

**APLIKASI SEGMENTASI COMPUTER MADE-IMAGE
MENGUNAKAN *HOMOTOPY TREE***

SKRIPSI

Oleh :

NURFAN DZIHAN

07650068



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2013**

**APLIKASI SEGMENTASI KOMPUTER – MADE IMAGE MENGGUNAKAN
*HOMOTOPY TREE***

SKRIPSI

**Diajukan Kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri
Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S. Kom)**

Oleh:

**NURFAN DZIHAN
NIM. 07650068**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2013**

**APLIKASI SEGMENTASI COMPUTER MADE-IMAGE
MENGUNAKAN HOMOTOPY TREE**

SKRIPSI

Oleh:

NURFAN DZIHAN

NIM. 07650068

Telah Diperiksa dan Disetujui untuk Diuji

Tanggal : 4 Juli 2013

Dosen Pembimbing I

Dosen Pembimbing II

M. Amin Hariyadi, M.T

Zainal Abidin, M.Kom

NIP. 19670018 200501 1 001

NIP. 19760613 200501 1 004

Mengetahui

Ketua Jurusan Teknik Informatika

Ririen Kusumawati, M.Kom

NIP. 19720309 200501 2 002

**APLIKASI SEGMENTASI COMPUTER MADE-IMAGE
MENGUNAKAN HOMOTOPY TREE**

SKRIPSI

**Oleh:
NURFAN DZIHAN
NIM. 07650068**

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan untuk
Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal : 11 Juli 2013

Susunan Dewan Penguji	Tanda Tangan
1. Penguji Utama : Irwan Budi Santoso, M.Kom NIP. 19770103 201101 1 004	()
2. Ketua : Dr. Cahyo Crysdian , M.Cs NIP. 19740424 200901 1 008	()
3. Sekretaris : M. Amin Hariyadi, M.T NIP. 19670018 200501 1 001	()
4. Anggota : Zainal Abidin, M.Kom NIP. 19760613 200501 1 004	()

**Mengetahui dan Mengesahkan
Ketua Jurusan Teknik Informatika**

Ririen Kusumawati, M.Kom
NIP. 19720309 200501 2 002

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Nurfan Dzihan

NIM : 07650068

Fakultas/Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Penelitian : **Aplikasi Segmentasi Computer Made-Image Menggunakan Homotopy Tree**

Menyatakan dengan sebenarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat orang lain, kecuali yang secara tertulis dikutip dalam naskah ini atau disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur jiplakan maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 10 Juli 2013

Yang Membuat Pernyataan,



Nurfan Dzihan
NIM. 07650068

The logo of Universitas Islam Negeri Maulana Malik Ibrahim Malang is a shield-shaped emblem. It features a green background with a white border. Inside the shield, there is a yellow Arabic calligraphic design. The text "UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM" is written in a light green font around the top and sides of the shield, and "PUSAT PERPUSTAKAAN" is written at the bottom. The word "MOTTO" is centered in the upper part of the shield.

MOTTO

"Jangan berhenti berupaya ketika menemui kegagalan. Karena kegagalan adalah cara Allah SWT mengajari kita tentang arti kesungguhan"

PERSEMBAHAN...

Syukur Alhamdulillahirabbil Aalamiin segala puja dan puji syukur kehadiran Allah SWT atas segala nikmat serta limpahan Rahmat dan Hidayah-Nya, sehingga saya dapat menyelesaikan tugas akhir skripsi ini. Semoga dapat bermanfaat bagi siapa saja yang telah membacanya. Saya persembahkan karya ini kepada :

Yang pertama kepada kedua orang tuaku M.Ducha Ilyas dan Siti Nihayatul Choiriyah yang selalu memberikan motivasi dan semangat kapanpun dimanapun dalam keadaan apapun.

Yang kedua kepada kakak dan adiku laili usdiana dan M.faris hasan yang juga selalu mensupport dan menyuntikan semangat.

Yang ketiga kepada teman-temanku komunitas Informatic community (IOC) yang telah memberikan kebersamaan dan kujadikan keluarga kedua selama ini.

Yang keempat kepada tim riset (didik, uma, kunti, rina, ipit, bara, chika uco, ratri) dan anak angkatan 09 yang telah membantu dan mengajari dengan sabar sehingga selesai skripsi ini, terimakasih ya rek...

Dan semua pihak yang tidak dapat saya sebutkan lagi, terima kasih atas bantuanya semua..

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Syukur Alhamdulillah penulis haturkan kehadiran Allah SWT yang telah melimpahkan rahmat, taufik serta hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi ini sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Ucapan terima kasih penulis sampaikan kepada :

1. Prof. Dr. H. Mudjia Rahardjo MSc selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Drh Bayyinatul Muctaromah, M.Si selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Ibu Ririen Kusumawati, M.Kom selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Bapak Ir. M. Amin Hariyadi, M.T selaku pembimbing dalam skripsi ini yang telah memberikan bimbingan dan pengarahan dalam proses penyelesaian skripsi ini.
5. Bapak Zainal Abidin, M.Kom selaku membinbing dalam menyelesaikan laporan skripsi ini..

6. Seluruh Dosen Universitas Islam Negeri Maulana Malik Ibrahim Malang, khususnya dosen Teknik Informatika beserta seluruh staf yang telah memberikan ilmu dan membantu dalam penyelesaian skripsi ini.
7. Kedua orang tua, kaka, dan adik yang selalu memberikan semangat dan doa dalam penyelesaian skripsi ini.
8. Seluruh teman-teman Informatic Community (IOC) yang selalu menemani penulis dan saling memberikan pelajaran bersama dalam belajar di kampus Universitas Islam Negeri Maulana Malik Ibrahim Malang
9. Teman-teman TIM Riset Image Processing yang telah memotivasi dan bersama-sama dalam proses penyelesaian skripsi ini.
10. Kepada seluruh pihak yang membantu penulisan skripsi ini, yang tidak dapat disebutkan satu persatu.

Dengan segala kerendahan hati, dalam menyelesaikan skripsi ini penulis menyadari bahwa masih banyak kekurangan dan kekeliruan, sehingga dengan sangat mohon saran dan kritik supaya penulis bisa menyempurnakan dimasa mendatang. semoga skripsi ini dapat bermanfaat dan berguna bagi yang membacanya. Semoga Allah SWT selalu melimpahkan rahmat, taufiq, hidayah dan inayahnya kepada kita semua. Amin

Wassalamu'alaikum Wr. Wb.

Malang, 10 Juli 2013

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGANTAR	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERNYATAAN.....	v
MOTTO	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
ABSTRAK.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4
1.5 Batasan Masalah.....	4
1.5 Metodologi Penelitian.....	5
1.6 Sistematika Penulisan Skripsi	5
BAB II TINJAUAN PUSTAKA.....	8
2.1 Pengolahan Citra Digital	8
2.1.1 Citra	8
2.1.1.1 Citra RGB	9
2.1.1.2 Citra Grayscale	10
2.1.1.3 Citra Biner.....	11
2.1.1.4 Pixel.....	12
2.1.2 Perbaikan Kualitas Citra	13
2.1.2.1 Median filter.....	14
2.1.2.1 CLAHE.....	16
2.1.3 Operasi Morfologi Citra.....	17
2.1.3.1 Dilasi	17

2.1.3.2 Erosi.....	18
2.1.5 Segmentasi Citra.....	19
2.2 Homotopy Tree	21
2.3 Matlab	25
2.3.1 Sejarah Matlab	26
2.3.2 Kelebihan Matlab	27
2.4 Paint	28
BAB III DESAIN DAN PERANCANGAN SISTEM	30
3.1 Deskripsi Sistem.....	30
3.2 Perancangan Sistem	30
3.2.1 Data Masukan	32
3.2.2 Data Keluaran	33
3.2.3 Desain Proses.....	33
3.2.3.1 Input Image	34
3.2.3.2 Proses Preprocessing	34
3.2.3.3 Proses Segmentasi	38
3.3 Desain <i>Interface</i> sistem.....	39
3.3.1 Desain Menu Utama.....	39
3.3.2 Menu <i>About</i>	40
3.3.3 Menu <i>Help</i>	40
BAB IV IMPLEMENTASI DAN PEMBAHASAN	42
4.1 Lingkup Uji Coba.....	42
4.2 Implementasi Desain dan Sistem	42
4.2.1 implementasi Antarmuka dan sistem menu utama.....	42
4.2.2 Implementasi <i>Preprocessing</i>	46
4.2.3 Implementasi Segmentasi Objek	49
4.2.4 Implementasi Form Bantuan.....	50
4.2.5 Implementasi Form <i>About</i>	51
4.3 Hasil Uji coba segmentasi <i>Computer Made-Image</i> menggunakan metode <i>homotopy tree</i>	51
4.4 Proses Segmentasi Perspektif Islam.....	56
BAB V KESIMPULAN DAN SARAN.....	59
5.1 Kesimpulan.....	59
5.2 Saran	60
DAFTAR PUSTAKA.....	61

DAFTAR TABEL

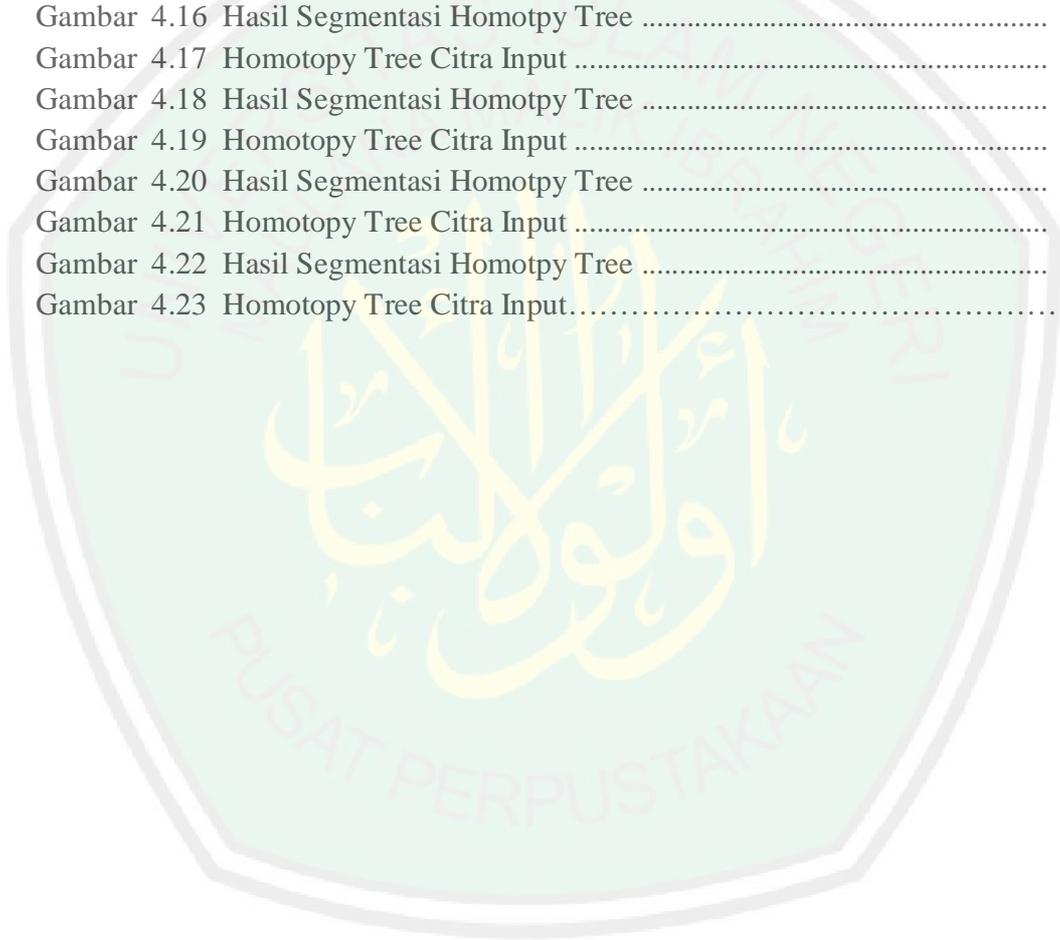
Tabel 4.1	Spesifikasi Lingkungan Uji Coba	42
-----------	---------------------------------------	----



DAFTAR GAMBAR

Gambar 2.1	Citra <i>RGB</i>	10
Gambar 2.2	Citra Grayscale	11
Gambar 2.3	Huruf B dan representasi dari derajat keabuan.....	12
Gambar 2.4	Blok diagram alur kerja Median Filter.....	15
Gambar 2.5	Contoh Penerapan Median filter.....	15
Gambar 2.6	Proses Dilasi pada <i>Grayscale Image</i>	18
Gambar 2.7	Proses Erosi Pada <i>Grayscale Image</i>	18
Gambar 2.8	(a) Citra Biner yang <i>Bernoise – Foreground</i> Berwarna Hitam dan <i>Background</i> Berwarna Putih, (b) Pengikisan dengan Elemen yang Berstruktur Segi Empat dengan Ukuran 3 x 3, (c) Hasil Dari Opening – Closing dengan Struktur Elemen yang Sama, (d) Hasil Dari Opening – Closing dengan Struktur Elemen yang Sama.....	23
Gambar 2.9	(a) Gambar Citra, (b) <i>Homotopy Tree</i> Dari Gambar a, (c) Transformasi <i>Homotopy</i> Dari Gambar a	24
Gambar 2.10	Citra <i>Greyscale</i> dengan <i>Homotopy Treenya</i> . X_0 Adalah Komponen yang Tidak Dibatasi <i>Background</i> dari X, yaitu $(X_0 \cup X_2 \cup X_2^c) = X^c$. Y <i>Homotopic</i> dengan X Karena Memiliki <i>Hotomopy Tree</i> yang Sama	25
Gambar 3.1	Diagram Alir Sistem	31
Gambar 3.2	Gambar Geometri	32
Gambar 3.3	<i>Homotopy Tree</i> dari citra <i>Geometri</i>	33
Gambar 3.4	FlowChart Proses Preprocessing	34
Gambar 3.5	FlowChart Proses Grayscale.....	35
Gambar 3.6	Diagram Alir <i>Median Filter</i>	36
Gambar 3.7	FlowChart Proses Clahe.....	37
Gambar 3.8	FlowChart Proses Segmentasi	38
Gambar 3.9	Desain <i>Interface Menu Utama</i>	39
Gambar 3.10	Desain <i>Gambar Menu About</i>	41
Gambar 3.11	Desain <i>Gambar Menu Bantuan</i>	41
Gambar 4.1	<i>Antar Muna Menu Utama</i>	43
Gambar 4.2	<i>Source code</i> untuk memanggil menu <i>About</i> Pembuat aplikasi	44
Gambar 4.3	<i>Source code</i> untuk memanggil menu <i>Help</i>	44
Gambar 4.4	<i>Source code</i> untuk memanggil menu <i>Exit</i>	44
Gambar 4.5	<i>Source code</i> Button Open Image.....	45
Gambar 4.6	<i>Source code</i> Button Simpan.....	45
Gambar 4.7	<i>Source code</i> Button Reset.....	45

Gambar 4.8	<i>Source code Grayscale</i>	46
Gambar 4.9	<i>Source code Median Filtering</i>	46
Gambar 4.10	<i>Source code Filter CLAHE</i>	46
Gambar 4.11	<i>Contoh citra geometri setelah di processing</i>	48
Gambar 4.12	<i>Source code Homotopy Tree</i>	49
Gambar 4.13	<i>Source code Menampilkan Objek</i>	50
Gambar 4.14	<i>Form Help</i>	50
Gambar 4.15	<i>Form Menu About pembuta aplikasi</i>	51
Gambar 4.16	Hasil Segmentasi Homotpy Tree	52
Gambar 4.17	Homotopy Tree Citra Input	52
Gambar 4.18	Hasil Segmentasi Homotpy Tree	53
Gambar 4.19	Homotopy Tree Citra Input	53
Gambar 4.20	Hasil Segmentasi Homotpy Tree	54
Gambar 4.21	Homotopy Tree Citra Input	54
Gambar 4.22	Hasil Segmentasi Homotpy Tree	55
Gambar 4.23	Homotopy Tree Citra Input.....	56



ABSTRAK

Nurfan Dzihan 2013. *Aplikasi Segmentasi Computer Made-Image menggunakan Homotopy Tree*. Skripsi. Jurusan Teknik Informatika, Fakultas Sains dan Teknologi. Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang. Pembimbing : (1) M. Amin Hariyadi, M.T, (2) Zainal Abidin, M.Kom

Kata Kunci : *Homotopy Tree*, segmentasi, *Paint*

Segmentasi adalah salah satu metode yang penting untuk mengubah citra *input* menjadi citra *output* berdasarkan atribut yang diambil dari citra tersebut. Segmentasi citra membagi citra ke dalam daerah intensitas masing-masing sehingga bisa membedakan antara objek dengan *background*-nya. Ada banyak metode untuk segmentasi citra salah satunya adalah *homotopy tree*.

