ekstraksi Fitur	49
5.4 Pengujian Klasifikasi dengan SVM	50
5.5 Pembahasan Hasil Pengujian	53
5.6 Pembahasan Penyebab Besar Kecilnya Akurasi	53
BAB VI. PENUTUP	
6.1 Kesimpulan	55
6.2 Saran	55
DAFTAR PUSTAKA	57

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Penghilangan Derau dengan Median 3x3.....	7
Gambar 2.2 Kalkulasi <i>Fractal Dimension</i> menggunakan metode <i>Box-counting</i>	13
Gambar 2.3 SVM Berusaha Mencari <i>Hyperplane</i> Terbaik...	17
Gambar 2.4 SVM Memperbesar Margin	18
Gambar 3.1 Diagram Alir Tahap Keseluruhan Proses.....	25
Gambar 4.1 Diagram Alir Proses Pra-pemrosesan	30
Gambar 4.2 Diagram Alir Proses Segmentasi.....	31
Gambar 4.3 Diagram Alir Ekstraksi Fitur ABC dan GLCM	32
Gambar 4.4 Diagram Alir Proses <i>Training</i> SVM	34
Gambar 4.5 Diagram Alir Proses <i>Testing</i> SVM.....	35
Gambar 4.6 Halaman Utama.....	36
Gambar 4.7 Antarmuka Pelatihan Citra Kanker Kulit Melanoma Secara Manual	38
Gambar 4.8 Antarmuka Pelatihan Citra Kanker Kulit Melanoma dengan <i>Load Data</i>	39
Gambar 4.9 Antarmuka Pengujian Citra Kanker Kulit Melanoma.....	41
Gambar 5.1 Citra Input	48
Gambar 5.2 Hasil Tahap Pra-pemrosesan.....	49

DAFTAR TABEL

	Halaman
Tabel 4.1 Kegunaan Menu Sistem	37
Tabel 5.1 Lingkungan Pengujian Sistem	47
Tabel 5.2 Hasil Proses Ekstraksi Fitur ABC dan GLCM....	50
Tabel 5.3 Hasil Proses <i>Training</i> SVM.....	51
Tabel 5.4 Pengaruh Banyak Data <i>Training</i> Terhadap Pengujian dan Waktu Komputasi menggunakan Kernel Polynomial.....	52
Tabel 5.5 Pengaruh Parameter Polynomial, Linear, dan Rbf terhadap Pengujian dan Waktu Komputasi menggunakan Data Latih 1.....	53

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Kanker kulit melanoma merupakan jenis kanker kulit yang paling mematikan. Melanoma dimulai di sel-sel kulit dan bisa menyebar ke organ lain dalam tubuh melalui darah. Kemunculan tahi lalat baru atau perubahan pada tahi lalat yang sudah ada merupakan pertanda umum gejala melanoma [2].

Penyebab utama melanoma karena kulit sering terkena sinar ultra violet . Umumnya dokter menggunakan metode Biopsi untuk mendiagnosis melanoma. Metode Biopsi adalah pengambilan potongan kecil jaringan tubuh untuk sampel diagnosis. Dengan cara memotong atau menggores sepotong kecil jaringan atau dengan menggunakan jarum suntik untuk mengambil sampel, kemudian sampel diuji oleh beberapa dokter terlatih untuk didiagnosis. Proses Biopsi ini menyakitkan dan membutuhkan waktu yang cukup lama. Dari penelitian yang telah dilakukan, diagnosis dini melanoma menunjukkan lebih dari 90% dapat disembuhkan dan jika terlambat mendiagnosis melanoma menunjukkan kurang dari 50% dapat disembuhkan. Karena itu jika melanoma tidak terdiagnosis sejak tahap awal, maka akan menyebabkan kematian pada penderita. Sehingga mendeteksi kanker kulit dari tahap awal merupakan hal yang tak terelakkan [1,2].

Pada pengolahan citra digital, ada beberapa metode untuk mendeteksi citra kanker kulit melanoma. Jika sudah didapatkan ciri citra melanoma maka proses klasifikasi dapat dilakukan. Salah satu metode yang sering mendapat perhatian dalam klasifikasi citra adalah *Support Vector Machine* (SVM). *Support Vector Machine* (SVM) berusaha menemukan *hyperplane* yang terbaik pada *input space*. Akhir-akhir ini *Support Vector Machine* (SVM) telah berhasil diaplikasikan untuk menyelesaikan masalah di dunia nyata dan secara umum memberikan solusi yang lebih baik dibandingkan metode lainnya . Penelitian yang

menggunakan metode *Support Vector Machine* diantaranya Analisis Diagnosis Pasien Kanker Payudara Menggunakan Regresi Logistik dan *Support Vector Machine* (SVM) Berdasarkan Hasil Mamografi [3] dan penelitian tentang klasifikasi cacat pada selongsong peluru dengan metode *Support Vector Machine* (SVM) [5].

Berdasarkan latar belakang serta kajian pada penelitian tersebut, maka pada tugas akhir ini telah meneliti tentang klasifikasi citra kanker kulit menggunakan metode *Support Vector Machine* (SVM) berbasis citra digital.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, masalah yang akan diselesaikan adalah sebagai berikut:

1. Bagaimana mengaplikasikan metode *Asymmetry*, *Border Irregularity*, *Colour Variation* (ABC) dan *Gray-Level Co-Occurrence Matrix* (GLCM) untuk mengekstraksi fitur citra melanoma.
2. Bagaimana mengaplikasikan metode *Support Vector Machine* (SVM) untuk mengklasifikasi melanoma dan bukan melanoma.
3. Bagaimana membangun prototipe perangkat lunak untuk mengklasifikasi melanoma dan bukan melanoma.

1.3 Batasan Masalah

Untuk memperjelas rumusan masalah pada tugas akhir ini, maka batasan masalahnya adalah :

1. Data citra yang digunakan diperoleh dari grup riset *Vision and Image Processing* (VIP) Lab (vip.uwaterloo.ca).
2. Pembuatan prototipe perangkat lunak menggunakan Matlab.
3. Data citra masukan pada Tugas Akhir ini adalah citra melanoma maupun yang diduga melanoma dengan format .jpg dengan ukuran citra bebas.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah :

1. Mengaplikasikan metode *Asymmetry*, *Border Irregularity*, *Colour Variation* (ABC) dan *Gray-Level Co-Occurrence Matrix* (GLCM) untuk tahap ekstraksi fitur citra melanoma.
2. Mengaplikasikan metode *Support Vector Machine* (SVM) untuk tahap klasifikasi citra melanoma dan bukan melanoma.
3. Mendapatkan prototipe perangkat lunak untuk klasifikasi citra melanoma dan bukan melanoma.

1.5 Manfaat

Manfaat dari tugas akhir ini adalah :

1. Membantu dokter dan masyarakat dalam mendiagnosis melanoma.
2. Sebagai referensi dan informasi tentang penggunaan metode *Support Vector Machine* (SVM).

1.6 Sistematika Penulisan Tugas Akhir

Sistematika penulisan Tugas Akhir ini disusun berdasarkan beberapa bab sebagai berikut:

1. BAB I Pendahuluan
Bab ini menjelaskan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat dari penelitian yang dilakukan, serta sistematika penulisan Tugas Akhir.
2. BAB II Kajian Teori
Bab ini menjelaskan tentang studi penelitian sebelumnya dan beberapa teori pendukung yang berkaitan dengan Tugas Akhir ini.
3. BAB III Metodologi Penelitian
Bab ini menjelaskan tentang langkah-langkah yang digunakan untuk menyelesaikan Tugas Akhir ini.

4. BAB IV Perancangan dan Implementasi

Bab ini menjelaskan tentang perancangan dan implementasi sistem, serta proses pembuatan sistem secara keseluruhan.

5. BAB V Pengujian dan Pembahasan Hasil

Bab ini menjelaskan tentang hasil uji coba sistem. Kemudian dicatat sebagai bahan untuk merumuskan beberapa kesimpulan dari Tugas Akhir ini.

6. BAB VI Kesimpulan dan Saran

Bab ini berisi tentang kesimpulan dari hasil penelitian dan saran untuk pengembangan dari Tugas Akhir ini.

BAB II

KAJIAN TEORI

2.1 Studi Penelitian Sebelumnya

Beberapa penelitian sebelumnya yang terkait dengan tugas akhir ini yaitu oleh Fourina Ayu Novianti dan Santi Wulan Purnami menganalisis diagnosis kanker payudara menggunakan regresi logistic dan *support vector machine* (SVM) berdasarkan hasil mamografi. Pada penelitian tersebut yang diklasifikasikan adalah penderita dan bukan penderita kanker payudara. Ketepatan klasifikasi yang didapatkan dari model regresi logistik pada kasus ini hanya sebesar 37% dan *Support Vector Machine* memberikan ketepatan klasifikasi di atas 95 %. Hal ini menunjukkan keunggulan metode *Support Vector Machine* yang menghasilkan akurasi yang tinggi [3].

Penelitian oleh Refta Listia, Agus Harjoko dengan judul Klasifikasi Massa pada Citra Mammogram Berdasarkan *Gray Level Cooccurrence Matrix* (GLCM). Pada penelitian tersebut metode ekstraksi ciri GLCM 4 arah dengan $d=1$ mampu mengklasifikasi citra mammogram ke dalam kelas normal, tumor ganas dan tumor jinak [4].

Penelitian oleh Tyara Herdha yaitu mengklasifikasi cacat pada selongsong peluru dengan metode *Support Vector Machine* (SVM). Berdasarkan hasil pengujian sistem pada penelitian tersebut dapat mengenali citra amunisi dengan tingkat akurasi sebesar 90% pada 10 data pelatihan baik dan cacat dengan panjang fitur 200 dengan waktu komputasi terbaik untuk pelatihan adalah 7.4 detik. [5].

2.2 Pengolahan Citra

Pengolahan citra adalah suatu metode yang digunakan untuk memroses citra dalam bentuk dua dimensi. Segala operasi untuk memperbaiki, menganalisa, atau melakukan transformasi suatu citra menjadi citra lain dengan menggunakan teknik tertentu disebut pemrosesan citra (*image processing*). Pemrosesan citra

telah diaplikasikan dalam berbagai bentuk, dengan tingkat kesuksesan yang cukup besar. Konsep-konsep pemrosesan citra dalam tugas akhir ini meliputi pra-pemrosesan, ekstraksi fitur, dan klasifikasi.

Tahap pra-pemrosesan bertujuan untuk menghilangkan *noise* dan mempersiapkan citra agar bisa diproses untuk proses ekstraksi fitur. Tahapan-tahapan yang dilakukan dalam pra-pemrosesan antara lain :

2.2.1 Median Filtering

Peningkatan kualitas citra menggunakan filter median. Filter ini memilih intensitas piksel yang ditengah, setelah piksel-piksel yang tercakup dalam filter diurutkan. Filter median sering digunakan karena filter ini memberikan kemampuan reduksi *noise* yang sangat baik dengan *blurring* yang sedikit. Filter median sangat populer dalam pengolahan citra. Filter ini dapat dipakai untuk menghilangkan derau bintik-bintik. Sebagai contoh tinjau jendela berupa kelompok piksel (berbentuk kotak diarsir) pada sebuah citra pada Gambar 2.1. Piksel yang sedang diproses adalah yang mempunyai intensitas 35. Urutkan piksel-piksel tersebut:

9 10 10 10 **10** 10 11 12 35

Median dari kelompok tersebut adalah 10 (dicetak tebal). Titik tengah dari jendela (35) sekarang diganti dengan nilai median (10) [6].

13	10	15	14	18
12	10	10	10	15
11	11	35	10	10
13	9	12	10	12
13	12	9	8	10

(a)

13	10	15	14	18
12	10	10	10	15
11	11	10	10	10
13	9	12	10	12
13	12	9	8	10

(b)

Gambar 2.1. (a) Penghilangan derau dengan median filter 3x3. Pixel bernilai 35 terkena derau . (b) 35 diganti dengan median dari kelompok 3x3 piksel [6].

