

**Segmentasi Paru-Paru pada Citra X-Ray Rongga Dada
Menggunakan Metode Active Shape Model Untuk Mengetahui
Diameter Maksimal Paru-Paru**

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:

DIDIK TRI SUSANTO

NIM. 07650076



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

HALAMAN PERSETUJUAN

Segmentasi Paru-Paru pada Citra X-Ray Rongga Dada Menggunakan Metode Active Shape Model untuk Mengetahui Diameter Maksimal Paru-Paru

SKRIPSI

Oleh:

Didik Tri Susanto

NIM: 07650076

Telah disetujui oleh:

Pembimbing I

Pembimbing II

Ir. M. Amin Hariyadi, M. T
NIP. 196701182005011001

Muhammad Faisal, M. T
NIP. 197405102005011007

Januari 2013

Mengetahui,

Ketua Jurusan Teknik Informatika

Ririen Kusumawati, M. Kom
NIP.197203092005012002

HALAMAN PENGESAHAN

**Segmentasi Paru-Paru pada Citra X-Ray Rongga Dada
Menggunakan Metode Active Shape Model untuk Mengetahui
Diameter Maksimal Paru-Paru**

SKRIPSI

Oleh:
DIDIK TRI SUSANTO
NIM. 07650076

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

Tanggal : 10 Januari 2013

Susunan Dewan Penguji	Tanda Tangan
1. Penguji Utama : <u>Dr. Cahyo Crysdian</u> NIP. 19740424 200901 1 008	()
2. Ketua : <u>Zainal Abidin, M.Kom</u> NIP. 19760613 200501 1 004	()
3. Sekretaris : <u>Ir. M. Amin Hariyadi, M.T</u> NIP. 1967018 200501 1 001	()
4. Anggota : <u>Muhammad Faisal, M. T</u> NIP. 19740510 200501 1 007	()

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 : Didik Tri Susanto

NIM : 07650076

Jurusan : Teknik Informatika

Judul Skripsi : Segmentasi Paru-Paru pada Citra X-Ray Rongga Dada
Menggunakan Metode Active Shape Model untuk Mengetahui
Diameter Maksimal Paru-Paru

Dengan ini menyatakan bahwa:

1. Isi dari skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama termaktub di isi dan tertulis di daftar pustaka dalam skripsi ini.
2. Apabila di kemudian hari ternyata skripsi saya terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Malang, 28 Januari 2013
Yang menyatakan,

Didik Tri Susanto
NIM: 07650076

MOTTO

*“Faktanya kita sering terjebak dengan kata **masih ada hari esok**, terjebak dengan keengganan bertindak karena terlena dengan yang namanya **waktu luang**”*

HALAMAN PERSEMBAHAN

Sujud syukurku kepada Allah Azza Wa Jalla yang telah merahmatkan banyak karunia-Nya kepada seluruh ciptaan-Nya. Sungguh Engkau Maha Mengetahui dan tiada hamba menguasai ilmu-Mu kecuali sangat sedikit. Maka hamba berharap Skripsi ini bisa mengantarkan hamba bisa lebih dekat dengan-Mu. Tentu tak lupa sholawat dan salam untuk Kekasih-Mu Rasulullah Muhammad SAW.

Untuk siapa saja karya ini dipersembahkan?

Sebenarnya kurang pantas jika karya ini dipersembahkan untuk mereka. Tapi sekasual mungkin saya mendedikasikan karya ini untuk Ayah dan Ibu yang sudah banyak berkorban agar saya bisa terus eksis di dunia perkuliahan. Tak lupa ada mbak dan mas ipar, keponakan, sepupu, dan keluarga besar tercinta di Lamongan sana.

Apa yang terjadi adalah saya mengerjakan riset ini sampai malnutrisi dengan jayanya bersama tim riset yang diantaranya ada Ucho, Ipit, Bara, Uma, Chika, Ratri, Cunti, Nurfan, dan Rina. Setelah perjuangan yang banyak mengandung unsur absurd akhirnya kita berhasil. Karya ini untuk kalian sobat!

Penghuni KOM sekaligus sesepuh DPA IOC apa kabar? Semoga yang belum bisa menyusul dan sukses bareng-bareng. Colek penghuninya, ada Jona, Ipunk, Eko, Alfi, Muse, Afif, Hento, Perdana, Aji, Nurfan, pank2, Mbah Miz, dan tak lupa Feby. Yes, you're awesome, all of you.

Laskar skripsi di tahun-tahun krusial ternyata jumlahnya lumayan. Ada Farhan, Susi, Ieta, Kilat, Fajri, dan semua dari angkatan 2007, 2008, dan 2009. Bahkan bebek jalan kayang pun usahanya masih lebih keren kalian kok.

Teman-teman redaksi di bisakomputer[dot]com, apa perlu skripsi ini publish di sana? Seriously, sepertinya tidak dan saya bangga bisa hadir di tengah-tengah orang kreatif seperti kalian.

Finally, buat para dosen khususnya dosen pembimbing skripsi, dosen wali akademik, karya ini adalah salah satu kontribusi terbesar dari beliau-beliau ini. Jazakumullah khairon.

Jadi untuk siapa lagi karya ini dipersembahkan?

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah melimpahkan karunia-Nya kepada penulis sehingga bisa menyelesaikan skripsi dengan judul “Segmentasi Paru-Paru pada Citra *X-Ray* Rongga Dada Menggunakan Metode *Active Shape Model* untuk Mengetahui Diameter Maksimal Paru-Paru”.

Sholawat dan salam semoga selalu tercurah kepada Nabi Muhammad SAW karena bimbingan beliau Islam telah disampaikan dan membawa umat dari kegelapan menuju zaman yang terang benderang.

Penyelesaian skripsi ini akan sulit terwujud tanpa bantuan dan sumbangsih berbagai pihak karena penulis memiliki keterbatasan kemampuan dan pengetahuan. Untuk itu, dengan segala kerendahan hati maka penulis mengucapkan terima kasih kepada:

1. Ir. M. Amin Hariyadi, M.T selaku pembimbing I yang telah meluangkan banyak waktu untuk membimbing dan mengarahkan dalam penyelesaian skripsi.
2. Muhammad Faisal, M.T selaku pembimbing II yang membantu penulis dalam integrasi Islam dan keilmuan, serta banyak memberikan petunjuk dan nasehat dalam penyelesaian skripsi.
3. Dr. Cahyo Chrysdian selaku dosen *computer vision* yang telah banyak memberi masukan dan dukungan pada pengerjaan skripsi ini.
4. Segenap Dosen Teknik Informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.

5. Keluarga besar Perpustakaan Pusat Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan dukungan pustaka kepada penulis untuk menyelesaikan penyusunan skripsi.
6. Rekan-rekan tim penelitian *image processing* yang sudah banyak memberikan dukungan dan bahu membahu dalam menyelesaikan proyek skripsi.
7. Bapak, ibu, kakak, keponakan, dan segenap keluarga besar penulis yang banyak berkontribusi dan memotivasi penulis untuk menyelesaikan skripsi.
8. Teman-teman jurusan Teknik Informatika dan untuk seluruh civitas akademika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah banyak membantu perjuangan dan berbagi pengalaman dalam kehidupan sehari-hari.
9. Semua pihak yang tidak mungkin penulis sebutkan satu persatu, atas segala yang telah diberikan kepada penulis dan dapat menjadi pelajaran.

Sebagai penutup, penulis menyadari masih banyak kekurangan dalam skripsi ini. Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya. Harapan penulis selanjutnya adalah semoga karya ini bermanfaat bagi kita semua. Aamiin.

Malang, 28 Januari 2013

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TULISAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xii
ABSTRAK	xiv
BAB I: PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	5
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
1.6 Metode Penelitian	6
1.7 Sistematika Penulisan Skripsi	7
BAB II: TINJAUAN PUSTAKA	9
2.1 Penelitian Terdahulu	9
2.2 Pengolahan Citra Digital	11
2.2.1 Pengertian Citra Digital	11
2.2.2 <i>Image Processing</i>	12
2.2.3 Perbaikan Citra (<i>Image Enhancement</i>)	14
2.2.3.1 Histogram	14
2.2.3.2 Histogram <i>Equalization</i>	16
2.2.3.3 Konvolusi	17
2.2.3.4 Penskalaan	18
2.3 Segmentasi Citra	19
2.3.1 Pengambangan (<i>Thresholding</i>)	19
2.3.2 Segmentasi Berbasis <i>Clustering</i>	20
2.3.3 <i>Template Matching</i>	21
2.4 <i>Active Shape Model</i>	21
2.4.1 <i>Point Distribution Model</i>	22
2.4.1.1 Pelabelan <i>Training Set</i>	22
2.4.1.2 <i>Aligning Shape</i>	22
2.4.1.3 Model Statistik	24
2.4.2 <i>Image Search</i>	26
2.4.2.1 Inisialisasi Awal <i>Shape</i>	26

2.4.2.2 Perhitungan <i>Suggested Movement</i>	27
2.4.2.3 Perubahan <i>Pose</i> dan <i>Shape Parameter</i>	28
2.4.2.4 Pembaruan <i>Pose</i> dan <i>Shape Parameter</i>	30
2.5 Validasi Segmentasi	31
2.6 Pencitraan <i>X-Ray</i>	32
2.7 Paru-Paru	33
2.8 <i>Cardiothoracic Ratio</i>	34
BAB III: ANALISIS DAN PERANCANGAN SISTEM	36
3.1 Deskripsi Sistem	36
3.2 Desain Sistem	36
3.3 Desain Data	39
3.3.1 Data Masukan	39
3.3.2 Data Proses	40
3.3.3 Data Keluaran	40
3.4 Desain Proses	40
3.4.1 Desain <i>Preprocessing Image</i>	41
3.4.2 Desain Proses Inisialisasi Model	43
3.4.3 Desain Proses Pencarian Kontur Paru-Paru	44
3.4.4 Desain Pengukuran Diameter Maksimal Paru-Paru	46
3.5 Desain Perhitungan Validasi Segmentasi	49
BAB IV: HASIL PEMBAHASAN	50
4.1 Lingkungan Uji Coba	50
4.2 Implementasi Sistem	50
4.2.1 <i>Form</i> Utama Aplikasi	51
4.2.2 Implementasi <i>Preprocessing</i> Citra	51
4.2.2.1 Implementasi Proses <i>Grayscale</i>	52
4.2.2.2 Implementasi Proses <i>Resizing</i>	53
4.2.2.3 Implementasi Proses <i>Histogram Equalization</i>	54
4.2.2.4 Implementasi Proses <i>Filtering</i>	57
4.2.2.5 Implementasi <i>Preprocessing</i> pada <i>Form</i> Utama	59
4.2.3 Implementasi Inisialisasi Model	61
4.2.4 Implementasi Pencarian Kontur Paru-Paru	62
4.2.5 Implementasi Segmentasi Hasil Pencarian Kontur	65
4.2.6 Implementasi Perhitungan Diameter Maksimal Paru-Paru	68
4.3 Implementasi Perhitungan Validasi	69
4.4 Hasil Uji Coba Validasi Segmentasi Paru-Paru	70
4.5 Hasil Uji Coba Diameter Maksimal Paru-Paru	78
4.6 Integrasi Penelitian dalam Islam	79

BAB V: PENUTUP	82
5.1 Kesimpulan	82
5.2 Saran	82
DAFTAR PUSTAKA	83
LAMPIRAN	84



DAFTAR TABEL

Tabel 4.1 Spesifikasi Lingkungan Uji Coba	50
Tabel 4.2 Hasil Uji Coba Segmentasi Paru-Paru Kanan dengan 15 Landmark....	71
Tabel 4.3 Hasil Uji Coba Segmentasi Paru-Paru Kanan dengan 20 Landmark....	72
Tabel 4.4 Hasil Uji Coba Segmentasi Paru-Paru Kanan dengan 25 Landmark....	73
Tabel 4.5 Hasil Uji Coba Segmentasi Paru-Paru Kiri dengan 18 Landmark... ..	74
Tabel 4.6 Hasil Uji Coba Segmentasi Paru-Paru Kiri dengan 23 Landmark... ..	75
Tabel 4.7 Hasil Uji Coba Segmentasi Paru-Paru Kiri dengan 28 Landmark... ..	76
Tabel 4.8 Perbandingan Rata-Rata Validasi Segmentasi Paru-Paru Kanan	77
Tabel 4.9 Perbandingan Rata-Rata Validasi Segmentasi Paru-Paru Kiri	77
Tabel 4.10 Hasil Uji Coba Perhitungan Diameter Maksimal Paru-Paru	78

DAFTAR GAMBAR

Gambar 2.1 Sistem koordinat pada citra digital.....	12
Gambar 2.2 Histogram citra <i>grayscale</i> 8 bit	15
Gambar 2.3 <i>Histogram Equalizations</i>	17
Gambar 2.4 Ilustrasi operasi konvolusi dengan kernel 3 x 3	17
Gambar 2.5 <i>Sample</i> profil normal sepanjang tepi objek pada setiap titik model	27
Gambar 2.6 Perbedaan antara citra asli dan citra hasil segmentasi.....	31
Gambar 2.7 Sampel citra <i>x-ray</i> rongga dada	33
Gambar 2.8 Gambar garis bantu untuk perhitungan CTR	34
Gambar 3.1 Diagram blok aplikasi segmentasi paru-paru	37
Gambar 3.2 Diagram alir sistem secara umum	38
Gambar 3.3 Desain <i>form</i> aplikasi utama	39
Gambar 3.4 Diagram alir <i>histogram equalization</i>	42
Gambar 3.5 Diagram alir penerapan konvolusi untuk <i>filtering</i>	43
Gambar 3.6 Diagram alir inisialisasi model.....	44
Gambar 3.7 Diagram alir proses <i>image search</i>	45
Gambar 3.8 Kotak pada model untuk membantu proses perhitungan diameter maksimal paru-paru	47
Gambar 3.9 Diagram alir pengukuran diameter maksimal paru-paru.....	47
Gambar 3.10 Diagram alir perhitungan validasi hasil segmentasi.....	49
Gambar 4.1 Implementasi <i>form</i> utama aplikasi	51
Gambar 4.2 Hasil <i>grayscale</i> pada citra masukan.....	52
Gambar 4.3 Hasil <i>histogram equalization</i> pada citra <i>grayscale</i>	56
Gambar 4.4 Hasil <i>filter sharpening</i> pada citra <i>histogram equalization</i>	57
Gambar 4.5 Hasil <i>filtering emboss</i> sekaligus tahap akhir <i>preprocessing</i>	58
Gambar 4.6 Kotak dialog untuk memilih data citra.....	59
Gambar 4.7 Hasil <i>preprocessing</i> citra pada panel utama.....	60
Gambar 4.8 Jendela dialog untuk menentukan jumlah <i>landmark</i> model.....	61
Gambar 4.9 Inisialisasi model pada paru-paru kanan	62
Gambar 4.10 Kotak dialog konvergensi pada proses pencarian kontur	63
Gambar 4.11 Hasil proses pencarian kontur untuk paru-paru kanan	64
Gambar 4.12 Hasil pencarian kontur untuk kedua kontur paru-paru	65
Gambar 4.13 Implementasi <i>form</i> dan hasil segmentasi	67
Gambar 4.14 Penempatan kotak untuk perhitungan diameter	68
Gambar 4.15 Informasi diameter maksimal paru-paru	69

ABSTRAK

Susanto, Didik T. 2013. **Segmentasi Paru-Paru pada Citra X-Ray Rongga Dada Menggunakan Metode Active Shape Model Untuk Mengetahui Diameter Maksimal Paru-Paru**. Skripsi, Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (1) Ir. M. Amin Hariyadi, M.T (2) Muhammad Faisal, M.T.

Kata Kunci: Segmentasi citra digital, X-Ray, Active Shape Model, Diameter maksimal

Segmentasi citra merupakan teknik untuk membagi suatu citra menjadi beberapa daerah (*region*) di mana setiap daerah memiliki kemiripan atribut. Segmentasi digunakan untuk keperluan analisis citra seperti pengenalan pola atau pengambilan informasi terkait citra segmentasi. *Active Shape Model* merupakan *statistical model* dari kumpulan *shape* yang terdiri dari *landmark* yang membentuk garis batas dari suatu objek sehingga dapat dimanfaatkan untuk segmentasi citra.

Penelitian ini adalah melakukan segmentasi paru-paru pada citra *x-ray* rongga dada menggunakan metode *Active Shape Model* dan dari hasil segmentasi dapat diketahui diameter maksimal paru-paru. Segmentasi dilakukan dengan mencari tepi kontur paru-paru dengan menempatkan model di dalam kontur. *Landmark* model yang digunakan adalah 15, 20, dan 25 *landmark* untuk paru-paru kanan, sedangkan paru-paru kiri menggunakan 18, 23, dan 28 *landmark*. 20 data citra percobaan didapatkan dari *public image databases* di <http://www.via.cornell.edu/databases>. Segmentasi dapat dilakukan dengan tingkat akurasi tertinggi 95,3455% untuk paru-paru kanan dengan jumlah *landmark* 25 dan 96,2363% untuk paru-paru kiri dengan jumlah *landmark* 28. Diameter maksimal paru-paru didapatkan dari jarak koordinat terluar hasil segmentasi dengan satuan *pixel*.

ABSTRACT

Susanto, Didik T. 2013. **Lung Segmentation on Chest X-Ray Image Using Active Shape Model to Determine Maximum Diameter of Lung.** Thesis. Information Technology Department Faculty of Science and Technology. Advisor: (1) Ir. M. Amin Hariyadi, M.T (2) Muhammad Faisal, M.T.

Keywords: image segmentation, chest x-ray, Active Shape Model, maximum diameter

Image segmentation is a technique for dividing an image into several regions (region) where each region have similar attributes. Segmentation used for image analysis such as pattern recognition or image retrieval related information segmentation. Active Shape Model is a statistical shape models from a landmarks collection that form the boundary of an object that can be used for image segmentation.

This research is to segment lung from chest x-ray radiograof using Active Shape Models methods and the results of segmentation id to determine maximum diameter of lungs. Segmentation is done by searching the edge contour of the lungs by placing model in the contour. Landmark models used are 15, 20, and 25 landmarks for the right lung, while the left lung using 18, 23, and 28 landmarks. 20 experimental image data obtained from public databases in <http://www.via.cornell.edu/databases> image. Segmentation can be performed with the highest degree of accuracy 95.3455% for the right lung by the number of landmark 25 and 96.2363% for the left lung with landmark number 28. Maximum diameter of the lung obtained from outer coordinates with a unit pixel segmentation results.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ilmu modern dan teknologi modern memakai matematika sebagai bahasa untuk menyatakan hukum alam secara tepat. Komputer digital elektronik merupakan alat yang sangat berharga untuk mempelajari akibat dari hukum tersebut. Seringkali prosedur eksak untuk memecahkan persoalan telah ditemukan, tetapi waktu yang diperlukan untuk melakukan perhitungan secara manual tidak memungkinkannya. Kadang kita perlu memecahkan persoalan yang sama berkali-kali dengan parameter yang berbeda, dan komputer sangat berguna untuk memecahkan permasalahan jenis ini.

Komputer tidak hanya mampu menghitung rumus-rumus matematis dengan kecepatan tinggi, tetapi juga sekumpulan perhitungan dilakukan berulang dengan kumpulan harga numerik yang berbeda, komputer bisa membandingkan hasilnya dan menentukan harga optimum yang dipakai. (Bartee, 1994).

Penggunaan teknologi dapat dianalogikan seperti pada *Al-Qur'an* surat *Al-Anbiya* ayat ke-80 yang berbunyi:

وَعَلَّمْنَاهُ صَنْعَةَ لَبُوسٍ لَّكُمْ لِيُحْصِنَكُمْ مِنْ بَأْسِكُمْ ۖ فَهَلْ أَنْتُمْ شَاكِرُونَ ﴿٨٠﴾

Artinya: “Dan telah Kami ajarkan kepada Dawud membuat baju besi untuk kamu, guna memelihara kamu dalam peperanganmu; Maka hendaklah kamu bersyukur (kepada Allah)” (Q.S Al-Anbiya. 80).

Menurut tafsir yang ada pada kitab *Al-Qurthubi*, ayat ini merupakan pokok landasan tentang upaya pembuatan alat-alat dan sebab-sebab. Allah *Ta'ala* telah mengabarkan tentang Nabi Dawud AS, bahwa ia membuat baju besi, teropong, dan makan dari hasil kerjanya sendiri sehingga pembuatan alat berguna untuk melindungi diri dari serangan lain. Dari tafsir tersebut dapat dipahami bahwa pembuatan alat pembantu atau teknologi itu penting untuk mempermudah aktifitas manusia dari segala bidang. (Tafsir Al-Qurthubi 11, 2008).

Teknik pengolahan citra digital merupakan salah satu produk perkembangan teknologi komputasi. Pengolahan citra digital sebagai bentuk teknologi modern telah lama ada. Dulu digitasi gambar pada koran untuk transmisi antar Atlantik melalui kabel kapal selam pada awal 1920-an.

Bagaimanapun juga dulu pengolahan citra digital yang sebenarnya adalah hal yang tidak mungkin hingga datangnya masa *hardware* komputer digital dengan skala besar. Motivasi awal pengembangan pengolahan citra digital berasal dari programmer antariksa di NASA pada tahun 1964. Laboratorium jet *propulsion* milik NASA menggunakan komputer komputer untuk memperbaiki distorsi citra permukaan bulan yang diambil oleh *Ranger 7 Probe*. Dan sekarang pengolahan citra digital mulai berkembang di bidang-bidang medis, militer, dan banyak lagi (Efford, 2000).

Penggunaan teknologi pengolahan citra pada bidang medis tentu berguna untuk mempermudah proses-proses medis sehingga efisiensi diagnosa dan penanganan medis bisa lebih cepat di atasi. Tentunya tindakan medis bisa dilakukan sebagai pencegahan dari penyakit-penyakit tertentu. Jika kita melihat

pentingnya menjaga kesehatan, Allah *Ta'ala* berfirman dalam surat *Al-Baqarah* ayat 168:

يَأْتِيهَا النَّاسُ كُلُّوا مِمَّا فِي الْأَرْضِ حَلَالًا طَيِّبًا وَلَا تَتَّبِعُوا خُطُوَاتِ الشَّيْطَانِ إِنَّهُ
لَكُمْ عَدُوٌّ مُبِينٌ

Artinya: “*Hai sekalian manusia, makanlah yang halal lagi baik dari apa yang ada di bumi dan janganlah kamu mengikuti langkah-langkah setan. Karena sesungguhnya setan itu adalah musuh yang nyata bagimu*” (Q.S. *Al-Baqarah*. 168).