Homotopy Tree adalah metode yang membagi objek berdasarkan kesamaan topologi dengan menggunakan struktur seperti pohon, terdapat akar (*root*), node dan bercabang. Kesamaan topologi pada *homotopy tree* diperoleh dengan melakukan klasifikasi derajat keabuan pada suatu citra, kemudian itu mengidentifikasi nilai piksel dan mengklasifikasikannya.

Dalam aplikasi ini objek yang digunakan adalah hasil dari pembuatan dengan menggunakan *software paint*. Pengenalan objek dengan menggunakan metode *homotopy tree* ini dilakukan dengan cara input citra dalam bentuk RGB, setelah itu diproses menjadi grayscale kemudian di segmentasi dengan mengenali nilai piksel-piksel dari objek tersebut, *hasilnya* dalam mengenali objek cukup bagus, dan terbentuk tree dengan baik.

ABSTRACT

Nurfan Dzihan 2013. Computer Applications Made-Image Segmentation using homotopy Tree. Thesis. Department of Informatics, Faculty of Science and Technology. State Islamic University (UIN) Maulana Malik Ibrahim Malang. Supervisor: (1) M. Amin Hariadi, M.T, (2) Zainal Abidin, M. Kom

Keywords: homotopy Tree, segmentation, Paint

Segmentation is one of important method to change the image of the input into output image based on attributes extracted from the image. Image segmentation divides the image into the intensity of each region so that it can distinguish between objects with the background. There are many methods for image segmentation is one of the homotopy tree.

Tree homotopy is a method that divides the object based on common topology by using a tree-like structure, there is a root (root), and branching nodes. The similarity of the homotopy tree topology obtained by performing classification degrees of gray in an image, then it identifies and classifies pixel value.

In this application object that is used the result of creation by using paint software. Object recognition using tree homotopy method is done by way of the image in the form of RGB input, after it is processed into grayscale and then on segmentation by recognizing the value of the pixels of the object, the results are pretty good at recognizing objects, and a well-formed tree.



BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Dalam sebuah terbitan ilmiah “Dalam minggu ketujuh, rangka mulai tersebar keseluruh tubuh dan tulang tulang mencapai bentuknya yang kita kenal. Pada akhir minggu ke tujuh dan minggu kedelapan, otot-otot menempati posisinya di sekeliling bentukan tulang” (Moore, developing Human, 6, edition, 1998). Hal ini sesuai dengan yang diberitakan Allah dalam surat Al Mu’minun : 12-14

وَلَقَدْ خَلَقْنَا الْإِنْسَانَ مِنْ سُلَالَةٍ مِّنْ طِينٍ ﴿١٢﴾ ثُمَّ جَعَلْنَاهُ نُطْفَةً فِي قَرَارٍ مَّكِينٍ ﴿١٣﴾
 ثُمَّ خَلَقْنَا النَّطْفَةَ عَلَقَةً فَخَلَقْنَا الْعَلَقَةَ مُضْغَةً فَخَلَقْنَا الْمُضْغَةَ عِظْمًا فَكَسَوْنَا
 الْعِظْمَ لَحْمًا ثُمَّ أَدْنَيْنَاهُ خَلْقًا آخَرَ فَتَبَارَكَ اللَّهُ أَحْسَنُ الْخَالِقِينَ ﴿١٤﴾

Artinya :

“Dan Sesungguhnya kami Telah menciptakan manusia dari suatu saripati (berasal) dari tanah, Kemudian kami jadikan saripati itu air mani (yang disimpan) dalam tempat yang kokoh (rahim), kemudian air mani kami jadikan segumpal darah, lalu segumpal darah itu kami jadikan segumpal daging, dan segumpal daging itu kami jadikan tulang belulang, lalu tulang belulang itu kami bungkus dengan daging. Kemudian kami jadikan dia makhluk yang (berbentuk) lain. Maka maha sucilah Allah, pencipta baik” (QS : Al Mu’minun :12-14)

Ayat –ayat diatas menerangkan tahap-tahap penciptaan manusia dari suatu keadaan ke keadaan lain yang menunjukkan akan kesempurnaannya sehingga Dia Jalla wa ‘Alaa saja yang berhak untuk diibadahi. Begitu pula penggambaran penciptaan Adam ‘Alaihis Salam yang Dia ciptakan dari suatu saripati yang berasal dari tanah berwarna hitam yang berbau busuk dan diberi bentuk.

Tanah tersebut diambil dari seluruh bagiannya, sebagaimana dikabarkan oleh Rasulullah Shallallahu ‘AlaihiWaSallam :

“Sesungguhnya Allah menciptakan Adam dari segenggam (sepenuh telapak tangan) tanah yang diambil dari seluruh bagiannya. Maka datanglah anak Adam (memenuhi penjuru bumi dengan beragam warna kulit dan tabiat). Di antara mereka ada yang berkulit merah, putih, hitam, dan di antara yang demikian. Di antara mereka ada yang bertabiat lembut, dan ada pula yang keras, ada yang berperangai buruk (kafir) dan ada yang baik”(Mukmin).

Melalui konsep embrilio yang tersurat di dalam Al-Quran dapat dipelajari bahwa Allah SWT menciptakan manusia melalui beberapa proses untuk memperoleh bentuk yang sempurna. Dan para dokter kandungan membuktikan bahwa semua yang disebutkan di dalam al-Quran dan hadist-hadist Rosululloh SAW tentang proses penciptaan manusia adalah sesuai dengan yang ditemukan pada ilmu modern (Kiptiyah,2007).

Seiring dengan perkembangan teknologi tentang pengolahan citra pada saat ini, telah berkembang salah satu teknologi yaitu *Image Processing*,

yaitu pemrosesan citra dengan menggunakan computer untuk mendapatkan citra yang lebih baik, tujuan dari memperbaiki citra sendiri adalah supaya mudah di iterpresentasikan oleh manusia atau mesin (Rinaldi Munir:3-5). Salah satu teknik yang digunakan untuk Pengolahan citra Digital adalah *Image Segmentation*.

Segmentasi citra digital adalah suatu proses pengelompokan citra menjadi beberapa region berdasarkan kriteria yang telah ditentukan. Segmentasi citra memiliki tujuan menemukan karakteristik yang dimiliki oleh citra. Semakin baik kualitas segmentasi maka semakin baik juga pengenalan polanya. Pada proses ini sangatlah penting karena segmentasi yang dilakukan harus tepat, supaya informasi yang diperoleh dapat di terjemahkan (Meilinda, 2009). Terdapat banyak sekali metode segmetasi citra yang salah satunya adalah metode segmetasi *homotopy tree*.

Homotopy tree adalah metode digunakan untuk mendapatkan tepi dari objek dalam suatu citra dengan mengenali bagian titik-titik warna yang hampir sama dengan ketetanggaanya, membagi objek dengan menggunakan struktur objek pohon, terdapat akar, node dan bercabang. Menurut Noorullah (Noorullah, 2009) dalam jurnalnya menyebutkan bahwa pohon yang berdekatan (*adjacent tree*) atau *homotopy tree* dari gambar biner merupakan sebuah graf yang menggambarkan bentuk dari latar depan dan latar belakang komponen terhubung.

kelebihan dari metode *Homotopy tree* adalah efektif bila digunakan dalam proses untuk mengenali objek yang akan disegmentasi, yakni dengan

cara mengenali perbedaan warna yang terdapat dalam sebuah citra dan mengetahui range nilai piksel dari objek yang akan disegmentasi. Dan diharapkan dengan metode ini nantinya dapat mendapatkan objek yang diinginkan.

Jadi dalam penelitian ini yang terpenting adalah bagaimana metode *homotopy tree* diterapkan dalam proses segmentasi citra dan mendapatkan informasi yang diinginkan.

1.2 RUMUSAN MASALAH

Berdasarkan uraian latar belakang dapat dirumuskan permasalahan pada penelitian ini bagaimana memperoleh objek geometri dari gambar yang dibuat dengan *software paint* dengan metode *homotopy tree* dengan jelas.

1.3 TUJUAN PENELITIAN

Tujuan dari penelitian ini adalah mendapatkan informasi yang diinginkan dari hasil segmentasi dengan *Computer Made-Image* menggunakan metode *homopoty tree*.

1.4 MANFAAT PENELITIAN

penelitian ini diharapkan dapat memberikan manfaat yaitu :

1. Mendapatkan objek dari suatu citra secara jelas.
2. Efisiensi waktu dalam menganalisa citra

1.5 BATASAN MASALAH

1. metode yang akan digunakan adalah segmentasi citra dengan *homotopy tree*.
2. objek yang digunakan adalah gambar citra berformat JPG.

1.6 METODOLOGI PENELITIAN

Dalam menyelesaikan Penelitian ini langkah-langkah yang harus dilakukan adalah :

1. Data yang di gunakan untuk penelitian adalah data citra yang dibuat dengan menggunakan *software Paint* dan berformat JPG.
2. Dilakukan proses terhadap citra tersebut yaitu dengan melakukan fungsi *grayscale* median filter, dan clahe.
3. Setelah data telah dilakukan Proses, langkah selanjutnya adalah melakukan segmentasi dengan metode *homotopy tree*.
4. Tree didapatkan setelah proses segmentasi dan mengenali objek yang diinginkan.

1.7 SISTEMATIKA PEYUSUNAN LAPORAN

Untuk mempermudah pembahasan dari laporan skripsi ini, maka sistematika yang akan digunakan seperti ini :

1. BAB I PENDAHULUAN

Segmentasi citra sebenarnya merupakan salah satu metoda pengolahan citra digital yang bertujuan untuk mempartisi atau membagi citra menjadi bagian-bagian citra yang lebih kecil dengan mengkonversikan kedalam bentuk

matrix. Bagian-bagian tersebut tidak selalu terdiri dari intensitas warna dan tekstur yang persis sama, sebab dalam kenyataan sehari-hari banyak atau hampir selalu ditemui suatu citra yang merupakan perpaduan dari berbagai intensitas warna dan tekstur yang kompleks. Segmentasi di sini akan sangat berperan, yang memudahkan pengamat citra untuk dapat membedakan bagian-bagian tertentu dari suatu citra yang mempunyai kesamaan atau kemiripan (homogen) berdasarkan metoda tertentu dengan peninjauan dari segi tertentu pula, seperti dari segi persamaan intensitas warna, threshold, tekstur dan lain-lain. Masing-masing bagian tersebut nantinya diharapkan dapat digunakan untuk diolah dan dianalisa lebih lanjut secara terpisah..

BAB II TINJAUAN PUSTAKA

Homotopy tree adalah metode suatu metode yang membagi objek berdasarkan kesamaan topologi dengan menggunakan struktur seperti pohon, terdapat akar (*root*), node dan bercabang. Metode ini digunakan untuk mendapatkan tepi dari objek dalam suatu citra dengan mengenali bagian titik-titik warna yang hampir sama dengan ketetanggaannya. Transformasi ketetanggaan merupakan penipisan dari citra awal dengan mengetahui nilai titik-titik dari setiap piksel sesuai dengan konfigurasi ketetanggaan yang ada dari warna citra tersebut. Karena adanya perubahan intensitas inilah sehingga mampu mendeteksi tepi-tepi objek dalam citra.

BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi analisa sistem yang akan dibuat serta perancangan *desain interface* dari sistem.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil dari perancangan sistem yang telah dibuat serta menganalisa *output* yang dihasilkan.

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran dari pembuatan sistem tersebut, supaya dapat bermanfaat untuk pengembangan aplikasi segmentasi *Computer Made-Image* dengan metode *homotopy tree* selanjutnya dan dengan objek yang lebih kompleks.



BAB II

TINJAUAN PUSTAKA

2.1 Pengolahan citra digital

Pengolahan citra digital adalah sebuah disiplin ilmu yang mempelajari hal-hal yang berkaitan dengan perbaikan kualitas citra, tranformasi gambar, melakukan pemilihan citra ciri yang optimal untuk bertujuan analisis, melakukan proses penarikan informasi atau deskripsi objek atau pengenalan objek yang terkandung pada citra, melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan proses data. Input dari pengolahan citra adalah citra sedangkan outputnya adalah citra hasil pengolahan (Wijanarko, 2009 : 5).

Pengolahan citra bertujuan untuk memperbaiki kualitas citra agar mudah diinterpretasikan oleh manusia atau mesin. Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain

2.1.1 Citra

Definisi citra menurut kamus Webster adalah suatu representasi, kemiripan, atau imitasi dari suatu obyek atau benda. Sebuah citra mengandung informasi tentang obyek yang direpresentasikan. Citra dapat dikelompokkan menjadi citra tampak dan citra tak tampak. Untuk dapat dilihat mata manusia, citra tak tampak harus dirubah menjadi citra tampak, misalnya dengan menampilkannya di monitor, dicetak di kertas dan sebagainya. Salah satu contoh citra tak tampak adalah citra digital.

Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu pita magnetik. Citra digital merupakan suatu larik dua dimensi atau suatu matriks yang elemen-elemennya menyatakan tingkat keabuan dari elemen gambar. Jadi informasi yang terkandung bersifat diskret. Citra digital tidak selalu merupakan hasil langsung data rekaman suatu sistem. Kadang-kadang hasil rekaman data bersifat kontinu seperti gambar pada monitor televisi, foto sinar-X, dan lain sebagainya. Dengan demikian untuk mendapatkan suatu citra digital diperlukan suatu proses konversi, sehingga citra tersebut selanjutnya dapat diproses dengan komputer.

2.1.1.1 Citra RGB

Suatu citra biasanya mengacu ke citra RGB. Citra (image) merupakan suatu yang menggambarkan objek dan biasanya dua dimensi. Citra merupakan suatu representasi, kemiripan atau imitasi dari suatu objek atau benda. Citra dalam komputer tidak lebih dari sekumpulan sejumlah piksel dimana setiap triplet terdiri atas variasi tingkat keterangan (brightness) dari elemen red, green dan blue. Representasinya dalam citra, triplet akan terdiri dari 3 angka yang mengatur intensitas dari Red (R), Green (G) dan Blue (B) dari suatu triplet. Setiap triplet akan merepresentasikan 1 piksel (picture element). Suatu triplet dengan nilai 67, 228 dan 180 berarti akan mengeset nilai R ke nilai 67, G ke nilai 228 dan B ke nilai 180. Angka-angka RGB ini yang seringkali disebut dengan color values. Pada format .bmp citra setiap

piksel pada citra direpresentasikan dengan dengan 24 bit, 8 bit untuk R, 8 bit untuk G dan 8 bit untuk B. Digambar 2.1 adalah citra RGB



Gambar 2.1 Citra berwarna

Sumber: <http://id.wikipedia.org>

2.1.1.2 Citra grayscale

Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat. Citra grayscale berbeda dengan citra "hitam-putih", dimana pada konteks komputer, citra hitam putih hanya terdiri atas 2 warna saja yaitu "hitam" dan "putih" saja. Pada citra grayscale warna bervariasi antara hitam dan putih, tetapi variasi warna diantaranya sangat banyak. Citra grayscale seringkali merupakan perhitungan dari intensitas cahaya pada setiap pixel pada spektrum elektromagnetik single band.