2.2.2 Metode Mapping Nilai Intensitas

Metode penguatan nilai kekontrasan menggunakan teknik *mapping* nilai intensitas. Intensitas piksel-piksel pada citra input dimap dengan batasan atau *threshold* tertentu, misal senilai *minS* untuk batas minimal dan *maxS* untuk batas maksimal. Maka piksel-piksel pada citra input yang memiliki intensitas dibawah batas minimal, nilainya akan diset sama dengan batas minimal. Begitu juga untuk piksel-piksel pada citra input yang memiliki intensitas diatas batas maksimal, nilainya akan diset sama dengan batas maksimal. Untuk piksel-piksel pada citra input yang memiliki intensitas diantara batas minimal dan batas maksimal, maka nilainya dipetakan/ diskalakan (*mapping*) untuk memenuhi rentang nilai-nilai keabuan yang lengkap (dari batas minimal sampai dengan batas maksimal) dengan persamaan [6]:

$$S = \frac{(R - \min R)(\max S - \min S)}{(\max R - \min R)} + \min S \quad (2.1)$$

Dengan,

S = citra output

maxS = batas maximal (untuk citra output)

minS = batas minimal (untuk citra output)

R = citra input

$maxR$ = intensitas terbesar dari piksel di citra input

$minR$ = intensitas terkecil dari piksel di citra input

2.2.3 Segmentasi Citra

Segmentasi menggunakan metode thresholding otsu, yaitu metode untuk segmentasi berdasarkan histogram citra. Metode otsu berasumsi bahwa citra yang akan dicari nilai thresholdnya memiliki 2 kelas piksel, yaitu background dan foreground. Cara kerja metode Otsu dengan memaksimalkan varians antar kelas (*between-class variance*). Varians antar kelas ini cocok digunakan untuk analisa diskriminan kelas secara statistik [7].

Algoritma dari metode thresholding Otsu untuk pemilihan satu threshold:

- Citra dibagi menjadi 2 piksel yaitu G1 dan G2
- Hitung bobot dari masing-masing kelompok piksel G1 dan G2

$$w1 = \sum_0^t p(i) \quad (2.2)$$

dan

$$w2 = \sum_t^n p(i) \quad (2.3)$$

- Hitung rata-rata intensitas μ_1 dan μ_2 untuk piksel di daerah G1 dan G2:

$$\mu_1 = [\sum_0^t p(i)x(i)]/w1 \quad (2.4)$$

dan

$$\mu_2 = [\sum_t^n p(i)x(i)]/w2 \quad (2.5)$$

- Hitung varians antar kelas (*between-class variance*) pada persamaan :

$$Vb = w1.w2((\mu_1 - \mu_2)^2) \quad (2.6)$$

- Ulangi langkah a sampai d, sehingga didapatkan nilai Vb terbesar. Maka itulah nilai k^* yang digunakan.

Setelah mendapatkan nilai ambang batas atau threshold maka kita gunakan untuk segmentasi citra. Metode Otsu hanya untuk menemukan threshold yang optimum. Teknik segmentasi menggunakan persamaan :

$$g(x,y) = \begin{cases} 1, & \text{jika } f(x,y) > k^* \\ 0, & \text{jika } f(x,y) \leq k^* \end{cases} \quad (2.7)$$

Hasil segmentasi adalah citra biner yang memiliki nilai intensitas hanya 0 dan 1. Nilai intensitas 0 menyatakan warna hitam dan (dalam Tugas Akhir ini dianggap background) sedangkan nilai intensitas 1 menyatakan warna putih (dianggap foreground atau objek).

2.2.4 Metode *Flood-Filling*

Metode *flood fill* disebut juga dengan *seed fill*. Algoritma *flood-fill* memiliki 3 parameter masukan, yaitu node awal, warna target, dan warna pengganti. Algoritma ini bekerja dengan mencari semua node yang terkoneksi dengan node awal sebagai sebuah jalur berwarna target, lalu mengubahnya dengan warna pengganti.

2.3 Ekstraksi fitur

Tujuan dari ekstraksi fitur adalah untuk mendapatkan karakteristik pembeda yang mewakili sifat utama dengan memisahkannya dari fitur yang tidak diperlukan pada proses klasifikasi. Ekstraksi fitur yang digunakan untuk penelitian ini adalah ekstraksi fitur ABC dan GLCM.

2.3.1 *Asymmetry*

Sebuah tahi lalat umumnya berbentuk simetris dan sirkular sedangkan melanoma biasanya tumbuh secara irregular dan asimetris. Pengukuran nilai asimetri bergantung pada suatu properti geometris yang disebut momen inersial. Momen inersial digunakan untuk mengkalkulasi dua fitur, yaitu *asymmetry index* AI dan *lengthening index* $\hat{A}$ [8].

1. *Asymmetry Index (AI)*

Pada pencarian nilai *asymmetri index*, titik *origin* (0,0) dari koordinat kartesius ditetapkan pada pusat massa G dari area luka L , dengan L didefinisikan oleh fungsi biner $z(i,j)$, dengan $z(i,j) = 1$ jika $(i,j) \in L$, selain itu 0. Momen inersia kuadrat $I(\varphi)$ dari suatu citra luka L yang berhubungan dengan sembarang sumbu yang melalui G menunjukkan sudut φ terhadap sumbu horizontal dari koordinat kartesius Δ , dan diberikan dalam persamaan :

$$I(\varphi) = \sum_{(i,j) \in L} D_{\varphi}^2(i,j) = \sum_{(i,j) \in L} [-i \times \sin\varphi + j \times \cos\varphi]^2 \quad (2.8)$$

$D_{\varphi}(i,j)$ adalah jarak antara piksel yang ditinjau (i,j) dan proyeksinya pada Δ , sepanjang normal garis Δ . Sumbu utama mayor didapat dari momen inersia terkecil dari L dan memiliki arah tegak φ_0 . Sumbu utama mayor tersebut bisa didapat dengan menghitung turunan dari persamaan tersebut dan mengeset turunannya dengan 0 seperti dalam persamaan berikut :

$$\frac{\partial I(\varphi)}{\partial \varphi} = 0 \Rightarrow \varphi_0 = \frac{1}{2} \tan^{-1} \left[\frac{2m_{11}^c}{m_{20}^c - m_{02}^c} \right] \quad (2.9)$$

m_{11}^c merepresentasikan momen standar, m_{20}^c merepresentasikan momen kuadrat yang berhubungan dengan sumbu horizontal koordinat kartesius G_x dan m_{02}^c merepresentasikan momen kuadrat yang berhubungan dengan sumbu vertical koordinat kartesius G_y . Sumbu utama minor dari L memiliki arah $\varphi_0 + \pi/2$ menghasilkan arah yang melintang terhadap L yang berarti memiliki momen inersia terbesar. Sumbu melintang dan tegak yang telah didapatkan bisa dilakukan untuk mengkalkulasi nilai *asymmetri index*. Dilakukan dengan melipat area luka L pada kedua sumbu tadi dan mengukur area *non overlap* seperti dalam persamaan

$$AI = \frac{1}{2} \sum_{k=1}^2 \frac{\Delta A_k}{A_L} \quad (2.10)$$

Subscript k mengidentifikasi sumbu utama (mayor vs. minor), ΔA_k merupakan area *non-overlap* saat luka dilipat dan $A_L (A_L = m_{00}^c)$ adalah area luka.

2. *Lengthening Index*

Pengukuran ini digunakan untuk mendeskripsikan pemanjangan sebuah luka, misalnya derajat *anisotropy* luka. Pemanjangan luka berhubungan dengan nilai eigen λ', λ'' dari matriks tensor inersia. *Lengthening Index* ini didefinisikan oleh perbandingan antara momen inersia λ' (terhadap sumbu mayor) dan momen inersia λ'' (terhadap sumbu minor).

$$\mathring{A} = \frac{\lambda'}{\lambda''} \quad (2.11)$$

$$\lambda' = \frac{m_{20}^c + m_{02}^c - \sqrt{(m_{20}^c - m_{02}^c)^2 + 4(m_{11}^c)^2}}{2} \quad (2.12)$$

$$\lambda'' = \frac{m_{20}^c + m_{02}^c + \sqrt{(m_{20}^c - m_{02}^c)^2 + 4(m_{11}^c)^2}}{2} \quad (2.13)$$

2.3.2 *Border Irregularity*

Luka tidak berbahaya umumnya mempunyai pembatas yang jelas. Sebuah melanoma sering menunjukkan pembatas yang tidak jelas atau kabur yang mensinyalkan pertumbuhan dan penyebaran kanker.

Iregularitas pembatas luka/ tepi area luka (*Border Irregularity Lesion*) ini sudah mempresentasikan faktor diagnostik yang sangat signifikan ketika menduga bahaya sebuah luka. Dalam Tugas Akhir ini, digunakan lima fitur untuk mengkuantifikasi *Border Irregularity*, antara lain: indeks kepadatan (*Compactness Index*) CI , dimensi fraktal (*Fractal Dimension*) fd , kecuraman tepi (*Edge Abruptness*) Cr , dan transisi pigmentasi (*Pigmentation Transition*) m_e dan v_e [8].

1. *Compactness Index*

Indeks kepadatan (*Compactness Index/ CI*) adalah pengukuran bentuk pembatas paling populer yang mengestimasi kebutuhan obyek 2D. Namun, pengukuran ini sangat sensitive

terhadap noise di sepanjang pembatas/ tepi area luka yang nilainya dikuatkan oleh kuadrat dari keliling luka. Berikut ini persamaan Indeks kepadatan (*Compactness Index/ CI*)

$$CI = \frac{P_L^2}{4\pi A_L} \quad (2.14)$$

P_L = keliling luka

2. *Fractal Dimension*

Fractal memiliki karakteristik kesamaan sendiri (*self-similarity*), artinya fractal memiliki sifat-sifat yang sama untuk berbagai skala/ukuran yang digunakan. Setiap bagian fractal yang memiliki skala berbeda tersebut memiliki sifat yang sama dengan keseluruhan fractal. Karakteristik ini yang menyebabkan fractal cocok digunakan untuk teknik kompresi. Karakteristik lain dari fractal adalah dimensinya. Ukuran dimensi pada umumnya merupakan bilangan bulat, seperti garis memiliki dimensi 1, bidang memiliki dimensi 2, dan kubus memiliki dimensi 3, dan seterusnya. Akan tetapi, dimensi fractal merupakan sesuatu yang aneh karena dapat bernilai pecahan. Dimensi fractal ini dapat digunakan sebagai ciri suatu citra. Dimensi fractal dapat dihitung dengan metode perhitungan kotak (*box-counting*). Metode ini membagi citra menjadi kotak-kotak dengan berbagai variasi ukuran (r).

Salah satu contoh pengukuran nilai r adalah $2k$, dengan $k = 0,1,2,\dots$ dst, dan $2k$ lebih kecil dari ukuran citra. Contoh untuk citra luka pada kulit, digunakan grid kotak yang membagi citra dengan masing-masing kotak berukuran $r \times r$ (Gambar 2.2). $N(r)$ adalah sebuah piksel yang berisi potongan pembatas/ tepi luka. Ukuran piksel berbeda (r berbeda) digunakan dan fd diperoleh dari gradient atau kemiringan garis regresi $\log(r)$ terhadap $\log(N(r))$.



Gambar 2.2. Kalkulasi *fractal dimension* (*fd*) menggunakan metode *box-counting*

Secara keseluruhan, hubungan antara $N(r)$ dan fd dapat diungkapkan sebagai berikut:

$$N(r) = \lambda r^{-fd} \quad (2.15)$$

3. *Edge Abruptness*

Luka dengan pembatas atau tepi yang irregular (*Edge Abruptness*) memiliki nilai variance jarak radial yang besar (contoh d_2 , jarak antara centroid G_L dan pembatas atau tepi C). untuk mengestimasi iregularitas pembatas dilakukan analisa nilai variance dari distribusi jarak radial

$$C_r = \frac{\frac{1}{P_L} \sum_{p \in C} (d_2(p, G_L) - m_d)^2}{m_d^2} \quad (2.16)$$

m_d adalah mean jarak d_2 anantara titik-titik pembatas dan centroid G_L (titik pusat luka).

4. *Pigmentation Transition*

Fitur penting ini menjelaskan transisi pigmentasi antara area luka dan kulit sekitarnya. Tepi area luka yang mengalami perubahan pigmentasi secara tajam mengindikasikan bahwa luka adalah melanoma, sedangkan perubahan secara perlahan (gradasi) mengindikasikan luka tidak berbahaya. Karena itu, dipertimbangkan komponen luminansi $lum(i, j)$ (Persamaan 2.17) dari warna citra asli sebagai tiga komponen warna yang berbobot sama. Kemudian diestimasi *gradient magnitude* dari komponen

intensitas lum sepanjang pembatas/ tepi luka C di kulit. Diperoleh set-set nilai *gradient magnitude* K , $e(k)$ ($1 \leq k \leq K$, dimana K adalah jumlah sampel tepi luka) yang mendiskripsikan secara local transisi antara luka dan kulit di setiap piksel tepi luka pada citra. Untuk menjelaskan transisi secara global, digunakan mean m_e dan variance v_e dari nilai magnitude gradient yang mendeskripsikan level kecuraman dan variasinya secara global (Persamaan 2.18).

$$lum(i, j) = \frac{1}{3} [r(i, j) + g(i, j) + b(i, j)] \quad (2.17)$$

$$m_e = \frac{1}{K} \sum_{k=1}^K e(k), v_e = \sum_{k=1}^K e^2(k) - m_e^2 \quad (2.18)$$

2.3.3 Color Variation

Salah satu tanda awal dari melanoma adalah munculnya variasi warna dalam warna. Karena melanoma tumbuh dalam sel penumbuh pigmen, umumnya berwarna coklat, coklat gelap, atau hitam, tergantung kondisi pigmen melanin pada kulit. Untuk membatasi diagnosis lebih lanjut, variasi warna pada sebuah luka dijelaskan dengan homogenitas warna (*Color Homogeneity*) C_h dan korelasi antara geometri dan fotometri (*Correlation between Photometry and Geometry*) C_{pg} [8].