Dari penggalan surat *Al-Baqarah* ayat 168, Allah memerintahkan untuk makan dari sesuatu yang baik karena makanan memiliki dampak yang besar untuk tubuh. Sari makanan dari makanan yang baik diserap tubuh akan digunakan tubuh untuk menjaga tubuh tetap sehat. Apabila makanan itu tidak baik, maka tubuh bisa menganggap itu sebagai racun sehingga berbahaya bagi kesehatan tubuh. Inilah point di mana Allah SWT berkehendak agar manusia menjaga kesehatan dengan makanan sebagai contohnya.

Rongga dada merupakan struktur tubuh yang penting karena di dalamnya terdapat organ-organ vital. Salah satu organ vital tersebut adalah paru-paru. Paru-paru dapat digunakan sebagai salah satu parameter identifikasi kelainan atau penyakit organ jantung melalui ukuran diameter terpanjang paru-paru. Metode pengukuran ini disebut dengan CTR (*Cardiothoracic Ratio*). CTR menggunakan perbandingan antara panjang diameter terpanjang paru-paru dengan diameter terpanjang jantung. Dengan mengetahui ukuran perbandingan tersebut, maka kelainan atau penyakit organ bisa diidentifikasi sejak dini.

Beberapa penelitian tentang segmentasi paru-paru pada citra *radiograph* berbasis *Active Shape Model* memiliki tingkat keberhasilan yang baik. Pengembangan *Active Shape Model* melalui penambahan *Minimal Path Searching* mampu memberikan akurasi yang baik dengan *distance error* antara 1.75 ± 0.33 piksel (Shengwen Guo dan Baowei Fei. 2009). Penggunaan pendekatan multiresolusi juga mampu melakukan segmentasi 68 citra secara efektif dari 80 citra yang diuji, serta meminimalisasi *CPU time* pada pengujiannya (Chunyan Wang, Shengwen Guo, dkk. 2008).

Mengingat pentingnya ukuran diameter paru-paru dalam identifikasi kelainan organ, dan kemampuan *Active Shape Model* dalam membantu proses segmentasi, maka pada penelitian ini dikembangkan sebuah aplikasi segmentasi paru-paru pada citra *x-ray* rongga dada menggunakan metode *Active Shape Model* untuk mengetahui diameter maksimal paru-paru.

1.2 Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut:

- 1.2.1 Bagaimana mengimplementasikan metode *Active Shape Model* untuk segmentasi paru-paru pada citra *x-ray* rongga dada?
- 1.2.2 Bagaimana mengukur diameter maksimal paru-paru dari hasil segmentasi citra *x-ray* rongga dada?

1.3 Batasan Masalah

Batasan Masalah dari penelitian ini adalah sebagai berikut:

- 1.3.1 Citra yang akan disegmentasi adalah citra *x-ray* sampel data *thorax* (rongga dada) dari *public image database*.
- 1.3.2 Citra *x-ray* rongga dada yang digunakan merupakan *file* hasil digitasi citra *x-ray*.
- 1.3.3 Metode yang digunakan adalah metode *Active Shape Model*.
- 1.3.4 Informasi terkait hasil segmentasi adalah hanya mendapatkan diameter maksimal paru-paru dalam satuan piksel.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

- 1.4.1 Mengimplementasikan metode *Active Shape Model* dalam segmentasi paru-paru pada citra *x-ray* rongga dada.
- 1.4.2 Mengukur diameter maksimal paru-paru yang didapatkan dari segmentasi paru-paru pada citra *x-ray* rongga dada.

1.5 Manfaat Penelitian

Manfaat yang bisa didapatkan dari penelitian ini adalah sebagai berikut:

- 1.5.1 Penggunaan metode *Active Shape Model* bisa digunakan untuk bidang medis, khususnya dalam analisa citra paru-paru.
- 1.5.2 Sebagai salah satu referensi pengembangan metode untuk pengolahan citra medis lainnya.

1.5.3 Menambah pengetahuan peneliti dalam pengolahan citra medis menggunakan metode *Active Shape Model*.

1.6 Metode Penelitian

Pada penelitian kali ini, peneliti menggunakan metode penelitian sebagai berikut:

1. Studi Literatur

Studi literature dilakukan dengan mempelajari teori-teori tentang *Digital Image Processing*, teori *Active Shape Model*, jurnal-jurnal terkait implementasi *Active Shape Model*, dan teori-teori tentang CTR untuk menghitung diameter maksimal paru-paru hasil segmentasi.

2. Pendefinisian dan Analisis Masalah

Mendefinisikan masalah dan melakukan analisis masalah untuk mendapatkan solusi yang tepat dalam proses segmentasi citra dan pengukuran diameter paru-paru.

3. Analisis Perancangan dan Implementasi Sistem

Dari analisa dan solusi yang telah didapatkan, selanjutnya membuat rancangan sistem melalui *interface*, alur proses, dan alur program. Setelah itu rancangan tersebut diaplikasikan dengan menggunakan *tool* yang telah tersedia.

4. Uji Coba dan Analisa Hasil Implementasi

Setelah sistem dibuat berdasarkan rancangan yang telah ditentukan, selanjutnya dilakukan uji coba untuk mengetahui kinerja sistem dan melakukan analisa output data agar melakukan evaluasi tingkat akurasi data.

5. Penulisan Laporan

Setelah semua tahapan selesai, penyusunan dan penulisan laporan dilakukan sebagai dokumentasi serta menjadi acuan untuk pengembangan penelitian selanjutnya.

1.7 Sistematika Penulisan Skripsi

BAB 1: Pendahuluan

Pendahuluan berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metode penelitian, dan sistematika penulisan skripsi.

BAB 2: Tinjauan Pustaka

Tinjauan pustaka merupakan argumentasi ilmiah yang dipakai sebagai referensi. Tinjauan pustaka bisa berasal dari berbagai sumber seperti jurnal-jurnal penelitian, blog, ataupun diskusi-diskus ilmiah lainnya.

BAB 3: Analisis dan Perancangan Sistem

Pada bab ini berisi penjelasan tentang analisa, desain dan perancangan sistem, serta implementasi sistem yang telah dibuat.

BAB 4: Hasil dan Pembahasan

Bab ini berisi hasil dari penelitian berupa gambar, grafik, dan tabel yang disertai dengan pembahasan baik secara teoritis maupun praktis sesuai dengan teori pada referensi yang telah ada.

BAB 5: Penutup

Bab 5 berisi kesimpulan dan saran. Kesimpulan merupakan pernyataan singkat yang dijabarkan dari hasil penelitian. Sedangkan saran dibuat berdasarkan pengalaman dan pertimbangan penulis serta anjuran dan rekomendasi untuk penelitian selanjutnya.



BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

a. Pengukuran Lebar *Cortical Bone* Berbasis *Active Shape Model*

Penelitian ini dilakukan oleh Gede Wirya Wardhana pada tahun 2009 untuk pengajuan tugas akhir di Institut Sepuluh November Surabaya. Tugas Akhir ini bertujuan untuk membangun suatu sistem pengukuran lebar *cortical bone* dengan metode *Active Shape Model*, dimana pengukuran dilakukan pada *boundary* atau tepi hasil *fitting* antara model statistik yang dihasilkan oleh metode ASM dengan objek dalam citra.

Hasil pengukuran lebar *cortical bone* dengan metode *active shape model* ini menunjukkan korelasi 90% dan mendekati pengukuran manual seiring dengan meningkatnya jumlah titik pada *training set*. (Wardhana, 2009)

b. *Lung Region Segmentation Based on Multi-resolution Active Shape Model*

Penelitian ini diajukan oleh Chunyan Wang, Shengwen Guo, Jianbo Wu, Qiong Liu, Xiaoming Wu. Segmentasi paru-paru pada citra radiografi dada merupakan titik kunci dalam diagnosa penyakit paru-paru. Karena kompleksitas dari struktur anatomi dan intensitasnya dalam radiografi dada menyebabkan *low-level* segmentation atau penajaman tepi citra medis kurang

mendapatkan hasil yang memuaskan. Pada jurnal ini disajikan cara segmentasi baru menggunakan *Active Shape Model*.

Jurnal ini mengembangkan metode *Active Shape Model* dengan menambahkan *automatic point insertion*, semi otomatis peletakan inialisasi awal dan menggunakan *framework multi-resolusi*. 80 citra radiografi digunakan untuk percobaan, dan hasilnya menunjukkan bahwa 68 citra tersegmentasi secara efektif. Penelitian ini juga menunjukkan bahwa pencarian menggunakan *framework multi-resolusi* memakai *CPU time* lebih sedikit dibandingkan yang original karena metode yang baru konvergensi dicapai pada iterasi yang sedikit.

Kesimpulannya, pengembangan *active shape model* menghasilkan efektifitas dan kecepatan yang lebih baik, *semiautomatic* dan *model-based method* untuk segmentasi paru-paru. (Chunyan Wang, dkk. 2008).

c. *A Minimal Path Searching Approach for Active Shape Model (ASM)-based Segmentation of The Lung*

Penelitian ini dilakukan oleh Shengwen Guo dan Baowei Fei untuk mengembangkan *Active Shape Model* dengan menambahkan *minimal path searching* pada deteksi tepi paru-paru. *Minimal Path* diaplikasikan pada proses pencarian dan untuk menjaga *smoothness shape*, sebuah *smooth constraint* ditambahkan pada *deformable model*.

Model shape didapatkan dengan menggunakan PCA dengan penambahan *smooth matrix* untuk menghasilkan *shape* yang lebih halus.

Pembentukan *Grey-level structure* juga dilakukan untuk membentuk profil pada arah normal tepian paru-paru.

Hasil perbandingan segmentasi menggunakan ASM-MP dengan segmentasi manual dari 72 data digitasi *radiograph*, menghasilkan jarak *error* sebesar $1,75 \pm 0,33$ piksel. Dengan demikian ASM-MP mampu melakukan segmentasi paru-paru pada citra digital radiografi dengan akurat. (Shengwen Guo dan Baowei Fei. 2009).

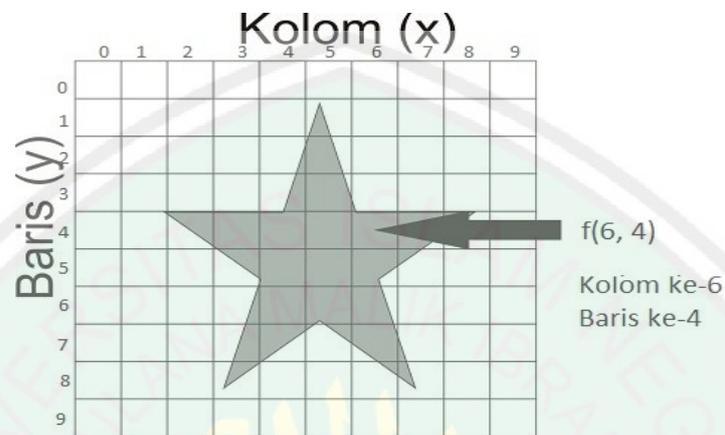
2.2 Pengolahan Citra Digital

2.2.1 Pengertian Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan data 2 dimensi. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu.

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitude f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y, dan nilai amplitude f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixels*. Istilah terakhir (*pixel*)

paling sering digunakan pada citra digital. (Putra, 2010) Gambar 2.1 menunjukkan posisi koordinat citra digital.



Gambar 2.1 Sistem kordinat pada citra digital

2.2.2 Image Processing

Image processing adalah suatu metode yang digunakan untuk memproses atau memanipulasi gambar dalam bentuk 2 dimensi. *Image Processing* dapat juga didefinisikan sebagai segala operasi untuk memperbaiki, menganalisa, atau mengubah suatu gambar. Konsep dasar pemrosesan suatu objek pada gambar menggunakan *image processing* diambil dari kemampuan indera penglihatan manusia yang selanjutnya dihubungkan dengan kemampuan otak manusia. (Gonzales & Wood. 2002: 1).

Dalam sejarahnya, *image processing* telah diaplikasikan dalam berbagai bentuk dengan tingkat kesuksesan yang cukup besar. Seperti berbagai cabang ilmu lainnya, *Image Processing* menyangkut pula berbagai gabungan cabang-cabang ilmu, diantaranya adalah optik, elektronik, matematika, fotografi, dan teknologi komputer.

Pada umumnya, tujuan dari *image processing* adalah metransformasikan atau menganalisis suatu gambar sehingga informasi baru tentang gambar dibuat lebih jelas.

Ada empat klasifikasi dasar dalam *image processing* yaitu *point processing*, *area processing*, *geometric processing*, dan *frame processing*.

(Jaya, 2009)

- a. *Point Processing* memproses nilai *pixel* suatu gambar berdasarkan nilai dari *pixel* tersebut. Contoh dari *point processing* adalah *adding*, *subtracting*, *contrast stretching*, dan lain sebagainya.
- b. *Area processing* memproses nilai *pixel* suatu gambar berdasarkan nilai *pixel* suatu gambar berdasarkan nilai *pixel* di sekelilingnya. Contoh dari *area processing* adalah *convolution*, *blurring*, *sharpening*, dan *filtering*.
- c. *Geometric processing* digunakan untuk mengubah posisi dari *pixel*. Contoh dari *geometric processing* adalah *scaling*, *rotation*, dan *mirroring*.
- d. *Frame processing* memproses nilai *pixel* dari dua buah gambar atau lebih. Contoh dari *frame processing* adalah *cross dissolve*, *morphing*, dan lain sebagainya.

Selain itu, masih ada tiga tipe *image processing* berdasarkan tingkatannya, yaitu:

- a. *Low-level processing*: proses-proses yang berhubungan dengan operasi primitif seperti *image pre-processing* untuk mengurangi *noise*, menambah kontras dan menajamkan gambar. Pada *low-level processing*, input dan outputnya berupa gambar.

- b. *Mid-level processing*: proses-proses yang berhubungan dengan segmentasi gambar (membagi gambar menjadi objek-objek), pengenalan (*recognition*) suatu objek individu. Pada *mid-level processing*, input pada umumnya berupa gambar tetapi outputnya berupa atribut yang dihasilkan dari proses yang dilakukan pada gambar tersebut seperti garis, garis kontur, dan objek-objek individu.
- c. *High-level processing*: proses-proses yang berhubungan dengan hasil dari *mid-level processing*.

2.2.3 Perbaikan Citra (*Image Enhancement*)

Perbaikan citra bertujuan meningkatkan kualitas tampilan citra untuk pandangan manusia atau untuk mengkonversi suatu citra agar memiliki format yang lebih baik sehingga citra tersebut menjadi lebih mudah diolah dengan komputer.

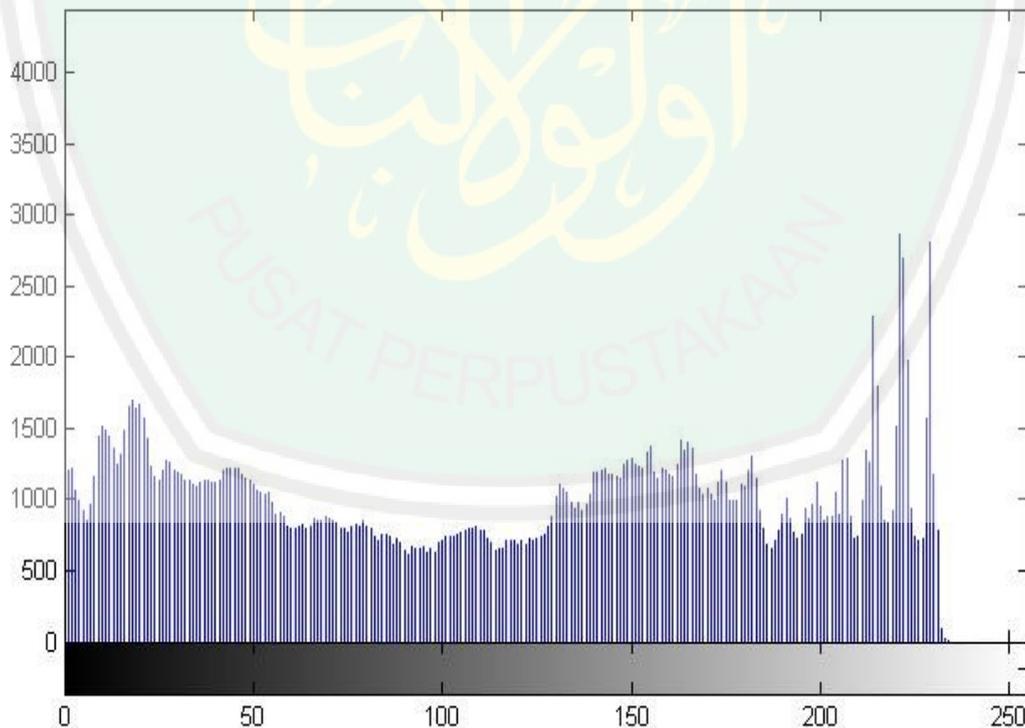
Perbaikan citra dapat dilakukan dengan operasi titik (*point operation*), operasi spasial (*spatial operation*), operasi geometri (*geometric operation*), dan operasi aritmatik (*arithmetic operation*).

2.2.3.1 Histogram

Histogram citra menunjuk pada histogram dari nilai intensitas piksel. Histogram menampilkan banyaknya piksel dalam suatu citra yang dikelompokkan berdasarkan level nilai intensitas piksel yang berbeda. Pada citra *grayscale 8-bit*, terdapat 256 level nilai intensitas yang berbeda maka pada *histogram* akan ditampilkan secara grafik distribusi dari masing-masing 256 piksel tersebut.

Proses pembentukan *histogram* dapat dilakukan dengan memeriksa setiap nilai piksel pada citra, kemudian hitung banyaknya nilai piksel tersebut dan disimpan di memori (Putra, 2010) . Contoh grafik *histogram* dari citra *grayscale 8-bit* ditunjukkan pada gambar 2.2.

Histogram juga bisa diterapkan untuk citra berwarna, dengan cara memisahkan terlebih dahulu 3 komponen warna *red*, *green*, dan *blue*, kemudian setiap komponen warna dibuat histogramnya. Histogram citra berwarna juga dapat ditampilkan dalam grafik 3D dengan salah satu sumbu menyatakan komponen ruang warna. Contoh grafik *histogram* dari citra *grayscale 8-bit* ditunjukkan pada gambar 2.2.



Gambar 2.2 *Histogram* citra *grayscale 8 bit*

2.2.3.2 Histogram Equalization

Histogram equalization merupakan teknik untuk menghasilkan histogram citra yang seragam. Teknik ini dapat dilakukan pada keseluruhan citra atau pada beberapa bagian saja.

Histogram hasil proses ekualisasi tidak akan seragam atau sama untuk seluruh intensitas. Teknik ini hanya melakukan distribusi ulang terhadap distribusi intensitas dari *histogram* awal. Jika *histogram* awal memiliki beberapa puncak dan lembah, maka *histogram* hasil ekualisasi akan tetap memiliki puncak dan lembah. Akan tetapi puncak dan lembah tersebut akan digeser. *Histogram* hasil ekualisasi akan lebih disebar (spreading). (Putra, 2010)

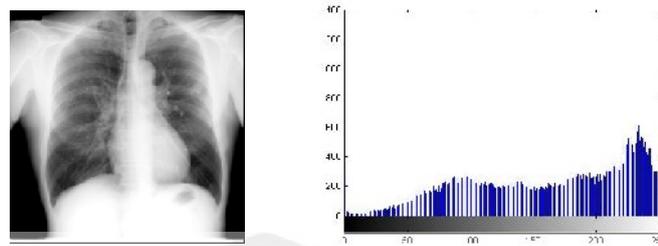
Distribusi ulang dapat dilakukan menggunakan persamaan berikut:

$$n(g) = \begin{cases} 0, & (g-1) * \frac{c(g)}{L-1} \\ \dots & \dots \end{cases} \quad (2.1)$$

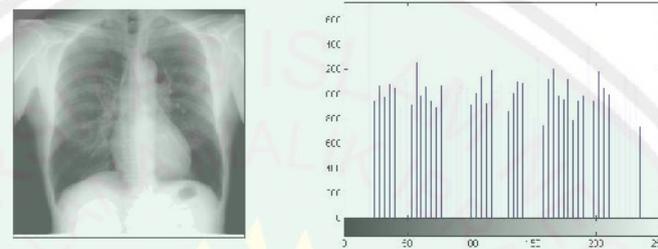
Dengan $n(g)$ adalah nilai piksel baru, N menyatakan banyaknya piksel pada citra (bila citra berukuran 8×8 maka N adalah 64), g menyatakan nilai *gray level* awal yang nilainya dari $1 \dots L-1$ (L menyatakan nilai *gray level* maksimum). Sedangkan $c(g)$ menyatakan banyaknya piksel yang memiliki nilai sama dengan g atau kurang yang secara matematis dapat dinyatakan sebagai:

$$c(g) = \sum_{i=0}^g h(i) \quad (2.2)$$

Dengan $g = 1, 2, \dots, L-1$, dan $h(i)$ menyatakan *histogram* awal. Contoh tampilan grafik antara *histogram gray level* yang belum diekualisasi dengan *histogram* yang telah diekualisasi tampak pada gambar 2.3.



(a)



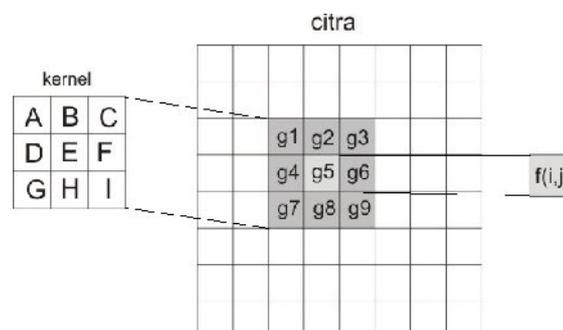
(b)

Gambar 2.3 *Histogram Equalization* (a) Citra asli beserta histogramnya (b) Citra telah mengalami ekualisasi histogram beserta histogramnya.

2.2.3.3 Konvolusi

Konvolusi merupakan operator sentral pengolah citra dan telah digunakan secara luas pada berbagai piranti lunak pengolah citra.