Citra grayscale disimpan dalam format 8 bit untuk setiap sample pixel, yang memungkinkan sebanyak 256 intensitas. Format ini sangat membantu dalam pemrograman karena manipulasi bit yang tidak terlalu banyak. Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing R, G

dan B menjadi citra grayscale dengan nilai X, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai R, G dan B sehingga dapat dituliskan menjadi:

$$X = (R + G + B) / 3$$
$$\text{Warna} = \text{RGB}(X, X, X)$$



Gambar 2.2 Citra grayscale
(Seetharaman, 2012)

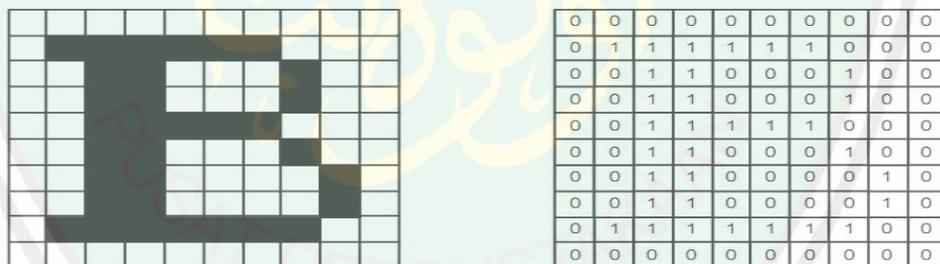
2.1.1.3 Citra Biner

Citra biner hanya mempunyai dua nilai derajat keabuan: hitam dan putih. Pixel-pixel objek bernilai 1 dan pixel-pixel latar belakang bernilai 0. Pada waktu menampilkan gambar, 0 adalah putih dan 1 adalah hitam. Jadi, pada citra biner, latar belakang berwarna putih sedangkan objek berwarna hitam

Meskipun komputer saat ini dapat memproses citra hitam-putih (greyscale) maupun citra berwarna, namun citra biner masih tetap

dipertahankan keberadaannya. Alasan penggunaan citra biner adalah karena ia memiliki sejumlah keuntungan sebagai berikut:

1. Kebutuhan memori kecil karena nilai derajat keabuan hanya membutuhkan representasi 1 bit. Kebutuhan memori untuk citra biner masih dapat berkurang secara berarti dengan metode pemampatan run-length encoding (RLE). Metode RLE akan dijelaskan kemudian.
2. Waktu pemrosesan lebih cepat dibandingkan dengan citra hitam-putih karena banyak operasi pada citra biner yang dilakukan sebagai operasi logika (AND, OR, NOT, dll) ketimbang operasi aritmetika bilangan bulat



Gambar 2.3 huruf B dan representasi dari derajat keabuan
(Rinaldi Munir : 2004)

2.1.1.4 Pixel

Piksel merupakan informasi terkecil dari sebuah citra yang terdiri dari 3 buah layer yaitu Red layer, Green layer dan Blue layer yang membentuk sebuah warna. Ketiga unsur warna tersebut (Red = Merah, Green = Hijau, Blue = Biru) dapat membentuk 2553 kombinasi warna. Ketiga unsur warna

pembentuk warna piksel ini dapat dipisah-pisahkan untuk membantu dalam pengolahan citra.

2.1.2 Perbaikan kualitas citra

Perbaikan kualitas citra adalah langkah awal dalam pengolahan sebuah citra. Perbaikan kualitas diperlukan karena seringkali citra yang diuji mempunyai kualitas yang buruk, misalnya citra mengalami derau (noise) pada saat pengiriman melalui saluran transmisi, citra terlalu terang/gelap, citra kurang tajam, kabur, dan sebagainya. Melalui operasi pemrosesan awal inilah kualitas citra diperbaiki sehingga citra dapat digunakan untuk aplikasi lebih lanjut, misalnya untuk aplikasi pengenalan (recognition) objek di dalam citra.

Yang dimaksud dengan perbaikan kualitas citra adalah proses memperjelas dan mempertajam ciri/fitur tertentu dari citra agar citra lebih mudah dipersepsi maupun dianalisis secara lebih teliti (Andrian, 2006).

Secara matematis, image enhancement dapat diartikan sebagai proses mengubah citra $f(x, y)$ menjadi $f'(x, y)$, sehingga ciri-ciri yang dilihat pada $f(x, y)$ lebih ditonjolkan. Image enhancement tidak meningkatkan kandungan informasi, melainkan jangkauan dinamis dari ciri agar bisa dideteksi lebih mudah dan tepat (Munir, 2004: 103).

Operasi-operasi yang digolongkan sebagai perbaikan kualitas citra cukup beragam antara lain, perubahan kecerahan gambar (image brightness), peregangan kontras (contrast stretching), perubahan histogram citra, pelembutan

citra (image smoothing), penajaman (sharpening), tepi (edge), pewarnaan semu (pseudocoloring), perubahan geometrik, dan sebagainya (Mulyono, 2008: 21).

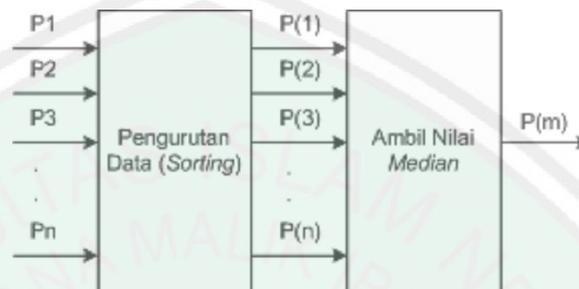
2.1.2.1 Median filter

Konsep dasarnya adalah dengan menemukan nilai *pixel* yang memiliki nilai intensitas dari suatu *pixel* yang berbeda dengan nilai *pixel* yang ada di daerah sekitarnya, dan menggantinya dengan nilai yang lebih cocok. Cara yang paling sederhana dalam mencapainya adalah dengan melakukan pencegahan atau pembatasan nilai *pixel*, sehingga suatu *pixel* tidak memiliki nilai intensitas yang diluar nilai yang ada di sekitarnya. (Davies, 2000)

Sesuai dengan namanya, *median filter* merupakan suatu metode yang menitik beratkan pada nilai *median* atau nilai tengah dari jumlah total nilai keseluruhan *pixel* yang ada di sekelilingnya. Dimisalkan terdapat data A=1, B=5, C=2, D=9, dan E=7, maka *median filter* akan mencari nilai tengah dari semua data yang telah diurutkan terlebih dahulu dari yang paling kecil hingga pada data yang paling besar dan kemudian diambil nilai tengahnya (1, 2, 5, 7, 9). *Median* dari deret tersebut adalah 5.

Pemrosesan *median filter* ini dilakukan dengan cara mencari nilai tengah dari nilai *pixel* tetangga yang mempengaruhi *pixel* tengah. Teknik ini bekerja dengan cara mengisi nilai dari setiap *pixel* dengan nilai *median* tetangganya. Proses pemilihan *median* ini diawali dengan terlebih

dahulu mengurutkan nilai-nilai *pixel* tetangga, baru kemudian dipilih nilai tengahnya.



Gambar 2.4 Blok Diagram Alur Kerja *Median Filter* (Sulistiyo, 2009)

Pengurutan akan menghasilkan nilai dari yang terkecil sampai nilai yang terbesar sesuai dengan $P(1) < P(2) < P(3) < P(n)$, sedangkan nilai m

sesuai dengan rumus $m = \frac{n+1}{2}$ dimana n bernilai ganjil.

43	38	25	45	38
57	45	98	33	46
65	54	57	45	43
74	58	67	62	58

43	38	25	45	38
57	45	45	33	46
65	54	57	45	43
74	58	67	62	58

Gambar 2.5 Contoh Penerapan *Median Filter* (Sulistiyo, 2009)

Hasil dari pengurutan data pada contoh (Gambar 2.4) didapatkan urutan 25, 33, 38, 45, 45, 45, 54, 57, 98. Dari hasil ini akan diambil nilai *median* yang memiliki nilai 45. (Sulistiyo, 2009)

2.1.2.2 CLAHE (Contras Adaptif limited Histogram Equalization)

CLAHE merupakan salah satu cara untuk menajamkan citra dimana proses untuk menampakkan struktur halus yang hilang karena adanya efek pengaburan. Penajaman dilakukan dengan suatu tapis lolos tinggi atau dengan sebuah operator yang invarian-geser, misalnya dengan mask yg berisi kombinasi bilangan positif-negatif yg sesuai. Dengan mask ini, perubahan derajat keabuan didalam citra menjadi lebih tajam.

CLAHE awalnya dikembangkan untuk pencitraan medis dan telah terbukti berhasil untuk peningkatan dari gambar dengan kontras rendah seperti film portal. CLAHE merupakan versi perbaikan dari AHE atau Adaptive Histogram Equalization, keduanya mengatasi keterbatasan standar pemerataan histogram. Menurut Zuiderveld (1994), masalah noise pada AHE dapat dikurangi dengan membatasi peningkatan kontras khususnya di daerah yang homogen. Daerah ini dapat dicirikan sebagai puncak yang tinggi pada histogram terkait dengan daerah kontekstual oleh karena banyak piksel tergabung di dalam kisaran abu-abu yang sama. Lereng yang terkait dengan skema penempatan gray-level menjadi terbatas dengan CLAHE. Hal ini dapat dicapai dengan hanya memungkinkan jumlah maksimum dari pixel di setiap kelompok data yang terkait dengan histogram lokal.

Keuntungan menggunakan CLAHE adalah perhitungan yang sederhana, mudah digunakan dan menghasilkan output yang bagus pada sebagian besar

citra. Citra yang menerapkan CLAHE memiliki noise yang sedikit dan bisa menghindari adanya saturasi kecerahan yang biasa terjadi pada Histogram Equalization.

2.1.3 Operasi morfologi citra

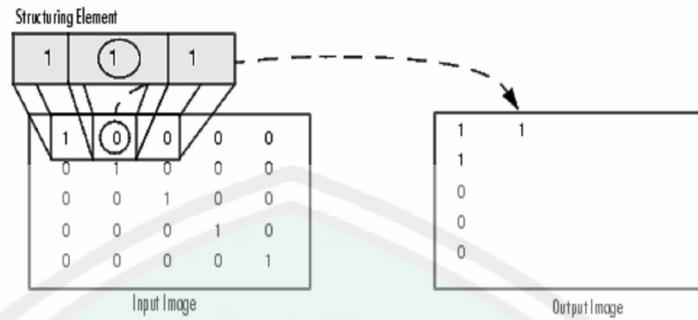
Morfologi adalah salah satu proses yang berhubungan dengan bentuk dan struktur suatu benda. Dalam pengolahan citra digital, istilah morfologi digunakan untuk mengolah struktur suatu objek dalam citra sedemikian rupa sehingga diperoleh struktur objek yang diinginkan.

Matematika morfologi adalah salah satu alat yang di gunakan untuk mengestrak komponen-komponen citra yang berguna dalam representasi dan dekssripsi dari suatu bentuk wilayah citra. Moorfologi juga berguna sebagai langkah awal maupun akhir dari proses analisis citra(wijanarto:2009).

2.1.3.1 Dilasi

Dilasi adalah operasi *morphologi* yang akan menambahkan piksel pada batas antar objek dalam suatu citra digital. Operasi ini menggunakan aturan sebagai berikut :

Untuk gambar *grayscale* maka nilai hasil operasi (output piksel) adalah nilai maksimal yang diperoleh dari himpunan piksel tetangganya. Dalam *binary image*, jika ada piksel tetangga yang bernilai 1 maka output piksel akan diset menjadi 1". (Tjokorda, 2006).

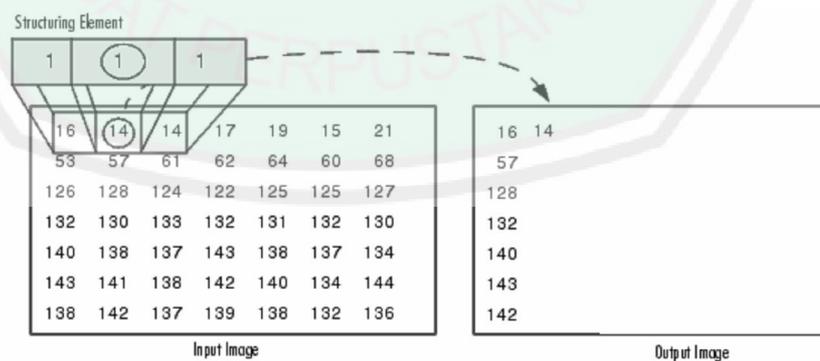


Gambar 2.6 Proses Dilasi pada *Grayscale Image* (Tjokorda, 2006)

2.1.3.2 Erosi

Erosi adalah operasi morfologi yang akan mengurangi pixel pada batas antar objek dalam suatu citra digital. Operasi ini menggunakan aturan sebagai berikut :

“Untuk gambar *grayscale* maka nilai hasil operasi (output piksel) adalah nilai minimal yang diperoleh dari himpunan piksel tetangganya. Dalam *binary image*, jika ada piksel tetangga yang bernilai 0 maka output piksel akan diset menjadi 0”. (Tjokorda, 2006)



Gambar 2.7 Proses Erosi pada *Grayscale Image* (Tjokorda, 2006)

2.1.4 Segmentasi Citra

Segmentasi citra bertujuan untuk membagi wilayah-wilayah yang homogen. Segmentasi salah satu metode yang penting untuk mengubah citra *input* menjadi citra *output* berdasarkan atribut yang diambil dari citra tersebut. Segmentasi citra membagi citra ke dalam daerah intensitas masing-masing sehingga bisa membedakan antara objek dengan *background*-nya. Diharapkan segmentasi memiliki tingkat keakuratan yang tinggi (wijanarto:2009:225). Algoritma dari segmentasi citra terbagi dalam dua macam yaitu :

1. Diskontinu (*discontinuity*) : citra dibagi berdasarkan perbedaan *graylevel* yang menyolok. Contoh proses segmentasi yang didasarkan pada sifat diskontinu antara lain deteksi titik, deteksi garis, deteksi tepi, dan *homotopy algorithm*
2. Keserupaan (*similarity*) : citra dibagi berdasarkan kemiripan *graylevel*. Contoh proses segmentasi yang didasarkan pada sifat keserupaan adalah *thresholding*, *region growing*, *region splitting*, dan *merging*.

Secara umum proses pendekatan yang sering digunakan dalam proses segmentasi citra adalah :

1. Thresholding

Proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk objek dan *background* dari citra secara jelas. Citra dari *thresholding* biasanya digunakan

lebih lanjut untuk proses pengenalan objek serta ekstraksi fitur. Metode *thresholding* secara umum dibagi menjadi 2 yaitu :

a. *thresholding* global

dilakukan dengan mempartisi histogram dengan menggunakan sebuah *thresholding*(batas ambang) global T, yang berlaku untuk seluruh bagian pada citra.

b. *thresholding* adaptif

dilakukan dengan membagi citra menggunakan beberapa sub citra. Lalu pada setiap sub citra, segmentasi dilakukan dengan *threshold* yang berbeda.

Thresholding diimplementasikan setelah dilakukan proses perbaikan kontras citra menggunakan fungsi *Contras-limited adaptive histogram equalizer* (CLAHE). *Thresholding* dikatakan global jika nilai *threshold* T hanya bergantung pada $f(x,y)$, yang melambangkan tingkat keabuan pada titik (x,y) dalam suatu citra.

Kesuksesan metode ini tergantung seberapa bagus teknik partisi histogram. Citra hasil *thresholding* dapat didefinisikan sebagai persamaan

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases}$$

Keterangan :

$f(x,y)$ = mewakili citra yang tersusun atas objek terang dan gelap

T = nilai *Threshold*

c. *egde based*

yaitu pengelompokan citra kedalam region-region tertentu secara langsung berdasar persamaan karakteristik suatu area citranya.

d. *egde based method*

pengelompokan citra kedalam wilayah berbeda yang terpisahkan karena adanya perbedaan perubahan warna tepi dan warna dasar citra yang mendadak.

Pendekatan pertama dan kedua merupakan contoh kategori pemisahan *image* berdasarkan kemiripan area citra, sedangkan pendekatan ketiga merupakan salah satu contoh pemisahan daerah berdasarkan perubahan intensitas yang cepat terhadap suatu daerah.

2.2 *Homotopy tree*

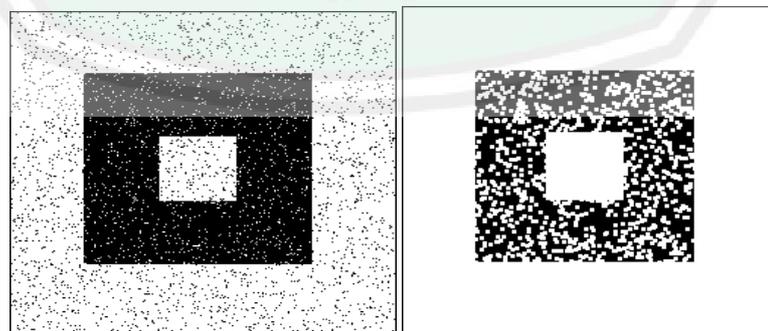
Homotopy berasal dari bahasa Yunani *homos* dan *topos*, *homos* berarti sama dan *topos* berarti topologi. Sehingga dapat disimpulkan bahwa *homotopy tree* adalah suatu metode yang membagi objek berdasarkan kesamaan topologi dengan menggunakan struktur seperti pohon, terdapat akar (*root*), node dan bercabang.

Homotopy Tree digunakan untuk mendapatkan tepi dari objek dalam suatu citra dengan mengenali bagian titik-titik warna yang hampir sama dengan ketetanggaannya. Transformasi ketetanggaan merupakan penipisan dari citra awal dengan mengetahui nilai titik-titik dari setiap piksel sesuai dengan konfigurasi ketetanggaan yang ada dari warna citra tersebut. Karena

adanya perubahan intensitas inilah sehingga mampu mendeteksi tepi-tepi objek dalam citra.