1. Color Homogeneity

Histogram dari luminansi warna pada luka dibagi dalam tiga interval yang panjangnya sama. Interval yang berelasi ke nilai luminansi ketiga terkecil mendefinisikan area gelap. *Level intermediate* berelasi ke yang lain dan tidak terlibat dalam kuantifikasi warna. Kemudian homogenitas warna dijelaskan sebagai jumlah transisi zona lebih terang/ zona lebih gelap ke zona lebih gelap/zona lebih terang ketika memindai luka secara horizontal dan vertical.

2. Correlation between Photometry and Geometry

Atribut ini mengevaluasi distribusi warna pada luka. Termasuk penjelasan evaluasi level warna dari centroid G_L menuju pembatas/ tepi luka. Nilai lebih besar menunjukkan luka tak-berbahaya karena luka tak-berbahaya memiliki aspek target, sedangkan nilai yang kecil menandakan bahaya melanoma.

$$C_{pg} = \frac{1}{A_L} \cdot \sum_{P \in L} \frac{(lum(p) - m_1) \cdot (d_2(p, G_L) - m_d)}{v_1 \cdot v_d} \quad (2.19)$$

m_d dan v_d adalah mean dan variance, d_2 adalah jarak antara titik pembatas dengan centroid, m_1 dan v_1 berhubungan dengan luminansi, P adalah semua elemen piksel dan L adalah area luka.

2.3.4 GLCM

GLCM (*Gray-Level Co-occurrence Matrix*) merupakan ekstraksi fitur yang sering digunakan untuk analisis tekstur. GLCM adalah suatu matriks yang elemen-elemennya merupakan jumlah pasangan piksel yang memiliki tingkat kecerahan tertentu, dimana pasangan piksel itu terpisah dengan jarak d , dan dengan suatu sudut inklinasi θ .

Suatu piksel yang bertetangga (i, j) dan memiliki jarak d diantara keduanya, dapat terletak di delapan arah yang berlainan baik horizontal, vertical, maupun diagonal.

Berikut adalah persamaan ekstraksi fitur GLCM [9]:

1. Entropi

Entropi dapat menunjukkan ketidakteraturan ukuran bentuk citra.

$$Entropy = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p(i, j) \log(p(i, j)) \quad (2.20)$$

dengan $P_{(i, j)}$ = Piksel baris ke i dan kolom ke j

2. Energi

Energi menyatakan ukuran konsentrasi pasangan dengan intensitas keabuan tertentu pada matriks, dimana (i, j)

menyatakan nilai pada baris i dan kolom j pada matriks kookurensi.

$$Energy = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p(i, j)^2 \quad (2.21)$$

dengan $P_{(i,j)}$ = Piksel baris ke $-i$ dan kolom ke $-j$.

3. Kontras

Menunjukkan ukuran penyebaran (momen inersia) elemen-elemen matriks citra.

$$Contrast = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |i - j|^2 p(i, j) \quad (2.22)$$

dengan $P_{(i,j)}$ = Piksel baris ke $-I$ dan kolom ke $-j$.

4. Homogenitas

Menunjukkan kehomogenan variasi intensitas dalam citra.

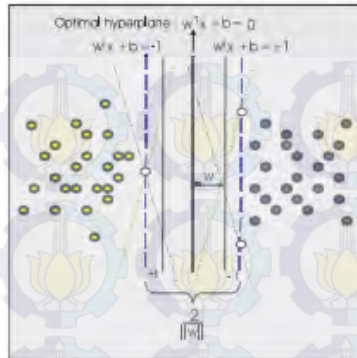
$$homogenitas = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{p(i, j)}{1 + |i - j|} \quad (2.23)$$

dengan $P_{(i,j)}$ = Piksel baris ke $-I$ dan kolom ke $-j$

2.4 Klasifikasi

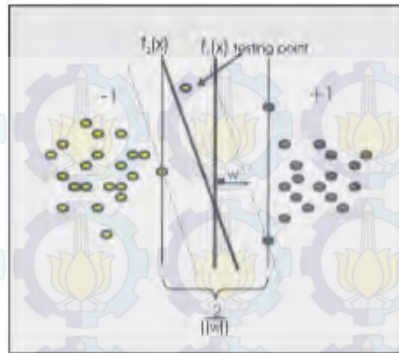
Klasifikasi merupakan proses pengelompokan suatu data berdasarkan ciri-ciri tertentu. Dalam tugas akhir ini metode yang digunakan untuk mengklasifikasi citra adalah *Support Vector Machine* (SVM) [9].

Support Vector Machine (SVM) merupakan sebuah metode klasifikasi linier dengan *feature set* yang telah ditentukan sebelumnya. Metode SVM akan mencari sebuah *hyperplane* linier dengan margin terbesar untuk memisahkan kelas yang ada. Margin terbesar, yang kemudian disebut sebagai *Maximum Marginal Hyperplane* (MMH) tersebut akan memberikan jarak terbesar antar kelas. Jarak antara *hyperplane* dengan sebuah sisi dari margin tersebut sama dengan jarak antara *hyperplane* dengan margin pada sisi lainnya.



Gambar 2.3. SVM Berusaha Mencari *Hyperplane* Terbaik [10].

Gambar 2.3 memperlihatkan bagaimana SVM bekerja untuk menemukan suatu *hyperplane* / fungsi pemisah dengan margin yang maksimal. Tujuannya untuk membuktikan bahwa memaksimalkan margin antara dua set objek akan meningkatkan probabilitas pengelompokkan secara benar dari data testing. Pada dasarnya jumlah fungsi pemisah ini tidak terbatas banyaknya. Misalkan dari jumlah yang tidak terbatas ini diambil dua fungsi, yaitu $f_1(x)$ dan $f_2(x)$ (Gambar 2.4). Fungsi f_1 mempunyai margin yang lebih besar dari pada fungsi f_2 . Setelah menemukan dua fungsi ini, sekarang suatu data baru masuk dengan keluaran -1 . Langkah selanjutnya adalah mengelompokkan apakah data ini ada dalam kelas -1 atau $+1$ menggunakan fungsi pemisah yang sudah ditemukan. Dengan menggunakan f_1 , akan dikelompokkan data baru ini di kelas -1 yang berarti fungsi yang dipilih benar mengelompokkannya. Sekarang jika menggunakan f_2 , data baru akan ditempatkan di kelas $+1$ yang berarti pengelompokkan yang dilakukan salah. Dari contoh sederhana ini dapat diketahui bahwa memperbesar margin bisa meningkatkan probabilitas pengelompokkan suatu data secara benar.



Gambar 2.4. Memperbesar margin dapat meningkatkan probabilitas pengelompokkan suatu data secara benar [10].

Dalam kasus real, biasanya data yang ada bersifat *non linearly separable data*. Sehingga perlu dilakukan perubahan pada metode SVM yang biasa. Metode ini mentransformasikan data ke dalam dimensi ruang fitur (*feature space*) sehingga dapat diipisahkan secara linear pada *feature space*. *Feature space* dalam prakteknya biasanya berdimensi lebih tinggi dari vektor input (*input space*). Hal ini mengakibatkan komputasi pada *feature space* mungkin akan sangat besar, karena ada kemungkinan *feature space* dapat memiliki jumlah fitur yang tidak terhingga. Selain itu sulit untuk mengetahui fungsi transformasi yang tepat. Untuk mengatasi masalah ini, pada SVM digunakan “*kernel trick*”. Untuk menyelesaikan masalah pada *non linearly separable data* harus melakukan modifikasi pada beberapa fungsi yang ada.

Pada mulanya *teknik machine learning* dikembangkan dengan asumsi kelinearan. Sehingga algoritma yang dihasilkan terbatas untuk kasus-kasus yang linear saja. Akan tetapi untuk menghadapi kasus yang tidak linear maka dapat menggunakan bantuan berbagai macam fungsi kernel. *Kernel trick* memberikan berbagai kemudahan, karena dalam proses pembelajaran SVM, untuk menentukan *support vector*, maka cukup dengan

mengetahui fungsi kernel yang dipakai, dan tidak perlu mengetahui wujud dari fungsi non-linear.

Ada banyak jenis kernel yang biasa digunakan dalam proses SVM. Pada tugas akhir ini metode yang telah diteliti untuk memetakan dari *input space* ke *feature space* ada tiga yaitu kernel polynomial, kernel linear, dan kernel gaussian/rbf [10]. Dari ketiga kernel tersebut dicari kernel mana yang memberikan hasil lebih akurat.

Fungsi kernel polynomial memiliki bentuk persamaan :

$$K(x_i, x_j) = ((x_i, x_j^T) + 1)^n \quad (2.24)$$

Fungsi kernel linear memiliki bentuk persamaan :

$$K(x_i, x_j) = x_i x_j^T \quad (2.25)$$

Fungsi kernel rbf memiliki bentuk persamaan :

$$K(x_i, x_j) = \exp \frac{\|x_i - x_j\|^2}{2\sigma^2} \quad (2.26)$$

$i, j = 1, 2, \dots, l$ dengan :

K = kernel

x_i = data inputan (*training*)

x_j = data inputan (*training/testing*) yang telah di transpos

σ = parameter bebas

l = banyaknya data

Setelah dilakukan transformasi dari *input space* ke *feature space* dilakukan, langkah selanjutnya adalah pencarian *hyperplane* klasifier. Untuk itu dilakukan langkah-langkah berikut [10]:

- a. Data yang tersedia dinotasikan sebagai $x \in R$ sedangkan label masing-masing dinotasikan $y_i \in \{-1, +1\}$ untuk $i=1, 2, \dots, l$ yang mana l adalah banyaknya data. Diasumsikan kedua *class* -1 dan $+1$ dapat terpisah secara sempurna oleh *hyperplane* berdimensi d , yang didefinisikan:

$$\vec{w} \cdot x + b = 0 \quad (2.27)$$

Sebuah *pattern* x_i yang termasuk *class* -1 (sampel negatif) dapat dirumuskan sebagai *pattern* yang memenuhi persamaan :

$$\vec{w} \cdot x + b = -1 \quad (2.28)$$

sedangkan pattern yang termasuk *class* +1 (sampel positif):

$$\vec{w} \cdot x + b = +1 \quad (2.29)$$

- b. Margin terbesar dapat ditemukan dengan memaksimalkan nilai jarak antara hyperplane dan titik terdekatnya, yaitu $1/\|w\|$. Hal ini dapat dirumuskan sebagai *Quadratic Programming* (QP) *problem*, yaitu mencari titik minimal dengan memperhatikan *constraint* :

$$\min_{\vec{w}} \tau(w) = \frac{1}{2} \|w\|^2 \quad (2.30)$$

dengan batasan :

$$y_i(x_i \cdot w + b) - 1 \geq 0 \quad (2.31)$$

- c. Problem diatas dapat dipecahkan dengan berbagai teknik komputasi, diantaranya *Lagrange Multipliers* sebagaimana ditunjukkan pada Persamaan :

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i (y_i((x_i \cdot w + b) - 1)) \quad (2.32)$$

α_i adalah *Lagrange Multipliers*, yang bernilai nol atau positif ($\alpha_i \geq 0$).

- d. Nilai optimal dari persamaan diatas dapat dihitung dengan meminimalkan L terhadap w dan b , dan memaksimalkan L terhadap α_i . Dengan memperhatikan sifat bahwa pada titik optimal gradient $L=0$, persamaan dapat dimodifikasi sebagai maksimalisasi problem yang hanya mengandung α_i saja, sebagaimana Persamaan :

$$= \max \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (2.33)$$

dengan batasan :

$$\alpha_i \geq 0 \quad (i = 1, 2, \dots, l) \quad (2.34)$$

$$\sum_{i=1}^1 \alpha_i y_i = 0 \quad (2.35)$$

Dari hasil dari perhitungan ini diperoleh α_i yang kebanyakan bernilai positif. Data yang berkorelasi dengan α_i yang positif inilah yang disebut sebagai *support vector*.