Kernel (*sliding window*) diletakkan pada setiap piksel dari citra input dan menghasilkan piksel baru. Nilai piksel baru dihitung dengan mengalikan setiap nilai piksel tetangga dengan bobot yang berhubungan pada kernel dan kemudian menjumlah hasil perkalian tersebut.



Gambar 2.4 Ilustrasi operasi konvolusi dengan kernel 3 x 3

Dari gambar 2.4, operasi konvolusi menggunakan *kernel* 3 x 3 adalah sebagai berikut:

$$f(i, j) = A_{g1} + B_{g2} + C_{g3} + D_{g4} + E_{g5} + F_{g6} + G_{g7} + H_{g8} + I_{g9} \quad (2.3)$$

Jika operasi konvolusi berada di luar batas ukuran citra, tetap dapat dilakukan dengan menambahkan sembarang piksel untuk piksel di luar batas yang umumnya digunakan atau ditambahkan nilai 0 (*zero padding*). Dengan cara tersebut, proses konvolusi tetap dapat dilakukan. (Putra, 2010)

2.2.3.4 Penskalaan

Penskalaan adalah operasi geometri yang memberikan efek memperbesar atau memperkecil ukuran citra input sesuai dengan variabel penskalaan citranya. Ukuran baru hasil penskalaan didapat melalui perkalian antara ukuran citra input dengan variabel penskalaan.

Proses penskalaan dapat dilakukukan dengan rumus:

$$P_o = S_p \times P_i \quad (2.4)$$

$$L_o = S_l \times L_i \quad (2.5)$$

Di mana (P_i , L_i) adalah ukuran citra *input*, (P_o , L_o) adalah ukuran citra *output*, dan (S_p , S_l) adalah variabel penskalaan yang diinginkan. Jika variabel penskalaan lebih besar dari 1, maka hasil penskalaannya akan memperbesar ukuran citra, sebaliknya jika variabel penskalaan lebih kecil dari 1, maka penskalaannya akan memperkecil ukuran citra. (Putra, 2010)

2.3 Segmentasi Citra

Segmentasi merupakan teknik untuk membagi suatu citra menjadi beberapa daerah (*region*) di mana setiap daerah memiliki kemiripan atribut. Di sini akan dijelaskan beberapa teknik segmentasi citra.

2.3.1 Pengambangan (*Thresholding*)

Proses pengambangan akan menghasilkan citra biner, yaitu citra yang memiliki dua nilai tingkat keabuan yaitu hitam dan putih. Secara umum proses pengambangan citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut.

$$g(x,y) = \begin{cases} 1 & f(x,y) \geq T \\ 0 & f(x,y) < T \end{cases}$$

Dengan $g(x,y)$ adalah citra biner dari citra *grayscale* $f(x,y)$, dan T menyatakan nilai ambang. Nilai T memegang peranan yang sangat penting dalam proses pengambangan. Kualitas hasil citra biner sangat tergantung pada nilai T yang digunakan.

Terdapat dua jenis pengambangan, yaitu pengambangan global (*global thresholding*) dan pengambangan secara local adaptif (*locally adaptive thresholding*). Pada pengambangan global, seluruh piksel pada citra dikonversikan menjadi hitam atau putih dengan satu nilai ambang T . Kemungkinan besar pada pengambangan global akan banyak informasi yang hilang karena hanya menggunakan satu nilai T untuk keseluruhan piksel. Untuk mengatasi masalah ini dapat digunakan pengambangan secara *local adaptif*. Pada pengambangan lokal,

suatu citra dibagi menjadi blok-blok kecil dan kemudian dilakukan pengambangan lokal pada setiap blok dengan nilai T yang berbeda. (Putra, 2010)

2.3.2 Segmentasi Berbasis *Clustering*

Segmentasi berbasis *cluster* menggunakan data multidimensi untuk mengelompokkan pixel citra ke dalam beberapa *cluster*. Pada umumnya piksel di-*cluster* berdasarkan kedekatan jarak antar piksel.

Segmentasi berbasis *cluster* ini mulai populer sejak diimplementasikan pada aplikasi OCR (*Optical Character Recognition*), pengenalan sidik jari hingga remote sensing. Keberhasilan dari proses segmentasi berbasis *cluster* ditentukan dari keberhasilan dalam mengelompokkan fitur-fitur yang berdekatan ke dalam satu cluster.

Metode-metode dalam segmentasi berbasis *cluster* di antaranya adalah iterasi, *K-means*, *Fuzzy C-means*, jaringan syaraf *Kohonen*, dan berbagai teknik *cluster* lainnya. (Putra, 2010)

2.3.3 *Template Matching*

Template Matching adalah proses mencari suatu objek (*template*) pada keseluruhan objek yang berada dalam suatu citra. *Template* dibandingkan dengan keseluruhan objek tersebut dan bila *template* cocok (cukup dekat) dengan suatu objek yang belum diketahui pada citra tersebut maka objek tersebut ditandai sebagai *template* (Putra, 2010) .

Perbandingan antara *template* dengan keseluruhan objek pada citra dapat dilakukan dengan menghitung selisih jaraknya, seperti berikut.

$$(x, y) = \sum \sum [(x, y) - (x - \Delta x, y - \Delta y)] \quad (2.6)$$

2.4 Active Shape Model

Active Shape Model merupakan *statistical model* dari kumpulan *shape* yang terdiri dari landmark-landmark yang membentuk garis batas dari suatu objek. Model tersebut menggunakan beberapa citra dalam *training set* di mana landmark dari objek yang sebenarnya telah didefinisikan. Setelah bentuk disesuaikan menggunakan algoritma *Procrustes*, variasi dari bentuk dideskripsikan menggunakan *Principal Component Analysis*. Analisis PCA menghasilkan sebuah model termasuk model statistik baik dari bentuk atau *edge profile*.

Dalam proses pencarian objek menggunakan ASM, *mean shape* ditempatkan pada perkiraan posisi yang tepat pada gambar. Algoritma yang ada akan mencoba untuk mencocokkan *mean shape* dengan objek pada gambar dengan mencari kecocokan terbaik dari profil tepian pada setiap landmark dan kemudian bergerak menuju profil yang cocok tersebut. Prosedur ini diulang untuk setiap landmark dan pada akhirnya model baru dihasilkan (Rizvandi, Pizuria, Philips. IEEE. Hal: 1).

2.4.1 Point Distribution Model

Pembentukan *model shape* dari *training set* yang ada terdapat pada bagian *Point Distribution Model*. Langkah-langkah yang ada pada *Point Distribution Model* adalah pelabelan *training set* untuk mendapatkan landmark model, penyesuaian bentuk (*aligning shape*) dari hasil *labelling*, dan model statistik untuk menghasilkan statistik dan variasi model. (Hamarneh, _____)

2.4.1.1 Pelabelan *Training Set*

Sebelum memulai penandaan pada *training set*, kita harus menentukan jumlah titik landmark terlebih dahulu. Di sini setiap *training set* ditempatkan suatu titik secara manual oleh *user*, kemudian titik-titik yang terlihat signifikan dapat didefinisikan. Ini penting untuk menjamin bahwa peletakan titik-titik tersebut dilakukan secara akurat.

Training set yang telah diberi label ini dapat dinotasikan dengan S . Terdiri dari bentuk sebanyak N di mana setiap *shape* memiliki n penanda. Dengan kata lain di sini didapatkan N buah titik koordinat penanda ke- j dan bentuk ke- i dalam *training set* oleh (x_{ij}, y_{ij}) dan vektor yang mendeskripsikan titik-titik berjumlah n dari bentuk ke- i dari *training set* yang dapat dituliskan dengan:

$$x_i = [x_{i0}, y_{i0}, x_{i1}, y_{i1}, \dots, x_{i(n-1)}, y_{i(n-1)}]^T; 1 \leq i \leq N \quad (2.7)$$

2.4.1.2 *Aligning Shape*

Penyesuaian bentuk atau *aligning shape* dilakukan untuk mempelajari variasi dari setiap posisi titik landmark pada *training image*. Proses penyesuaian

diselesaikan dengan mengubah pose (skala, rotasi, dan translasi) dari *shape* untuk mencapai bentuk yang sesuai.

Cootes menyebutkan penyesuaian bentuk ini juga dilakukan untuk memastikan bahwa *shape* berada pada *co-ordinate frame* yang sama, dan teknik yang bisa dilakukan adalah menggunakan *Procrustes Analysis*. Setiap *shape* disesuaikan sehingga meminimalisasi jarak antara *shape* yang satu dengan *shape* yang lain ke *mean shape*.

Misalkan terdapat dua bentuk, x_1 dan x_2 yang berpusat pada $(0, 0)$, Skalasi dan rotasi akan dilakukan terhadap x_1 dengan (s, θ) untuk meminimalkan $|sAx_1 - x_2|$, di mana A merupakan rotasi dari x dengan sudut sebesar θ . Maka:

$$= (\cdot) / | | \quad (2.8)$$

$$= (\Sigma -) / | | \quad (2.9)$$

$$= + \quad (2.10)$$

$$= \tan (/) \quad (2.11)$$

Jika kedua bentuk tersebut tidak berpusat pada $(0, 0)$ maka harus dilakukan translasi terlebih dahulu supaya kedua bentuk berpusat pada $(0, 0)$. (Cootes et al., 2004) .

Berikut adalah langkah-langkah penyesuaian atau pengaturan bentuk pada suatu kumpulan *shape* yang berjumlah N (Cootes et al., 1994) :

1. Rotasikan, skalakan, dan translasikan setiap *shape* untuk penyesuaian dengan *shape* pertama dalam kumpulan *shape* tersebut.
2. Melakukan iterasi berikut:
 - a. Hitung *mean shape* dari *shape* yang telah disesuaikan atau diatur

- b. Normalisasi orientasi, skala, dan titik asal dari mean yang telah didapatkan ke arah dan *shape* yang sesuai.
 - c. Ulang penyesuaian atau pengaturan *shape* tersebut untuk setiap *shape* dengan *mean* yang telah di dapatkan sebelumnya.
3. Iterasi selesai sampai proses mendapatkan hasil *shape* yang konvergen.

Dalam diktat *Active Shape Model, Modelling Shape Variations and Gray Level Information and an Application to Image Search and Classification* oleh Ghassan Hamarneh, normalisasi pose berarti penskalaan (*scaling*) model *shape* agar jarak antara dua titik menjadi konstan, perotasian (*rotating*) model *shape* agar garis yang menghubungkan dua titik landmark yang belum dispesifikasikan bisa terarah, dan translasi (*translation*) *shape* untuk memusatkan pada koordinat yang sama.

2.4.1.3 Model Statistik

Jika sudah mendapatkan sekumpulan titik-titik x_i sejumlah s yang telah diselaraskan dalam *co-ordinate frame* yang sama, vektor-vektor tersebut membentuk sebuah distribusi pada *nd dimensional space*. Jika distribusi ini dapat dimodelkan, dapat dihasilkan data baru yang sesuai dengan data yang ada pada *training set* dan dapat memeriksa sebuah bentuk apakah bentuk tersebut merupakan bentuk yang sesuai dengan bentuk-bentuk yang ada pada *training set*.

Untuk menyederhanakan permasalahan ini, dibutuhkan cara untuk mereduksi data dari bentuk $2n$ menjadi lebih sederhana. Pendekatan yang cukup efektif adalah menggunakan *Principal Component Analysis* (PCA) terhadap data

tersebut. Bentuk data berupa “cloud” dari titik-titik pada ruang 2D. PCA menghitung axis utama dari “cloud”, memungkinkan satu atau beberapa titik-titik utama menggunakan model yang kurang dari parameter $2n$ (Cootes. 2000).

Dalam jurnal *Active Shape Model – Their Training and Application* oleh Cootes dan Taylor (1994), dinotasikan *mean* vektor sebagai \bar{x} dan diferensial vektor antara vektor x_i dan x sebagai dx_i , maka dapat dituliskan

$$dx = x - \bar{x} \text{ dan } \bar{x} = -\sum (2.12)$$

Kovarian matrik dan penanda atau *landmarks* dari tiap-tiap bentuk dapat dituliskan dengan:

$$= -\sum (x_i - \bar{x})(x_j - \bar{x}) (2.13)$$

$$SP_k = \lambda_k P_k, \lambda_k \geq \lambda_{k+1} (2.14)$$

Jika P adalah matrix $2n \times t$ dari t unit *eigenvector* sesuai dengan *eigenvalue* yang paling besar, maka semua data dari *training set* x dapat diperkirakan dengan persamaan:

$$x \approx \bar{x} + (2.15)$$

Di mana b adalah vektor dengan t dimensi yang didapatkan dari:

$$b = (b_1, \dots, b_t) (2.16)$$

Vektor b mendefinisikan sekumpulan parameter dari model yang *deformable*. Dengan memvariasikan nilai dari elemen pada vektor b , variasi bentuk x dapat ditimbulkan menggunakan persamaan 2.15. Dengan memberikan batasan $\pm \lambda_i^{-1}$ pada parameter b_i , di mana λ_i merupakan *eigenvalue* ke- i , maka dapat dipastikan bahwa bentuk yang dihasilkan sesuai dengan bentuk-bentuk yang ada pada *training set* (Cootes. 2000).

Jumlah *eigenvector* yang disimpan, t , dapat dipilih sehingga model dapat merepresentasikan sejumlah proporsi (misal 98%) dari total *variance* pada data dan sisanya dianggap sebagai *noise*. jika λ_i merupakan *eigenvalue* ke- i , maka total *variance* dari *training data* adalah jumlah dari semua *eigenvalue*:

$$= \sum \quad (2.17)$$

Dipilih t *eigenvalue* yang paling besar sehingga:

$$\sum \geq \quad (2.18)$$

Di mana f_v merepresentasikan sejumlah proporsi (misal 98%) dari total *variance* pada data.

2.4.2 Image Search

Hamarneh mengungkapkan, ide dasar untuk memulai pencarian suatu objek dalam citra adalah dengan inialisasi awal, kemudian memeriksa ketetanggaan *landmark* dari inialisasi tersebut untuk mencari lokasi terbaik dari *landmark*. Kemudian melakukan perubahan *shape* dan *pose* dari inialisasi ke pencocokan yang terbaik menuju lokasi terbaru. Itu menunjukkan bentuk (*shape*) dimodelkan dengan cara yang model-model tersebut hanya bisa bervariasi dengan cara yang terkontrol.

2.4.2.1 Inialisasi Awal *Shape*

Dijelaskan pada jurnal *Deformable Shape Description Using Active Shape Model* oleh Rizvandi, dkk, pengguna harus menentukan inialisasi *shape* secara manual. *Shape* tersebut ada baiknya diletakkan sedekat mungkin dengan objek dalam citra. Jika *shape* X_i merupakan *shape* yang dipilih secara manual oleh

pengguna, itu bisa dianggap sebagai translasi, rotasi, dan skalasi dari *mean shape*

diperoleh dengan penyesuaian:

$$= (,) [\uparrow] + \quad (2.19)$$

Dimana,

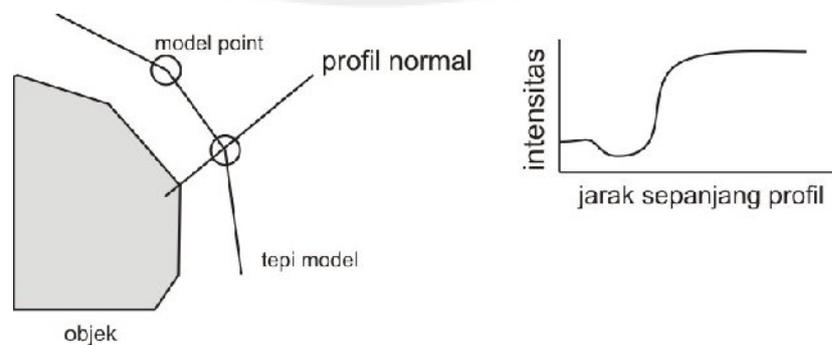
$$(,) = \begin{matrix} \cos & - \sin \\ \sin & \cos \end{matrix}$$

$$t_i = [t_{xi}, t_{yi}, t_{xi}, t_{yi}, \dots, t_{xi}, t_{yi}]^T$$

2.4.2.2 Perhitungan *Suggested Movement*

Diberikan suatu nilai estimasi awal dari posisi untuk kumpulan titik-titik yang dicoba untuk dicocokkan (*fitting*) pada suatu objek citra dan disesuaikan untuk mencari kumpulan *adjustments* yang akan menggerakkan setiap-setiap titik ke arah posisi yang lebih baik.

Dalam prakteknya, pencarian dilakukan sepanjang profil normal pada model di setiap titik. Jika kita mengharapkan garis batas model sesuai dengan tepi objek, kita bisa dengan mudah menentukan lokasi tepi terkuat sepanjang profil. Posisi ini memberikan lokasi baru kepada titik model. Sampel profil normal dapat dilihat pada gambar 2.5.



Gambar 2.5 *Sample* profil normal sepanjang tepi objek pada setiap titik model

2.4.2.3 Perubahan *Pose* dan *Shape Parameters*

Di dalam jurnal *Active Shape Model – Their Training and Application* milik Tim Cootes (1994), Perhitungan perubahan posisi dan *shape parameters* ini bertujuan untuk melihat proses *adjustment* dari posisi dan *shape parameters* dari *model points* untuk bergerak dari lokasi awal dalam *image frame* ke arah *suggested location* yang baru ($X+dX$) yang dapat diolah ketika masih dalam kondisi awal dari *shape constraint* pada model. Jika estimasi awal dari model berpusat pada (X_c, Y_c) dengan orientasi θ dan skala s maka dapat pertama kali dihitung bagaimana cara memperbaharui parameter ini untuk mencocokkan model dengan gambar. Ini dicapai dengan mencari translasi (dX_c, dY_c) , rotasi $d\theta$ dan factor skala $(1+ds)$ yang memetakan titik-titik X pada point-point yang dihasilkan oleh $(X+dx)$ menggunakan metode *Aligning Shape (Aligning Two Shapes)*.

Setelah melakukan proses *adjustment*, ini akan menyisakan suatu *residual adjustment* yang hanya dapat dicapai dengan melakukan *deforming shape* dari model. Di sini akan dihitung *adjustment dx* dalam koordinat lokal untuk melihat sebab bergeraknya titik X oleh dX ketika dikombinasikan dengan efek dari parameter skala, rotasi, dan translasi yang baru. Posisi awal dari titik-titik dalam gambar dinotasikan dengan persamaan 2.19.

Untuk menghitung *residual adjustment dx* dalam model koordinat lokal maka didapatkan persamaan sebagai berikut:

$$dx = M((s(1+ds))^{-1}, -(\theta+d\theta))[y]-x \quad (2.20)$$

di mana

$$y = M(s, \theta)[x]+dX-dX_c$$

persamaan 2.20 memberikan cara untuk menghitung perkiraan pergerakan pada titik-titik x dalam model koordinat lokal. Pergerakan ini tidak secara general konsisten terhadap *shape model*. Dalam rangka untuk menerapkan *shape constrains*, ditransformasikan dx ke dalam ruang *model parameter*, yaitu db , yaitu perubahan *model parameter* yang digunakan untuk melakukan *adjustment* terhadap *model points* mendekati dx dimana dapat dinotasikan sebagai:

$$x = \bar{x} + Pb \quad (2.21)$$

dan akan dicari db sehingga

$$x + dx \approx \bar{x} + P(db) \quad (2.22)$$

Selama hanya ada $t < 2n$, *mode of variation* berlaku dan dx dapat berpindah atau bergerak sehingga titik dalam $2n$ memiliki derajat yang berbeda. Di sini akan hanya dicapai suatu perkiraan untuk persyaratan deformasi.

Pengurangan persamaan 2.21 dengan persamaan 2.22 menjadikan:

$$dx \approx P(db) \quad (2.23)$$

sehingga:

$$db = P^T dx \quad (2.24)$$

Selama $P^T = P^{-1}$ sebagai kolom dari P satu sama lain orthogonal. Ini bisa dilihat bahwa persamaan 2.23 ekuivalen untuk menggunakan *least-squares approximation* untuk menghitung *shape parameter adjustment db*. (Cootes. 1994)

2.4.2.4 Pembaharuan *Pose* dan *Shape Parameters*

Setelah proses perubahan pose dan *shape parameters*, diterapkan suatu persamaan untuk memperbaharui parameter dalam suatu skema iteratif (Hamarneh.).

$$X_c \rightarrow X_c + dX_c \quad (2.25)$$

$$Y_c \rightarrow Y_c + dY_c \quad (2.26)$$

$$\theta \rightarrow \theta + d\theta \quad (2.27)$$

$$s \rightarrow s(1+ds) \quad (2.28)$$

$$b \rightarrow b+db \quad (2.29)$$

Cootes (1994) menjelaskan, sebelumnya perubahan yang dijelaskan sangat cepat atau besar, perubahan ini memiliki kecenderungan untuk memperbesar *shape variations*. Dapat dijamin bahwa model hanya berubah atau mengalami deformasi ke dalam suatu *shape* yang konsisten dengan training set dengan menempatkan limit dari nilai dari b_k . Shape dapat dipertimbangkan untuk diterima jika mahalanobis distance D_m kurang dari konstanta yang pantas, D_{max} .

Vektor b hendaknya berada dalam *hyperellipsoid*, jika perubahan b menggunakan persamaan 2.23 ini memungkinkan untuk mencapai shape yang memenuhi $D_m > D_{max}$ dan titik-titik tersebut berada di luar *ellipsoid*, b dan di *rescale* untuk berada dalam titik terdekat menggunakan:

$$b_k \rightarrow \text{---} (k=1, \dots, t) \quad (2.30)$$

2.5 Validasi Segmentasi

True Positive (TP) merupakan nilai “positif” yang benar diklasifikasikan sebagai positif, *True Negative* (TN) merupakan nilai “negatif” yang benar diklasifikasikan sebagai negatif, *False Negative* (FN) merupakan nilai “positif” yang salah diklasifikasikan sebagai negatif, dan *False Positive* (FP) merupakan nilai “negatif” yang salah diklasifikasikan sebagai positif. (Southall B, dkk.)

Nilai-nilai tersebut dapat menghasilkan tiga persamaan berupa akurasi, sensitifitas atau *True Positive Rate* (TPR), dan Spesifitas atau *True Negative Rate*.