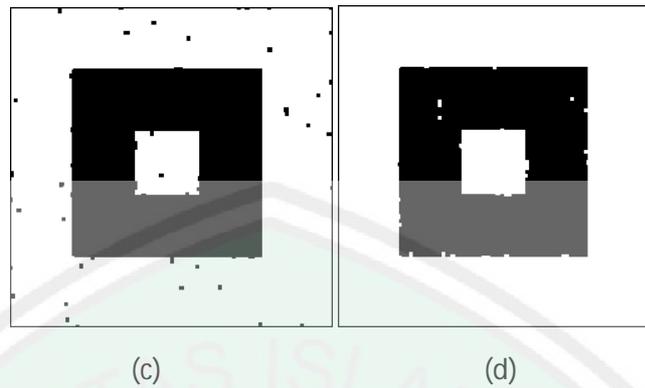
Penipisan homotopik dapat merepresentasikan citra biner dalam bentuk dasarnya dimana obyek diurai menjadi himpunan bagian terkecil yang merupakan rangka dari obyek tersebut. Rangka obyek ini akan dapat memberikan informasi bentuk dasar, dinamika pertumbuhan dan perubahan posisi relatif terhadap acuan. Dalam operasi perangkaan ini dibutuhkan transformasi homotopi yaitu tetap menjaga konektivitas atau antar hubung dari komponen-komponen yang ditipiskan sehingga dapat mencerminkan alur-alur bentuk pada citra.

Seperti contoh pada Gambar 2.8 dalam penyelesaian masalah pada *noise* citra biner dimana gambar a merupakan operasi pengikisan sederhana (sebagai langkah pertama dalam pembukaan/opening) menghapus komponen yang positif pada noise, tetapi komponen yang negative sebenarnya dibesarkan, seperti dalam gambar b, sedangkan pendekatan standar pada opening-closing atau closing-opening ditunjukkan pada gambar c dan d.



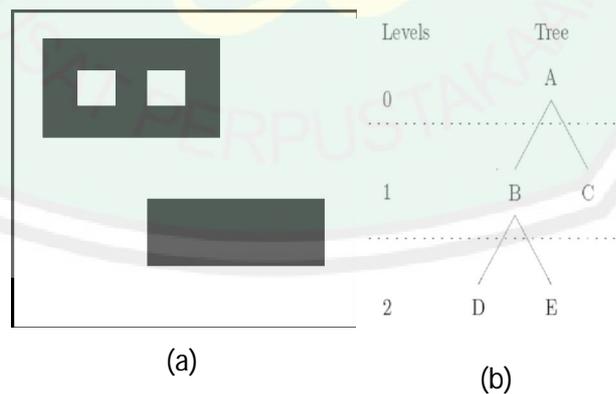
(a)

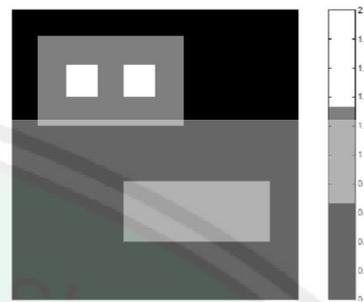
(b)



Gambar 2.8 (a) Citra biner yang bernoise – foreground berwarna hitam dan background berwarna putih, (b) pengikisan dengan elemen yang berstruktur segi empat dengan ukuran 3 x 3, (c) hasil dari opening – closing dengan struktur elemen yang sama, (d) hasil dari opening–closing dengan struktur elemen yang sama (Renato, 2004)

Kesamaan topologi pada *homotopy tree* diperoleh dengan melakukan klasifikasi derajat keabuan pada suatu citra, kemudian itu mengidentifikasi nilai piksel dan mengklasifikasikannya. Seperti gambar berikut :





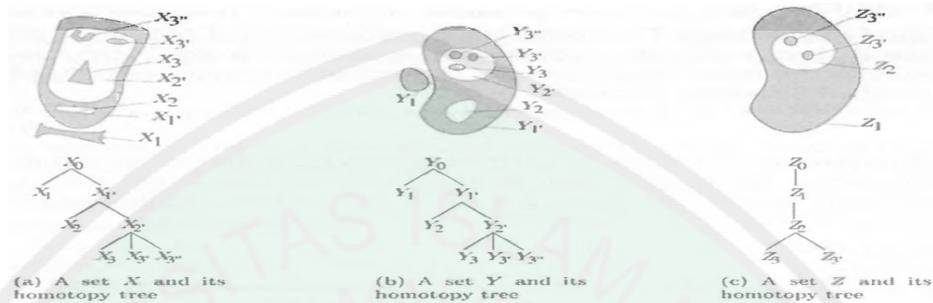
(c)

Gambar 2.9 (a) gambar citra, (b) *homotopy tree* dari gambar **a**,
(c) transformasi *homotopy* dari gambar **a** (Renato,
2004)

Menurut Noorullah dalam jurnalnya yang berjudul *Innovative Thinning and Gradient Algorithm For Binary and Grey Tone Image Using First In First Out Linear Data Structure* menyebutkan bahwa pohon yang berdekatan (*adjacent tree*) atau *homotopy tree* dari gambar biner merupakan sebuah *graf* yang menggambarkan bentuk dari latar depan (*foreground*) dan latar belakang (*background*) komponen yang terhubung. (Noorullah, 2009)

Masing-masing node dari *tree* mengacu pada komponen yang tersambung baik X atau X^c . Akar dari pohon adalah komponen yang dihubungkan secara tak terbatas dari X^c sementara anak-anak dari sebuah node adalah komponen yang terhubung baik X atau X^c yang keduanya berdekatan dengan komponen yang dihubungkan sesuai dengan node tersebut. Dengan asumsi bahwa akar berada pada tingkat 0, node yang muncul pada level terakhir selalu mengacu pada komponen yang terhubung dari X . Beberapa gambar dengan *homotopy tree*nya terlihat pada Gambar 2.9.

Dua gambar dikatakan *homotopic* jika memiliki *homotopy tree* yang sama/identik.



Gambar 2.10 citra *greyscale* dengan *homotopy tree*nya. X_0 adalah komponen yang tidak dibatasi *background* dari X , yaitu $(X_0 \cup X_2 \cup X_2')^c = X^c$. Y *homotopic* dengan X karena memiliki *homotopy tree* yang sama (Soille, 2004)

2.3 Matlab

Matlab merupakan salah satu perangkat lunak (program) yang banyak digunakan dalam penghitungan teknikal. Program Matlab mengintegrasikan komputasi, visualisasi dan pemrograman dalam suatu lingkungan program yang mudah digunakan, dimana masalah dan solusi diekspresikan ke dalam suatu notasi matematika yang mudah dipahami. Matlab adalah sebuah program berbasis operasi matematika, yang dalam hal ini berupa matriks. (The Math Works, 2005)

MATLAB merupakan suatu sistem interaktif yang memiliki elemen data dalam suatu *array* sehingga tidak lagi kita dipusingkan dengan masalah dimensi. Hal ini memungkinkan kita untuk memecahkan banyak masalah teknis yang terkait dengan komputasi, khususnya yang berhubungan dengan *matrix* dan formulasi vektor, yang mana masalah tersebut merupakan momok

apabila kita harus menyelesaikannya dengan menggunakan bahasa level rendah seperti Pascall, C dan Visual Basic. (Miftahul Huda, 2008)

2.3.1 Sejarah Matlab

Pada awalnya MatLab hanya dikenal sebagai “*Matrik Laboratory*”, tetapi sesuai dengan perkembangannya MatLab adalah bahasa yang canggih untuk komputasi teknik. Di dalamnya terdapat kemampuan penghitungan, visualisasi, dan pemograman dalam suatu lingkungan yang mudah untuk digunakan karena permasalahan dan pemecahannya dinyatakan dalam notasi matematika biasa. Kegunaan MatLab secara umum adalah sebagai berikut:

1. Matematika dan komputasi
2. Perkembangan algoritma
3. Pemodelan, simulasi, dan pembuatan prototype
4. Analisa data, eksplorasi dan visualisasim
5. Pembuatan aplikasi, termasuk pembuatan antaramuka grafis.

Matlab bukan merupakan compiler tetapi punya sifat compiler. Ada dua file yang dapat diciptakan :

1. .mat
2. .m

Semua perintah pada MATLAB ditulis dengan huruf kecil dan variabel pada MATLAB bersifat *case sensitive* (membedakan huruf kecil dengan huruf besar).

MATLAB pada awalnya ditulis untuk memudahkan akses perangkat lunak matrik yang telah dibentuk oleh LINPACK dan EISPACK. Saat ini perangkat MATLAB telah menggabung dengan LAPACK dan BLAS *library*, yang merupakan satu kesatuan dari sebuah seni tersendiri dalam perangkat lunak untuk komputasi *matrix*. (Tri Budi, 2007)

2.3.2 Kelebihan Matlab

Sebagai sebuah sistem, MATLAB tersusun dari 5 bagian utama:

1. ***Development Environment***. Merupakan sekumpulan perangkat dan fasilitas yang membantu anda untuk menggunakan fungsi-fungsi dan file-file MATLAB. Beberapa perangkat ini merupakan sebuah *graphical user interfaces* (GUI). Termasuk didalamnya adalah MATLAB *desktop* dan *Command Window*, *command history*, sebuah *editor* dan *debugger*, dan *browsers* untuk melihat *help*, *workspace*, *files*, dan *search path*.
2. **MATLAB *Mathematical Function Library***. Merupakan sekumpulan algoritma komputasi mulai dari fungsi-fungsi dasar seperti *sum*, *sin*, *cos*, dan *complex arithmetic*, sampai dengan fungsi-fungsi yang lebih kompleks seperti *matrix inverse*, *matrix eigenvalues*, *Bessel functions*, dan *fast Fourier transforms*.
3. **MATLAB *Language***. Merupakan suatu *high-level matrix/array language* dengan *control flow statements*, *functions*, *data structures*,

input/output, dan fitur-fitur *object-oriented programming*. Ini memungkinkan bagi kita untuk melakukan kedua hal baik "pemrograman dalam lingkup sederhana " untuk mendapatkan hasil yang cepat, dan "pemrograman dalam lingkup yang lebih besar" untuk memperoleh hasil-hasil dan aplikasi yang kompleks.

4. **Graphics.** MATLAB memiliki fasilitas untuk menampilkan *vector* dan *matrices* sebagai suatu grafik. Didalamnya melibatkan *high-level functions* (fungsi-fungsi level tinggi) untuk visualisasi data dua dimensi dan data tiga dimensi, *image processing*, *animation*, dan *presentation graphics*. Ini juga melibatkan fungsi level rendah yang memungkinkan bagi anda untuk membiasakan diri untuk memunculkan grafik mulai dari bentuk yang sederhana sampai dengan tingkatan *graphical user interfaces* pada aplikasi MATLAB anda.

MATLAB Application Program Interface (API). Merupakan suatu *library* yang memungkinkan program yang telah anda tulis dalam bahasa C dan *Fortran* mampu berinteraksi dengan MATLAB. Ini melibatkan fasilitas untuk pemanggilan *routines* dari MATLAB (*dynamic linking*), pemanggilan MATLAB sebagai sebuah *computational engine*, dan untuk membaca dan menuliskan *MAT-files*. (Santoso, 2008)

2.4 Paint

Paint adalah program yang digunakan untuk menggambar, warna,dan mengedit gambar. Dapat digunakan untuk mewarnai seperti sketsa digital

untuk membuat gambar sederhana dan proyek kreatif atau untuk menambahkan teks dan desain untuk gambar yang lain, seperti gambar yang diambil dengan kamera digital.

Paint (sebelumnya Paintbrush for Windows) adalah program lukisan grafis sederhana ataupun rumit yang telah disertakan dengan semua versi dari Microsoft Windows. Gambar-gambar ini dapat berupa hitam-putih atau warna, dan dapat disimpan sebagai file bitmap. Anda dapat mencetak gambar, menggunakannya untuk latar belakang desktop atau menyisipkannya ke dalam dokumen lain.

Bahkan Paint dapat menggunakan untuk melihat dan mengedit foto scan. Paint dapat juga digunakan untuk bekerja dengan gambar, seperti .Jpg, .gif, atau .Bmp. File. Gambar Paint dapat disisipkan ke dokumen lain yang sedang dikerjakan, atau menggunakannya sebagai latar belakang desktop. Paint sering disebut sebagai MS Paint atau Microsoft Paint. Program ini membuka dan menyimpan file sebagai bitmap Windows (24-bit, 256 warna, 16 warna, dan monokrom). Karena kesederhanaannya, aplikasi ini dengan cepat menjadi salah satu aplikasi yang paling.



BAB III

DESAIN DAN PERANCANGAN SISTEM

3.1 Deskripsi sistem

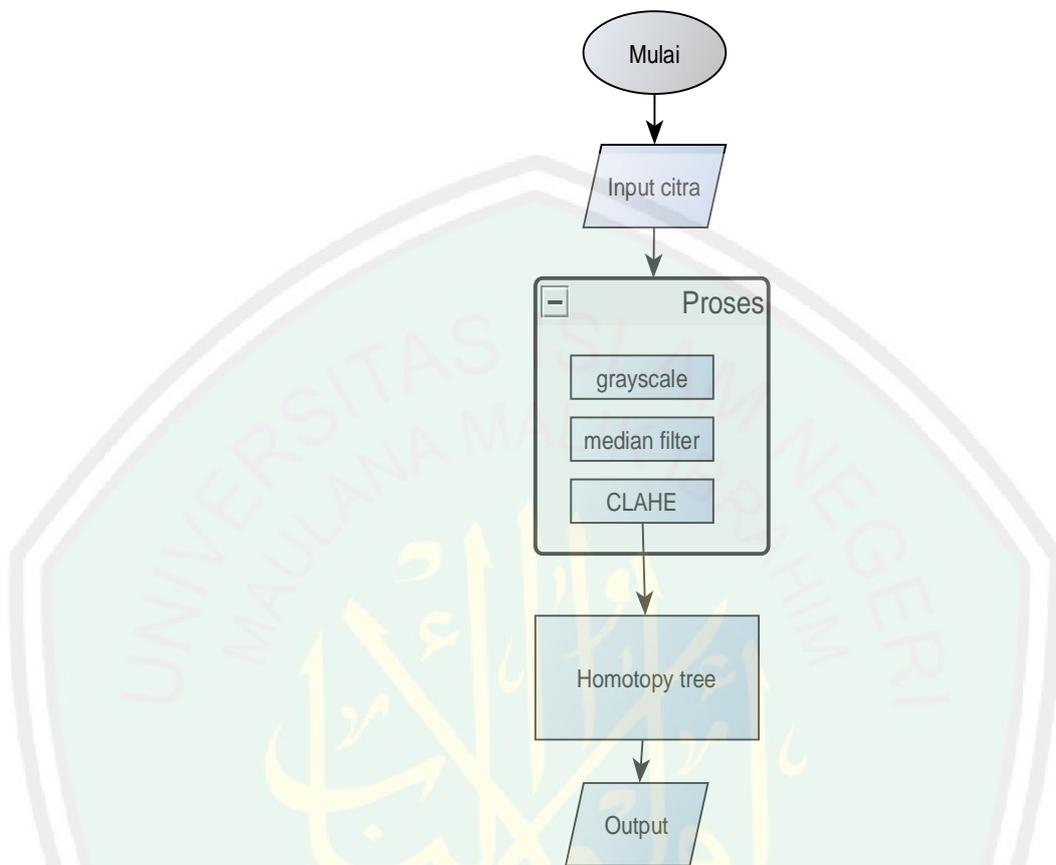
Pada penelitian ini sistem yang akan dibuat adalah tentang aplikasi segmentasi *computer made image* menggunakan *homotopy tree*. Fokus dari penelitian ini adalah segmentasi objek-objek geometri yang dibuat dengan *software paint*.

Sistem yang akan di buat menggunakan bahasa pemrograman Matlab. Tujuannya adalah mendapatkan mengetahui jumlah objek dalam suatu citra berupa bentuk-bentuk geometri yang dibuat dengan *software paint*.

3.2 Perancangan sistem

Perancangan sistem ini adalah implementasi dari Segmentasi *Computer made-image* dengan Metode *Homotpy tree*. Pada proses awal adalah input data berupa citra berformat JPG, selanjutnya selanjutnya setelah itu proses preprocessing citra, pada proses ini adalah proses untuk meningkatkan kontras dari citra setelah proses ini selesai dilanjutkan dengan proses segmentasi pada bagian yang di inginkan. Dalam hal ini adalah *Objek computer made-image* .

Secara umum diagram sistem sebagai berikut :



Gambar 3.1 diagram alir sistem

Pada gambar 3.1 sesuai diagram pertama yang dilakukan input citra berupa objek yang dibuat dengan *software paint* setelah itu dilakukan proses, dan langkah yang terakhir adalah segmentasi dengan metode *homotopy tree* pada objek yang di inginkan yaitu bentuk-bentuk geometri bangun ruang. Pada proses ini, dilakukan pengklasifikasian nilai pixel sehingga menghasilkan citra segmentasi.