BAB III

METODOLOGI PENELITIAN

Tahap-tahap yang akan dilakukan pada tugas akhir adalah studi literature, analisa dan penerapan metode SVM untuk klasifikasi kanker kulit, membangun prototipe perangkat lunak, uji coba dan evaluasi, penulisan tugas akhir.

3.1 Studi literatur

Pada tahap pertama ini dilakukan pengkajian tentang melanoma, segmentasi, perbaikan citra, dll. Pengkajian ini dilakukan dengan membaca jurnal, buku, maupun artikel. Selain itu pada tahap ini juga melakukan diskusi dengan dosen, teman-teman, dan orang-orang yang berkompeten dalam tugas akhir ini.

3.2 Analisa dan Penerapan Metode SVM untuk Klasifikasi Kanker Kulit Melanoma

Pada tahap ini akan dirancang sebuah prototipe perangkat lunak klasifikasi melanoma dan bukan melanoma dengan menggunakan metode *Support Vector Machine* (SVM). Gambaran umum dari proses pengerjaan Tugas Akhir ini adalah pengambilan data, pra-pemrosesan, segmentasi, ekstraksi fitur, dan klasifikasi (dapat dilihat pada Gambar 3.1).

1. Pengambilan data

Pada tahapan ini dilakukan pengambilan citra melanoma yang diperoleh dari grup riset *Vision and Image Processing* (VIP) Lab (vip.uwaterloo.ca).

2. Pra-pemrosesan

Pada tahapan ini dilakukan pra-pemrosesan untuk menghilangkan *noise* dan mempersiapkan citra agar dapat diproses untuk proses ekstraksi fitur.

3. Segmentasi

Proses ini bertujuan untuk membagi suatu citra menjadi wilayah-wilayah yang homogen berdasarkan kriteria

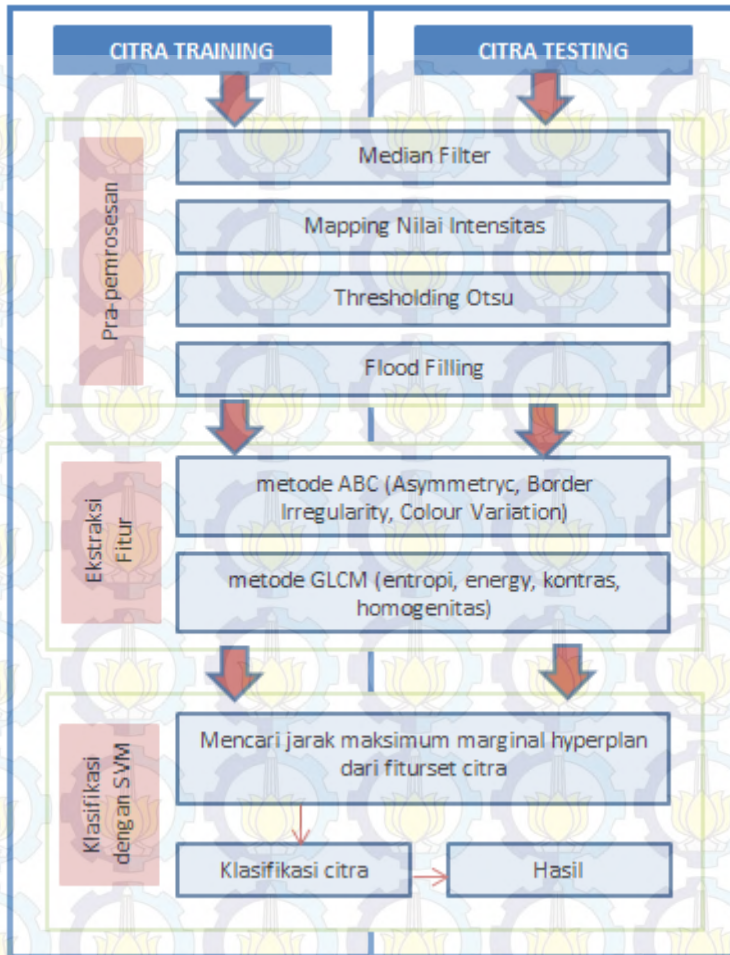
keserupaan tertentu antara tingkat keabuan suatu piksel dengan tingkat keabuan piksel-piksel tetangganya.

4. Ekstraksi fitur

proses untuk mendapatkan ciri-ciri citra melanoma dengan menggunakan ekstraksi fitur *Asymmetry*, *Border Irregularity*, *Colour Variation* (ABC) dan *Gray-Level Co-Occurrence Matrix* (GLCM) yaitu energi, entropi, kontras, homogenitas.

5. Klasifikasi

proses mengklasifikasi atau membandingkan ciri citra dengan menggunakan metode SVM.



Gambar 3.1 Diagram Alir Keseluruhan Proses.

3.3 Membangun prototipe perangkat lunak

Pembuatan prototipe perangkat lunak berdasarkan analisis algoritma yang telah dibuat dengan menggunakan Matlab. Serta pembuatan *interface* yang bersifat *userfriendly* sehingga dapat

memudahkan pengguna dalam mengaplikasikan perangkat lunak ini.

3.4 Uji coba dan evaluasi

Menguji coba perangkat lunak dengan data-data yang diperoleh kemudian dianalisis dan dievaluasi hasilnya. Tingkat akurasi dihitung dari

$$\text{Akurasi} = \frac{\text{jumlah citra yang terklasifikasi dengan benar}}{\text{jumlah total citra uji}} 100\%$$

Selain itu akan ditinjau kondisi-kondisi yang mempunyai kinerja terbaik atau pada saat metode klasifikasi SVM memberikan hasil yang baik atau memberikan hasil yang buruk.

3.5 Penulisan tugas akhir

Penulis menulis proses dan semua hasil dalam pengerjaan tugas akhir ini.

BAB IV

PERANCANGAN DAN IMPLEMENTASI

Pada bab ini menjelaskan mengenai perancangan sistem dan hasil implementasi semua proses yang telah dirancang sebelumnya. Pembahasan perancangan sistem diawali dengan perancangan data, perancangan proses, serta perancangan antar muka.

4.1 Perancangan Data

Data yang digunakan dalam mengklasifikasi citra melanoma dan bukan melanoma ini dibagi menjadi 3 macam, yaitu data masukan yang didapatkan dari pengguna perangkat lunak, data proses yang dibutuhkan dan dihasilkan selama proses eksekusi perangkat lunak, dan data keluaran yang memberikan hasil proses eksekusi perangkat lunak untuk pengguna yang menjalankannya, yaitu citra terdeteksi sebagai melanoma atau bukan melanoma.

4.1.1 Data Masukan

Data masukan yang diberikan oleh pengguna adalah berupa citra masukan. Data citra masukan ini adalah citra luka melanoma maupun yang diduga melanoma dengan format .jpg dengan ukuran citra bebas. Citra masukan diharapkan adalah hasil foto dari luka dengan resolusi citra harus bagus dan berwarna.

Perlu diingat bahwa dalam metode adaptif ini, tidak diperlukan parameter apapun yang harus diatur oleh pengguna pada saat awal penggunaan aplikasi. Semua parameter telah diatur secara adaptif didalam sistem aplikasi.

4.1.2 Data Proses

Terdapat beberapa proses utama yang akan dilalui data citra yaitu pemrosesan awal citra (*image preprocessing*), segmentasi citra (*image segmentation*), ekstraksi fitur ABC dan GLCM, proses pelatihan sistem (*system training*), dan proses pengujian sistem (*system testing*).

4.1.3 Data Luaran

Tahapan atau proses paling akhir adalah pengujian sistem. Data hasil dari proses tersebut adalah prediksi apakah citra uji yang berada pada testing set merupakan melanoma atau hanya luka biasa yang tidak berbahaya.

4.2 Perancangan Proses

Pada subbab ini akan diuraikan perancangan proses dalam pembuatan perangkat lunak. Perancangan proses ini bertujuan untuk memperjelas hubungan antar proses beserta langkah-langkah yang dilakukan pada setiap proses. Secara garis besar proses yang dilakukan adalah :

1. Data input, berupa citra melanoma atau yang diduga melanoma.
2. Proses pra-pemrosesan, adalah serangkaian proses untuk mempersiapkan citra input sebelum proses segmentasi, ekstraksi fitur dengan ABC dan GLCM serta klasifikasi dengan *Support Vector Machine*, misalnya penghilangan noise.
3. Proses Segmentasi, adalah serangkaian proses untuk mempersiapkan citra input sebelum proses ekstraksi fitur dengan ABC dan GLCM serta klasifikasi dengan *Support Vector Machine*, misalnya merubah citra menjadi citra biner.
4. Proses Ekstraksi Fitur dengan ABC dan GLCM, merupakan proses-proses yang dilakukan terhadap citra input yang sudah di proses sebelumnya untuk memperoleh ciri citra melanoma yang akan digunakan untuk proses berikutnya.
5. Proses Klasifikasi dengan *Support Vector Machine*, merupakan proses pengklasifikasian antara citra melanoma dan citra bukan melanoma. Ada 2 proses yang dilakukan pada proses ini, yaitu :
 - a. Proses pelatihan (*training*), adalah proses untuk melatih sistem dengan data masukan yang telah diolah sebelumnya sehingga mampu mengenali apabila diberi masukan yang baru.

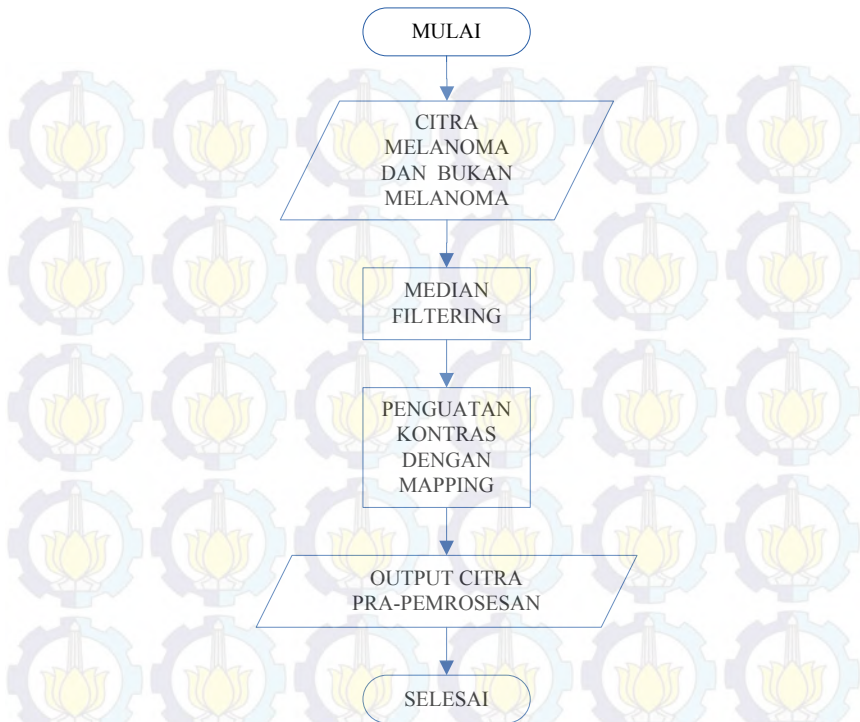
- b. Proses pengujian (*testing*), yaitu proses membandingkan fitur masukan baru dengan fitur yang ada pada referensi yang sebelumnya sudah dilatih kepada sistem.

4.2.1 Data Input

Data citra masukan dari proses ini adalah citra dermatoskopis yaitu citra melanoma dan citra bukan melanoma. Citra-citra tersebut akan diproses untuk menentukan apakah citra tersebut melanoma atau hanya citra luka yang tidak berbahaya.

4.2.2 Proses Pra-pemrosesan

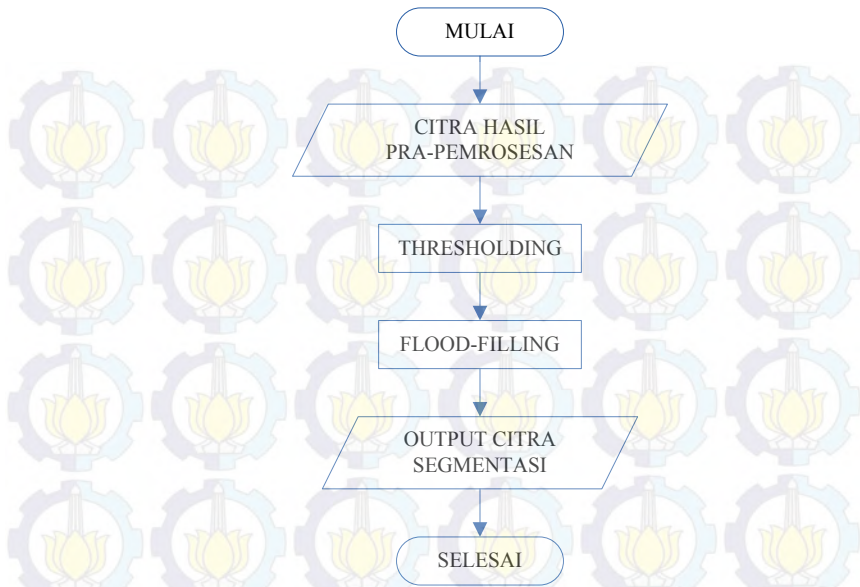
Proses pra-pemrosesan merupakan tahap pertama yang dilakukan sebelum memperoleh fitur dari suatu citra. Dalam pra-pemrosesan terdapat dua subproses yaitu median filtering dan mapping nilai intensitas citra. Gambar menunjukkan diagram alir proses pra-pemrosesan.