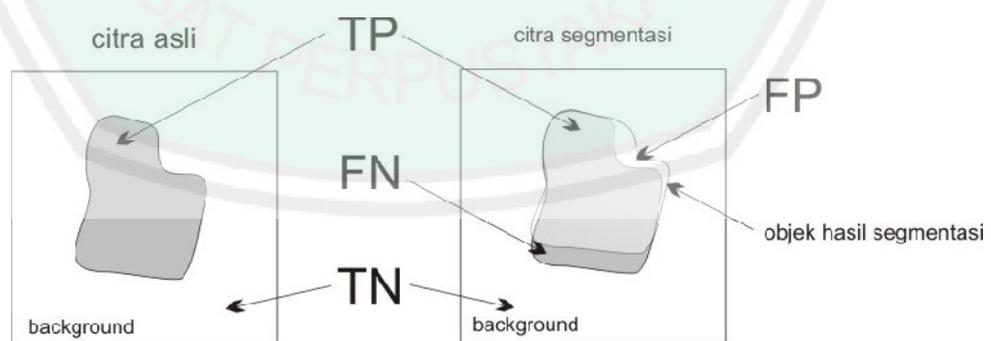
Persamaan tersebut adalah sebagai berikut:

$$\text{Akurasi (ACC)} = (TP + TN) / (TP + FN + FP + TN) \quad (2.31)$$

$$\text{Sensitifitas (TPR)} = TP / (TP + FN) \quad (2.32)$$

$$\text{Spesifitas (TNR)} = TN / (FP + TN) \quad (2.33)$$

Validasi citra segmentasi dan citra referensi berupa pembagian daerah TP, TN, FN, dan FP pada dapat dilihat pada gambar 2.6



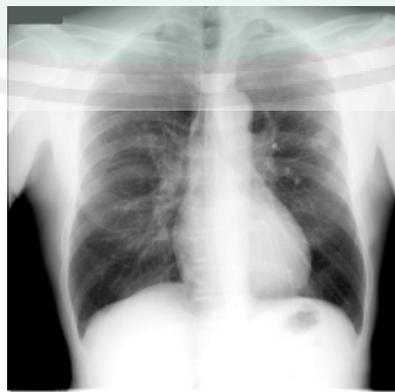
Gambar 2.6 Pembagian daerah TP, TN, FN, dan FP antara citra segmentasi dan citra referensi

2.6 Pencitraan X-Ray

X-ray merupakan salah satu dari sumber tertua radiasi elektromagnetik (EM) yang digunakan sebagai pencitraan. Biasanya *x-ray* digunakan untuk bidang diagnosa medis, tapi belakangan diketahui juga digunakan dalam bidang industri dan astronomi.

X-ray untuk pencitraan medis dan industrial didapatkan dengan menggunakan tabung *x-ray* yang berisi katoda dan anoda. Katoda yang dipanaskan akan melepaskan elektron bebas. elektron-elektron tersebut mengalir dengan kecepatan tinggi menuju anoda yang bermuatan positif. Ketika elektron menumbuk *nucleus*, energi dilepaskan dengan bentuk radiasi *x-ray*. Energi *x-ray* tersebut dikontrol oleh voltase yang dihasilkan anoda, dan jumlah *x-ray* ditentukan oleh filamen katoda.

Pada digital radiografi, citra digital didapatkan oleh dua cara yaitu dengan digitalisasi film *x-ray* atau menggunakan sinar *x-ray* yang melewati tubuh pasien dan sinar tersebut jatuh tepat pada perangkat yang mengkonversi *x-ray* menjadi cahaya. (Gonzalez. 2002. Hal: 9). Contoh citra *x-ray* dapat dilihat pada gambar 2.7



Gambar 2.7 Sampel citra *x-ray* rongga dada
Sumber: *public medical image databases*

2.7 Paru-Paru

Paru-paru adalah dua organ yang berbentuk seperti bunga karang besar yang terletak di dalam torak pada sisi lain jantung dan pembuluh darah besar. Paru-paru memanjang mulai dari akar leher menuju diafragma dan secara kasar berbentuk kerucut dengan puncak di sebelah atas dan alas di sebelah bawah.

Paru-paru dibagi menjadi lobus-lobus. Paru-paru sebelah kiri mempunyai dua lobus, yang dipisahkan oleh belahan miring. *Lobus superior* terletak di atas dan di depan *lobus inferior* yang berbentuk kerucut. Paru-paru sebelah kanan mempunyai tiga lobus. Lobus bagian bawah dipisahkan oleh *fisura oblik* dengan posisi yang sama terhadap *lobus inferior* kiri. Sisa paru lainnya dipisahkan oleh suatu *fisura horizontal* menjadi lobus atas dan lobus tengah. Setiap lobus selanjutnya dibagi menjadi segmen-segmen yang disebut *bronko-pulmoner*, mereka dipisahkan satu sama lain oleh sebuah dinding jaringan *koneknif*, masing-masing satu arteri dan satu vena.

Bronkus utama kiri dan kanan tidak simetris. *Bronkus* kanan lebih pendek dan lebih lebar dan merupakan kelanjutan dari *trakea* yang arahnya hampir vertikal. Sebaliknya *bronkus* kiri lebih panjang dan lebih sempit dan merupakan kelanjutan dari *trakea* dengan sudut yang lebih tajam (Dorce Mengkidi. 2006. Hal: 15).

Salah satu ayat Al-Qur'an dari surat *At-Tiin* ayat 8 yang berbunyi

لَقَدْ خَلَقْنَا الْإِنْسَانَ فِي أَحْسَنِ تَقْوِيمٍ ﴿٨﴾

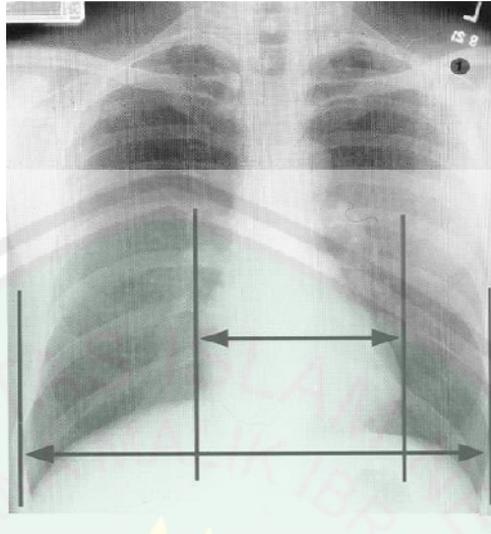
Artinya: “Sesungguhnya Kami telah menciptakan manusia dalam bentuk yang sebaik-baiknya”

Pada tafsir *Juz 'Amma Al-Qurtubi* dijelaskan, dalam bentuk yang sebaik-baiknya berarti bentuknya yang lurus dan kemudaannya yang seimbang. Demikian seperti apa yang dilakukan ahli tafsir. Dia adalah ciptaan yang sebaik-baiknya karena Allah Ta'ala menciptakan segala sesuatu itu bersandar pada tujuannya. Keindahan bentuknya, susunannya yang mengagumkan, kepala dan aya yang ada di dalamnya, dada dan apa yang dihipunnya, perut dan apa yang ditindaknya, dan kedua kaki dengan apa yang dipikulnya.

2.8 Cardiothoracic Ratio

Postero-anterior standar radiografi dada telah diterapkan pada departemen radiologi. *Cardiothoracic ratio* seperti yang telah dijelaskan oleh Danzer, garis vertikal ditarik pada depan film melalui pembesaran *procardiac spinosus vertebra*. Jumlah jarak maksimal dari baris ini ke kanan dan kiri batas jantung adalah diameter transversal. Nilai ini dibagi dengan lebar terbesar dada, yang diukur dari dalam margin dari tulang rusuk, untuk memberikan rasio kardioraks. Rasio kardioraks telah dikoreksi untuk tahap inspiarium seperti dijelaskan oleh Onat. Pembesaran jantung terjadi jika *cardiothoracic ratio* bernilai > 0.50 . (Kadir Babaoglu. 2007).

Gambar citra *thorax* dan garis bantu untuk perhitungan *cardiothoracic ratio* dapat dilihat pada gambar 2.8



Gambar 2.8 Gambar garis bantu untuk perhitungan CTR
Sumber: A Shugaba, dkk (2012).

Perhitungan CTR ini sangat berguna untuk mendeteksi penyakit jantung terutama yang ditandai dengan adanya pembesaran ukuran jantung (*cardiomegally*). Kemungkinan penyebab CTR lebih dari 50% diantaranya (Lailyana, 2009) :

- a. Kegagalan jantung (*cardiac failure*).
- b. *Pericardial effusion*.
- c. *Left or right ventricular hypertrophy*.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

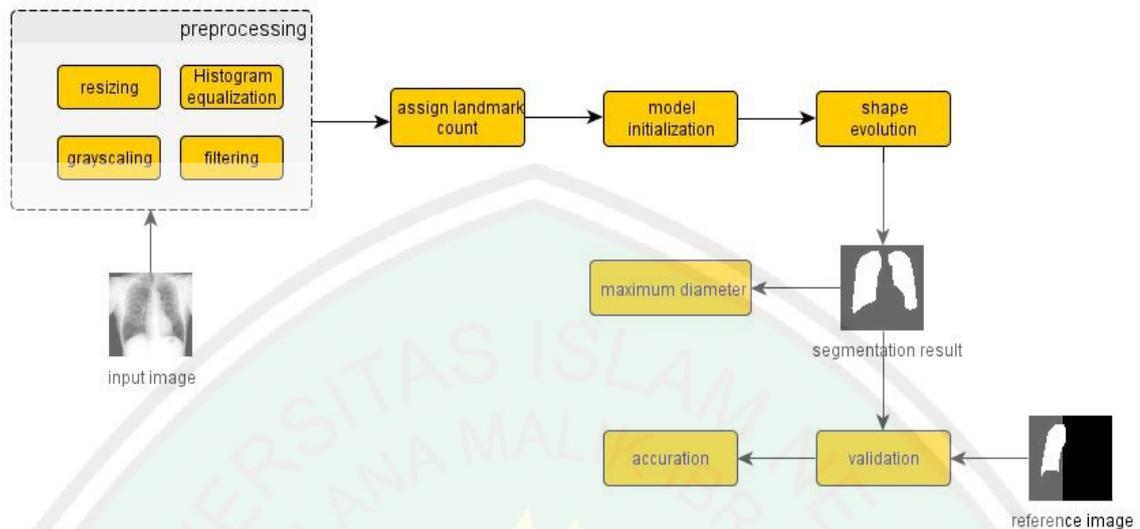
3.1 Deskripsi Sistem

Subbab ini membahas mengenai deskripsi sistem yang dibuat pada skripsi ini. Sistem ini dirancang untuk mengolah sebuah citra *x-ray* rongga dada digital untuk dianalisis sehingga bisa mendapatkan diameter maksimal paru-paru dengan menggunakan metode *Active Shape Model* sebagai metode utama pencarian model citra.

Sistem dibuat dengan model berbasis *desktop*. Tujuannya adalah komputerisasi proses segmentasi dan pengukuran citra *x-ray* rongga dada agar lebih efisien dibandingkan dengan pencarian manual. Sistem ini dibangun menggunakan bahasa pemrograman *Java* sebagai aplikasi segmentasi citra dan menggunakan aplikasi Matlab sebagai alat validasi citra hasil segmentasi dengan citra referensi.

3.2 Desain Sistem

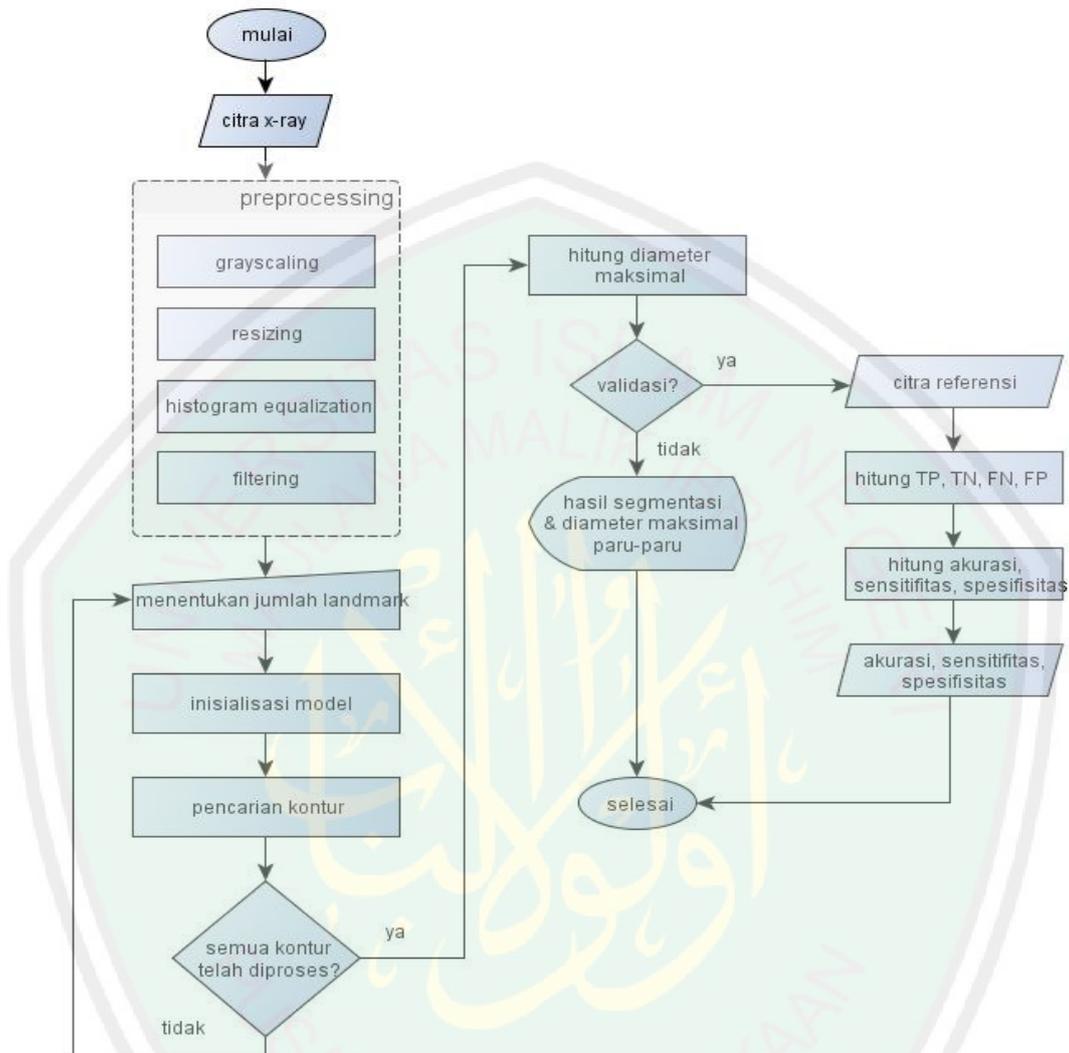
Desain sistem ini meliputi desain data dan desain proses. Desain data merupakan rancangan dan penjelasan data yang diperlukan untuk dapat menerapkan metode *Active Shape Model*. Desain Proses merupakan rancangan proses untuk mengimplementasikan metode *Active Shape Model* berupa algoritma yang dapat digambarkan dengan diagram. Diagram blok aplikasi segmentasi paru-paru dapat dilihat pada gambar 3.1.



Gambar 3.1 Diagram blok aplikasi segmentasi paru-paru

Sebelum menjalankan proses pencarian kontur, citra digital terlebih dahulu dilakukan *preprocessing* sebagai tahap awal pengolahan citra. Setelah tahap *preprocessing* selesai, pencarian kontur paru-paru dengan menggunakan metode *Active Shape Model* bisa dilakukan. Pencarian kontur dan segmentasi citra dilakukan satu persatu dimulai dari pencarian kontur untuk paru-paru kanan kemudian paru-paru kiri. Inisialisasi model dilakukan secara manual di dalam kontur paru-paru. Segmentasi dapat dilakukan setelah paru-paru kanan dan kiri sudah ditemukan oleh model yang diinisialisasi. Hasil segmentasi kemudian dilakukan perhitungan diameter maksimal paru-paru dan divalidasi terhadap citra referensi.

Dari penjelasan diagram blok, dapat digambarkan diagram alir sistem seperti digambarkan pada gambar 3.2.



Gambar 3.2 Diagram alir sistem

Diagram blok sistem pada gambar 3.1 dan diagram alir sistem pada gambar 3.2 merupakan rancangan sistem secara keseluruhan. Dari kedua diagram tersebut dapat dirancang sebuah *form* utama yang memiliki menu dan fitur yang mendukung setiap proses dalam sistem yang dirancang. Form utama tersebut bisa dilihat pada gambar 3.3.



Gambar 3.3 Desain *form* aplikasi utama

3.3 Desain Data

Dalam sistem ini terdapat tiga bagian utama data, yaitu data masukan, data proses, dan data keluaran yang akan digunakan pada proses segmentasi menggunakan metode *Active Shape Model*.

3.3.1 Data Masukan

Data masukan untuk citra percobaan dan citra untuk validasi diambil dari *public medical image databases* pada situs *Cornell University, Vision and Image Analysis Group* di <http://www.via.cornell.edu/databases/> (Ginneken, dkk. 2005). Citra percobaan yang digunakan sebanyak 30 data dan jumlah landmark model yang digunakan sebanyak 33 landmark, 43 landmark, dan 53 landmark. Validasi hasil segmentasi menggunakan data hasil segmentasi uji coba dengan citra referensi.

3.3.2 Data Proses

Data citra masukan yang akan diproses terlebih dahulu dilakukan *preprocessing* berupa konversi ke format *grayscale*, *resizing* citra menjadi ukuran 512 x 512, dan proses perbaikan kualitas citra. *Preprocessing* ini digunakan untuk proses *image search*.

Shape model yang digunakan untuk proses *image search* dibentuk dari inisialisasi model pada kontur yang dilakukan secara manual menggunakan metode labelling image. Inisialisasi *landmark* model pada kontur paru-paru dibatasi oleh jumlah *landmark* yang telah ditentukan sebelumnya.

3.3.3 Data Keluaran

Data keluaran yang dihasilkan adalah segmentasi citra hasil *fitting* antara *shape model* dengan citra masukan. Nilai akurasi, sensitifitas, dan spesifisitas, serta informasi diameter maksimal paru-paru hasil segmentasi dalam satuan *pixel*.

3.4 Desain Proses

Pada subbab ini akan dijelaskan mengenai desain proses yang bertujuan untuk mengetahui hubungan antar proses beserta langkah-langkahnya pada setiap proses dalam membangun perangkat lunak untuk segmentasi citra menggunakan *Active Shape Model*.

Ada tahapan proses yang dirancang untuk aplikasi segmentasi yaitu *preprocessing*, inisialisasi model, pencarian kontur paru-paru, dan pengukuran diameter maksimal paru-paru hasil segmentasi.

3.4.1 Desain *Preprocessing Image*

Pada sistem terdapat proses *preprocessing* pada citra masukan sebelum dilakukan proses *image search* ataupun sebelum proses *labeling training set*. *Preprocessing* ini dibutuhkan untuk menormalisasi citra masukan agar seluruh citra yang akan diproses memiliki kesamaan sehingga tidak memerlukan perubahan variabel-variabel sistem secara keseluruhan. *Preprocessing* yang digunakan adalah sebagai berikut:

1. *Grayscale image*

Tahap normalisasi awal adalah merubah citra input menjadi citra *grayscale* atau citra 8-bit. Dengan merubah citra menjadi mode *grayscale*, maka pengolahan komponen piksel menjadi lebih mudah karena rentang warna pada komponen RGB menjadi sama yaitu 0 – 255.

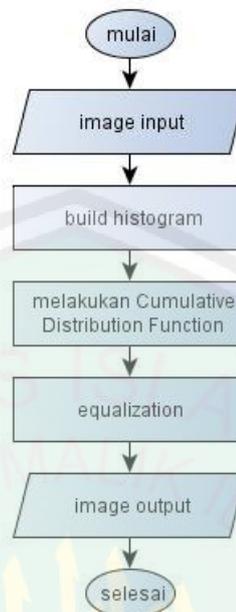
2. *Resizing image*

Resizing image dilakukan pada citra input yang berukuran 256 x 256 menjadi 512 x 512. Penskalaan hingga dua kali ini dimaksudkan untuk mendapatkan resolusi yang lebih baik sehingga diharapkan inisialisasi model dan *output* citra menjadi lebih jelas.

3. *Histogram equalization*

Histogram equalization merupakan salah satu metode perbaikan citra (*image enhancement*) melalui pendistribusian ulang histogram agar lebih merata.

Diagram alir *histogram equalization* bisa digambarkan pada gambar 3.4.



Gambar 3.4 diagram alir histogram equalization

4. Filtering

Konvolusi digunakan sebagai operasi spasial (*filtering*) untuk mengolah citra input menjadi output citra yang diinginkan. Pada penelitian ini proses konvolusi menggunakan dua macam *filtering*, yaitu *filter sharpening* untuk penajaman citra dan *filter emboss* untuk membentuk tepi kontur agar lebih kuat.

Kernel yang digunakan untuk *filter sharpening* dan *filter emboss* adalah:

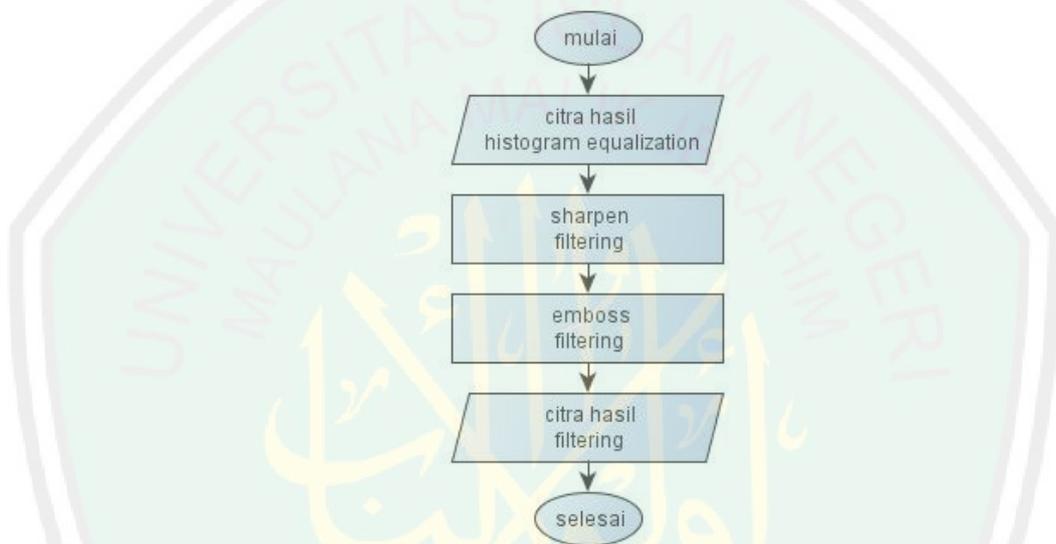
$$\begin{matrix} 1 & 1 & 1 \\ 1 & 7 & 1 \\ 1 & 1 & 1 \end{matrix}$$

(*Sharpening mask*)

$$\begin{matrix} -2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{matrix}$$

(Emboss mask)

Proses penerapan konvolusi pada sistem ini dapat digambarkan dalam diagram alir pada gambar 3.5.