3.2.1 Data Masukan

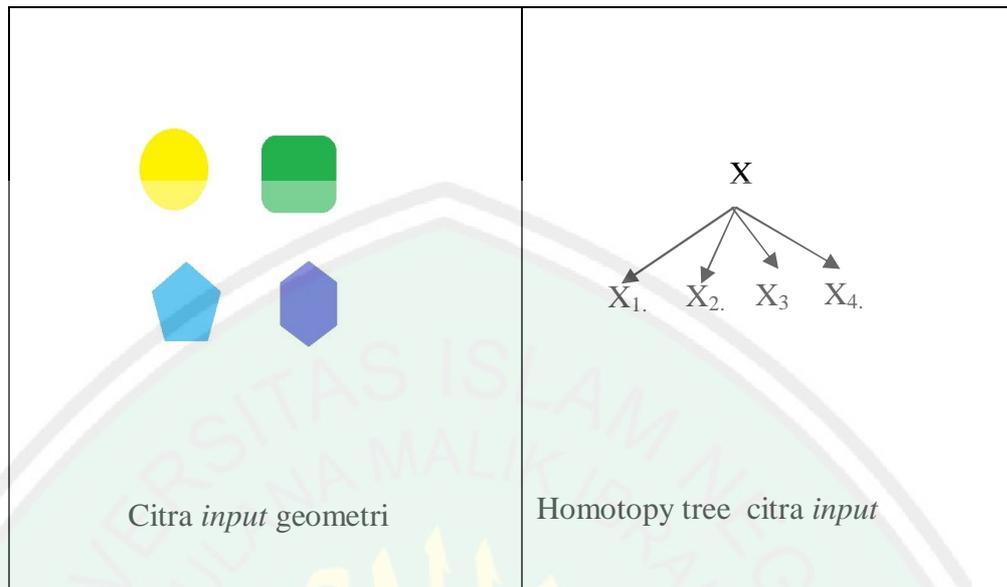
Data yang akan digunakan untuk mengimplemetasikan penelitian ini adalah berupa data yang dibuat dengan *software paint*. data ini berupa file yang berformat *joint photographic experts group* (*.jpeg).



Gambar 3.2 gambar geometri
(Dibuat dengan *software paint*)

sebelum data input diproses menggunakan metode *homotopy tree*, harus di lakukan proses *grayscale* terlebih dahulu. Tujuannya adalah supaya mudah di lakukan segmentasi dengan metode *homopy tree*.

Proses homopy tree dari sebuah citra *x-ray* dapat dilihat dari gambar 3.2 sebagai berikut



Gambar 3.3 *homotopy tree* dari citra geometri

Dari gambar 3.3 dapat di ketahui tree dari citra *X-ray thorax* yaitu X sebagai *background* yang merupakan *root* dari topologi *Homotopy Tree*. Sedangkan untuk *node* atau cabangnya yakni X_1 sebagai citra geometri bulat, X_2 sebagai citra geometri limas, X_3 sebagai cita geometri kotak, X_4 sebagai citra geometri persegi lima.

3.2.2 Data keluaran

Data yang akan dihasilkan dari segmetasi dengan metode *homotopy tree* , berupa objek geometri hasil segmentasi.

3.2.3 Desain Proses

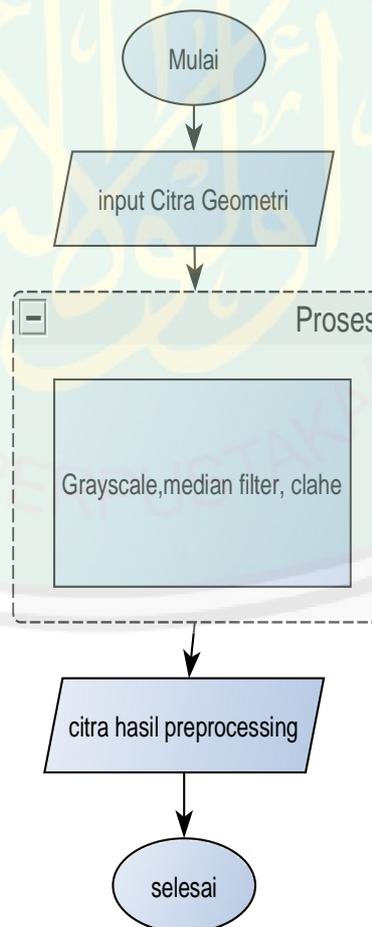
Desain proses ini akan menjelaskan tentang semua proses yang akan di lakukan secara detail untuk mendapatkan objek citra geometri menggunakan metode *homotopy tree*.

3.2.3.1 Input image

Input image adalah langkah pertaman dalam memulai proses penelitian ini. Data yang akan di inputkan berupa data citra digital berformat *joint photographic experts group* (*.jpeg). data ini masih berupa RGB.

3.2.3.2 Proses Preprocessing

Preprocessing adalah proses untuk meningkatkan kualitas citra dan dapat meningkatkan kemungkinan dalam tahap pengolahan citra selanjutnya. Gambar 3.4 adalah diagram alir secara umum preprocessing

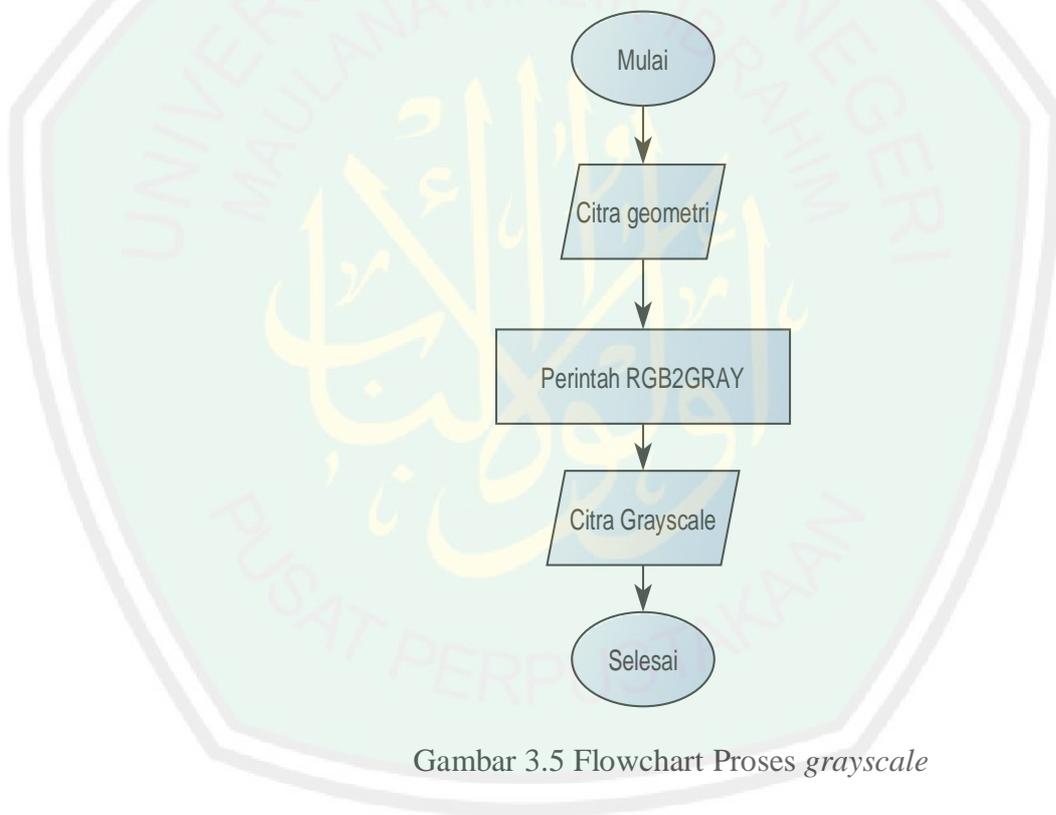


Gambar 3.4 Flowchart Proses preprocessing

a. Proses Citra

Proses Citra bertujuan untuk mengubah citra dari RGB menjadi *grayscale*. Proses ini bertujuan supaya citra dapat mudah untuk di proses pada tahap pengolahan citra selanjutnya.

Proses perubahan dari citra RGB menjadi *grayscale* dapat dilihat di gambar 3.4

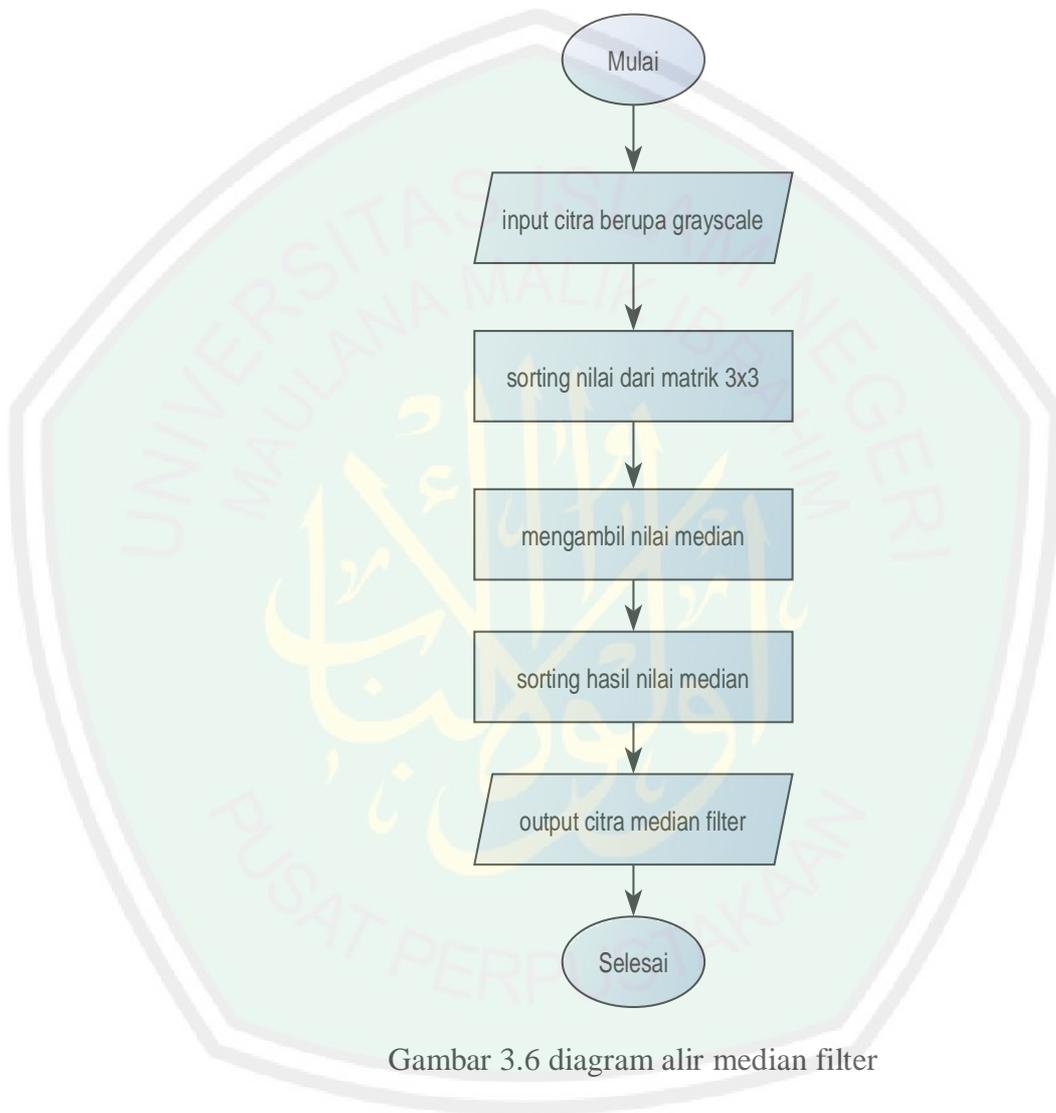


Gambar 3.5 Flowchart Proses *grayscale*

b. Filtering

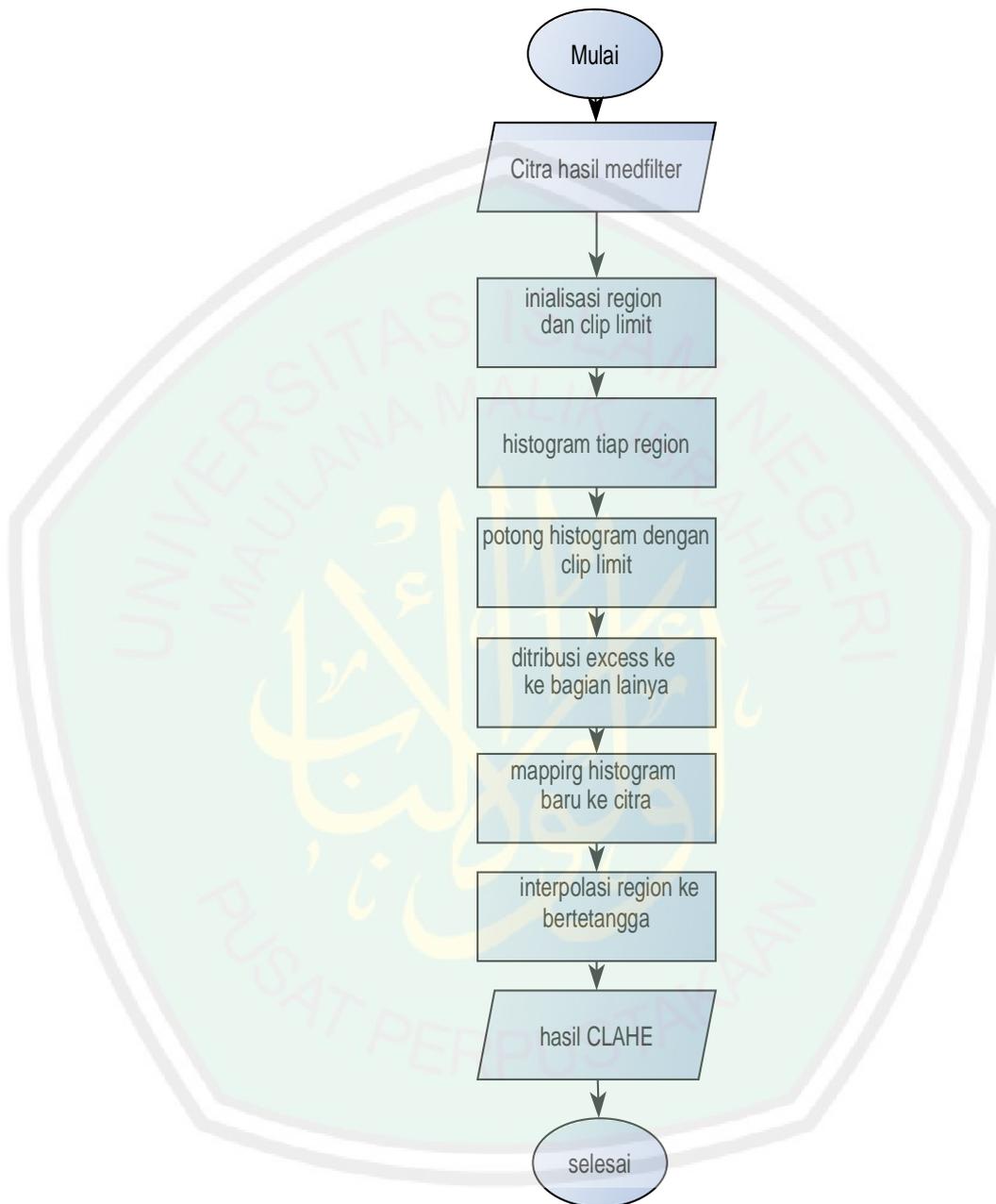
Proses filtering adalah proses yang bertujuan untuk meningkatkan kualitas dari citra. Dalam proses ini yang akan dilakukan adalah median filter. Proses ini bertujuan untuk mengurangi derau yang di akibatkan oleh derau acak seperti *salf and papper noise*, sehingga citra akan lebih mudah untuk diproses

pada tahap selanjutnya. Diagram alir dari median filter dapat di lihat pada gambar 3.6