Gambar 4.1. Diagram Alir Proses Pra-pemrosesan.

4.2.3 Proses Segmentasi

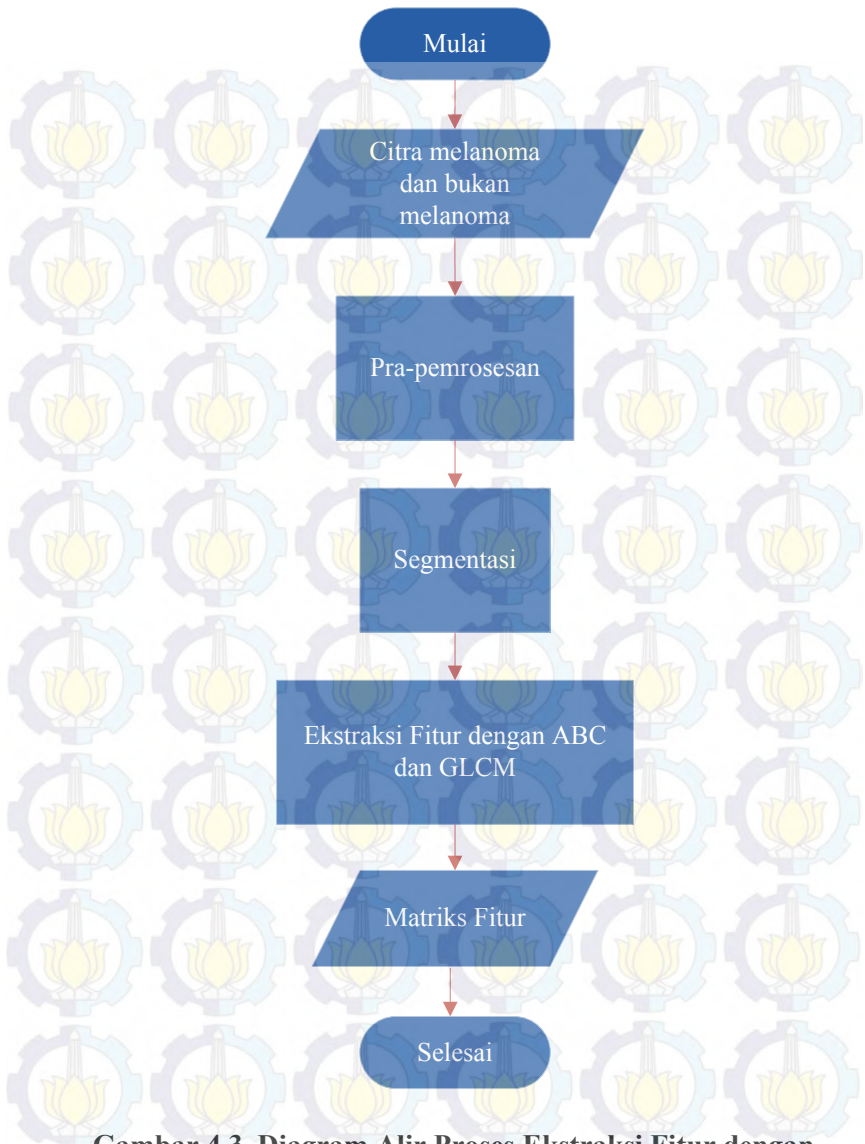
Setelah mendapatkan hasil keluaran dari proses pra-pemrosesan, maka citra output dari proses pra-pemrosesan tersebut diproses untuk mendapatkan daerah luka melalui proses segmentasi. Proses segmentasi dengan thresholding kemudian dengan *flood-filling*. Gambar menunjukkan diagram alir proses segmentasi.



Gambar 4.2. Diagram Alir Proses Segmentasi

4.2.4 Proses Ekstraksi Fitur ABC dan GLCM

Pada proses ekstraksi fitur ini akan dihitung masing-masing nilai ABC yaitu *Asymmetry*, *Border Irregularity*, *Color Variation*, dan GLCM yaitu Energi, Entropi, Kontras, Homogenitas. Gambar menunjukkan diagram alir dari ekstraksi fitur.



Gambar 4.3. Diagram Alir Proses Ekstraksi Fitur dengan ABC dan GLCM.

4.2.5 Proses Klasifikasi dengan SVM

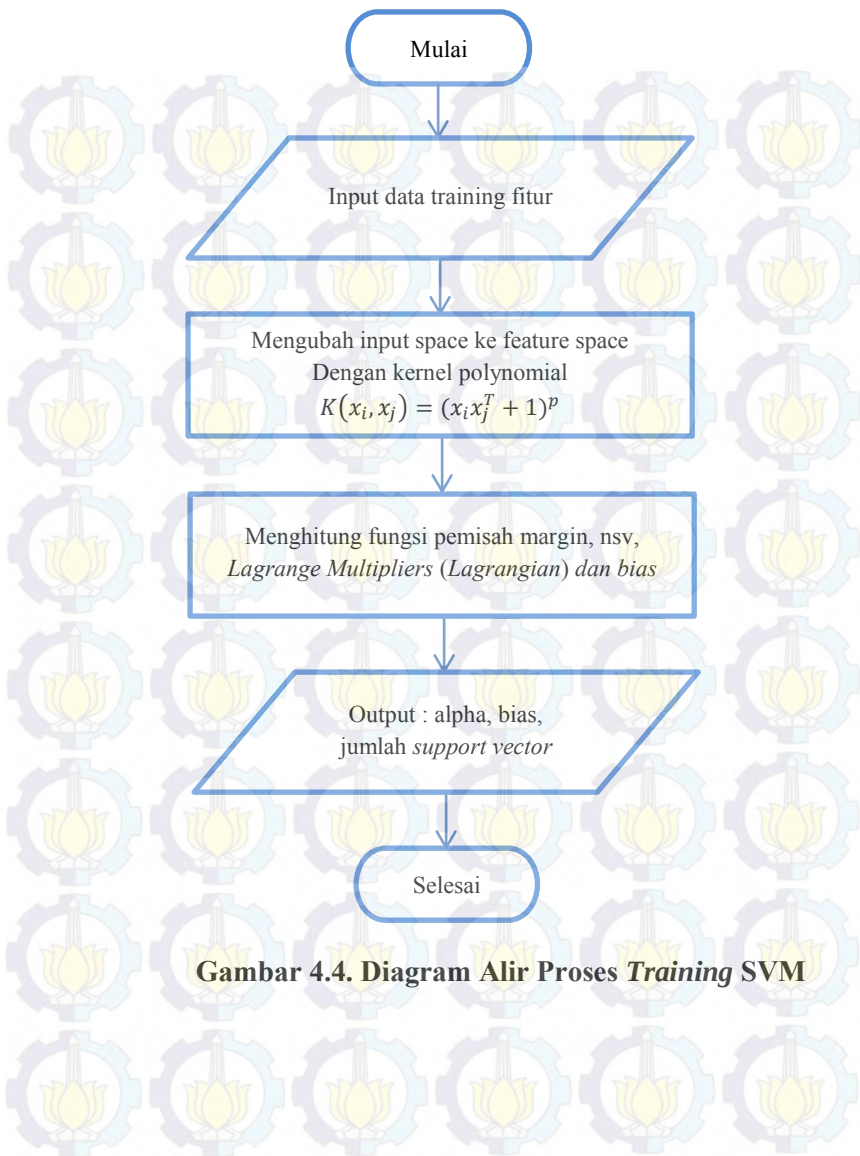
Tahapan setelah proses ekstraksi fitur dengan ABC dan GLCM adalah proses klasifikasi dengan metode *Support Vector Machine*. Proses klasifikasi ini dibagi menjadi dua proses. Proses pertama adalah proses *training* dari *dataset* berupa fitur yang telah direduksi matriksnya. dan proses kedua adalah proses *testing* data dimana akan di klasifikasi sejumlah citra *testing* yang telah ditentukan apakah citra tersebut citra berjenis melanoma atau bukan melanoma.

1. Proses *Training*

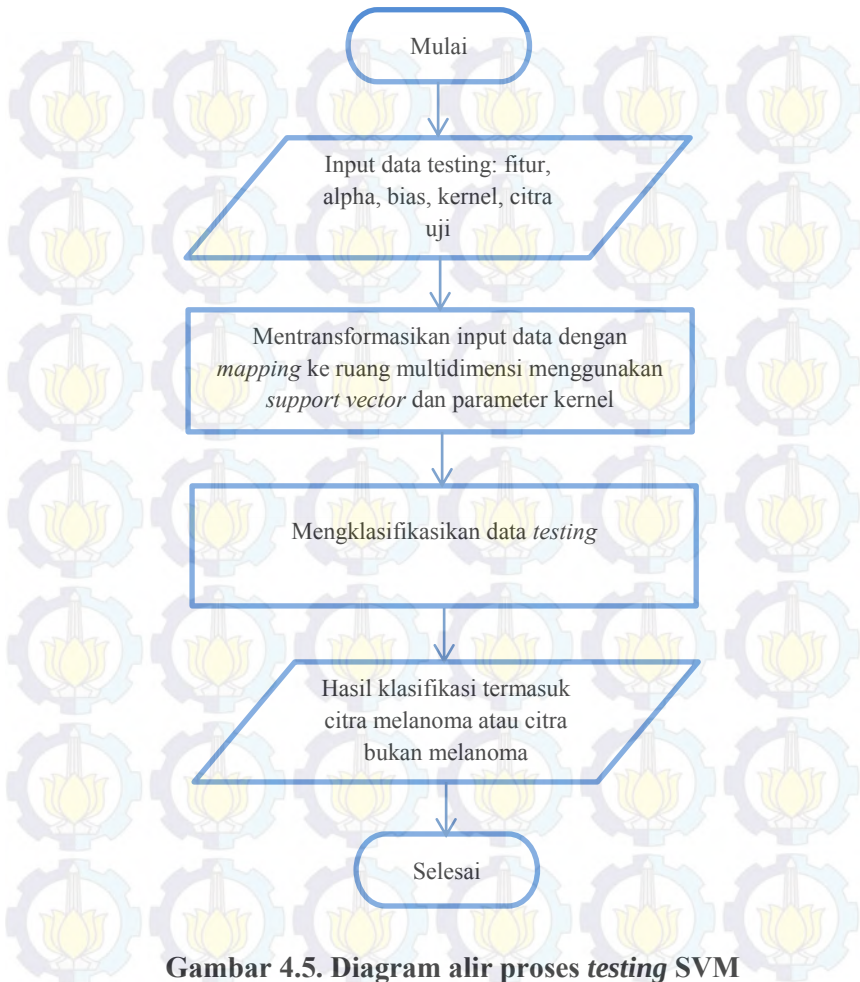
Proses ini merupakan proses dimana harus dilakukan *training dataset* yang bertujuan untuk menemukan fungsi pemisah, margin, jumlah *support vector*, *Lagrange Multipliers* (*Lagrangian*) dan bias. Parameter yang dicari antara lain α , b dan nsv . Sedangkan parameter yang diinputkan adalah fitur, target (dalam hal ini jenis citra) dan jenis kernel. Parameter α disini digunakan untuk menentukan jumlah dari *support vector*. *Support vector* merupakan suatu data yang terletak pada *hyperplane*, untuk menentukan *support vector* dapat dilihat dari nilai α yang melebihi nilai toleransi. Selain nilai α dalam proses ini juga dicari nilai margin dan bias (b). Nilai margin adalah jarak antara *hyperplane* tersebut dengan *pattern* terdekat dari masing masing *class*. Diagram alir proses *training* dapat dilihat pada Gambar 4.4.

2. Proses *Testing*

Pada proses sebelumnya telah diperoleh nilai α , bias, dan jumlah *support vector*. Proses selanjutnya adalah mengklasifikasi apakah suatu citra adalah citra melanoma atau citra bukan melanoma. Jika suatu citra yang di *testing* menghasilkan keluaran satu maka citra tersebut merupakan citra melanoma. Sedangkan jika citra tersebut menghasilkan output minus satu maka citra tersebut merupakan citra bukan melanoma. Diagram alir proses *testing* dapat dilihat pada Gambar 4.5.