Gambar 3.5 Diagram alir penerapan konvolusi untuk *filtering*

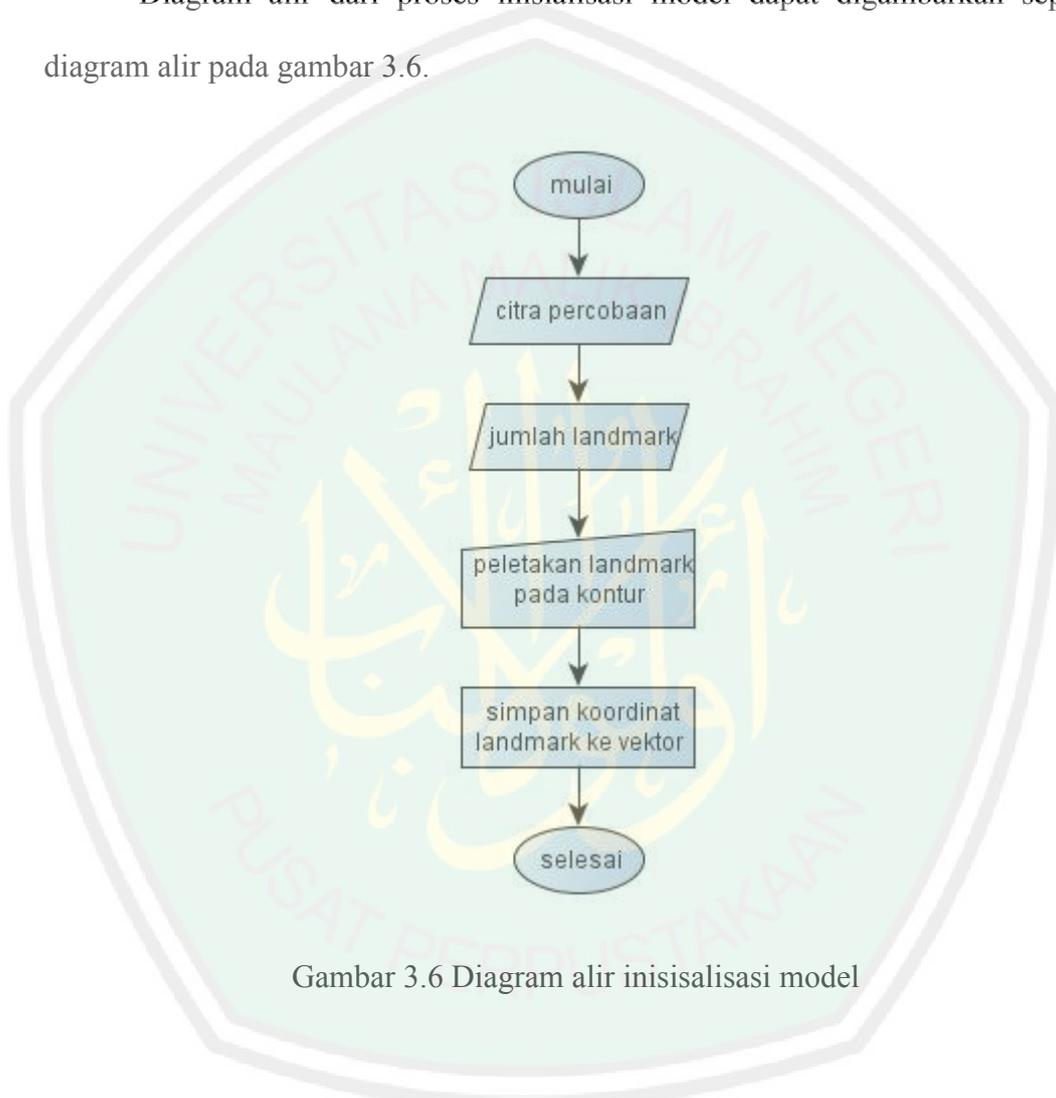
3.4.2 Proses Inisialisasi Model

Inisialisasi model dilakukan untuk menempatkan model pada kontur untuk digunakan sebagai data image search. Proses inisialisasi model dilakukan secara manual dengan menentukan jumlah landmark yang digunakan dan penempatan landmark digunakan metode labelling image. Langkah-langkahnya adalah sebagai berikut:

1. Memasukkan citra percobaan.
2. Menentukan jumlah *landmark* model untuk bagian kontur paru-paru.

3. Meletakkan *landmark* di dalam kontur secara manual.
4. Menyimpan data *landmark* ke dalam vektor.

Diagram alir dari proses inisialisasi model dapat digambarkan seperti diagram alir pada gambar 3.6.



Gambar 3.6 Diagram alir inisialisasi model

3.4.3 Desain Proses Pencarian Kontur Paru-Paru

Image search merupakan sebuah proses yang digunakan untuk mencari tepi suatu kontur melalui pergerakan *landmark* berdasarkan *gray level* citra. Langkah-langkahnya adalah sebagai berikut:

1. Masukan pertama adalah citra percobaan yang telah di-*preprocessing*.
2. Masukan kedua adalah koordinat vektor model inisialiasi

3. Perhitungan *suggested movement* untuk setiap *landmark*.
4. Memperbaharui *pose* dan *shape parameters* menggunakan persamaan 2.25, 2.26, 2.27, 2.28, dan 2.29:

$$X_c \rightarrow X_c + dX_c$$

$$Y_c \rightarrow Y_c + dY_c$$

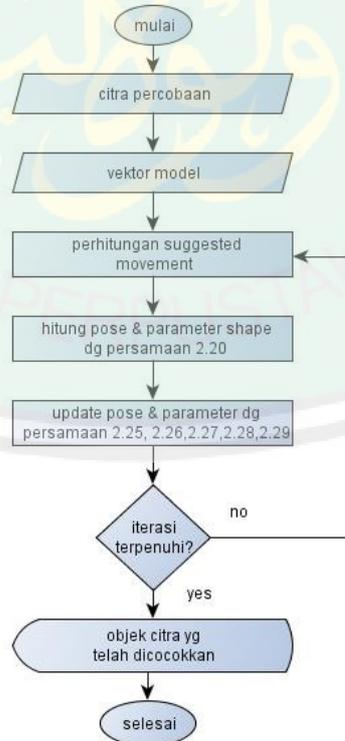
$$\theta \rightarrow \theta + d\theta$$

$$s \rightarrow s(1+ds)$$

$$b \rightarrow b+db$$

5. Jika belum konvergen, kembali ke langkah nomor 3.

Diagram alir untuk proses pencarian citra dapat digambarkan seperti pada gambar 3.7



Gambar 3.7 Diagram alir *image search*

3.4.4 Desain Pengukuran Diameter Maksimal Paru-Paru

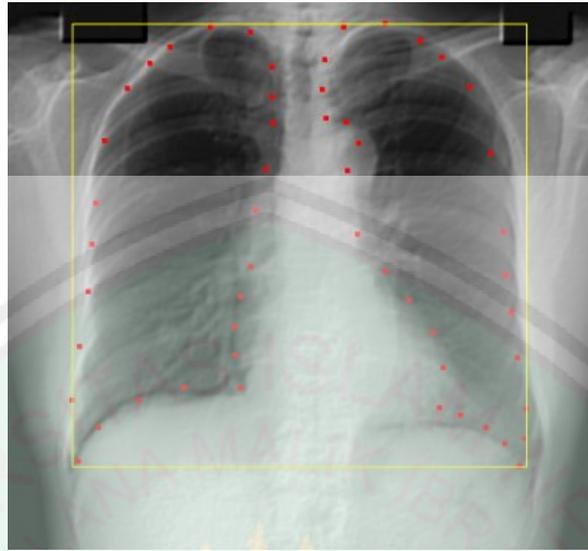
Pengukuran diameter maksimal paru-paru pada hasil pencarian kontur menggunakan *Active Shape Model* dapat dihitung menggunakan jarak koordinat model terluar paru-paru kiri dengan koordinat terluar paru-paru kanan. Untuk memudahkan proses perhitungan, dapat digambarkan sebuah kotak yang didapatkan dari batas-batas terluar dari model. Cara ini sebelumnya digunakan oleh Kim Le dalam jurnalnya, *A Design of A Computer-Aided Diagnostic Tool For Chest X-Ray Analysis* (2011).

Algoritma untuk pembuatan kotak dan perhitungan diameter paru-paru adalah sebagai berikut:

1. Model hasil pencarian paru-paru dijadikan input berupa koordinat-koordinat model.
2. Urutkan koordinat secara *ascending*.
3. Ambil nilai koordinat x terbesar dan terkecil, serta nilai koordinat y terbesar dan terkecil.
4. Tentukan empat titik sudut sebagai penghubung garis pada kotak.

Misalkan:

- a. $T_1 = (x_{\min}, y_{\min})$
 - b. $T_2 = (x_{\max}, y_{\min})$
 - c. $T_3 = (x_{\min}, y_{\max})$
 - d. $T_4 = (x_{\max}, y_{\max})$
5. Hitung diameter maksimal paru-paru dengan persamaan $d = |x_{\max} - x_{\min}|$ (lebar kotak). Diameter maksimal paru-paru dalam satuan piksel.



Gambar 3.8 Kotak pada model untuk membantu proses perhitungan diameter maksimal paru-paru.

Diagram alir untuk pengukuran diameter maksimal paru-paru digambarkan pada gambar 3.9.



Gambar 3.9 Diagram alir pengukuran diameter maksimal paru-paru

3.5 Perhitungan Validasi Segmentasi

Validasi hasil segmentasi dilakukan dengan membandingkan hasil segmentasi melalui *image search* menggunakan *active shape model* dengan hasil segmentasi manual yang didapatkan dari data publik.

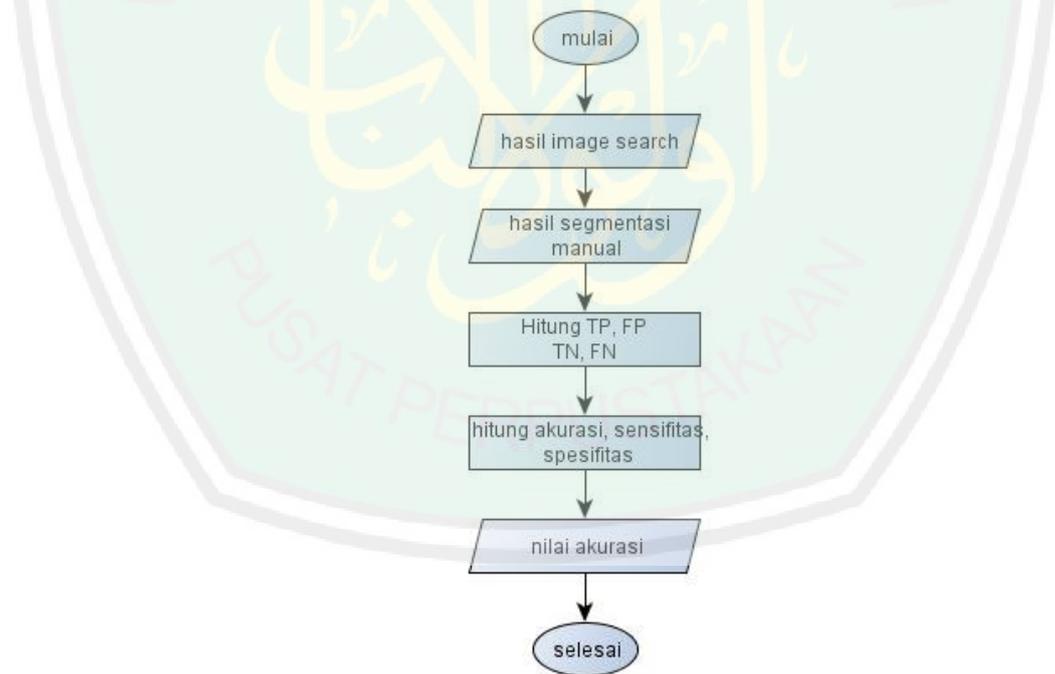
Perhitungan Akurasi, sensitifitas, dan spesifisitas dilakukan dengan persamaan 2.31, 2.32, dan 2.33:

$$\text{Akurasi (ACC)} = (TP + TN) / (TP + FN + FP + TN)$$

$$\text{Sensitifitas (TPR)} = TP / (TP + FN)$$

$$\text{Spesifisitas} = TN / (FP + TN)$$

Proses ini bisa digambarkan melalui diagram alir pada gambar 3.10.



Gambar 3.10 Diagram alir perhitungan validasi hasil segmentasi

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini dibahas mengenai hasil uji coba dan implementasi program yang telah dirancang. Uji coba dilakukan untuk mengetahui apakah program dan metode yang digunakan sudah berjalan sebagai mana mestinya dengan lingkungan uji coba yang telah ditentukan sesuai dengan skenario pengujian yang ada.

4.1 Lingkungan Uji Coba

Subbab ini menjelaskan tentang lingkungan uji coba terhadap aplikasi yang telah dirancang. Lingkungan uji coba meliputi perangkat lunak dan perangkat keras yang digunakan selama proses pengujian. Spesifikasi perangkat keras dan perangkat lunak dalam lingkungan uji coba disajikan dalam tabel 4.1 sebagai berikut:

Tabel 4.1 Spesifikasi Lingkungan Uji Coba

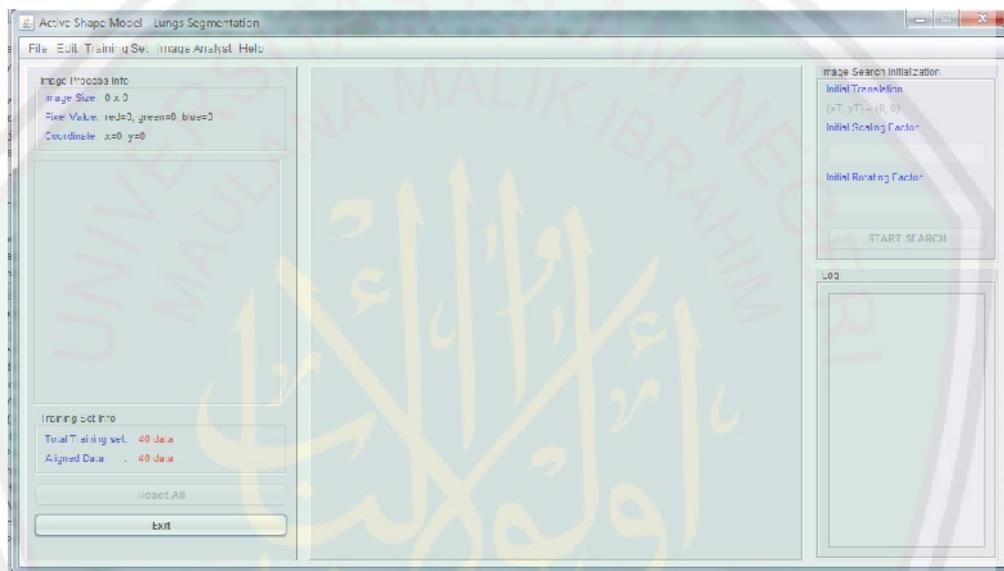
Perangkat Keras	Processor: Dual Core 2.20GHz RAM: 3GB VGA: Mobile Intel(R) 4 Series Express Chipset Family Input device: Keyboard, mouse
Perangkat Lunak	OS: Windows 7 Professional Edition SP 1 Perangkat Pengembang: JDK 7, IDE Netbeans 7.1. Perangkat Validasi: Matlab R2008a

4.2 Implementasi Sistem

Implementasi sistem merupakan proses pembangunan rancangan atau desain yang telah dirancang sebelumnya sekaligus melakukan uji coba keberhasilan sistem yang dibangun.

4.2.1 Form Utama Aplikasi

Form utama pada gambar 4.1 merupakan *form* aplikasi yang pertama kali muncul saat pengguna menggunakan aplikasi ini. Pada *form* ini terdapat kumpulan menu seperti *open file*, *preprocessing*, inialisasi model, perhitungan diameter maksimal dan menampilkan hasil segmentasi.



Gambar 4.1 Implementasi *Form* utama aplikasi

4.2.2 Implementasi *Preprocessing* Citra

Preprocessing citra dilakukan untuk normalisasi dan pengolahan awal citra untuk mendapatkan citra yang bisa mempermudah proses penggunaan metode *Active Shape Model*. *Preprocessing* dilakukan pada citra input saat *labelling* atau saat proses *image search*. Dalam penelitian ini ada 4 macam *preprocessing* yang digunakan yaitu *grayscale*, *resizing*, *histogram equalization*, dan *filtering*.

4.2.2.1 Implementasi Proses *Grayscale*

Grayscale dilakukan untuk merubah komponen warna citra input menjadi mode *grayscale* atau 8-bit. Untuk kemudahan penggunaan, *class grayscale.java* dibuat untuk menangani proses *grayscale*. Berikut method utama yang melakukan proses *grayscale*:

```
public BufferedImage set_grayscale_image(BufferedImage buffer)
{
    width = buffer.getWidth();
    height = buffer.getHeight();
    size = width * height;
    grayscale = new float[size];

    BufferedImage im = new
    BufferedImage(width,height,BufferedImage.TYPE_BYTE_GRAY);
    WritableRaster raster = im.getRaster();

    for(int i=0; i<width; i++)
    {
        for(int j=0; j<height; j++)
        {
            c = buffer.getRGB(i,j);
            r = (c & 0x00ff0000) >> 16;
            g = (c & 0x0000ff00) >> 8;
            b = c & 0x000000ff;
            float gray = (float) (0.3 * r + 0.59 * g + 0.11 * b);
            raster.setSample(i, j, 0, gray);
        }
    }

    return im;
}
```

Class ini menerima masukan citra dalam bentuk *BufferedImage* dan keluaran berupa *BufferedImage* karena masih ada tahap *preprocessing* yang akan dilakukan selanjutnya. Hasil dari *grayscale* pada salah satu citra input dapat dilihat pada gambar 4.2.



Gambar 4.2 Hasil *grayscale* pada citra masukan

4.2.2.2 Implementasi Proses *Resizing*

Java sudah menyediakan class yang bisa digunakan untuk pengolahan citra termasuk operasi *resizing*. Dengan memanfaatkan *class* itu dapat dimasukkan parameter *height* dan *width* suatu citra sehingga proses bisa lebih mudah dilakukan.

Pada proses ini, citra di-*resize* menjadi 512 x 512 sehingga parameter input untuk *width* adalah 512 dan *height* adalah 512. Masukan citra maupun keluaran citra pada proses ini menggunakan tipe data *BufferedImage*. *Method* utama untuk proses *resizing* pada *class Resizing.java* adalah sebagai berikut:

```
public BufferedImage resize(BufferedImage img, int newX, int newY)
{
    BufferedImage dimg = new BufferedImage(newX, newY,
    BufferedImage.TYPE_BYTE_GRAY);
    Graphics2D g = dimg.createGraphics();
    g.setRenderingHint(RenderingHints.KEY_INTERPOLATION,
    RenderingHints.VALUE_INTERPOLATION_BILINEAR);
    g.drawImage(img, 0, 0, newX, newY, null);

    return dimg;
}
```

4.2.2.3 Implementasi Proses *Histogram Equalization*

Sesuai dengan diagram alir pada gambar 3.5, class ini mengatur pembuatan histogram, pengaturan *cumulative distribution function* (cdf), dan equalisasi histogram. Sama dengan bagian *preprocessing* sebelumnya, input dan output citra dalam class ini menggunakan format *BufferedImage*.

Pembuatan histogram dapat dilihat melalui *source code* berikut:

```
public int [] histogram(float[] grayScale)
{
    int [] pixNum = new int [512];
    int size = grayScale.length;
    for(int c = 0; c<512; c++)
    {
        int sum = 0;
        for(int i = 0; i<size; i++) if(grayScale[i]==c) sum++;
        pixNum[c] = sum;
    }
    return pixNum;
}
```

Hasil dari pembuatan histogram kemudian digunakan untuk pembentukan *Cumulative Distributif Function* yang diimplementasikan di dalam *source code* berikut:

```
public int [] getCDF(int [] histogram)
{
    int [] cdf = new int [512];
    int cum = 0;
    for(int i = 0; i<512; i++)
    {
        cum += histogram[i];
        cdf[i] = cum;
    }
    return cdf;
}
```

Di dalam *cumulatve distributif function* juga mengklasifikasikan CDF maksimal dan CDF minimal dari hasil pembentukan CDF sebelumnya. Implementasinya dapat dilihat pada *source code* berikut:

```

public int getMinCDF(int [] cdf)
{
    int minCDF = 513;
    for(int i = 0; i<512; i++)
    {
        if(cdf[i]<minCDF && cdf[i]!=0)
        {
            minCDF = cdf[i];
        }
    }
    return minCDF;
}
public int getMaxCDF(int [] cdf)
{
    int maxCDF = 0;
    for(int i = 0; i<512; i++)
    {
        if(cdf[i]>maxCDF)
        {
            maxCDF = cdf[i];
        }
    }
    return maxCDF;
}

```

Selanjutnya proses equalisasi citra dilakukan melalui *source code* berikut

ini:

```

public float[] equalization(int [] cdf, int pictSize)
{
    int min = getMinCDF(cdf);
    float e [] = new float[512];
    for(int i = 0; i<512; i++)
    {
        e[i] = (float)((float)cdf[i]-min)/(float)pictSize)*255;
    }
    for(int i = 0; i<512; i++)
    {
        if(e[i]<0) e[i]=0;
        if(e[i]>255) e[i]=255;
    }
    return e;
}
public float [] picEqualized(float [] grayScale, float [] equalization,
int w, int h)
{
    int size = w*h;
    float [] newGS = new float[size];
    int counter = 0;
    for(int i = 0; i<w; i++)
    {
        for(int j = 0; j<h; j++)
        {
            newGS [counter] = equalization[(int)grayScale[counter]]; //convert
            counter++;
        }
    }
}

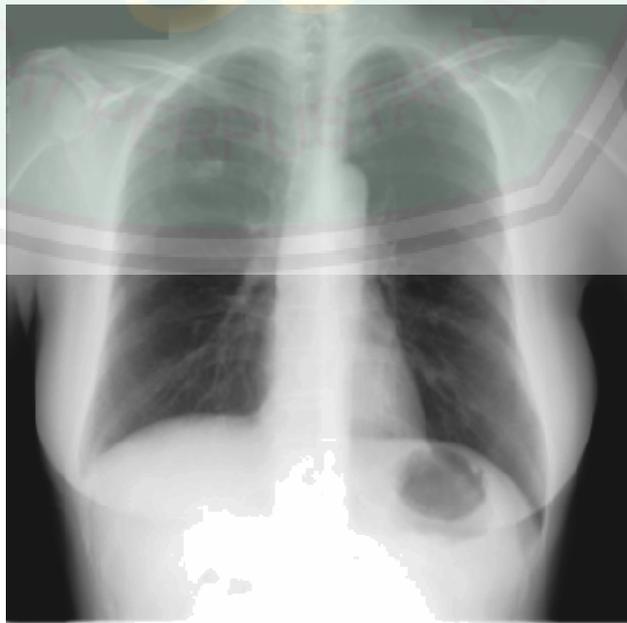
```

Selanjutnya, nilai hasil equalisasi yang sudah didapatkan disusun ulang menjadi sebuah citra dalam bentuk *BufferedImage* agar dapat digunakan dalam tahap *preprocessing* selanjutnya. Implementasinya adalah sebagai berikut:

```
public BufferedImage drawImage(float [] newGS, int w, int h) throws
IOException
{
    int size = w*h;
    int counter = 0;
    BufferedImage im = new BufferedImage(w,h,BufferedImage.TYPE_BYTE_GRAY);
    WritableRaster raster = im.getRaster();

    for(int i = 0; i<w; i++)
    {
        for(int j = 0; j<h; j++)
        {
            raster.setSample(i, j, 0, newGS[counter]);
            counter++;
        }
    }
    return im;
}
```

Proses *histogram equalization* diimplementasikan dalam sebuah *class* *histogram_equalization.java*. Hasil *histogram equalization* yang dilakukan pada citra *grayscale* bisa dilihat pada gambar 4.3.