Gambar 3.6 diagram alir median filter

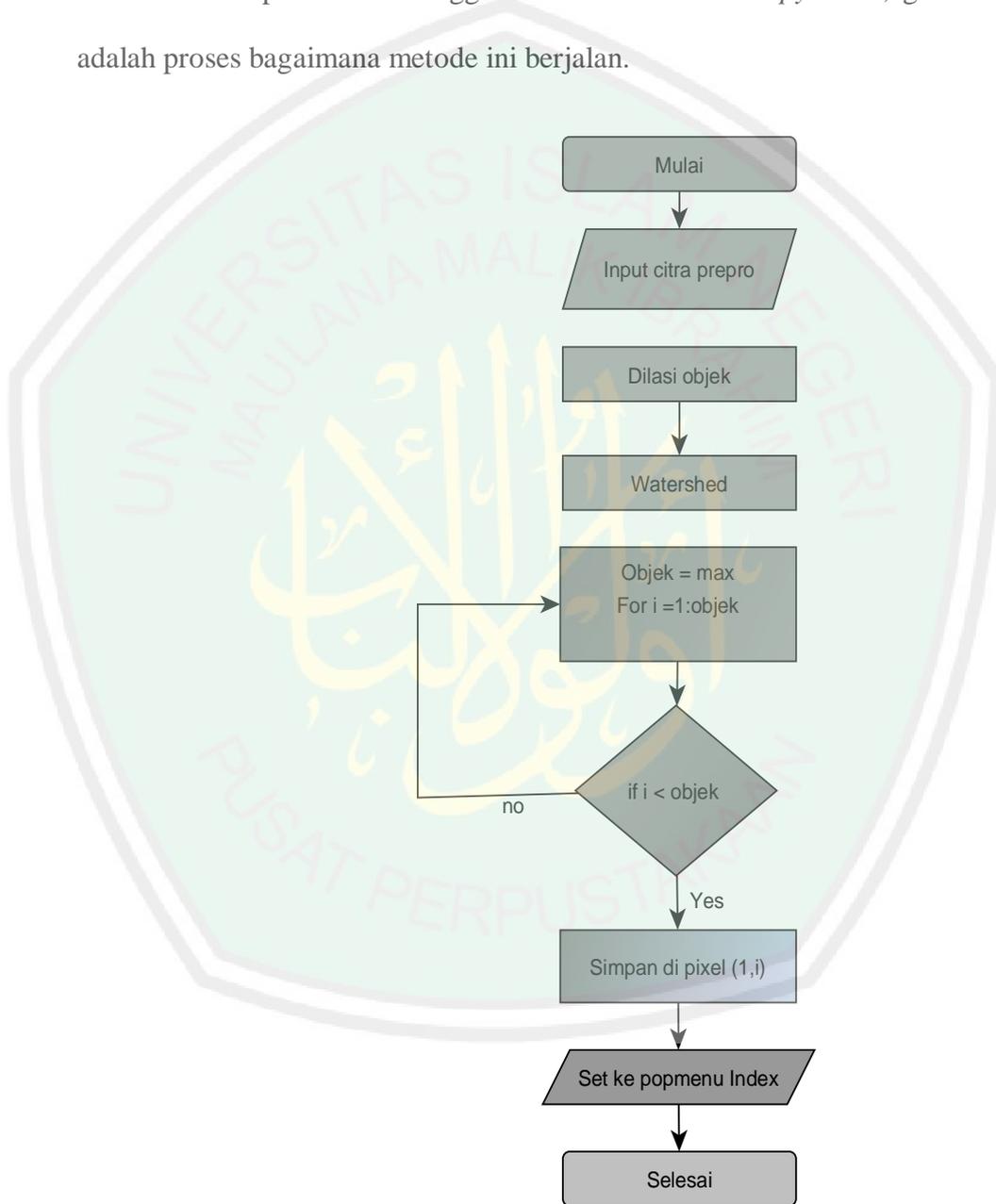
Proses filter selanjutnya yang akan dilakukan adalah CLAHE (Contrast Limited Adaptive histogram Equalization), filter ini bertujuan meningkatkan kecerahan dari citra sehingga citra dapat lebih mudah diproses ke tahap selanjutnya. Gambar 3.7 adalah diagram alir proses CLAHE



Gambar 3.7 Flowchart Proses CLAHE

3.2.3.3 Proses Segmentasi

Segmentasi adalah proses membagi objek citra menjadi objek-objek tertentu. Pada proses ini menggunakan metode *Homotopy tree*, gambar 3.8 adalah proses bagaimana metode ini berjalan.



Gambar 3.8 Flowchart Proses Segmentasi

Proses segmentasi diawali dengan mengubah citra menjadi *grayscale*. Setelah itu akan dilakukan pengenalan terhadap objek-objek yang dengan perbedaan warna derajat keabuan.

3.3 Desain *Interface* sistem

3.3.1 Desain Menu Utama

Desain *interface* sistem ini yang akan digunakan untuk memudahkan user dalam menjalankan aplikasi ini. Dalam desain aplikasi ini mempunyai beberapa menu yaitu Menu Utama Aplikasi, Menu about pembuat, Menu bantuan, Menu tentang pembuat.

FIL	ABOUT	HELP	EXIT			
OPEN		Original Citra	Hasil Proses	Hasil Segmentasi		
PROSES		Axes 1	Axes 2	Axes3		
SEGMENTA						
POPMENU						
VIEW						
RESET						

Gambar 3.9 Desain Interface Menu utama

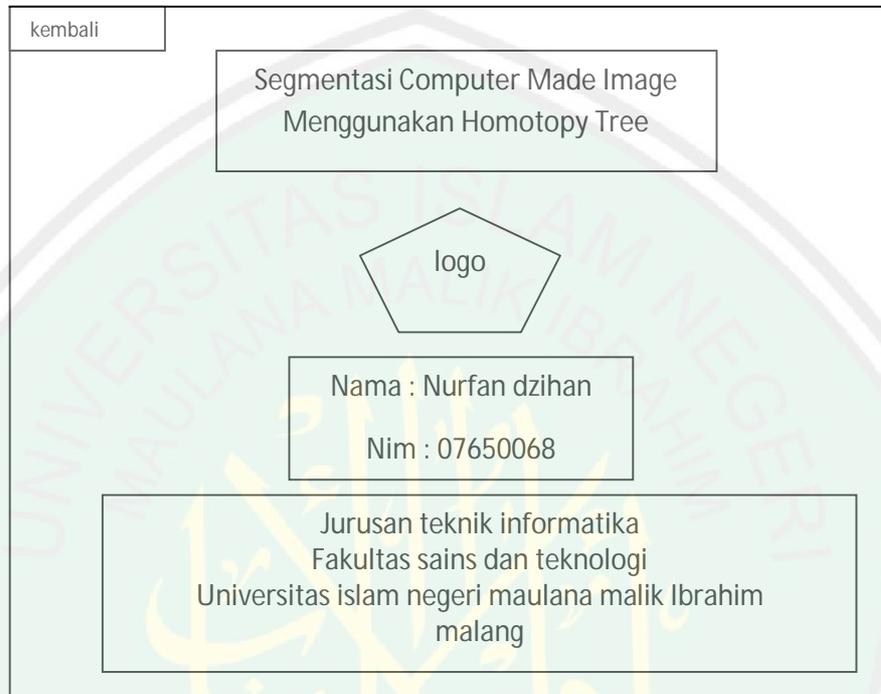
Penjelasan dari menu proses gambar 3.9 sebagai berikut :

- a. File : Menu ini berisi menu *save* yang digunakan untuk menyimpan citra hasil segmentasi
- b. About : Menu ini berfungsi untuk menuju ke menu informasi tentang pembuat program.
- c. Help : Menu ini berfungsi untuk melihat bantuan tentang penggunaan aplikasi
- d. Exit : Berfungsi untuk keluar dari program
- e. Open Image : *push button* berfungsi memanggil citra yang akan diproses.
- f. Proses : *push button* berfungsi proses preprocessing citra
- g. Segmentasi : *push button* yang berfungsi untuk proses segmentasi citra sehingga akan mendapatkan bagian yang diinginkan
- h. PopMenu : berfungsi untuk menampilkan jumlah objek dari hasil segmentasi dan terbentuk Tree.
- i. Axes 1,2,3 : berfungsi untuk menampilkan citra hasil. Axes1 untuk menampilkan citra asli, axes 2 menampilkan hasil citra hasil Proses, axes 3 menampilkan citra hasil Segmentasi.

3.3.2 Menu *About*

Pada menu ini hanya berisi menu kembali yang fungsinya kembali ke menu utama. menu about ini terdapat axes1 yang berfungsi untuk

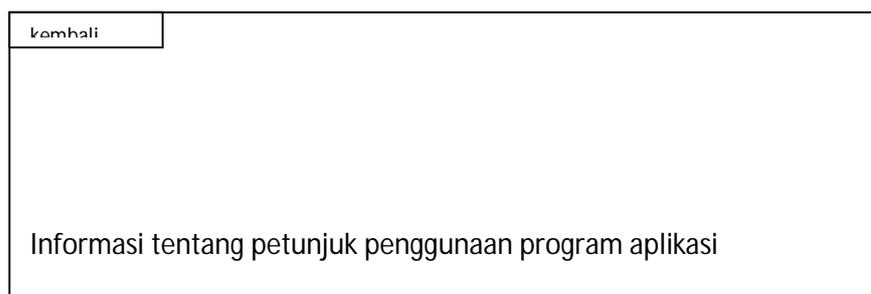
menampilakn logo. Selain itu, menu ini berisi tentang segala informasi tentang pembuat program penelitian Gambar 3.10 adalah desain menu about.



Gambar 3.10 Desain Gambar menu *about*

3.3.3 Menu *Help*

Pada menu ini hanya berisi tombol *back* yang berfungsi untuk kembali ke menu utama. Menu ini berisi tentang segala proses dan tata cara menggunakan aplkasu tersebut. Gambar 3.11 adalah desain dari menu bantuan.



Gambar 3.11 Desain gambar menu bantuan



BAB IV

HASIL DAN PEMBAHASAN

4.1 Lingkup Uji Coba

Pada subbab ini menjelaskan tentang lingkungan perangkat uji coba selama penelitian di lakukan. Perangkat yang digunakan adalah perangkat lunak dan keras. Spesifikasi perangkat keras dan lunak tersebut dapat dilihat dari table 4.1.

Perangkat Lunak	Perangkat Keras
Windows 8 pro Matlab 7.6.0 (r2008a)	Proc : AMD E350 1.6Ghz Ram: 4GB VGA : ATI 6310 Input : keyboard, mouse

Tabel 4.1 Spesifikasi Perangkat

4.2 Implementasi Desain dan Sistem

Implementasi desain dan sistem adalah proses pembangunan program “Aplikasi Segmentasi *Computer Made-Image* menggunakan Metode *Homotopy Tree*” yang telah dibahas pada bab III. Dalam sistem ini terdapat 2 proses yaitu proses *preprocessing* dan proses segmentasi.

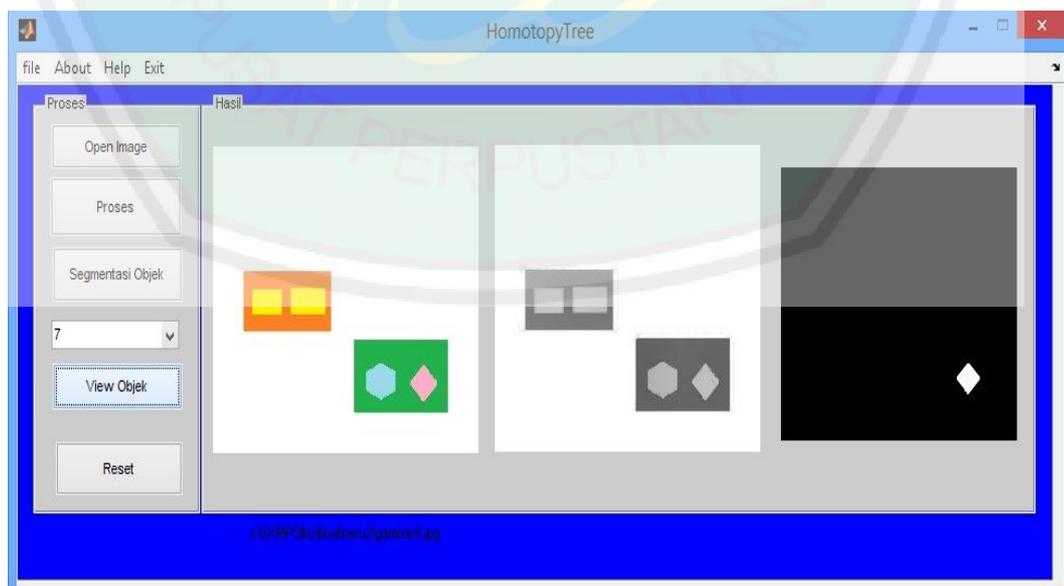
4.2.1 Implementasi Antarmuka dan Sistem Menu Utama

Antar muka menu utama aplikasi “Aplikasi Segmentasi Computer made image menggunakan *homotopy tree*” seperti pada **Gambar 4.1** merupakan tampilan yang muncul pertama kali setiap menjalankan aplikasi ini. Menu utama ini menghubungkan dengan menu-menu lain yang ada di dalam aplikasi. Pada tampilan antarmuka terdapat empat menu yaitu,

- a. Menu Bantuan : berisi cara penggunaan dari aplikasi sehingga *user* mengetahui menu-menu di dalam aplikasi.
- b. Menu Pembuat : berisi tentang pembuat aplikasi
- c. Menu Keluar : untuk keluar dari aplikasi

Dan terdapat lima *button* (tombol) untuk proses segmentasi citra yaitu,

1. Open image : untuk memilih atau memasukkan data citra geometri yang akan dilakukan proses segmentasi menggunakan *Homotpy tree*.
2. Proses : untuk menjalankan proses *preprocessing*.
3. Segmentasi : untuk menjalankan proses segmentasi.
4. Popmenu : Menampilkan jumlah objek dalam citra geometri
5. Reset : Menghapus citra pada axes1, axes2 dan axes3.
6. View objek : menampilkan objek yang di diinginkan.



Gambar 4.1 Antarmuka menu utama

Source code untuk memanggil menu About pembuat aplikasi dapat dilihat pada **Gambar 4.2**, menu *Help* dapat dilihat pada **Gambar 4.3**, dan menu *Exit* dapat dilihat pada **Gambar 4.4**.

Source code button Open Image dan *Simpan* dapat dilihat pada **Gambar 4.5**, **Gambar 4.6**. *Source code button Reset* dan *View Objek* dapat dilihat pada **Gambar 4.7** dan **Gambar 4.8**

```
function About_Callback(hObject, eventdata, handles)
delete(handles.figure1);
about=openfig('tentangpembuat.fig');
handles=guihandles(about);
guidata(about,handles);
% hObject    handle to About (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

Gambar 4.2 *Source code* untuk memanggil menu About pembuat aplikasi

```
function Help_Callback(hObject, eventdata, handles)
delete(handles.figure1);
help=openfig('bantuan.fig');
handles=guihandles(help);
guidata(help,handles);
% hObject    handle to Help (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

Gambar 4.3 *Source code* untuk memanggil menu *Help*

```
function Exit_Callback(hObject, eventdata, handles)
close(handles.figure1);
% hObject    handle to Exit (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

Gambar 4.4 *Source code* menu *Exit*

```

function Open_Callback(hObject, eventdata, handles)
global data
[nama_file,direktori]=uigetfile({'*.jpg;*.bmp;*.png;*.tiff;
*.gif;*.pgm'});
if ~isequal(nama_file,0);
    data=imread(fullfile(direktori,nama_file));
    axes(handles.axes1);
    imshow(data);
    guidata(hObject,handles);
else
    return;
end
set(handles.info,'String',fullfile(direktori,nama_file));
% hObject    handle to Open (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

```

Gambar 4.5 Source code button Open Image

```

function Simpan_Callback(hObject, eventdata, handles)
r = getframe(handles.axes3);
[im,map]=frame2im(r);
[Namafile, PathName] = uiputfile('*.jpg', 'Save As');
Name=fullfile(PathName, Namafile);
imwrite(im, Name, 'jpg');
% hObject    handle to Simpan (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

```

Gambar 4.6 Source code button Simpan

```

function Reset_Callback(hObject, eventdata, handles)
proyek=guidata(gcbo);
set(proyek.figure1,'CurrentAxes',proyek.axes1);
cla;
set(proyek.figure1,'CurrentAxes',proyek.axes2);
cla;
set (proyek.figure1,'CurrentAxes',proyek.axes3);
cla;
% hObject    handle to Reset (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

```

Gambar 4.7 Source Code button Reset

4.2.2 Implementasi *Preprocessing*

Implementasi *preprocessing* bertujuan memperbaiki citra supaya dapat diproses lebih lanjut dan menghasilkan nilai yang lebih baik. Tahapan *preprocessing* sesuai dengan rancangan sistem yang telah dijelaskan pada Bab 3. Implementasi *preprocessing* dapat dijelaskan sebagai berikut:

- ***Grayscale***

Citra *x-ray thorax* terlebih dahulu dilakukan proses normalisasi yaitu proses konversi citra RGB ke citra *grayscale*. Penggunaan global bertujuan agar data dari citra *x-ray thorax* dapat diakses oleh fungsi lain. Citra RGB diekstrak menjadi tiga nilai yaitu, nilai Red, nilai Green dan nilai Blue. Nilai R dikalikan dengan nilai 0.2989, nilai G dikalikan dengan 0.5870 dan nilai B dikalikan dengan 0.1140. Kemudian nilai ketiganya dijumlahkan dan menghasilkan citra *grayscale* dengan rentang nilai 0-255. **Gambar 4.8** merupakan *source code* dari proses normalisasi.