Gambar 4.4. Diagram Alir Proses Training SVM



Gambar 4.5. Diagram alir proses *testing* SVM

4.3 Implementasi

Perancangan program yang telah dibangun selanjutnya diimplementasikan pada bahasa pemrograman dengan menggunakan *software* Matlab. Pembahasan dalam implementasi sistem meliputi implementasi antarmuka (*interface*) sistem,

implementasi tahap pra-pemrosesan, segmentasi citra, implementasi tahap ekstraksi fitur dengan ABC dan GLCM, dan terakhir tahap klasifikasi dengan metode SVM.

4.3.1 Implementasi Antarmuka

Pada tugas akhir ini, antarmuka sistem dibangun dengan menggunakan form dan kontrol yang terdapat pada Matlab. Adapun antarmuka-antarmuka yang diimplementasikan untuk menunjang penelitian tugas akhir ini adalah sebagai berikut.

4.3.1.1 Halaman Utama

Halaman utama merupakan antarmuka yang berisi menu-menu untuk menampilkan antarmuka-antarmuka lainnya dalam sistem. Hasil implementasi halaman utama dapat dilihat pada Gambar 4.6 berikut.



Gambar 4.6. Halaman Utama.

Halaman utama dibuat dalam bentuk yang sederhana, terdiri dari 3 bagian utama, diantaranya :

1. Bagian *title bar*
2. Bagian *menu bar*
3. Bagian *main window*

Title bar merupakan bagian yang menunjukkan judul dari antarmuka yang sedang ditampilkan. Di bawah *title bar* terdapat *menu bar* yang berisi sederetan menu-menu yang digunakan oleh sistem. Adapun kegunaan menu-menu yang ditampilkan pada antarmuka utama sistem disajikan dalam Tabel 4.1 berikut ini :

Tabel 4.1. Kegunaan Menu Sistem

Menu	Kegunaan
Pelatihan Manual	Melakukan pelatihan agar sistem dapat mengenali melanoma dan bukan melanoma melalui input citra secara manual.
Pelatihan Load Data	Melakukan pelatihan agar sistem dapat mengenali melanoma dan bukan melanoma dengan data masukan yang telah diproses.
Pengujian	Melakukan pengujian, pengamatan, & penilaian pada sistem, seberapa baik hasil yang didapatkan pada data masukan tertentu.
Keluar	Keluar dari sistem

Main window merupakan bagian antarmuka yang digunakan untuk menampilkan berbagai antarmuka lain di dalam sistem.

4.3.1.2 Antarmuka Pelatihan

Antarmuka Pelatihan berguna untuk melakukan pelatihan terhadap citra melanoma atau yang diduga melanoma. Pada antarmuka Pelatihan ini dibagi menjadi dua sub-menu, yakni antarmuka Pelatihan Manual dan Pelatihan Load Data. Pelatihan Manual Melakukan pelatihan agar sistem dapat mengenali melanoma dan bukan melanoma melalui *input* citra secara manual. Sedangkan Pelatihan *Load* Data melakukan pengujian, pengamatan, & penilaian pada sistem, seberapa baik hasil yang didapatkan pada data masukan tertentu. Sama seperti antarmuka pada halaman utama, antarmuka pelatihan manual maupun pelatihan *load* data juga memiliki menu file yang dapat memudahkan dalam proses uji coba. Hasil implementasi antarmuka pelatihan citra melanoma secara manual maupun pelatihan load data dapat dilihat pada Gambar 4.7 dan Gambar 4.8 berikut :



Gambar 4.7. Antarmuka Pelatihan Citra Kanker Kulit Melanoma Secara Manual



Gambar 4.8. Antarmuka Pelatihan Citra Kanker Kulit Melanoma dengan *Load Data*

Pada antarmuka Pelatihan Manual terdapat 5 buah tombol dan 1 *radio button* yang memiliki fungsi yang berbeda-beda. Fungsi dari tombol dan *radio button* tersebut adalah sebagai berikut :

1. Tombol input citra digunakan untuk mengambil citra melanoma atau yang diduga melanoma yang tersimpan pada komputer.
2. Tombol pra-pemrosesan untuk melakukan pemrosesan dan ekstraksi fitur ABC dan GLCM terhadap citra yang akan dilatih. Citra hasil pemrosesan tersebut akan ditampilkan pada *axes* yang telah dibuat pada halaman Pelatihan Manual.
3. Tombol simpan digunakan untuk melakukan penyimpanan citra hasil pemrosesan pada Pelatihan Manual. Penyimpanan dibedakan berdasarkan jenis citra melanoma atau citra bukan melanoma. apabila tombol simpan dipilih maka akan menyebabkan perubahan pada *textfield* jumlah data training.

4. Tombol training SVM digunakan untuk melatih data masukan yang telah diolah sebelumnya sehingga mampu mengenali apabila diberi masukan yang baru.
5. Tombol uji digunakan untuk menampilkan halaman pengujian.
6. *Radio button* jenis citra training digunakan untuk melakukan pilihan jenis citra yang akan disimpan. Pilihan jenis citra terdiri dari jenis citra melanoma dan bukan melanoma.

Sedangkan pada antarmuka Pelatihan Load Data, terdapat 3 tombol dan 1 *radio button* yang memiliki fungsi yang berbeda-beda pula. Berikut fungsi dari tombol dan *radio button* yang ada pada antarmuka Pelatihan Load Data.

1. Tombol registrasi digunakan untuk ekstraksi data dengan menggunakan metode ABC dan GLCM, sehingga didapatkan fitur yang akan digunakan untuk proses *training*.
2. Tombol training SVM digunakan untuk melatih data masukan yang telah diolah sebelumnya sehingga mampu mengenali apabila diberi masukan yang baru.
3. Tombol uji digunakan untuk menampilkan halaman pengujian.
4. *Radio button* pilih jumlah data training digunakan untuk memilih banyaknya data yang akan dilatih dengan metode ABC dan GLCM. Pilihan jumlah data yang akan dilatih adalah 64 citra melanoma dan 56 citra bukan melanoma, 56 citra melanoma dan 49 citra bukan melanoma, 48 citra melanoma dan 42 citra bukan melanoma

4.3.1.3 Antarmuka Pengujian

Antarmuka Pengujian digunakan untuk melakukan pengujian terhadap sistem berdasarkan algoritma SVM. Hasil implementasi antarmuka Pengujian dapat dilihat pada Gambar 4.9 berikut :



Gambar 4.9. Antarmuka Pengujian Citra Kanker Kulit Melanoma

Pada antarmuka Pengujian terdapat tiga tombol yang memiliki fungsi hampir sama pada antarmuka Pelatihan Manual. Fungsi dari tombol-tombol tersebut adalah sebagai berikut :

1. Tombol input citra digunakan untuk mengambil citra melanoma atau yang diduga melanoma yang tersimpan pada komputer.
2. Tombol pra-pemrosesan untuk melakukan pemrosesan dan ekstraksi fitur ABC dan GLCM terhadap citra yang akan diuji. Citra hasil pemrosesan tersebut akan ditampilkan pada *axes* yang telah dibuat pada halaman Pengujian.
3. Tombol Pengujian digunakan untuk melakukan pengujian citra yang telah dipilih sebelumnya dengan menggunakan metode SVM, sehingga didapatkan keputusan apakah citra yang diuji berjenis melanoma atau bukan melanoma. Hasil pengujian tersebut akan ditampilkan pada *textfield* Hasil Pengujian

4.3.2 Implementasi Tahap Input Data

Proses input data membutuhkan interaksi pengguna untuk mencari dan mengambil citra yang dibutuhkan didalam media penyimpanan. Proses ini diimplementasikan menjadi sebuah program ke dalam fungsi yang telah tersedia di matlab. Fungsi dijalankan adalah ketika tombol “Ambil Citra” dipilih.

4.3.3 Implementasi Tahap Pra-pemrosesan

Pada tugas akhir ini, tahap pra-pemrosesan terdiri dari beberapa tahap yang mempengaruhi proses selanjutnya serta hasil yang didapatkan. Dalam pra-pemrosesan ini citra difilter menggunakan median filter untuk menghilangkan noise. Setelah itu tahap penguatan nilai kekontrasan citra dengan *mapping* nilai intensitas.

4.3.3.1 Implementasi Proses *filtering* dengan Median Filter

Proses ini bertujuan untuk menghilangkan noise yang ada pada citra. Proses tersebut diimplementasikan pada sebuah program sebagai berikut :

```
citra2 = citra1;  
for i = 1 : 3  
    citra2(:,:,i) = medfilt2(citra1(:,:,i));  
end
```

4.3.3.2 Implementasi Proses Penguatan Nilai Kekontrasan Citra dengan *Mapping* Nilai Intensitas

Proses ini bertujuan untuk menguatkan nilai kekontrasan citra, sehingga perbedaan antara daerah luka dengan daerah kulit sekitar luka semakin besar. Proses tersebut diimplementasikan pada sebuah program sebagai berikut :


```

citra3=rgb2gray(citra2);
citra4=imadjust(citra3, stretchlim(citra3),[]);

```

Hasil dari proses ini adalah jelasnya perbedaan antara daerah luka dan kulit sehingga mempermudah deteksi tepi dari luka.

4.3.4 Implementasi Tahap Segmentasi

Pada proses segmentasi ini, dilakukan pengubahan citra *gray* hasil pra-pemrosesan menjadi citra biner, dengan menentukan nilai *threshold* yang baik (menggunakan metode *thresholding*) sehingga citra biner yang dihasilkan dapat membedakan daerah luka dan daerah kulit dengan baik. Kemudian dilakukan metode *flood-filling* untuk menghilangkan piksel-piksel yang terisolasi pada daerah luka maupun daerah kulit.

4.3.4.1 Implementasi Proses Segmentasi dengan *Thresholding* Otsu

Proses *thresholding* Otsu adalah proses segmentasi citra yang bertujuan untuk memisahkan antara luka dan kulit. Proses tersebut diimplementasikan menjadi sebuah program kedalam fungsi berikut :

```

level=graythresh(citra4);

```

Fungsi *graythresh* merupakan fungsi pada matlab yang digunakan untuk menentukan *threshold* dengan menggunakan metode Otsu. Sedangkan variabel *level* merupakan *threshold* yang didapatkan dari *thresholding* Otsu. Setelah didapatkan *threshold* dari *thresholding* Otsu, fungsi yang digunakan adalah mendapatkan citra biner hasil *thresholding* Otsu yang disajikan pada fungsi berikut :

```

citra5=im2bw(citra4,level);

```

4.3.4.2 Implementasi Proses *Flood-Filling*

Flood-filling digunakan untuk menghilangkan piksel-piksel yang terisolasi pada daerah luka maupun pada daerah kulit. Proses tersebut diimplementasikan menjadi sebuah program kedalam fungsi berikut :

```
citra8 = imfill(citra7, 'holes');
```

Variable *citra7* merupakan variable hasil proses *thresholding otsu*. Kode program selengkapnya dapat dilihat pada lampiran A1.

4.3.5 Implementasi Tahap Ekstraksi Fitur dengan ABC dan GLCM

Proses selanjutnya adalah tahap ekstraksi fitur dengan ABC dan GLCM. Proses ini merupakan proses registrasi atau pembuatan fitur-fitur sebagai pengetahuan untuk sistem agar nantinya dapat mengklasifikasikan citra uji dengan baik. Proses registrasi ini diimplementasikan menjadi sebuah program ke dalam fungsi :

```
function [ hasilpre ] = ekstraksi( im )
```

Luaran dari fungsi ini adalah matriks fitur dari ABC yaitu *asymmetry*, *Border Irregularity*, *Color Variation*, dan GLCM yaitu entropi, energy, kontras, homogenitas. Kode program selengkapnya dapat dilihat pada lampiran A2.

4.3.6 Implementasi Tahap Klasifikasi Citra dengan SVM

Setelah dilakukan proses registrasi, maka proses selanjutnya adalah proses klasifikasi. Klasifikasi dengan metode SVM ini terdiri dari dua proses yaitu proses *training* dan proses *testing*.

4.3.6.1 Proses *Training*

Merupakan proses untuk menemukan fungsi pemisah, margin, jumlah *support vector*, *Lagrange Multipliers* (*Lagrangian*) dan

bias. Parameter yang dicari antara lain α , b dan nsv . Sedangkan parameter yang diinputkan adalah fitur, label/target (dalam hal ini jenis peluru) dan jenis kernel.