Gambar 4.3 Hasil *histogram equalization* pada citra *grayscale*

4.2.2.4 Implementasi Proses *Filtering*

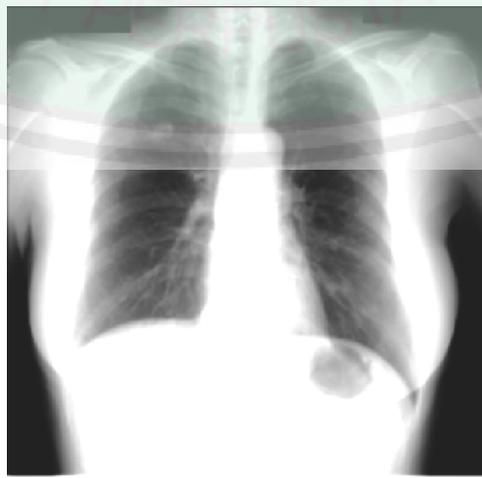
Filtering pada *preprocessing* ini menggunakan *class Convolution* yang sudah tersedia di dalam JDK. *Filtering* yang digunakan adalah filter *sharpen* dan filter *emboss*. Karena *filtering* melibatkan operasi konvolusi, maka dibutuhkan suatu *kernel* atau *mask* untuk pengolahannya. Operasi konvolusi untuk filter *sharpen* menggunakan *mask* sebagai berikut:

$$\begin{matrix} 1 & 1 & 1 \\ 1 & 7 & 1 \\ 1 & 1 & 1 \end{matrix}$$

Implementasi operasi konvolusi untuk filter *sharpen* dapat dilihat pada *source code* berikut:

```
public void sharpening(BufferedImage buffer)
{
    convol = new convolution(new float[]{0.1f, 0.1f, 0.1f, 0.1f, 0.7f,
    0.1f, 0.1f, 0.1f, 0.1f}, buffer);
    this.buffer_image_op = convol.convolve();
}
```

Hasil dari *filtering* menggunakan filter *sharpening* pada citra yang telah dilakukan *histogram equalization* dapat dilihat pada gambar 4.4.



Gambar 4.4 Hasil filter *sharpening* pada citra *histogram equalization*

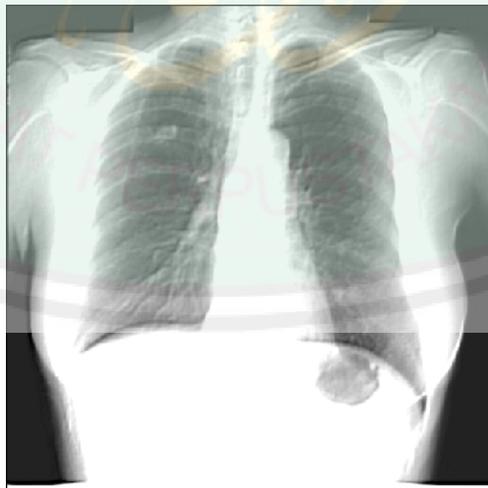
Setelah citra dilakukan proses *filtering* menggunakan *filter sharpening*, selanjutnya adalah dilakukan proses *filtering* menggunakan *filter emboss*. Operasi konvolusi untuk *filter emboss* menggunakan *mask* sebagai berikut:

$$\begin{matrix} -2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{matrix}$$

Implementasi operasi konvolusi menggunakan filter *emboss* dapat dilihat pada *source code* berikut:

```
public void emboss(BufferedImage buffer)
{
    convol = new convolution(new float[] { -2, 0, 0, 0, 1, 0, 0, 0, 2 },
buffer);
    this.buffer_image_op = convol.convolve();
}
```

Hasil dari *filtering* menggunakan *filter emboss* pada citra yang telah dilakukan *filter sharpening* sekaligus tahap akhir *preprocessing* dapat dilihat pada gambar 4.5.



Gambar 4.5 Hasil *filtering emboss* sekaligus tahap akhir *preprocessing*

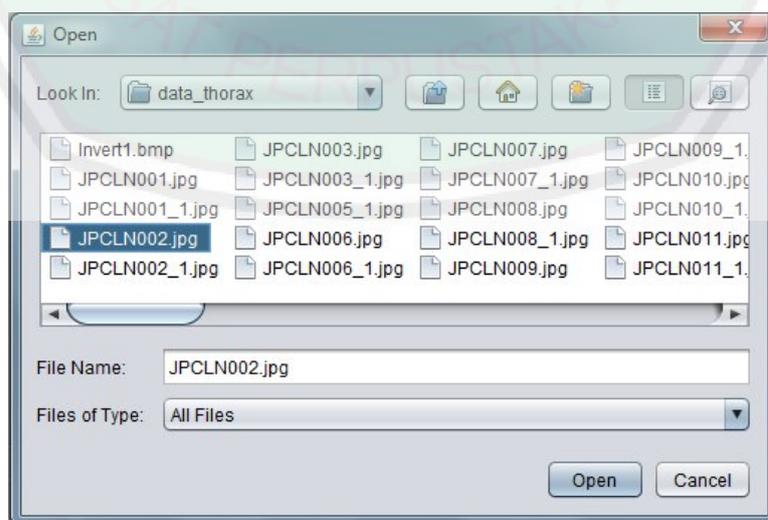
4.2.2.5 Implementasi *Preprocessing* pada Form Utama

Citra percobaan dimasukkan melalui menu *Open Image* yang ada di dalam menu *file* atau bisa mengakses *shortcut* dengan kombinasi tombol *ctrl + o*. Selanjutnya data bisa dipilih sesuai letak data citra melalui kotak dialog yang muncul. *Method* yang menangani proses *open image* bisa dilihat pada *source code* berikut:

```
private void menui_openActionPerformed(java.awt.event.ActionEvent evt) {
    image_path = browse.get_path(main_app.this);
    if(!image_path.equals(""))
    {
        load = new image_load(image_path);
        main_buffer = load.get_BufferImage();
        image_op.resize_image(main_buffer, 256, 256);
        display_image(image_op.get_buffer_image_op(), PANEL_TYPE_THUMB);

        menui_preprocessing.setEnabled(true);
        menui_open.setEnabled(false);
        button_reset.setEnabled(true);
    }else
    {
        JOptionPane.showMessageDialog(null, "Silahkan muat gambar");
    }
}
```

Dari *method* tersebut, sebuah jendela untuk memilih data citra yang akan digunakan bisa dilihat pada gambar 4.6.

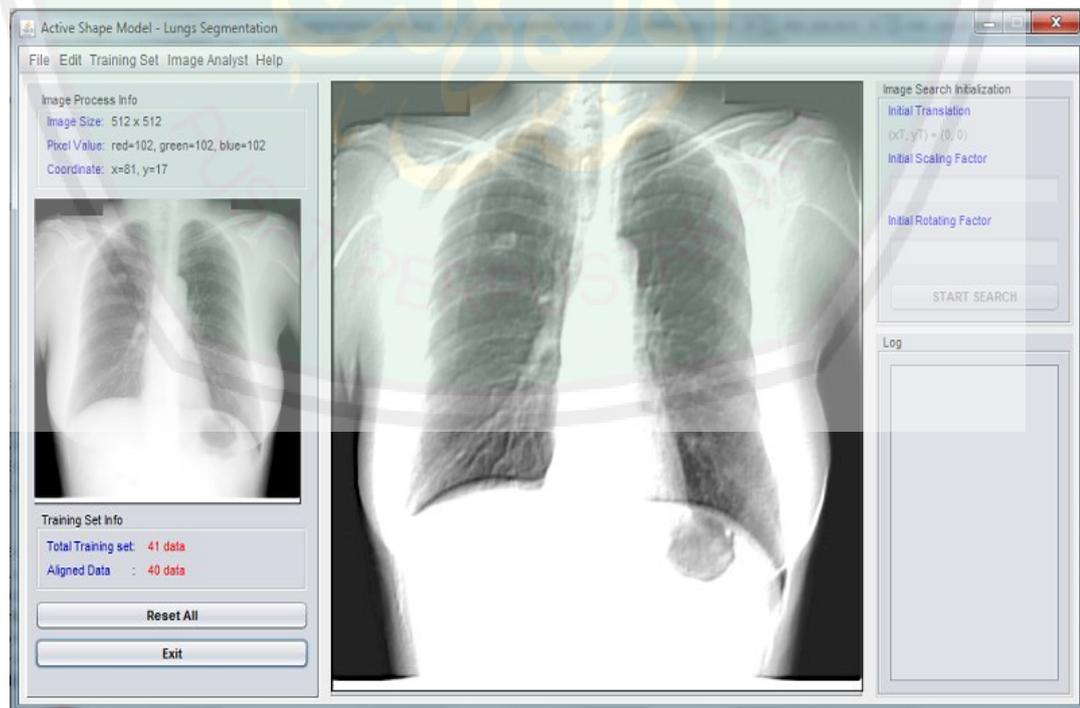


Gambar 4.6 Kotak dialog untuk memilih data citra

Implementasi *preprocessing* pada *form* utama dilakukan dengan menggunakan *menu preprocessing*. Di dalam *menu preprocessing* terdapat pemanggilan *method* dari *class grayscaleing*, *resizing*, *histogram equalization*, dan *filtering*. *Source code* implemetasi dapat dilihat sebagai berikut:

```
image_op.grayscale_image(main_buffer);
image_op.resize_image(image_op.get_buffer_image_op(), 512, 512);
image_op.setHisteq(image_op.get_buffer_image_op());
image_op.sharpening(image_op.get_buffer_image_op());
image_op.emboss(image_op.get_buffer_image_op());
main_buffer = image_op.get_buffer_image_op();
```

Hasil *preprocessing* citra masukan dimunculkan pada panel utama yang terdapat pada *form* utama. Citra pada panel utama tersebut yang nantinya akan dilakukan proses inisialisasi dan *image search*. Hasil *preprocessing* dapat dilihat pada gambar 4.7.



Gambar 4.7 Hasil *preprocessing* citra pada panel utama

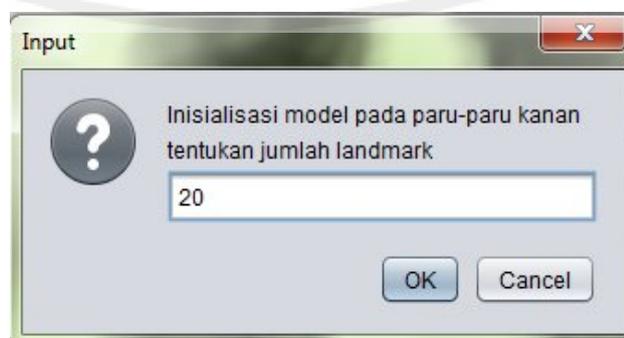
4.2.3 Implementasi Inisialisasi Model

Inisialisasi awal model dilakukan dengan memberikan *landmark* model secara manual pada paru-paru. Proses ini dilakukan satu persatu pada setiap objek paru-paru. Sebelum proses inisialisasi dilakukan, dibutuhkan data masukan berupa jumlah *landmark* yang akan diberikan. Inisialisasi pada kontur paru-paru dimulai dari paru-paru kanan dan inisialisasi paru-paru kiri dilakukan setelah *image search* paru-paru kanan selesai.

Model ditempatkan secara manual pada objek sesuai dengan jumlah *landmark* yang telah ditentukan tadi. Penempatan *landmark* model dilakukan di dalam objek dan sedekat mungkin dengan tepi objek agar proses pencarian bisa lebih optimal. Implementasi penempatan *landmark* model dapat dilihat pada *source code* berikut:

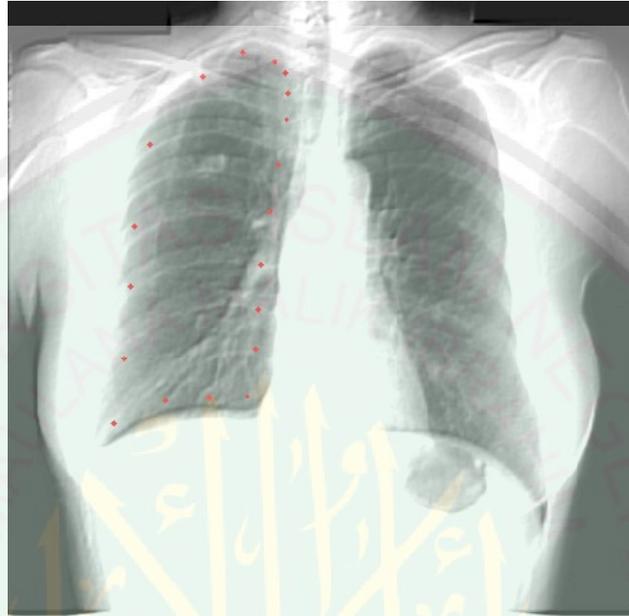
```
point = new points(e.getX(), e.getY());
kanvas.drawPoint(point);
initial_model_new.add(e.getX());
initial_model_new.add(e.getY());
land_count ++;
```

Data masukan untuk penentuan jumlah *landmark* diimplementasikan menggunakan jendela dialog seperti pada gambar 4.8.



Gambar 4.8. Jendela dialog untuk menentukan jumlah *landmark* model

Hasil inialisasi model secara manual di dalam kontur paru-paru dapat dilihat pada gambar 4.9.



Gambar 4.9. Inialisasi model pada paru-paru kanan

4.2.4 Implementasi Pencarian Kontur Paru-Paru

Setelah proses inialisasi awal model selesai dilakukan, maka proses *image search* bisa dilakukan. Perhitungan *suggested move* dilakukan sepanjang garis *profile* pencarian.

Perhitungan *suggested move* dilakukan dengan memeriksa *gray value* di sepanjang *profile* mencari perbedaan terbesar untuk memindahkan *landmark*.

Implementasinya pada *source code* `image_search.java` adalah sebagai berikut:

```
int d1 = Math.abs(this.greylevel_matrix[py][px] -
this.greylevel_matrix[py1][px1]);
int d2 = Math.abs(this.greylevel_matrix[pyy][pxx] -
this.greylevel_matrix[py2][px2]);

if(d1 > 20)
{
    map.put(d1,px1+" "+py1);
```

```

}
if(d2 > 20)
{
    map.put(d2, px2+" "+py2);
}

```

Perhitungan *pose* dan parameter dilakukan ketika sebuah iterasi selesai dilakukan dan *landmark* berada pada posisinya yang baru. Dengan menggunakan metode pada *aligning shape*, maka *pose* dan parameter dapat dihitung.

Implementasinya adalah sebagai berikut:

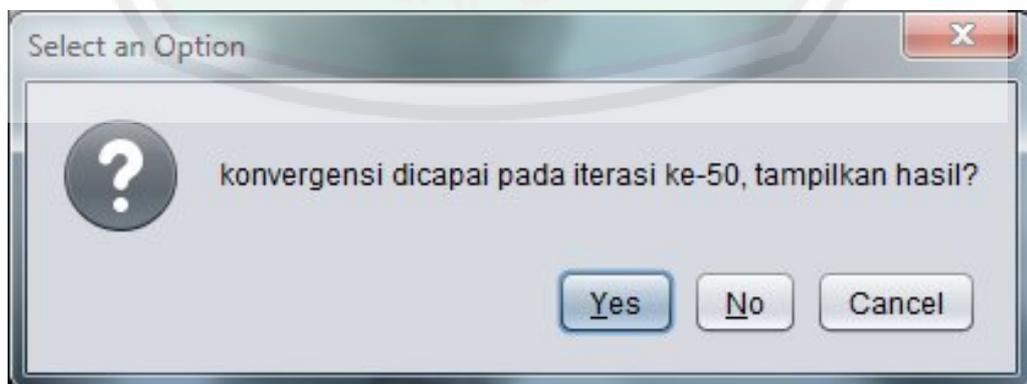
```

// find center of mass new model
transform = new transformation(vect_temp);
transform.center_of_shape();
centerx = (int) transform.get_center_x();
centery = (int) transform.get_center_y();

//align old model with new model
align.set_shape(initial_model, 1);
align.set_shape(vect_temp, 2);
align.translation();
align.scaling_teta_factor();
align.imsearch_calculate_scale_rotate();

```

Konvergensi dicapai ketika seluruh titik dianggap konvergen atau iterasi telah memenuhi batas maksimal iterasi yang telah ditentukan sebelumnya. Kotak dialog akan muncul jika konvergensi telah terpenuhi seperti pada gambar 4.10.



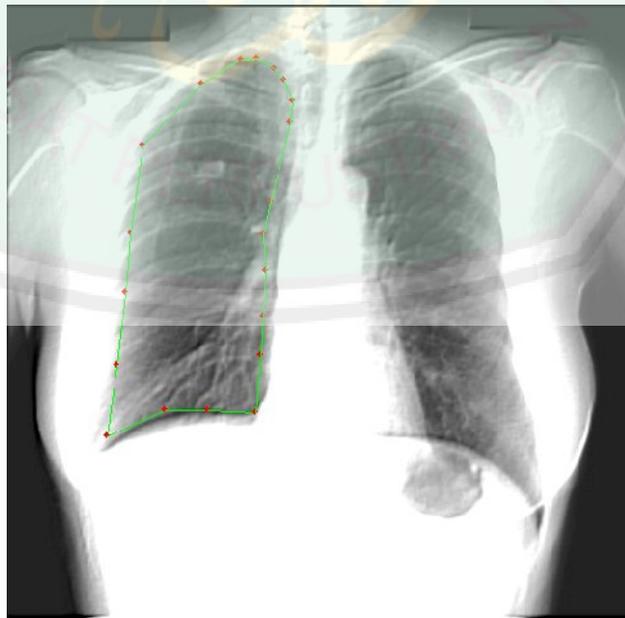
Gambar 4.10 Kotak dialog konvergensi pada proses pencarian kontur

Implementasinya untuk penentuan konvergensi adalah sebagai berikut:

```
if(iteration==500 || converge_status == (model_size/2))
{
    convergen = true;
    iteration_converge = iteration;
    tmap.clear();
    tmap_point.clear();
    converge_status=0;
}
```

Proses *image search* paru-paru kiri dapat dilakukan setelah paru-paru kanan. Menggunakan cara yang sama seperti *image search* paru-paru kanan, inisialisasi model juga diletakkan secara manual di dalam kontur setelah menentukan jumlah *landmark* untuk model paru-paru kiri. Hasil *image search* paru-paru kiri ditampilkan bersama hasil *image search* paru-paru kanan sehingga nampak tampak kedua kontur paru-paru telah melalui proses *image search*.

Hasil proses pencarian kontur paru-paru kanan bisa dilihat pada gambar 4.11.



Gambar 4.11 Hasil proses pencarian kontur untuk paru-paru kanan

Hasil pencarian kontur kontur paru-paru kanan dan kiri dapat dilihat pada gambar 4.12.



Gambar 4.12. Hasil pencarian kontur untuk kedua kontur paru-paru

4.2.5 Implementasi Segmentasi Hasil Pencarian Kontur

Hasil akhir dari *image search* berupa model baru yang sudah dicocokkan dengan objek yang dicarinya. Proses segmentasi dilakukan untuk digunakan dalam perhitungan akurasi dengan data citra hasil segmentasi manual yang sudah disediakan.

Cara segmentasi yang digunakan adalah dengan menggambar ulang model hasil *image search* sesuai koordinat *landmark* yang didapatkan. Model digambarkan pada objek *canvas* yang memiliki ukuran sama dengan citra proses lalu model paru-paru dibedakan warnanya dengan *background* menggunakan warna hitam.