```
function Proses_Callback(hObject, eventdata, handles)
global data t
t = rgb2gray(data);
% hObject    handle to Proses (see GCBO)
% eventdata  reserved - to be defined in a future
version of MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

Gambar 4.8 *Source code grayscale*

- ***Filtering***

Proses *filtering* adalah proses untuk memperbaiki citra sebelum memasuki tahap segmentasi, misalnya untuk mengurangi *noise* dan memperkuat objek yang akan diproses. Ada dua filter yang digunakan yaitu,

a. *Median Filtering*

Median filter merupakan filter setelah proses *grayscale*. **Gambar 4.9** *source code* dari Median Filter. Median filter digunakan untuk mengurangi derau yang di akibatkan oleh derau acak seperti *salf and papper noise*.

```
function Proses_Callback(hObject, eventdata, handles)
global data t
t = medfilt2(data,[3 3]);
axes(handles.axes2);
imshow(t)
% hObject    handle to Proses (see GCBO)
% eventdata  reserved - to be defined in a future
version of MATLAB
% handles    structure with handles and user data
(see GUIDATA)
```

Gambar 4.9 *Source code Media Filtering*

Gambar 4.9 adalah gambar *source code* aplikasi untuk Median Filter. Perintah *medfilt2(data[3 3])* digunakan untuk merubah citra input menjadi citra yang lebih halus,. *Imshow(t)* merupakan perintah untuk menampilkan citra hasil proses *medfilt2*.

b. *Contrast Limited Adaptive Histogram Equalization*

CLAHE adalah filter untuk memperbaiki tingkat kecerahan dari citra.

Source code dari CLAHE dapat dilihat pada **Gambar 4.10**.

```
function Proses_Callback(hObject, eventdata, handles)
global data t
b=adapthisteq(data,'ClipLimit',0.05,'Range','full','Dist
ribution','exponential' );
axes(handles.axes2);
imshow(b);
% hObject    handle to Proses (see GCBO)
% eventdata  reserved - to be defined in a future
version of MATLAB
% handles    structure with handles and user data (see
GUIDATA)
```

Gambar 4.10 *Source code filter CLAHE*

Clahe adalah variabel fungsi, dan *adapthisteq* adalah fungsi dari CLAHE. Hasil adalah variabel dari hasil *high-boost filter* dan yang akan diperbaiki. *Cliplimit* adalah faktor kontras yang mencegah over-saturasi gambar khususnya di daerah-daerah homogen, karena banyak *pixel* jatuh dalam kisaran level yang sama yaitu abu-abu dan *cliplimit* memiliki nilai *default* yaitu 0.01. *Range* merupakan salah satu string, mengontrol berbagai data output, pilihannya ada dua yaitu *original* dan *full*, *default* dari *range* adalah *full*. *Distribution* untuk membentuk histogram, dapat memilih salah satu dari tiga string yaitu *uniform*, *rayleigh*, dan *exponential*. Hasil *preprocessing* dari citra geometri dapat dilihat pada **Gambar 4.11**.



Gambar 4.11 Contoh citra geometri setelah di *preprocessing*

4.2.3 Implementasi Segmentasi Objek

Setelah melalui proses *preprocessing*, dilakukan proses dilasi dari strel dan deteksi tepi supaya tepi objek jelas, setelah itu dilakukan watershed fungsinya adalah proses segmentasi objek dengan mencari nilai max dari citra setelah itu proses mendeteksi objek dengan mengenali nilai pixelnya dan setelah dikenali disimpan ke popmenu dalam bentuk index gambar sehingga bisa mengenali banyak objek dalam citra geometri. Proses segmentasi dapat dilihat pada **Gambar 4.12**. sebagai berikut :

```
function Segmentsi_Callback(hObject, eventdata, handles)
global t hasil
Canny = edge(t, 'canny');
SE = strel('disk', 3);
t = imdilate(Canny, SE);
hasil = watershed(t);
axes(handles.axes3);
kolom = max(hasil);
objek = max(kolom);
for i=1:objek
    pixel(1, i)= i;
    pixel(2, i)= 0;
end
set(handles.NilaiPixel, 'String', pixel(1,:));
% hObject handle to Segmentsi (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles handles to Segmentsi (see GCBO)
%   - 'uicontrol' handles to the uicontrol handles;
%   - 'axes' handles to the axes in the figure;
%   - 'image' handles to the image in the figure;
```

Gambar 4.12 *Source code Homotopy Tree*

Setelah proses segmentasi, proses menampilkan objek-objek dalam citra geometri dengan memanggil index atau Tree dari hasil segmentasi sehingga menghasilkan objek yang dipanggil akan menjadi putih (255) dan yang tidak dipanggil akan menjadi hitam (0), *source code* dapat dilihat pada **Gambar 4.13**

```

function View_Callback(hObject, eventdata, handles)
global hasil
img = get(handles.NilaiPixel, 'Value');
for i=1:size(hasil,1)
    for j=1:size(hasil,2)
        if hasil(i,j) == img
            objek(i,j) = 255;
        else objek(i,j) = 0;
        end
    end
end
axes(handles.axes3);
imshow(objek);

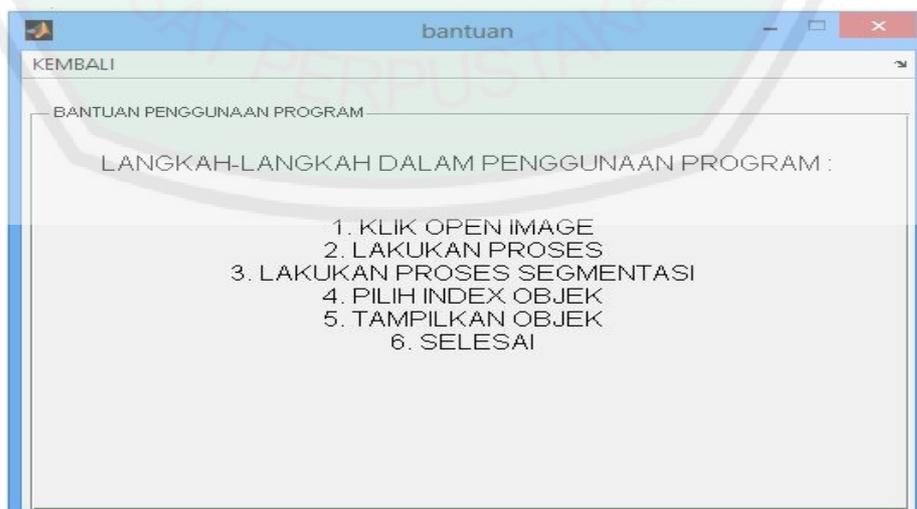
```

Gambar 4.13 Source code Menampilkan Objek

4.2.4 Implementasi *form* bantuan

Pada *form* ini menampilkan bantuan untuk menggunakan program ini, sehingga memudahkan pengguna. Dalam menu bantuan ini hanya terdapat menu kembali yang digunakan untuk kembali ke menu utama.

Berikut adalah *form help*

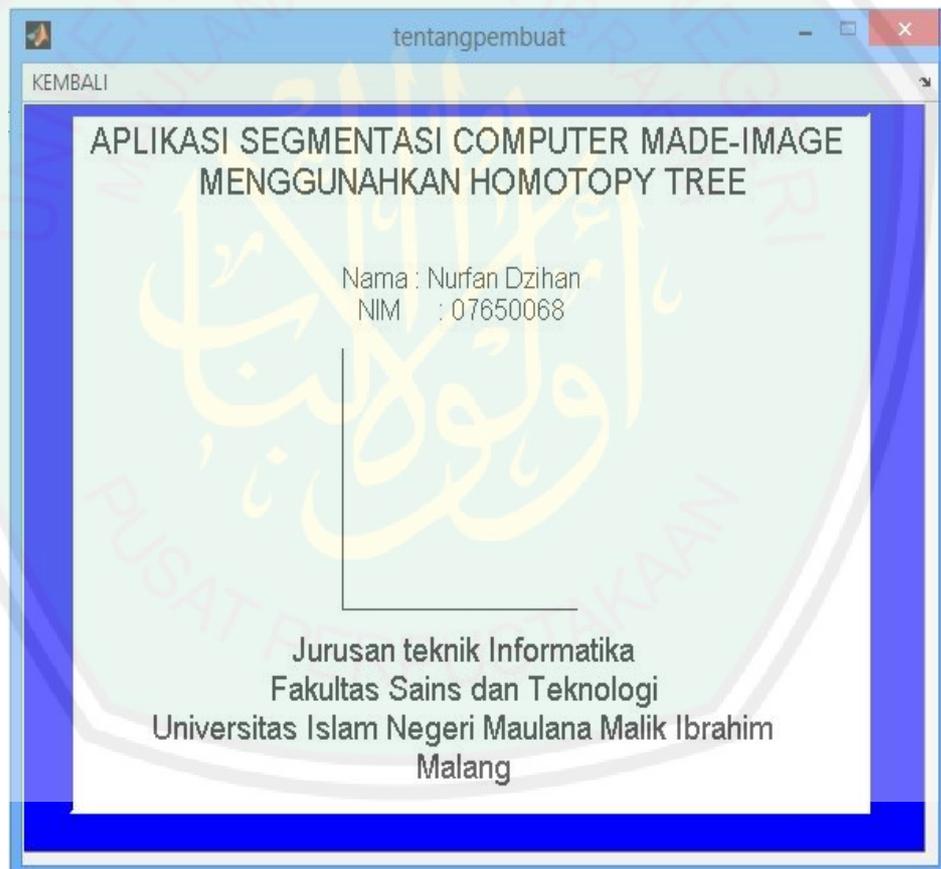


Gambar 4.14 form help

4.2.5 implementasi *form about*

Pada form ini berisi tentang profil dari pembuat dari aplikasi “Segmentasi *computer made-image* menggunakan *homotopy tree*”. Pada *form* ini hanya terdapat tombol kembali yang digunakan untuk kembali ke menu utama.

Berikut adalah *form* menu *about*

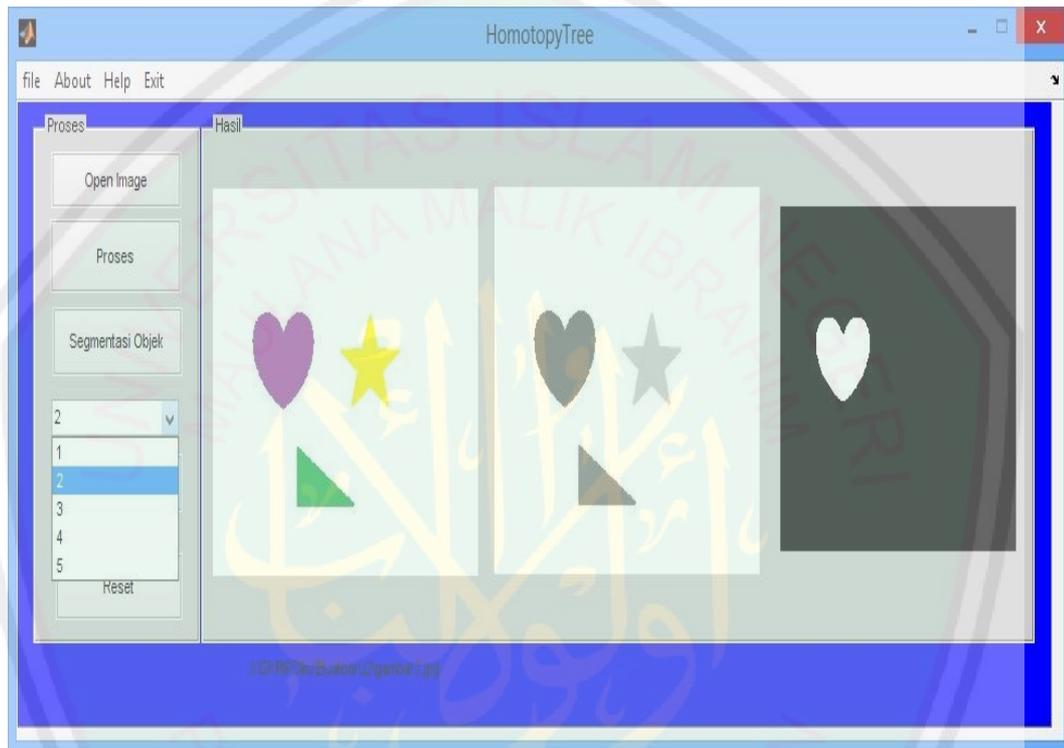


Gambar 4.15 *form* menu *about* pembuat aplikasi

4.3 Hasil uji coba segmetasi *computer made-image* menggunakan metode *homotopy tree*

Hasil dari uji coba ini adalah bagaimana mendapatkan objek-objek dalam citra geometri menggunakan metode homotopy tree

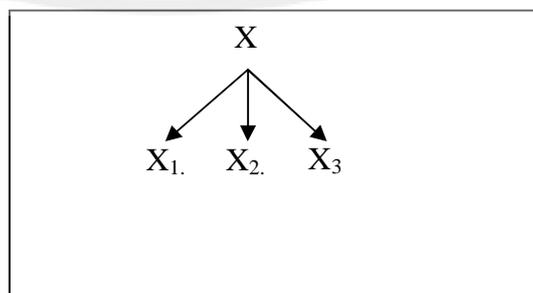
1. hasil Uji coba citra geometri dengan 3 objek pada Gambar 4.16



Gambar 4.16 Hasil Segmentasi *Homotopy Tree*

Dari gambar 4.16 telah terbentuk Tree, dapat di lihat pada Gambar

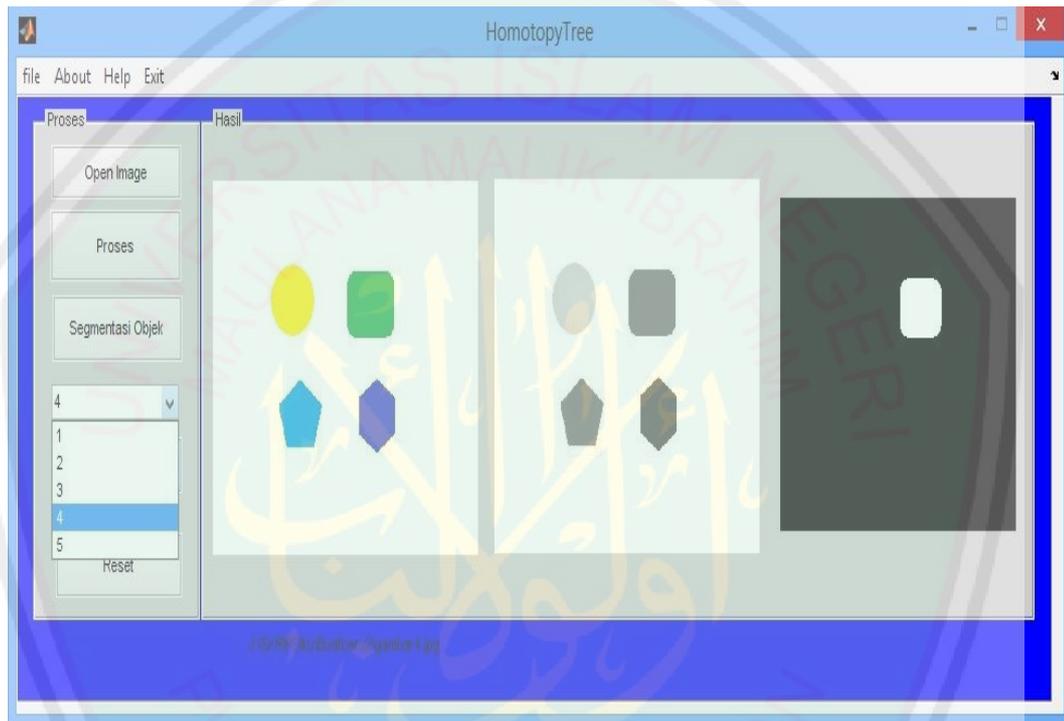
4.17



Gambar 4.17 Homotopy tree citra *input*

Dimana X adalah **Background** , X_1 adalah gambar bentuk hati, X_2 adalah gambar segitiga dan X_3 adalah gambar bintang