Proses training SVM ini diimplementasikan dengan fungsi matlab:

```
svm_struct = svmtrain(A, B,
    'Kernel_Function', 'poly');
```

Kode program selengkapnya dapat dilihat pada lampiran A3.

4.3.6.2 Proses Testing

Proses *testing* ini adalah mengklasifikasi apakah suatu citra adalah citra melanoma atau bukan melanoma. Jika suatu citra yang di testing menghasilkan keluaran satu maka citra tersebut merupakan citra melanoma. Sedangkan jika citra tersebut menghasilkan output minus satu maka citra tersebut merupakan citra bukan melanoma.

Proses pengujian diimplementasikan menjadi sebuah program kedalam fungsi testing berikut :

```
hasil = svmclassify(svm_struct, [AA BB CC
    glcm])

    if hasil == 1
        set(handles.hasil, 'String',
    'Melanoma');
        fprintf('Melanoma');
        fprintf('\n');
    else
        set(handles.hasil, 'String', 'Bukan
    Melanoma');
        fprintf('Bukan melanoma');
        fprintf('\n');
    end
```

Kode program selengkapnya dapat dilihat pada lampiran A4.

BAB V

PENGUJIAN DAN PEMBAHASAN HASIL

Bab ini menjelaskan mengenai proses pengujian yang dilakukan terhadap sistem klasifikasi citra melanoma dan bukan melanoma dengan menggunakan metode *Support Vector Machine*. Hasil pengujian kemudian dibahas untuk mengetahui unjuk kerja sistem secara keseluruhan dalam menjalankan fungsi yang diinginkan. Uji coba dimulai dengan menguji sistem pemrosesan citra, dengan tujuan mendapatkan citra yang tersegmentasi. Setelah itu uji coba dilakukan pada dataset, dengan menjalankan proses pembelajaran sistem dan juga pelatihan sistem, dengan tujuan untuk mengukur performa *Support Vector Machine*.

5.1 Lingkungan Pengujian Sistem

Lingkungan pengujian dari sistem klasifikasi citra melanoma dan bukan melanoma meliputi perangkat keras dan lunak komputer. Detail dari perangkat keras dan lunak yang digunakan dapat dilihat pada Tabel 5.1

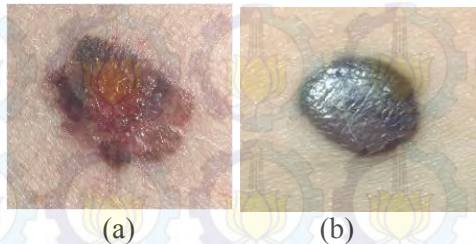
Tabel 5.1. Lingkungan Pengujian Sistem

Perangkat Keras	Prosesor : Intel(R) Core(TM) i5-2410M CPU @ 2.30GHz 2.30GHz
	Memory : 4.00 GB
Perangkat Lunak	Sistem Operasi : Windows 8 Pro 32-bit
	Tools : MATLAB R2013a

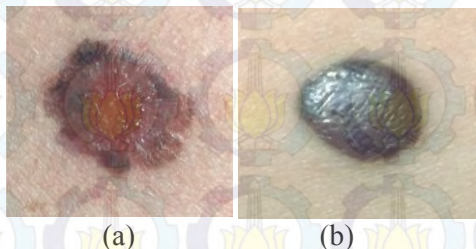
5.2 Pengujian Tahap Pra-pemrosesan

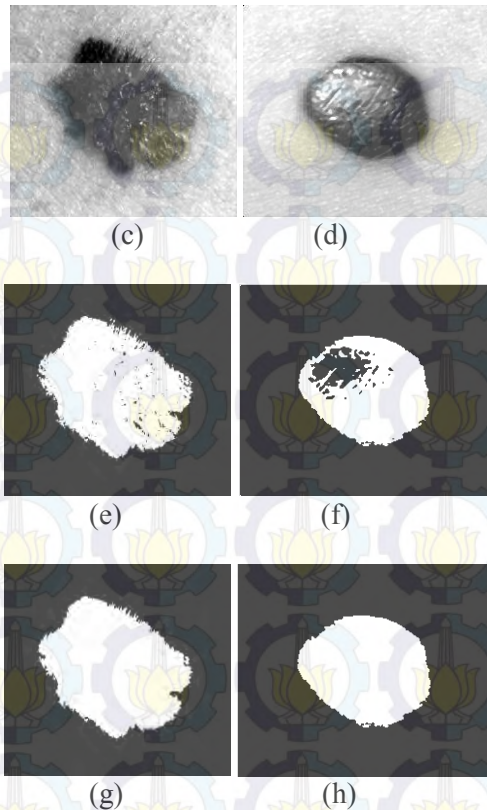
Pengujian ini dilakukan terhadap proses-proses pada tahap pengolahan citra. Pengujian bertujuan untuk mengetahui bahwa proses-proses pada tahap pengolahan citra sudah benar. Pada tahap ini terdapat 4 proses yaitu : median *filtering*, *mapping* nilai intensitas, segmentasi citra dengan *thresholding* otsu, dan *flood filling*.

1. Pengujian proses median *filtering*
Pengujian proses ini bertujuan untuk mengetahui bahwa sistem telah berhasil menghapus *noise* pada citra.
2. Pengujian proses *mapping* nilai intensitas
Pengujian proses ini bertujuan untuk mengetahui bahwa sistem telah berhasil menguatkan nilai kontrasan citra.
3. Pengujian proses segmentasi citra dengan *thresholding* otsu
Pengujian proses ini bertujuan untuk mengetahui bahwa system telah berhasil memisahkan antara *foreground* dan *background*.
4. Pengujian proses *flood filling*
Pengujian proses ini bertujuan untuk menghilangkan piksel-piksel yang terisolasi.



Gambar 5.1. Citra input (a) citra melanoma, (b) citra bukan melanoma





Gambar 5.2. Hasil tahap pra-pemrosesan

(a) Median *filtering* citra melanoma, (b) median *filtering* citra bukan melanoma, (c) *mapping* nilai intensitas citra melanoma, (d) *mapping* nilai intensitas citra bukan melanoma, (e) segmentasi citra melanoma, (f) segmentasi citra bukan melanoma, (g) *flood filling* citra melanoma, (h) *flood filling* citra bukan melanoma

5.3 Pengujian Tahap Ekstraksi Fitur

Pengujian proses ini menggunakan data citra yang telah diproses pada pra-pemrosesan. Tahap ekstraksi fitur ini

menggunakan metode ABC yaitu *Asymmetry*, *Border Irregularity*, *Colour Variation* dan GLCM yaitu entropi, energi, kontras, homogenitas. Data pengujian dari proses ekstraksi fitur dengan citra input pada gambar 5.1 dapat dilihat pada tabel 5.2 berikut:

Tabel 5.2. Hasil Proses Ekstraksi Fitur dengan ABC dan GLCM

Ekstraksi Fitur		Citra Melanoma	Citra Bukan Melanoma
ABC	<i>Asymmetry</i>	0.3016	0.3968
	<i>Border Irregularity</i>	0.1515	0.1764
	<i>Colour Variation</i>	0.1334	0.2147
GLCM	Entropi	3.6171	3.4156
	Energi	0.0005	0.0008
	Kontras	78.6318	110.1517
	Homogenitas	0.3348	0.3416

5.4 Pengujian Tahap Klasifikasi dengan SVM

5.4.1 Proses *Training*

Pengujian sistem menggunakan 80 data melanoma dan 70 data bukan melanoma. Data dibagi menjadi data pelatihan dan data uji. Data pelatihan dibagi ke dalam 3 kelompok data, yakni :

1. Data pertama yang terdiri dari 64 citra melanoma dan 56 citra bukan melanoma
2. Data kedua yang terdiri dari 56 citra melanoma dan 49 citra bukan melanoma
3. Data ketiga yang terdiri dari 48 citra melanoma dan 42 citra bukan melanoma

Sedangkan data uji merupakan sisa dari jumlah keseluruhan data yang tidak digunakan untuk data *training*.

Pengujian proses ini bertujuan untuk mengetahui data yang diperlukan dalam proses berikutnya seperti alpha, bias, jumlah support vector, kernel dan sebagainya. Pengujian proses ini menggunakan fitur yang telah diperoleh dari proses ekstraksi fitur dengan ABC dan GLCM.

Data luaran yang dihasilkan akan disimpan dalam file matlab dengan format .mat. Data-data yang disimpan akan berbentuk informasi seperti yang terlihat pada Tabel 5.3.

Tabel 5.3. Hasil Proses Training SVM

<i>Field</i>	<i>Value</i>	<i>Min</i>	<i>Max</i>
<i>SupportVectors</i>	<i>78x7 double</i>	<i>-2.2301</i>	<i>4.4690</i>
<i>Alpha</i>	<i>78x1 double</i>	<i>-0.9375</i>	<i>1.3333</i>
<i>Bias</i>	<i>0.673927759565599</i>	<i>0.6739</i>	<i>0.6457</i>
<i>KernelFunction</i>	<i>@poly kernel</i>		
<i>KernelFunctionArgs</i>	<i>0x0 cell</i>		
<i>GroupNames</i>	<i>120x1 double</i>	<i>-1</i>	<i>1</i>
<i>SupportVectorIndices</i>	<i>78x1 double</i>	<i>1</i>	<i>120</i>
<i>ScaleData</i>	<i>1x1 struct</i>		
<i>FigureHandles</i>	<i>[]</i>		

Tabel diatas merupakan hasil proses *training* SVM dimana contoh yang diambil adalah data *training* sejumlah 64 citra melanoma dan 56 citra bukan melanoma dengan parameter kernel polynomial. Informasi ini disimpan dalam file matlab sebagai pengetahuan sistem untuk proses berikutnya.

5.4.2 Proses Testing

Pengujian sistem menggunakan data uji yang merupakan sisa dari jumlah keseluruhan data yang tidak dipakai untuk training. Pengujian proses ini untuk menentukan apakah suatu citra yang diuji merupakan citra melanoma atau citra bukan melanoma. hasil dari pengujian system dapat dilihat pada Tabel 5.4

Tabel 5.4. Pengaruh Banyak Data *Training* Terhadap Pengujian dan Waktu Komputasi menggunakan Kernel Polynomial.

Data Latih	Akurasi	Waktu Komputasi (detik)	
		<i>Training</i>	<i>Testing</i>
1	90%	71.67 detik	0.513 detik
2	88.88%	62.02 detik	0.520 detik
3	73.33%	52.73 detik	0.516 detik

Berdasarkan Tabel 5.4 dengan parameter kernel polynomial dapat dilihat bahwa banyaknya data pelatihan dapat mempengaruhi hasil pengujian atau akurasi. Data 1 yaitu data pelatihan 64 citra melanoma dan 56 citra bukan melanoma dengan waktu komputasi 71.67 detik menghasilkan akurasi paling tinggi, yaitu 90%

Dengan menggunakan data pelatihan data 1 yaitu 64 citra melanoma dan 56 citra bukan melanoma pengujian selanjutnya adalah dengan menggunakan parameter yang berbeda yaitu parameter linear, rbf, dan polynomial (dapat dilihat pada Tabel 5.4). Hasil dari pengujian ini dapat dilihat pada Tabel 5.5.

Tabel 5.5. Pengaruh Parameter Polynomial, Linear dan Rbf Terhadap Pengujian dan Waktu Komputasi dengan menggunakan Data Latih 1

Kernel	Akurasi	Waktu Komputasi (detik)	
		Training	Testing
Polynomial	90%	71.67 detik	0.513 detik
Linear	80%	69.69 detik	0.512 detik
Rbf	73.33%	67.55 detik	0.528 detik

Berdasarkan Tabel 5.5 dengan data pelatihan 64 citra melanoma dan 56 citra bukan melanoma, parameter polynomial menghasilkan tingkat akurasi paling tinggi yaitu 90%.

5.4 Pembahasan Hasil Pengujian

Pembahasan hasil pengujian difokuskan pada hasil pengujian dan proses pengenalan citra kanker kulit melanoma. Pada tugas akhir ini pembahasan hasil pengujian ini digunakan untuk mengetahui kinerja sistem klasifikasi citra melanoma dan bukan melanoma menggunakan metode SVM. Dari hasil yang telah didapatkan pada subbab sebelumnya, sistem mampu mengenali citra melanoma dan bukan melanoma dengan cukup baik. Pada subbab ini akan dijelaskan cara menghitung tingkat akurasi sistem dengan menggunakan Persamaan 5.1.

$$\text{Akurasi} = \frac{\text{jumlah citra baik benar} + \text{jumlah citra cacat benar}}{\text{jumlah citra uji}} \times 100 \quad (5.1)$$

5.5 Pembahasan Penyebab Besar Kecilnya Akurasi

Penyebab utama rendahnya akurasi sistem klasifikasi citra kanker kulit melanoma dengan SVM antara lain :

1. Terletak pada kualitas citra yang diakuisisi kurang begitu baik. Hal tersebut dikarenakan terlalu banyak *noise* dan perbedaan intensitas cahaya yang tidak merata. Sehingga proses segmentasi kurang maksimal.
2. Jumlah Banyaknya Data *Training*, semakin banyak data *training* yang di latih, maka semakin banyak informasi yang tersimpan. Sehingga ketika dilakukan pengujian, akurasi yang diberikan lebih tinggi.