Canvas tersebut ditampilkan pada *form* tersendiri dan dapat dipanggil melalui menu dalam form utama setelah proses *image search* berakhir. Vektor model hasil *image search* dijadikan sebagai masukan dan selanjutnya titik-titik koordinat dari vektor menjadi acuan dalam penggambaran pada *canvas*. Implementasinya ada pada *source code* dalam class *segmentation_canvas.java* sebagai berikut:

```
public void paint(Graphics g)
{
    int point = 1;
    g2d = (Graphics2D)g;

    GeneralPath gp = new GeneralPath();
    GeneralPath gp2 = new GeneralPath();

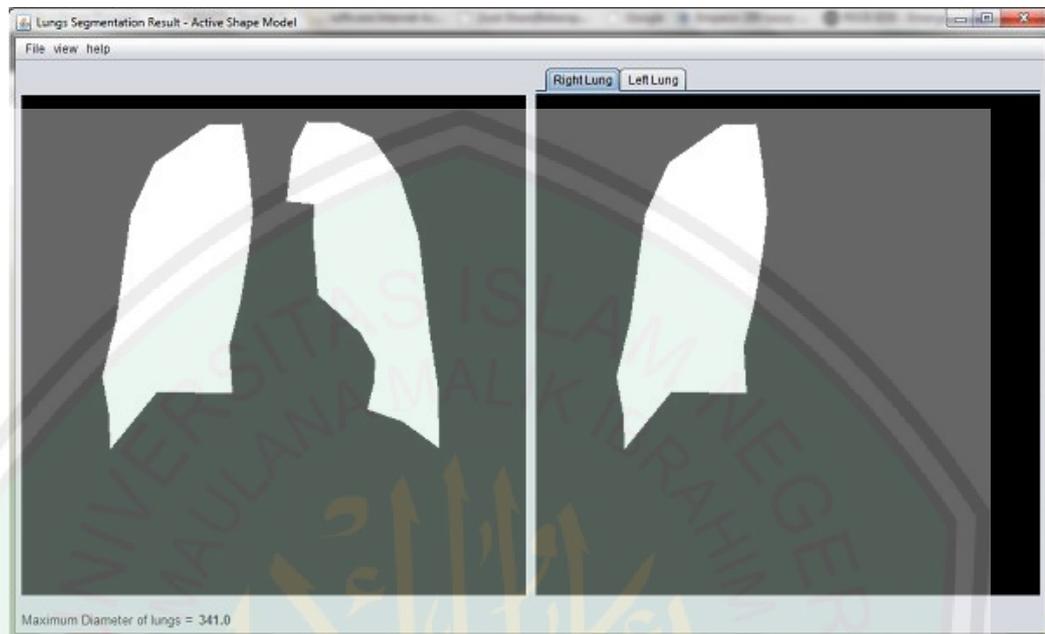
    gp.moveTo(Double.parseDouble(model_shape.get(right_lung_start).toString()), Double.parseDouble(model_shape.get(right_lung_start+1).toString()));
    // right lung

    gp2.moveTo(Double.parseDouble(model_shape.get(left_lung_start).toString()), Double.parseDouble(model_shape.get(left_lung_start+1).toString()));

    int subs_right = Math.abs(Config.last_point_contour_right - Config.first_point_contour_right);
    for(int i=2; i<subs_right+2; i+=2)
    {
        gp.lineTo(Double.parseDouble(model_shape.get(right_lung_start+i).toString()), Double.parseDouble(model_shape.get(right_lung_start+(1+i)).toString()));
    }
    int subs_left = Math.abs(Config.last_point_contour_left - Config.first_point_contour_left);
    for(int j=2; j<subs_left+2; j+=2)
    {
        gp2.lineTo(Double.parseDouble(model_shape.get(left_lung_start+j).toString()), Double.parseDouble(model_shape.get(left_lung_start+(1+j)).toString()));
    }
    g2d.setColor(Color.black); //coloring shapes with black
    g2d.draw(gp);
    g2d.fill(gp);

    g2d.draw(gp2);
    g2d.fill(gp2);
}
```

Implementasi *form* dan hasil segmentasi yang ditampilkan dapat dilihat pada gambar 4.13.



Gambar 4.13 Implementasi *Form* dan hasil segmentasi

Hasil segmentasi ini bisa disimpan menjadi sebuah citra hasil segmentasi. Pada *form* ini terdapat menu *save* yang berfungsi untuk menyimpan citra pada *canvas* menjadi *file* citra. Fungsi tersebut terdapat pada *method* berikut:

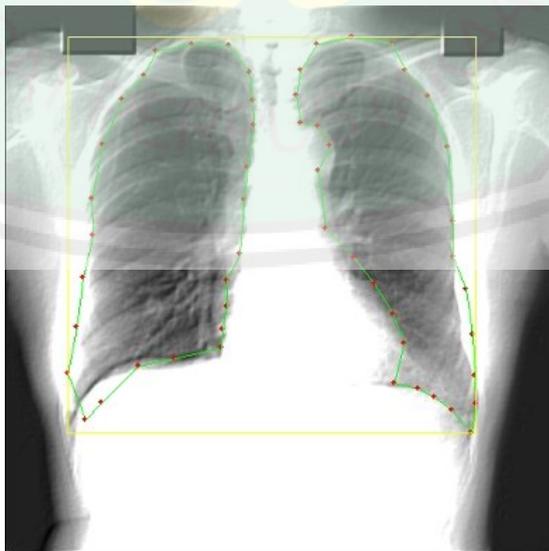
```
private void menu_saveActionPerformed(java.awt.event.ActionEvent evt) {
    Dimension size = kanvas.getSize();
    BufferedImage buffer = new BufferedImage(size.width, size.height,
        BufferedImage.TYPE_BYTE_GRAY);
    Graphics2D graphics = buffer.createGraphics();
    kanvas.paintAll(graphics);
    graphics.dispose();
    FileOutputStream fos = null;
    try
    {
        fos = new FileOutputStream("segmentation.jpg");
        ImageIO.write(buffer, "jpg", fos);
        fos.close();
        JOptionPane.showMessageDialog(null, "data citra telah disimpan");
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
```

4.2.6 Implementasi Perhitungan Diameter Maksimal Paru-Paru

Diameter maksimal paru-paru hasil segmentasi didapatkan dengan membentuk kotak dari nilai koordinat terkecil dan terbesar model hasil *image search*. Titik-titik ditetapkan sesuai dengan algoritma pada subbab 3.2.2.6 dan garis dibentuk melalui titik-titik tersebut. Implementasinya pada *class max_distance.java* adalah sebagai berikut:

```
private void create_box()
{
    List<Integer>listx = new ArrayList<>();
    List<Integer>listy = new ArrayList<>();
    for(int i=0;i<this.model.size();i+=2)
    {
        listx.add(Integer.parseInt(this.model.get(i).toString()));
        listy.add(Integer.parseInt(this.model.get(i+1).toString()));
    }
    Collections.sort(listx);
    Collections.sort(listy);
    x_max = listx.get((this.model.size()/2)-1);
    x_min = listx.get(0);
    y_max = listy.get((this.model.size()/2)-1);
    y_min = listy.get(0);
}
```

Hasil penempatan titik tersebut digambarkan pada gambar 4.14.

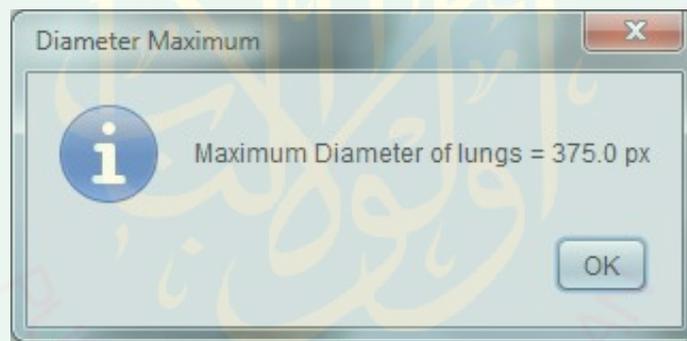


Gambar 4.14 Penempatan kotak untuk perhitungan diameter

Jarak atau diameter maksimal paru-paru dihitung dari selisih koordinat x terkecil dengan koordinat x terbesar. Implementasinya pada *source code* adalah sebagai berikut:

```
private void calculate_distance_byPixel()
{
    distance_byPixel = Math.abs(x_max - x_min);
}
```

Fungsi pemanggilan informasi diameter maksimal paru-paru terdapat pada menu maximum diameter dalam form utama. Menu dapat diakses jika *image search* telah selesai diproses dan model akhir telah didapatkan. Informasi diameter maksimal paru-paru dapat dilihat pada gambar 4.15.



Gambar 4.15 Informasi diameter maksimal paru-paru

4.3 Implementasi Perhitungan Validasi

Hasil segmentasi paru-paru dijadikan format biner dan dipisah antara paru-paru kanan dengan paru-paru kiri. Citra masukan selanjutnya adalah citra referensi yang dijadikan format biner dan disesuaikan dengan paru-paru hasil segmentasi. Ada empat klasifikasi yang didapatkan yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN).

Klasifikasi tersebut dicari menggunakan program matlab sebagai berikut:

```

for i=1:size(C,1)
    for j=1:size(C,2)
        if C(i,j)==2
            TP=TP+1;
        elseif C(i,j)==1
            FN=FN+1;
        end
    end
end
a=sum(sum(A==1));
FP=a-TP;
TN=(512*512)-(TP+FN+FP);

```

Nilai akurasi, sensitifitas, dan spesifisitas dapat dihitung setelah keempat klasifikasi (TP, FP, TP, TN) ditemukan. Implementasinya pada matlab adalah sebagai berikut:

```

akurasi_r=100*(TP+TN)/(TP+FN+FP+TN)
sensitifitas_r=100*(TP/(TP+FN))
spesifitas_r=100*(TN/(FP+TN))

```

4.4 Hasil Uji Coba Validasi Segmentasi Paru-Paru

Validasi hasil segmentasi paru-paru menggunakan metode *Active Shape Model* dengan citra referensi dilakukan pada 20 citra percobaan dengan jumlah *landmark* yang dipakai adalah 33, 43, dan 53. *Landmark* sejumlah 33 dibagi menjadi 15 titik untuk paru-paru kanan dan 18 titik untuk paru-paru kiri, *landmark* sejumlah 43 dibagi menjadi 20 titik untuk paru-paru kanan dan 23 titik untuk paru-paru kiri, sedangkan *landmark* sejumlah 53 dibagi menjadi 25 titik untuk paru-paru kanan dan 28 untuk paru-paru kiri.

Hasil uji coba segmentasi paru-paru kanan dengan menggunakan 15 *landmark* dapat dilihat pada tabel 4.2

Tabel 4.2 Hasil uji coba segmentasi paru-paru kanan dengan 15 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	95,6223	84,0996	97,5080
2	JPCLN002	96,0072	81,3289	97,8993
3	JPCLN003	96,6057	84,9038	98,1259
4	JPCLN005	93,4669	76,3329	96,2104
5	JPCLN006	94,8714	81,905	97,0857
6	JPCLN007	93,1602	80,9133	95,8336
7	JPCLN008	92,7536	82,4542	95,4585
8	JPCLN009	95,1721	84,9523	97,1665
9	JPCLN010	93,8442	76,3285	99,5306
10	JPCLN011	93,6157	79,1923	98,2345
11	JPCLN012	92,474	78,0929	95,7665
12	JPCLN013	95,5051	84,3117	99,613
13	JPCLN014	98,0919	89,9383	99,8648
14	JPCLN015	93,6981	78,1862	98,0865
15	JPCLN016	95,129	77,5641	98,9207
16	JPCLN017	94,4759	82,9565	96,953
17	JPCLN019	96,3585	88,7664	98,071
18	JPCLN020	97,2778	89,1359	99,0671
19	JPCLN021	91,4909	75,977	95,1382
20	JPCLN022	97,5327	87,2362	99,7356
Rata-rata		94,8856	82,4119	97,2552

Tabel 4.2 merupakan tabel hasil validasi uji coba segmentasi paru-paru kanan pada 20 citra percobaan yang menggunakan 15 landmark dengan citra referensi yang sesuai dengan data uji coba. Hasil validasi menunjukkan rata-rata akurasi sebesar 94,8856%, rata-rata sensitifitas sebesar 82,4119%, dan rata-rata spesifisitas sebesar 97,2552%.

Hasil uji coba segmentasi paru-paru kanan dengan menggunakan 20 landmark dapat dilihat pada tabel 4.3.

Tabel 4.3 Hasil uji coba segmentasi paru-paru kanan dengan 20 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	94,1227	83,8157	96,4609
2	JPCLN002	96,3943	83,1161	98,1007
3	JPCLN003	96,3139	83,8041	97,9298
4	JPCLN005	93,7645	77,3639	96,3958
5	JPCLN006	94,2219	79,7733	96,7741
6	JPCLN007	95,3678	86,9863	97,2027
7	JPCLN008	93,8686	85,1121	96,1702
8	JPCLN009	95,9373	87,201	97,6525
9	JPCLN010	94,685	79,4602	96,9833
10	JPCLN011	95,5723	85,6094	97,4041
11	JPCLN012	96,3337	86,0844	97,9113
12	JPCLN013	95,0123	82,5328	97,2811
13	JPCLN014	95,1	83,9049	97,151
14	JPCLN015	96,0293	85,4729	97,8878
15	JPCLN016	95,8828	80,7599	99,0729
16	JPCLN017	93,927	80,5041	96,9176
17	JPCLN019	96,5714	90,0739	98,0044
18	JPCLN020	97,3087	89,5425	99,0361
19	JPCLN021	93,2907	79,8527	96,562
20	JPCLN022	96,0396	83,1677	98,9089
Rata-rata		95,2871	83,7068	97,4903

Tabel 4.3 merupakan tabel hasil validasi uji coba segmentasi paru-paru kanan pada 20 citra percobaan yang menggunakan 20 landmark dengan citra referensi. Hasil validasi menunjukkan rata-rata akurasi sebesar 95,2871%, rata-rata sensitifitas sebesar 83,7068%, dan rata-rata spesifisitas sebesar 97,4903%. Uji coba ini menghasilkan rata-rata akurasi, sensitifitas, dan spesifisitas yang lebih tinggi daripada uji coba menggunakan 15 landmark.

Hasil uji coba segmentasi paru-paru kanan dengan menggunakan 25 landmark dapat dilihat pada tabel 4.4

Tabel 4.4 Hasil uji coba segmentasi paru-paru kanan dengan 25 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	95,5021	87,5147	97,3159
2	JPCLN002	96,7499	84,7168	98,2942
3	JPCLN003	96,666	85,2478	98,1454
4	JPCLN005	95,8725	84,9046	97,637
5	JPCLN006	94,7121	81,5285	97,0302
6	JPCLN007	94,7201	85,1845	96,8075
7	JPCLN008	94,5198	86,5885	96,6108
8	JPCLN009	95,295	85,4481	97,2148
9	JPCLN010	94,9661	80,5844	97,1328
10	JPCLN011	94,9936	83,6662	97,0822
11	JPCLN012	93,7451	82,1007	97,2852
12	JPCLN013	94,1134	83,662	96,7635
13	JPCLN014	95,8153	86,5321	98,7521
14	JPCLN015	92,7142	86,5148	95,2275
15	JPCLN016	96,1558	84,6678	96,7061
16	JPCLN017	97,9788	84,4852	94,5977
17	JPCLN019	95,2389	85,6516	97,4069
18	JPCLN020	97,0119	89,487	98,6465
19	JPCLN021	92,8913	80,3458	95,7583
20	JPCLN022	94,2497	79,9357	97,2245
	Rata-rata	95,3458	84,4883	97,082

Tabel 4.4 merupakan hasil validasi uji coba segmentasi paru-paru kanan pada 20 citra percobaan yang menggunakan 25 landmark dengan citra referensi. Hasil validasi menunjukkan rata-rata akurasi sebesar 95,3458%, rata-rata sensitifitas sebesar 84,4883%, dan rata-rata spesifisitas sebesar 97,082%.

Hasil uji coba segmentasi paru-paru kiri dengan menggunakan 18 landmark dapat dilihat pada tabel 4.5

Tabel 4.5 Hasil uji coba segmentasi paru-paru kiri dengan 18 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	95,6223	84,0996	97,508
2	JPCLN002	97,0287	84,4274	98,3741
3	JPCLN003	97,4098	86,7475	98,5835
4	JPCLN005	97,9141	88,4609	98,9073
5	JPCLN006	95,2805	82,0687	97,3129
6	JPCLN007	95,0153	81,8063	97,2791
7	JPCLN008	95,6882	87,2617	97,4769
8	JPCLN009	96,574	86,209	98,0942
9	JPCLN010	95,2095	76,1581	97,417
10	JPCLN011	95,7336	84,3324	97,5732
11	JPCLN012	95,1607	81,7356	97,2235
12	JPCLN013	96,0705	82,8665	97,8382
13	JPCLN014	96,8102	84,819	98,934
14	JPCLN015	94,9921	80,6722	97,1761
15	JPCLN016	92,9775	61,8844	98,4492
16	JPCLN017	91,4726	61,1166	96,7114
17	JPCLN019	97,4655	87,6914	99,1561
18	JPCLN020	97,1443	85,9267	99,137
19	JPCLN021	96,0037	82,9961	98,1626
20	JPCLN022	96,2982	84,2715	98,428
Rata-rata		95,7935	81,7775	97,9871

Tabel 4.5 merupakan tabel hasil validasi uji coba segmentasi paru-paru kiri pada 20 citra percobaan yang menggunakan 18 landmark dengan citra referensi. Hasil validasi menunjukkan rata-rata akurasi sebesar 95,7935%, rata-rata sensitifitas sebesar 81,7775%, dan rata-rata spesifisitas sebesar 97,9871%.

Hasil uji coba segmentasi paru-paru kanan dengan menggunakan 23 landmark dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil uji coba segmentasi paru-paru kanan dengan 23 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	97,1375	89,3904	98,4089
2	JPCLN002	97,0806	84,3056	98,4618
3	JPCLN003	97,3278	86,2127	98,5559
4	JPCLN005	97,3701	85,4806	98,624
5	JPCLN006	96,1109	85,1018	97,809
6	JPCLN007	96,4664	86,8307	98,1144
7	JPCLN008	95,261	86,1283	97,1939
8	JPCLN009	96,4375	85,915	97,9689
9	JPCLN010	95,8839	79,3108	97,8112
10	JPCLN011	97,2038	88,9435	98,5637
11	JPCLN012	93,6546	81,3492	96,2375
12	JPCLN013	97,0512	86,5114	98,4846
13	JPCLN014	97,151	97,151	97,527
14	JPCLN015	96,5866	86,5053	98,1335
15	JPCLN016	93,5528	63,7805	98,8036
16	JPCLN017	94,0521	70,0376	98,1583
17	JPCLN019	97,5056	87,5003	99,2539
18	JPCLN020	97,1088	86,0728	99,0524
19	JPCLN021	94,5168	75,7101	97,9341
20	JPCLN022	96,3726	84,5775	98,4572
	Rata-rata	96,1915	83,8407	98,1776

Tabel 4.6 merupakan tabel hasil validasi uji coba segmentasi paru-paru kiri pada 20 citra percobaan yang menggunakan 23 landmark dengan citra referensi. Hasil validasi menunjukkan rata-rata akurasi sebesar 96,1915%, rata-rata sensitifitas sebesar 83,8407%, dan rata-rata spesifisitas sebesar 98,1776%.

Hasil uji coba segmentasi paru-paru kanan dengan menggunakan 28 landmark dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil uji coba segmentasi paru-paru kanan dengan 28 landmark

No	Citra percobaan	Akurasi (%)	Sensitifitas (%)	Spesifisitas (%)
1	JPCLN001	96,9082	88,4846	98,2945
2	JPCLN002	97,9744	88,915	98,953
3	JPCLN003	97,4583	86,9108	98,6221
4	JPCLN005	97,1298	84,0522	98,5165
5	JPCLN006	96,8166	87,8829	98,1887
6	JPCLN007	96,653	88,1438	98,0783
7	JPCLN008	95,4395	86,8243	97,2519
8	JPCLN009	96,4687	85,9259	98,0086
9	JPCLN010	95,9324	79,032	97,9365
10	JPCLN011	96,1021	85,3807	97,8478
11	JPCLN012	95,6053	88,1802	98,2835
12	JPCLN013	94,2874	87,7256	97,2715
13	JPCLN014	95,2003	83,143	97,7391
14	JPCLN015	97,3712	85,3671	98,0172
15	JPCLN016	98,5621	88,2761	97,2817
16	JPCLN017	92,2119	64,6332	96,5926
17	JPCLN019	97,1298	86,4965	98,9724
18	JPCLN020	96,7167	84,5521	98,8745
19	JPCLN021	93,7546	75,0662	96,8606
20	JPCLN022	95,0035	80,7658	97,4482
	Rata-rata	96,2363	84,2879	97,952

Tabel 4.6 merupakan tabel hasil validasi uji coba segmentasi paru-paru kiri pada 20 citra percobaan yang menggunakan 28 landmark dengan citra referensi. Hasil validasi menunjukkan rata-rata akurasi sebesar 96,2363%, rata-rata sensitifitas sebesar 84,2879%, dan rata-rata spesifisitas sebesar 97,952%.

Dari hasil uji coba segmentasi paru-paru kanan menggunakan *landmark* sejumlah 15 *landmark*, 20 *landmark*, dan 25 *landmark*, dapat dilihat perbandingannya pada tabel 4.8

Tabel 4.8 Perbandingan rata-rata validasi segmentasi paru-paru kanan

No.	Jumlah landmark	Rata-rata akurasi	Rata-rata sensitifitas	Rata-rata spesifisitas
1	15	94,8856%	82,4119%	97,2552%
2	20	95,2871%	83,7068%	97,4903%
3	25	95,3455%	84,4883%	97,082%

Tabel 4.8 menunjukkan bahwa nilai rata-rata akurasi, sensitifitas, dan spesifisitas percobaan kedua lebih tinggi dibandingkan percobaan kedua. Hasil percobaan ketiga menunjukkan nilai rata-rata akurasi dan sensitifitas lebih tinggi daripada percobaan pertama dan kedua namun menghasilkan nilai spesifisitas terendah dibandingkan dengan percobaan pertama dan kedua.

Dari hasil uji coba segmentasi paru-paru kiri menggunakan *landmark* sejumlah 18 *landmark*, 23 *landmark*, dan 28 *landmark*, dapat dilihat perbandingannya pada tabel 4.9

Tabel 4.9 Perbandingan rata-rata validasi segmentasi paru-paru kiri

No.	Jumlah landmark	Rata-rata akurasi	Rata-rata sensitifitas	Rata-rata spesifisitas
1	18	94,7956%	83,6883%	97,082%
2	23	96,1915%	83,8407%	98,1776%
3	28	96,2363%	84,2879%	97,952%

Tabel 4.8 menunjukkan bahwa nilai rata-rata akurasi, sensitifitas, dan spesifisitas percobaan kedua lebih tinggi dibandingkan percobaan kedua. Hasil

percobaan ketiga menunjukkan nilai rata-rata akurasi, sensitifitas, dan spesifisitas lebih tinggi daripada percobaan pertama dan kedua.