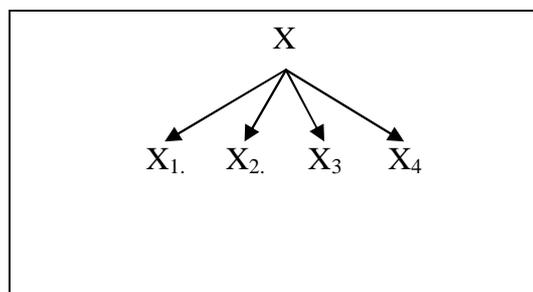
2. Hasil uji coba Citra geometri dengan 4 objek pada Gambar 4.18



Gambar 4.18 Hasil Segmentasi Dengan Metode Homotopy tree

Dari gambar 4.18 telah terbentuk Tree, dapat di lihat pada **Gambar**

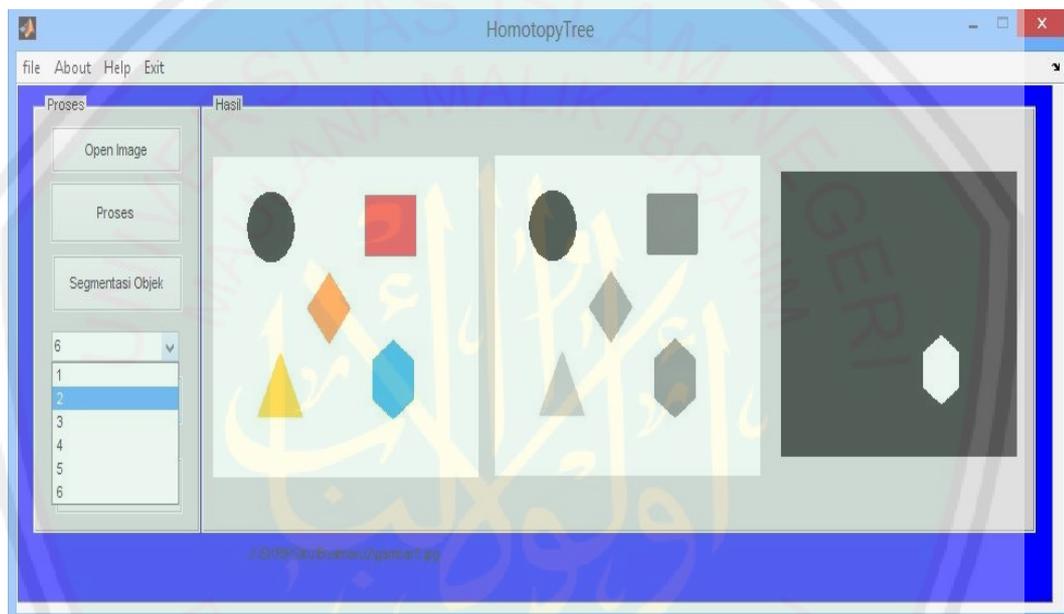
4.19



Gambar 4.19 Homotopy tree citra input

Dimana X adalah *Background* , X_1 adalah gambar bentuk lingkaran, X_2 adalah gambar bentuk limas warna biru muda dan X_3 adalah gambar bentuk kotak, dan X_4 adalah gambar bentuk limas warna biru Tua.

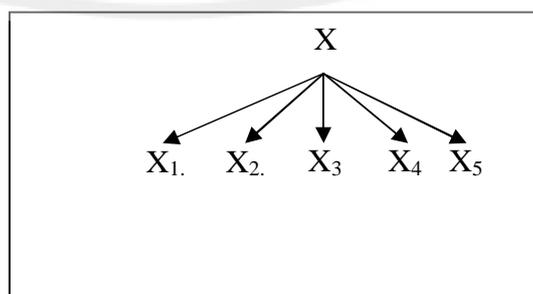
3. Hasil Uji coba citra geometri dengan 5 objek pada gambar 4.20



Gambar 4.20 Hasil Segmentasi Dengan Metode Homotopy tree

Dari gambar 4.20 telah terbentuk Tree, dapat di lihat pada Gambar

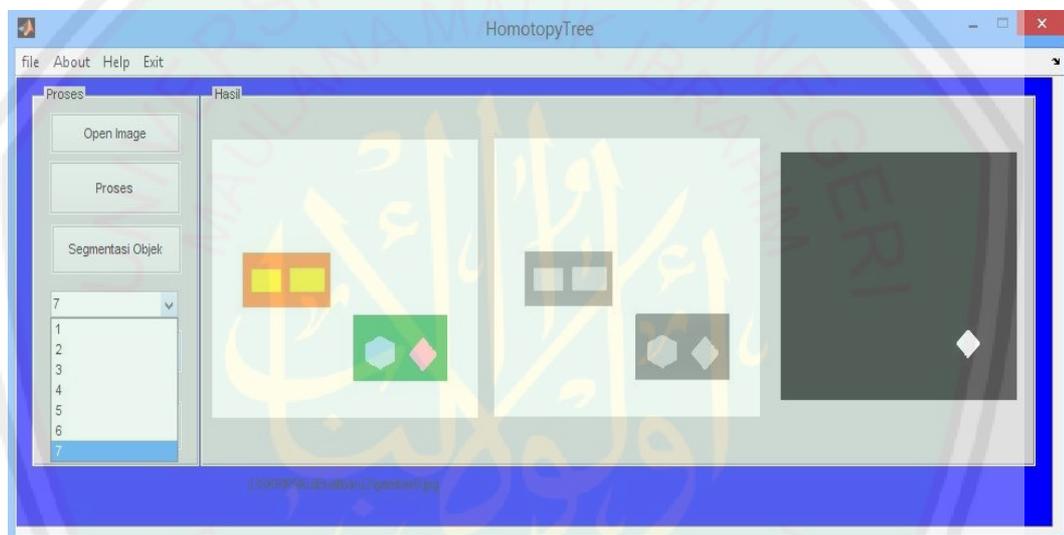
4.21



Gambar 4.21 Homotopy tree citra *input*

Dimana X adalah *Background* , X_1 adalah gambar bentuk lingkaran, X_2 adalah gambar bentuk segitiga dan X_3 adalah gambar bentuk ketupat, dan X_4 adalah gambar bentuk Kotak. Dan X_5 adalah gambar bentuk Limas segi lima

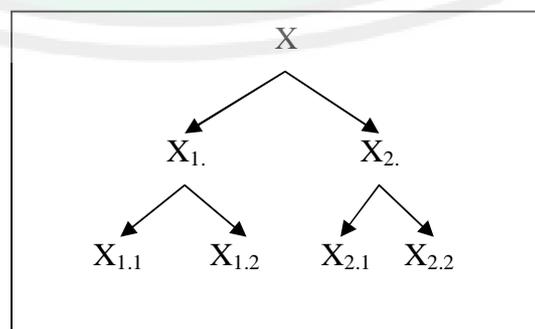
4. Hasil Uji coba citra geometri dengan 6 objek pada gambar 4.22



Gambar 4.22 Hasil Segmentasi Dengan Metode Homotopy tree

Dari gambar 4.22 telah terbentuk Tree, dapat di lihat pada **Gambar**

4.23



Gambar 4.23 Homotopy tree citra input

Dimana **X** adalah *Background* , **X1** adalah gambar bentuk Kotak Orange, **X2** adalah gambar bentuk Kotak hijau dan **X1,1** adalah gambar bentuk kotak warna kuning, dan **X1,2** adalah gambar bentuk Kotak warna kuning. Dan **X2,1** adalah gambar bentuk Limas segi lima, dan **X2,2** adalah bentuk ketupat.

4.4 Segmentasi menurut perspektif islam

Kemajuan sains dan teknologi telah memberikan kemudahan-kemudahan dan kesejahteraan bagi hidup manusia. Ilmu pengetahuan dan teknologi merupakan sosok yang tidak dapat dipisahkan satu sama lainnya. Ilmu adalah sumber teknologi yang mampu memberikan kemungkinan munculnya berbagai rekayasa dan ide-ide. Adapun teknologi atau aplikasi dari ilmu pengetahuan dapat ditunjukkan dalam hasil nyata yang lebih canggih dan dapat mendorong manusia untuk lebih maju lagi. Dan umat islam harus selalu mempelajari ilmu pengetahuan karena dasar-dasar filosofis untuk mengembangkan teknologi itu bisa dikaji dan digali didalam Al-Quran. Sebagaimana Rosululloh bersabda :

طَلِبُ الْعِلْمِ فَرِيضَةٌ عَلَى كُلِّ مُسْلِمٍ

Artinya:

“Menuntut ilmu wajib atas tiap muslim (baik muslimin maupun muslimah).” (Riwayat Ibnu Majah diambil dari *software* HaditsWeb)

Proses segmentasi untuk pembacaan objek tertentu merupakan teknologi komputer dalam dunia medis. Segmentasi adalah proses memisahkan suatu objek

yang satu dengan yang lain untuk mempermudah manusia menganalisis objek tertentu.

Islam telah menerangkan secara rinci dan jelas sekitar arkanul Islam, iman, Al-Qur'an, ilmu dan cabang-cabangnya, amal, dakwah kepada Allah, jihad, manusia dan hubungan kemasyarakatan, akhlak, peraturan yang berhubungan dengan harta, hal-hal yang berkaitan dengan hukum, negara dan masyarakat, pertanian dan perdagangan, sejarah dan kisah-kisah, agama-agama sebelum Islam yang terdapat dalam ayat-ayat Al-Qur'an dan yang tidak terdapat di dalam Al-Qur'an diterangkan secara rinci di dalam Al-Hadits. Karena Al-Hadits merupakan penjabaran makna tersurat dan tersirat dari isi kandungan Al-Qur'an.

Tidak ada satu amalan atau aturan yang mendatangkan kebajikan bagi umat manusia dalam kehidupan dunia dan akhirat melainkan telah dijelaskan di dalamnya. Tidak pula ada satu amalan pun yang membahayakan kehidupan mereka melainkan telah diperingatkan untuk ditinggalkan dan diajuhi, dan tidak ada satu pihak yang mampu menciptakan atau membuat aturan dan perundang-undangan selengkap, sesempurna, seadil, dan sejujur syariat Islam yang diturunkan oleh Allah, sebagaimana firman-Nya:

وَتَمَّتْ كَلِمَتُ رَبِّكَ صِدْقًا وَعَدْلًا لَا مُبَدِّلَ لِكَلِمَاتِهِ ۗ وَهُوَ السَّمِيعُ الْعَلِيمُ ﴿١١٥﴾

Artinya:

“Telah sempurnalah kalimat Tuhanmu (Al-Quran) sebagai kalimat yang benar dan adil. tidak ada yang dapat merobah robah kalimat-kalimat-Nya dan dia lah yang Maha Mendengar lagi Maha Mengetahui.” (QS. Al-An'am [6]: 115)

لَا يَأْتِيهِ الْبَطْلُ مِنْ بَيْنِ يَدَيْهِ وَلَا مِنْ خَلْفِهِ ۗ تَنْزِيلٌ مِّنْ حَكِيمٍ حَمِيدٍ ﴿٤٢﴾

Artinya:

“Yang tidak datang kepadanya (*Al Quran*) kebatilan baik dari depan maupun dari belakangnya, yang diturunkan dari Rabb yang Maha Bijaksana lagi Maha Terpuji.” (QS. Fushshilat [41]: 42)

Islam dengan jelas memisahkan suatu aturan yang satu dengan lain, supaya manusia sebagai khalifah di bumi dapat menjalankan segala sesuatu sesuai dengan syariat Islam. Jika dihubungkan dengan segmentasi maka keduanya memiliki hubungan yaitu memisahkan antara yang satu dengan yang lain.

Pemrosesan citra digital memerlukan satu proses preprocessing yang selanjutnya akan digunakan untuk proses yang lain. Proses tersebut adalah segmentasi. Segmentasi merupakan langkah pertama dan menjadi kunci yang penting dalam suatu pengenalan objek (*object recognition*). Proses segmentasi merupakan suatu proses untuk memisahkan antara satu obyek dengan obyek lainnya. Dengan proses segmentasi masing-masing obyek pada citra dapat diambil secara terpisah sehingga dapat digunakan sebagai masukan proses yang lain. Segmentasi citra merupakan suatu teknik pengelompokan (*clustering*) untuk citra. Dengan kata lain, merupakan suatu proses pembagian citra ke dalam wilayah (*region*) yang mempunyai kesamaan fitur antara lain : tingkat keabuan (*gray scale*), teksture(*texture*), warna(*color*), gerakan(*motion*). Dengan segmentasi kita dapat meneliti objek tertentu yang kita inginkan.



BAB V

PENUTUP

5.1 KESIMPULAN

Setelah dilakukan ujicoba berdasarkan aplikasi segmentasi *computer made image* menggunakan metode *homotopy tree* yang telah dilakukan, maka dapat ditarik kesimpulan sebagai berikut yaitu :

1. Sistem melakukan beberapa percobaan dan hasilnya dapat mengenali objek dengan baik dari citra yang telah dibuat dengan bantuan *software paint* yaitu bentuk bidang-bidang geometri.
2. Metode *homotopy tree* ini bisa memisahkan objek dalam satu *background*.
3. Mampu mengenali objek-objek yang ada dalam citra geometri.

5.2 SARAN

Pengembangan aplikasi ini masih memiliki banyak kekurangan dan memiliki keterbatasan yang dapat dijadikan acuan dalam pengembangan penelitian di masa yang akan datang, sehingga dapat disarankan beberapa hal sebagai berikut :

1. Tahap *preprocessing* pada sistem ini hanya menggunakan *Median filtering* dan *adaphisteq*, sehingga perlu adanya pengembangan untuk memilih *preprocessing* yang lebih baik.

2. Aplikasi ini hanya terbatas memiliki kemampuan untuk segmentasi objek dalam citra geometri sehingga perlu adanya pengembangan aplikasi pada pemilihan objek Citra yang lebih kompleks misalnya citra hasil *rontgen*.



DAFTAR PUSTAKA

- Munir, Rinaldi. 2004. Pengolahan Citra Digital Dengan Pendekatan Algoritmik. Bandung : Informatika Bandung
- Prasetyo, Eko. 2011. Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab: Yogyakarta : ANDI Yogyakarta.
- Sutoyo, T. Mulyanto, edy. Suhartono, Vincent. Dwi Mulyati, Oky. Wijanarto. Teori Pengolahan Citra Digital, Yoyakarta: ANDI Yogyakarta.
- Kiptiyah, M. Si. 2007. Embriologi Dalam Al-Qur'an "Kajian Pada Proses Penciptaan Manusia". Malang, UIN-Malang Press
- Wijaya, Marvin CH dan Prijono, Agus. 2007. Pengolahan Citra Digital Menggunakan Matlab. Bandung : Informatika Bandung
- Van Ginneken, B. 2001. Computer-Aided Diagnosis in Chest Radiography. Belanda
- Gonzalez Rafael C and Richard E. Woods. 2009. Digital Image Processing Using MATLAB. Amerika : Gatesmark Publishing
- Pierre J. Garrigues and Laurent El Ghaoui Robin Strand. An Homotopy Algorithm for the Lasso with Online Observations. Digital Geomethry and Mathematical Morphology
- Karl Schmedders. Homotopy Path-Following with Easy Homotopy : Solving Nonlinear Equations for Economic Model
- Nicholas Sven Shorter. Unsupervised Building Detection From Irregularly Spaced Lidar and Aerial Imagery
- Ranwez Vincent and Soille Pierre. 2001. Order Independent Homotopic Thinning for Binary and Grey Tone Anchored Skeletons. LIRMM, Dpt. Informatique Fondamentale et Applications 161 rue Ada, F-34392 Montpellier. France and Italy
- Keshet Renato. 2004. Homotopy Semilattices. Journal of HP Laboratories Israel

R. M. Noorullah and A. Damodaram, Innovative Thinning And Gradient Algorithm For Binary And Grey Tone Images Using First In First Out Linear Data Structure

Soille, P. 2004, Morphological Image Analysis : Principle and Application, Springer-Verlag, Berlin and New York.

Deschenes, F., dan Ziou, D., 2003, Homotopy-Based Computation of Defocus Blur and Affine Transform, Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03).

Dasgupta, N., Ji, S., Carin, L., 2006, Homotopy-Based Semi Supervised Hidden Markov Tree For Texture Analysis, in Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

http://en.wikipedia.org/wiki/Paint_software

diakses pada tanggal 10 Juli 2013

<http://kuliahinformatika.wordpress.com/2010/02/13/buku-ta-thresholding-citra/>
diakses pada tanggal 28 Maret 2011