LAMPIRAN A

A1. Kode Program Untuk *Flood Filling*

```
dim = size(citra5);  
citra6 = xor(citra5, ones(dim));  
  
baris = round(dim(1)/2);  
kolom = find(citra6(baris,:), 1);  
tepi = bwtraceboundary(citra6, [baris, kolom],  
    'w');  
n = size(tepi);  
citra7 = zeros(dim);  
i = 0;  
while i < n(1)  
    i = i+1;  
    citra7(tepi(i,1), tepi(i,2)) = 1;  
end
```


LAMPIRAN A (LANJUTAN)

A2. Kode Fungsi Pelatihan dengan metode ABC dan GLCM

```
function [ hasilpre ] = ekstraksi( im )
[xx yy] = find(citra8 == 1);
M00 = sum(citra8(:));
M01 = findImageMoment(citra8, 0, 1);
M11 = findImageMoment(citra8, 1, 1);
M10 = findImageMoment(citra8, 1, 0);
M20 = findImageMoment(citra8, 2, 0);
M02 = findImageMoment(citra8, 0, 2);
xbar = M10 / M00;
ybar = M01 / M00;
miu11 = M11-xbar*M01;
miu20 = M20-xbar*M10;
miu02 = M02-ybar*M01;
miuaks20 = miu20/M00;
miuaks02 = miu02/M00;
miuaks11 = miu11/M00;

Asi = 2*miuaks11/(miuaks20-miuaks02);
teta = 0.5*atan(Asi);
asym = 0.5*((teta/M00)+((teta+pi/2)/M00));

lamda = (miuaks20+miuaks02-sqrt((miuaks20-
miuaks02)+4*(miuaks11^2)))/2;
lamda2 = (miuaks20+miuaks02+sqrt((miuaks20-
miuaks02)+4*(miuaks11^2)))/2;
lengt = lamda/lamda2;

PL = find(citra7 == 1);
[row col] = size(PL);
compact = (power(row,2))/(4*pi*M00);
rr = 9;
lamda = 1;
fractal = fractalDimension(citra7, rr, lamda);
```

LAMPIRAN A (LANJUTAN)

```

dd = zeros(dim);
    for i=1:dim(1)
        for j=1:dim(2)
            if citra7(i,j)==1
                Phyta = (i-ybar)^2+(j-xbar)^2;
                dd(i,j) = sqrt(Phyta);
            end
        end
    end
ddd = dd(find(dd));
d = ddd(:);
md = mean(d);
matriksAtasRumus = d-ones(row,1)*md;
    for i=1:row
        matriksAtasRumus(i) =
matriksAtasRumus(i)^2;
    end
jmlMatriksAtas = sum(matriksAtasRumus);
Edge = ((1/row)*jmlMatriksAtas)/md^2;

DeltaF = sqrt((tepi(1)-xbar)^2+(tepi(2)-
ybar)^2);
[aaa bbb] = size(tepi);
me = sum(DeltaF)/aaa;
ve = (sum((DeltaF)^2)-me^2)/aaa;

I = citral;
[r c k] = size(I);
ICh = zeros(r,c,k);
threshold = zeros(k,2);
    for i=1:k
        threshold(i,1) = max(max(I(:, :, i)));
        threshold(i,2) = min(min(I(:, :, i)));
    end
    for o=1:k
        for i=1:r
            for j=1:c

```

LAMPIRAN A (LANJUTAN)

```

        if(I(i,j,o) > threshold(o,2) &&
I(i,j,o) <= threshold(o,2) + ((threshold(o,1)-
threshold(o,2))/3))
            ICh(i,j,o) = 1;
        end
        if(I(i,j,o) > threshold(o,2) +
((threshold(o,1)-threshold(o,2))/3) &&
I(i,j,o) <= threshold(o,2)+2*((threshold(o,1)-
threshold(o,2))/3))
            ICh(i,j,o) = 2;
        end
        if(I(i,j,o) > threshold(o,2) +
2*((threshold(o,1)-threshold(o,2))/3) &&
I(i,j,o) <= threshold(o,1))
            ICh(i,j,o) = 3;
        end
    end
end
end
transition = zeros(k,2,2);
for o=1:k
    for i=1:r
        for j=1:c
            if (j~=c)
                if(ICh(i,j,o)==2 &&
ICh(i,j+1,o)==1)
                    transition(o,1,1)=transition(o,1,1)+1;
                end
                if(ICh(i,j,o)==1 &&
ICh(i,j+1,o)==2)
                    transition(o,2,1) =
transition(o,2,1) + 1;
                end
            end
        end
    end
end
for j=1:c

```


LAMPIRAN A (LANJUTAN)

```

for i=1:r
    if (i~=r)
        if (Ich(i,j,o)==2&&Ich(i+1,j,o)==1)
            transition(o,1,2)=transition(o,1,2)+1;
        end
        if (Ich(i,j,o)==1&&Ich(i+1,j,o)==2)
            transition(o,2,2)=transition(o,2,2)+1;
        end
    end
end
end
for i=1:k
    Ch(i) =
        abs((transition(i,1,1)/transition(i,1,2))-
            (transition(i,1,2)/transition(i,1,1)));
    end
    cchh = (Ch(1)+Ch(2)+Ch(3))/3;

    lum = zeros(dim);
    for i=1:dim(1)
        for j=1:dim(2)
            if I(i,j)>=1

                lum(i,j)=1/3*(citra1(i,j,1)+citra1(i,j,2)+citra1
                    (i,j,3));
                Phyta = (i-ybar)^2+(j-xbar)^2;
                dp(i,j) = sqrt(Phyta);
            end
        end
    end
    lum = lum(find(lum));
    lum = lum(:);
    [jmlP kol] = size(lum);

```

LAMPIRAN A (LANJUTAN)

```

ml = mean(lum);
matriksAtasLum = lum-ones(jmlP,1)*ml;
dp(round(ybar),round(xbar)) = 0.1;
dp = dp(find(dp));
dp = dp(:);
mdp = mean(dp);

matriksAtasDp = dp-ones(size(dp,1),1)*mdp;
vd = var(dp);
v1 = var(lum);
for i=1:jmlP
    matriksAtasCpg(i) =
    matriksAtasLum(i)*matriksAtasDp(i);
end
jmlMatriksAtasCpg = sum(matriksAtasCpg);
Cpg = jmlMatriksAtasCpg/(vd*v1*M00);

AA = (lengt+asym)/2;
AA = real(AA);
BB = (compact+fractal+me+ve+Edge)/5;
CC = (Cpg+cchh)/2;
glcm = cooccxture(citra3, 1,1, 256);
hasilpre=[AA BB CC glcm];

end

```

LAMPIRAN A (LANJUTAN)

```
function m = coocccxture(f,dx,dy,L)
mat = matcoocc(f,dx,dy,L);
mat = mat/sum(sum(mat));
mat0 = find(mat==0);
matlog = mat;
matlog(mat0) = 1;
m(1) = -sum(sum(mat.*log10(matlog)));
m(2) = sum(sum(mat.^2));
[a,b] = meshgrid(1:L, 1:L);
m(3) = sum(sum((b-a).^2.*mat));
m(4) = sum(sum(mat./(1+abs(b-a))));
```


LAMPIRAN A (LANJUTAN)

A3. Kode Program untuk *Training SVM*

```
svm_struct = svmtrain(A, B, 'Kernel_Function',  
'Poly');  
save('SVMstruct.mat', 'svm_struct');
```

A4. Kode Program untuk *Testing SVM*

```
hasil = svmclassify(svm_struct, [AA BB CC glcm])  
  
if hasil == 1  
    set(handles.hasil, 'String',  
    'Melanoma');  
    fprintf('Melanoma');  
    fprintf('\n');  
else  
    set(handles.hasil, 'String', 'Bukan  
Melanoma');  
    fprintf('Bukan melanoma');  
    fprintf('\n');  
end
```

LAMPIRAN B

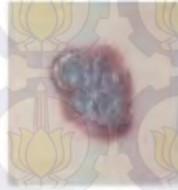
B.1 Contoh Citra Melanoma



melanoma3



melanoma12



melanoma35



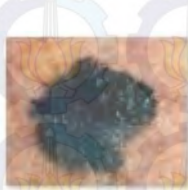
melanoma41



melanoma74

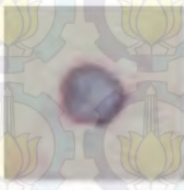


melanoma13



melanoma18

B.2 Contoh Citra Bukan Melanoma



notmelanoma1



notmelanoma18



notmelanoma20



notmelanoma26



notmelanoma15



notmelanoma5



notmelanoma8

DAFTAR LAMPIRAN

	Halaman
LAMPIRAN A	59
A.1 Kode Program untuk <i>Flood Filling</i>	59
A.2 Kode Fungsi Pelatihan dengan Metode ABC dan GLCM	60
A.3 Kode Program untuk <i>Training SVM</i> ...	66
A.4 Kode Program untuk <i>Testing SVM</i>	66
LAMPIRAN B	67
B.1 Contoh Citra Melanoma	67
B.2 Contoh Citra Bukan Melanoma	67

DAFTAR PUSTAKA

- [1] Immagulate, I., Vijaya, M.S. 2015. "*Categorization of Non-Melanoma Skin Lesion Diseases Using Support Vector Machine and Its Variants*". International Journal of Medical Imaging 3(2): 34-40.
- [2] Jaleel, J.A., Salim, S., Aswin, R.B. 2013. "*Computer Aided Detection of Skin Cancer*". International Conference on Circuits, Power and Computing Technologies.
- [3] Novianti, F.A., Purnami, S.W. 2012. "Analisis Diagnosis Pasien Kanker Payudara Menggunakan Regresi Logistik dan Support Vector Machine (SVM) Berdasarkan Hasil Mamografi". JURNAL SAINS DAN SENI ITS Vol. 1, No. 1, ISSN: 2301-928X
- [4] Listia, R., Harjoko, A. 2014. "Klasifikasi Massa pada Citra Mammogram Berdasarkan *Gray Level Cooccurrence Matrix* (GLCM)". IJCCS, Vol.8, No.1, pp. 59~68
- [5] Herdha, Tyara (2015). "Klasifikasi Kecatatan Peluru dengan Menggunakan Metode *Support Vector Machine* (SVM)" Tugas Akhir Jurusan Matematika ITS.
- [6] M. Rinaldi (2004). "Pengolahan Citra Digital dengan Pendekatan Algoritmik" Teknik Informatika ITB. Bandung.
- [7] Gonzales, R.C. dan Woods, Richard E. (2010). "*Digital Image Processing*". New Jersey. Prentice Hall.
- [8] Barhoumi W, Zagrouba E. 2002. "*A PRELIMINARY APPROACH FOR THE AUTOMATED RECOGNITION OF MALIGNANT MELANOMA*"
- [9] Immagulate I., Vijaya M. S. 2015. "*Categorization of Non-Melanoma Skin Lesion Diseases Using Support Vector Machine and Its Variants*". International Journal of Medical Imaging.
- [10] Santosa, Budi. (2007). "Data Mining, Teknik Pemanfaatan Data untuk Keperluan Bisnis". Yogyakarta. Graha Ilmu.

BIODATA PENULIS



Penulis bernama lengkap Kusuma Wahaninggar, yang biasa dipanggil Uus. Penulis dilahirkan di Ponorogo pada tanggal 30 Agustus 1992. Penulis lulus dari SDN 2 Mangkujayan Ponorogo, SMPN 1 Ponorogo, SMAN 2 Ponorogo dan melanjutkan pendidikan di Matematika ITS pada tahun 2011.

Semenjak kuliah penulis hobby mempelajari bahasa pemrograman. Bahasa Pemrograman yang pernah penulis pelajari adalah C, C++, dan Java. Oleh karena itu, penulis memilih bidang minat Ilmu Komputer.

Semasa menempuh jenjang pendidikan S-1, penulis juga aktif dalam kegiatan non-akademis diantaranya aktif di organisasi kemahasiswaan Matematika ITS. Untuk mendapatkan informasi yang berhubungan dengan Tugas Akhir ini dapat ditujukan ke alamat email : Kusumawahaninggar@gmail.com