4.5 Hasil Uji Coba Diameter Maksimal Paru-Paru

Perhitungan diameter maksimal paru-paru dilakukan dengan menghitung jarak koordinat terdalam dan koordinat terluar paru-paru hasil segmentasi menggunakan *image search Active Shape Model*.

Hasil uji coba perhitungan diameter maksimal paru-paru pada 20 citra percobaan berdasarkan nilai rata-rata akurasi tertinggi dapat dilihat pada tabel 4.10.

Tabel 4.10 Hasil uji coba perhitungan diameter maksimal paru-paru

No.	Citra Percobaan	Diameter maksimal
1	JPCLN001	389
2	JPCLN002	359
3	JPCLN003	340
4	JPCLN005	360
5	JPCLN006	359
6	JPCLN007	392
7	JPCLN008	409
8	JPCLN009	379
9	JPCLN010	355
10	JPCLN011	383
11	JPCLN012	390
12	JPCLN013	374
13	JPCLN014	374
14	JPCLN015	379
15	JPCLN016	374
16	JPCLN017	360
17	JPCLN019	390
18	JPCLN020	397
19	JPCLN021	369
20	JPCLN022	379

4.6 Integrasi Penelitian dalam Islam

Luasnya cakupan Islam juga memasuki ranah kesehatan manusia. Baik kesehatan jasmani maupun rohani. Apapun itu, Islam menganjurkan untuk selalu menjaga kesehatan. Misalkan pada perintah untuk memakan makanan yang halal dan baik seperti yang disebutkan pada Surat Al-Baqarah ayat ke 168.

يَأْتِيهَا النَّاسُ كُلُّوا مِمَّا فِي الْأَرْضِ حَلَالًا طَيِّبًا وَلَا تَتَّبِعُوا خُطُوَاتِ الشَّيْطَانِ إِنَّهُ لَكُمْ عَدُوٌّ مُبِينٌ ﴿١٦٨﴾

Artinya: “Hai sekalian manusia, makanlah yang halal lagi baik dari apa yang ada di bumi dan janganlah kamu mengikuti langkah-langkah setan. Karena sesungguhnya setan itu adalah musuh yang nyata bagimu (Q.S Al-Baqarah 2:168)”

Tak hanya diperintahkan makanan yang halal tapi juga baik. Makanan yang baik adalah makanan yang bersih dan juga menyehatkan serta diolah dengan baik pula sehingga dapat menjaga kesehatan manusia melalui apa yang masuk ke tubuh manusia.

Pentingnya kesehatan tidak hanya mengenai penjagaan kualitas makanan yang dikonsumsi tetapi juga penjagaan kesehatan tubuh dengan menggunakan alat-alat kesehatan. Karena bersifat pencegahan, maka dapat digolongkan sebagai golongan yang berjaga-jaga seperti yang ada pada surat Al-Syu'ara ayat 56 sebagai berikut:

وَإِنَّا لَجَمِيعٌ حَادِرُونَ ﴿٥٦﴾

Artinya: “dan sesungguhnya kami benar-benar golongan yang selalu berjaga-jaga (Q.S Asy-Syu'ara 26: 56)”.

Kesehatan dan penyakit serta penyebabnya terkadang dapat dirasakan secara jelas oleh indera dan kadang-kadang hanya dirasakan oleh alat bukti yang diberikan oleh berbagai gejala (*symptoms*) untuk itu diperlukan pengetahuan tentang terjadinya gejala-gejala penyakit dan kaitannya dengan kesehatan. (Avicenna, 980-2037).

Pengetahuan-pengetahuan tentang penyakit dan kesehatan kemudian membuat alat bantu medis berkembang hingga saat ini. Pengembangan teknologi ke arah komputerisasi atau era digital dapat membantu pendeteksian dini dari gejala-gejala penyakit tertentu sehingga jika kaum muslim memperhatikan Surat *Asy-Syu'ara* ayat 56 maka pengembangan teknologi medis merupakan implementasi dari golongan yang berjaga-jaga khususnya di bidang medis dan kesehatan.

Salah satu tujuan dilakukannya segmentasi citra adalah untuk membagi suatu citra menjadi beberapa daerah di mana setiap daerah memiliki kemiripan atribut. Dari hasil segmentasi kemudian dapat diolah untuk mendapatkan informasi tertentu. *Al-Quran* telah menyebutkan pada surat *Ar-Ra'd* ayat 8 bahwa setiap apa yang diciptakan Allah memiliki ukurannya tersendiri.

اللَّهُ يَعْلَمُ مَا تَحْمِلُ كُلُّ أُنْثَىٰ وَمَا تَغِيصُ الْأَرْحَامُ وَمَا تَزْدَادُ ۗ وَكُلُّ شَيْءٍ عِنْدَهُ

بِمِقْدَارٍ ﴿٨﴾

Artinya: "Allah mengetahui apa yang dikandung oleh setiap perempuan, dan kandungan rahim yang kurang sempurna dan yang bertambah. Dan segala sesuatu pada sisi-Nya ada ukurannya. (Q.S Ar-Ra'd 13:8)"

Disebutkan pula pada Surat Al-Furqon ayat 2 sebagai berikut:

الَّذِي لَهُ مُلْكُ السَّمَوَاتِ وَالْأَرْضِ وَلَمْ يَتَّخِذْ وَلَدًا وَلَمْ يَكُنْ لَهُ شَرِيكٌ فِي الْمُلْكِ
وَخَلَقَ كُلَّ شَيْءٍ فَقَدَرَهُ تَقْدِيرًا ﴿٢﴾

Artinya: “Tuhan, yang bagi Dia lah kerajaan seluruh langit dan bumi, dan Dia tidak mengambil anak, dan tidak ada pada-Nya sekutu dalam kerajaan-Nya, dan Dia lah yang menjadikan segala sesuatu, lalu diukurnya menurut ukuran tertentu. (Q.S Al-Furqon 25: 2)”

Sebuah objek atau kontur dari sebuah citra memiliki ukuran tersendiri seperti batas tepi objek, nilai pada piksel, lokasi objek, dan besarnya area objek. Diperlukan perhitungan tertentu untuk menemukan objek yang diinginkan dan memisahkannya dari objek yang lain pada citra digital. Perhitungan-perhitungan tersebut terkumpul menjadi suatu algoritma atau metode tertentu yang di dalamnya terdapat parameter-parameter dengan ukuran atau nilai tertentu. Tingkat keberhasilan algoritma tersebut bergantung pada ketepatan ukuran atau nilai variabel yang diberikan sehingga sesuai dengan surat *Ar-Ra'd* ayat 13 dan surat *Al-Furqon* ayat 2 pada *Al-Quran* bahwa ukuran itu menjadi hal penting.

BAB V

PENUTUP

5.1 Kesimpulan

Dari perancangan, implementasi, dan uji coba sistem yang telah dilakukan, maka dapat disimpulkan beberapa hal sebagai berikut:

- a. Sistem yang dibangun mampu melakukan segmentasi paru-paru menggunakan metode *Active Shape Model* dengan tingkat akurasi tertinggi paru-paru kanan mencapai 95,3455% untuk 25 *landmark* dan paru-paru kiri mencapai 96,2363% untuk 28 *landmark*.
- b. Semakin banyak jumlah *landmark* yang digunakan, akurasi dan sensitifitas yang diperoleh semakin baik.
- c. Diameter maksimal paru-paru didapatkan melalui titik terluar hasil segmentasi.

5.2 Saran

Untuk penelitian lebih lanjut, ada beberapa hal yang disarankan, yaitu:

- a. Menambahkan data *training* dan *landmark* agar memperoleh variasi model yang lebih banyak.
- b. Di dalam penelitian ini masih ada kesulitan dalam menentukan selisih *gray value* pada profil untuk mendapatkan tepi. *Preprocessing* citra yang lebih baik diperlukan untuk mendapatkan tepi kontur dengan baik.

DAFTAR PUSTAKA

- Babaoglu, K., Yilmaz, E., & dkk. (2007). *Predictive Value of Cardiothoracic Ratio as a Marker of Severity of Aortic Regurgitation and Mitral Regurgitation*. Istanbul: Istanbul University.
- Bartee, T. C. (1994). *Dasar Komputer Digital Edisi ke-6*. Jakarta: Erlangga.
- Cootes, T. F., & Taylor, C. J. (2004). *Statistical Model of Appearance for Computer Vision*. University of Manchester.
- Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1994). *Active Shape Models, Their Training and Application*. University of Manchester.
- Dorce, M. (Gangguan Fungsi Paru dan Faktor-faktor yang Mempengaruhinya pada Karyawan PT. Semen Tanosa Pengkep Sulawesi Selatan). 2006. Semarang: Universitas Diponegoro.
- Efford, N. (2000). *Digital Image Processing: a Practical Introducing Using Java*. USA: Pearson Education Limited.
- Farid, H. *Fundamentals of Image Processing*. Dartmouth College.
- ginneken, B. v., Stegmann, M., & Loog, M. (2006). Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Medical Image Analysis* , 10, 19-40.
- Gonzalez, R. C., & Woods, R. E. (2002). *Digital Image Processing*. USA: Prentice-hall,inc.
- Hamarneh, G. (____). *Active Shape Model, Modelling Shape Variations and Gray Level information and an Application to Image Search and Classification*. Sweden: Chalmers University of Technology.
- Hariyadi, M. A., & Lailyana, E. (2011). Max-Tree Filtering dan Geometri Kontur Aktif untuk Menentukan Segmentasi Paru-Paru pada Citra X-Ray thorax. *4E*.
- Jaya, S. S. (2009). *Perancangan dan Pembuatan Aplikasi Online Caricature Generator*. Surabaya: Universitas Kristen Petra.

- Lailyana, E. (2009). *Segmentasi Paru-Paru pada Citra X-Ray Menggunakan Level Set*. Surabaya: Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh November.
- Le, K. (2011). A Design of A Computer-Aided Diagnostic Tool for Chest X-Ray Analysis. *IJCST* , 212.
- Putra, D. (2010). *Pengolahan Citra Digital*. Yogyakarta: Andi Publisher.
- Qurthubi, A. (2008). *Tafsir Al-Qurthubi 11*. Jakarta: Pustaka Azzam.
- Rizvandii, N. B., Pizurica, A., & Philips, W. (t.thn.). Deformable Shape Description Using Active Shape Model (ASM).
- Shengwen Guo, B. F. (2009). A Minimal Path Searching Approach for Active Shape Model (ASM)-based Segmentation of Lung. *Medical Imaging* , 7259.
- Shugaba, A., Umar, M., & dkk. (2012). Cardiothoracic ratio of non - hypertensive patients at Jos University Teaching Hospital (JUTH). *1* (7).
- Wang, C., Guo, S., Wu, J., Liu, Q., & Wu, X. (2008). Lung Region Segmentation Based on Multi-Resolution Active Shape Model. *IFMBE Proceedings* , 19, 260.
- Wardhana, G. W. (2009). *Pengukuran Lebar Cortical Bone Berbasis Active Shape Model*. Surabaya: Institut Teknologi 10 November.

Lampiran 1

HASIL DATA VALIDASI SEGMENTASI PARU-PARU

1. Validasi Segmentasi Paru-Paru Kanan dengan 15 Landmark

No	Citra Percobaan	validasi			
		TP	FP	TN	FN
1	JPCLN001	42952	5238	207216	6738
2	JPCLN002	26767	1872	224467	9038
3	JPCLN003	26070	3868	221926	10280
4	JPCLN005	30144	6037	217671	8292
5	JPCLN006	35707	2608	219576	4253
6	JPCLN007	42415	4555	208682	6492
7	JPCLN008	44222	9118	198201	10603
8	JPCLN009	36463	6194	212405	7082
9	JPCLN010	31992	1052	223058	6042
10	JPCLN011	36844	3759	209156	12385
11	JPCLN012	37599	8806	199204	16535
12	JPCLN013	38478	836	215176	7654
13	JPCLN014	39531	295	217824	4494
14	JPCLN015	34104	4147	212581	11312
15	JPCLN016	38080	2316	212277	9471
16	JPCLN017	3848	6574	209177	7907
17	JPCLN019	42828	4126	209770	5420
18	JPCLN020	42098	2005	212910	5131
19	JPCLN021	37911	10319	201927	11987
20	JPCLN022	40304	571	215372	5897

2. Validasi Segmentasi Paru-Paru Kanan dengan 20 Landmark

No	Citra Percobaan	validasi			
		TP	FP	TN	FN
1	JPCLN001	43362	4828	207706	6248
2	JPCLN002	27672	967	222503	11002
3	JPCLN003	26624	3314	222248	9958
4	JPCLN005	34538	1643	222017	3946
5	JPCLN006	35293	3288	218622	4941
6	JPCLN007	40628	6342	207456	7718
7	JPCLN008	44949	9439	197505	10251
8	JPCLN009	39185	3472	214308	5179
9	JPCLN010	33485	706	223640	4313
10	JPCLN011	33450	7153	207440	14101
11	JPCLN012	38582	6566	203559	13437
12	JPCLN013	38990	324	216786	6044
13	JPCLN014	39779	590	215348	6427
14	JPCLN015	33807	4444	212444	11449
15	JPCLN016	36879	2007	214472	8786
16	JPCLN017	38452	6608	207772	9312
17	JPCLN019	42668	4286	210488	4702
18	JPCLN020	42710	2067	212379	4988
19	JPCLN021	40982	7248	203574	10340
20	JPCLN022	39740	2339	212022	8043

3. Validasi Segmentasi Paru-Paru Kanan dengan 25 Landmark

No	Citra Percobaan	Validasi			
		TP	FP	TN	FN
1	JPCLN001	41678	6512	204922	9032
2	JPCLN002	28274	949	223157	9764
3	JPCLN003	24300	5638	218061	14145
4	JPCLN005	33437	2744	220606	5357
5	JPCLN006	35626	2955	218806	4757
6	JPCLN007	41159	5811	207648	7526
7	JPCLN008	4741	6974	198845	8911
8	JPCLN009	37746	4911	212291	7196
9	JPCLN010	31612	2579	220372	7581
10	JPCLN011	36316	4287	210381	11160
11	JPCLN012	36177	8971	199766	17230
12	JPCLN013	35984	3330	216918	5912
13	JPCLN014	36299	4070	214886	6889
14	JPCLN015	33831	4420	214603	9290
15	JPCLN016	34403	4483	212250	11008
16	JPCLN017	33503	11557	202371	14713
17	JPCLN019	41410	5544	208253	6937
18	JPCLN020	41862	2915	212449	4918
19	JPCLN021	39179	9051	204330	9584
20	JPCLN022	36055	6024	211015	9050

4. Validasi Segmentasi Paru-Paru Kiri dengan 18 Landmark

No	Citra Percobaan	Validasi			
		TP	FP	TN	FN
1	JPCLN001	33572	3047	219286	6239
2	JPCLN002	24513	2884	229020	5727
3	JPCLN003	23629	2266	230121	6128
4	JPCLN005	22881	1759	231504	6000
5	JPCLN006	28508	6503	218947	8186
6	JPCLN007	32471	4994	215383	9296
7	JPCLN008	37311	7540	209115	8178
8	JPCLN009	28331	4931	222172	6710
9	JPCLN010	25452	868	230006	5818
10	JPCLN011	29609	6582	211649	14304
11	JPCLN012	29101	7049	211727	14267
12	JPCLN013	29927	719	223663	7835
13	JPCLN014	32887	3320	218434	7503
14	JPCLN015	30329	4080	221805	5930
15	JPCLN016	24517	4118	218137	15372
16	JPCLN017	23580	7352	216210	15002
17	JPCLN019	33898	1886	221602	4758
18	JPCLN020	34241	2421	219680	5802
19	JPCLN021	30970	4131	220698	6345
20	JPCLN022	33400	3247	219713	5784

5. Validasi Segmentasi Paru-Paru Kiri dengan 23 Landmark

No	Citra Percobaan	Validasi			
		TP	FP	TN	FN
1	JPCLN001	34198	2421	220196	5329
2	JPCLN002	24206	995	231240	5703
3	JPCLN003	22594	3301	226936	9313
4	JPCLN005	22708	1932	228498	9006
5	JPCLN006	27913	6875	218271	9085
6	JPCLN007	33147	4318	217580	7099
7	JPCLN008	36276	9234	206836	9798
8	JPCLN009	31138	2124	224795	4087
9	JPCLN010	24179	2620	226942	8403
10	JPCLN011	26152	10039	209761	16192
11	JPCLN012	26658	8189	208638	18659
12	JPCLN013	30138	508	224331	7167
13	JPCLN014	33694	2136	219756	6558
14	JPCLN015	30101	4308	221144	6591
15	JPCLN016	25067	2666	220176	14235
16	JPCLN017	26809	4123	219743	11469
17	JPCLN019	34119	1665	221486	4874
18	JPCLN020	33787	2112	220778	5467
19	JPCLN021	30518	4583	217252	9791
20	JPCLN022	33299	3437	219336	6072

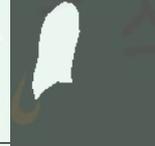
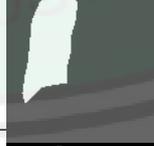
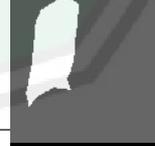
6. Validasi Segmentasi Paru-Paru Kiri dengan 28 Landmark

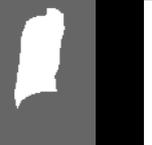
No	Citra Percobaan	Validasi			
		TP	FP	TN	FN
1	JPCLN001	220013	5512	219263	6262
2	JPCLN002	28070	3888	225409	4777
3	JPCLN003	24301	1594	233203	3046
4	JPCLN005	22969	1671	230255	7249
5	JPCLN006	28185	6603	220335	7021
6	JPCLN007	33570	3895	217657	7022
7	JPCLN008	36976	8534	206688	9946
8	JPCLN009	30593	2669	224652	4230
9	JPCLN010	24138	2661	227996	7349
10	JPCLN011	27878	8313	210471	15482
11	JPCLN012	26149	8698	208015	19282
12	JPCLN013	27888	2758	223073	8425
13	JPCLN014	28888	6942	215880	10434
14	JPCLN015	28350	6059	218432	9303
15	JPCLN016	21368	6365	217037	17374
16	JPCLN017	23224	7708	218504	12708
17	JPCLN019	33488	2296	221132	5228
18	JPCLN020	34246	2416	220324	5158
19	JPCLN021	28044	7057	217728	9315
20	JPCLN022	31027	5709	218019	7389

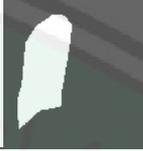
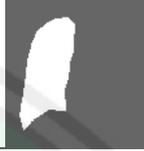
Lampiran 2

HASIL SEGMENTASI PARU-PARU

1. Hasil Segmentasi Paru-paru Kanan

No	Nama Citra	Citra Asli	Preprocessing	Landmark 15	Landmark 20	Landmark 25
1	JPCLN001					
2	JPCLN002					
3	JPCLN003					
4	JPCLN005					
5	JPCLN006					
6	JPCLN007					
7	JPCLN008					
8	JPCLN009					

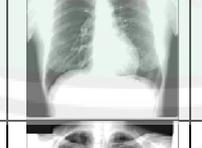
No	Nama Citra	Citra Asli	Preprocessing	Landmark 15	Landmark 20	Landmark 25
9	JPCLN010					
10	JPCLN011					
11	JPCLN012					
12	JPCLN013					
13	JPCLN014					
14	JPCLN015					
15	JPCLN016					
16	JPCLN017					
17	JPCLN019					
18	JPCLN020					

No	Nama Citra	Citra Asli	Preprocessing	Landmark 15	Landmark 20	Landmark 25
19	JPCLN02_1					
20	JPCLN02_2					



2. Hasil Segmentasi Paru-paru Kiri

No	Data Citra	Citra Asli	Preprocessing	Landmark 18	Landmark 23	Landmark 28
1	JPCLN001s					
2	JPCLN002					
3	JPCLN003					
4	JPCLN005					
5	JPCLN006					
6	JPCLN007					
7	JPCLN008					
8	JPCLN009					
9	JPCLN010					
10	JPCLN011					

No	Data Citra	Citra Asli	Preprocessing	Landmark 18	Landmark 23	Landmark 28
11	JPCLN01 2					
12	JPCLN01 3					
13	JPCLN01 4					
14	JPCLN01 5					
15	JPCLN01 6					
16	JPCLN01 7					
17	JPCLN01 9					
18	JPCLN02 0					
19	JPCLN02 1					
20	JPCLN02 2					

Lampiran 3

DAFTAR DIAMETER MAKSIMAL PARU-PARU

1. 33 Landmark

No.	Citra Percobaan	Diameter maksimal
1	JPCLN001	397
2	JPCLN002	343
3	JPCLN003	341
4	JPCLN005	344
5	JPCLN006	369
6	JPCLN007	405
7	JPCLN008	398
8	JPCLN009	366
9	JPCLN010	367
10	JPCLN011	402
11	JPCLN012	401
12	JPCLN013	397
13	JPCLN014	400
14	JPCLN015	383
15	JPCLN016	397
16	JPCLN017	387
17	JPCLN019	407
18	JPCLN020	373
19	JPCLN021	373
20	JPCLN022	399

2. 43 Landmark

No.	Citra Percobaan	Diameter maksimal
1	JPCLN001	397
2	JPCLN002	343
3	JPCLN003	345
4	JPCLN005	368
5	JPCLN006	367
6	JPCLN007	396
7	JPCLN008	398
8	JPCLN009	384
9	JPCLN010	366
10	JPCLN011	368
11	JPCLN012	393
12	JPCLN013	400
13	JPCLN014	402
14	JPCLN015	388
15	JPCLN016	397
16	JPCLN017	388
17	JPCLN019	403
18	JPCLN020	396
19	JPCLN021	397
20	JPCLN022	400

3. 53 Landmark

No.	Citra Percobaan	Diameter maksimal
1	JPCLN001	389
2	JPCLN002	359
3	JPCLN003	340
4	JPCLN005	360
5	JPCLN006	359
6	JPCLN007	392
7	JPCLN008	409
8	JPCLN009	379
9	JPCLN010	355
10	JPCLN011	383
11	JPCLN012	390
12	JPCLN013	374
13	JPCLN014	374
14	JPCLN015	379
15	JPCLN016	374
16	JPCLN017	360
17	JPCLN019	390
18	JPCLN020	397
19	JPCLN021	369
20	JPCLN022	379