



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502

SEGMENTASI CORTICAL BONE PADA CITRA DENTAL PANORAMIC RADIOGRAPH MENGUNAKAN DYNAMIC MULTI-SCALE LINE DETECTION

MOHAMMAD ANWAR HIDAYAT
0511144000137

Dosen Pembimbing
Dr. Agus Zainal Arifin, S.Kom., M.Kom.
Dini Adni Navastara, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - KI141502

SEGMENTASI CORTICAL BONE PADA CITRA DENTAL PANORAMIC RADIOGRAPH MENGUNAKAN DYNAMIC MULTI-SCALE LINE DETECTION

MOHAMMAD ANWAR HIDAYAT
0511144000137

Dosen Pembimbing I
Dr. Agus Zainal Arifin, S.Kom., M.Kom.

Dosen Pembimbing II
Dini Adni Navastara, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

[Halaman ini sengaja dikosongkan]



UNDERGRADUATE THESIS - KI141502

**CORTICAL BONE SEGMENTATION ON DENTAL
PANORAMIC RADIOGRAPH USING DYNAMIC
MULTI-SCALE LINE DETECTION**

MOHAMMAD ANWAR HIDAYAT
05111440000137

Supervisor I
Dr. Agus Zainal Arifin, S.Kom., M.Kom.

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

DEPARTMENT OF INFORMATICS
Faculty of Information and Communication Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN
SEGMENTASI CORTICAL BONE PADA CITRA DENTAL
PANORAMIC RADIOGRAPH MENGGUNAKAN
DYNAMIC MULTI-SCALE LINE DETECTION

TUGAS AKHIR

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

Oleh:

MOHAMMAD ANWAR HIDAYAT
NRP: 05111440000137

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dr. Agus Zainal Arifin, S.Kom., M.Kom.
NIP. 197208091995121001

.....
(Pembimbing 1)

Dini Adni Navastara, S.Kom., M.Sc.
NIP. 198510172015042001

.....
(Pembimbing 2)

SURABAYA
MEI, 2018

[Halaman ini sengaja dikosongkan]

SEGMENTASI CORTICAL BONE PADA CITRA DENTAL PANORAMIC RADIOGRAPH MENGGUNAKAN DYNAMIC MULTI-SCALE LINE DETECTION

Nama Mahasiswa : Mohammad Anwar Hidayat
NRP : 05111440000137
Jurusan : Departemen Informatika, FTIK – ITS
Dosen Pembimbing 1 : Dr. Agus Zainal Arifin, S.Kom., M.Kom.
Dosen Pembimbing 2 : Dini Adni Navastara, S.Kom., M.Sc.

ABSTRAK

Dental panoramic radiograph adalah citra x-ray 2D dari rahang mulut seseorang. Selain informasi kondisi dari gigi, citra dental panoramic radiograph dapat digunakan sebagai diagnosis awal terhadap kemungkinan penyakit osteoporosis melalui ketebalan cortical bone. Dengan bantuan komputer, penghitungan ketebalan cortical bone dapat dilakukan dengan otomatis. Untuk menghitung ketebalan cortical bone, komputer harus memilih dan membedakan area cortical bone dari background. Dikarenakan komputer tidak dapat melakukan hal tersebut secara otomatis, dibutuhkan teknik segmentasi untuk memisahkan kedua area ini.

Beberapa teknik segmentasi telah dikembangkan. Salah satu teknik yang digunakan untuk segmentasi cortical bone adalah dengan menghitung fitur line strength menggunakan teknik line detection. Seperti teknik segmentasi lainnya, pengembangan teknik ini biasanya optimal untuk ukuran citra tertentu saja. Padahal, citra pada penerapan medis dapat memiliki ukuran yang berbeda. Oleh karena itu, dibutuhkan teknik segmentasi yang dapat secara dinamis menyesuaikan terhadap kondisi tersebut.

Tugas akhir ini mengusulkan metode segmentasi cortical bone menggunakan dynamic multi-scale line detection. Secara umum, sistem dibagi menjadi 5 (lima) tahap. Tahap pertama yaitu pemilihan ROI. Tahap kedua yaitu perhitungan parameter. Tahap ketiga yaitu perhitungan line strength. Tahap keempat yaitu segmentasi. Dan tahap kelima yaitu perbaikan citra.

Uji coba pada tugas akhir ini dilakukan pada 30 citra gray-scale dari dental panoramic radiograph. Hasil uji coba menunjukkan bahwa sistem mampu melakukan segmentasi pada citra dengan ukuran berbeda dengan nilai standard deviasi F_1 score di bawah 5% atau 0,05. Hasil uji coba juga menunjukkan nilai rata-rata F_1 score adalah 0,890.

Kata kunci: Cortical Bone, Dental Panoramic Radiograph, Dynamic, Line Detection, Multi-scale

CORTICAL BONE SEGMENTATION ON DENTAL PANORAMIC RADIOGRAPH USING DYNAMIC MULTI-SCALE LINE DETECTION

Student Name : Mohammad Anwar Hidayat
Registration Number : 05111440000137
Department : Informatics, Faculty o ICT – ITS
First Supervisor : Dr. Agus Zainal Arifin, S.Kom.,
M.Kom.
Second Supervisor : Dini Adni Navastara, S.Kom.,
M.Sc.

ABSTRACT

Dental panoramic radiograph is 2D x-ray image that captures entire structure of mouth. Beside information about the condition of teeth, dental panoramic radiograph image can be used for early diagnosis of the possibility of osteoporosis through cortical bone width. With help of computer, calculation of cortical bone width can be done automatically. To perform the calculation, computer has to distinct cortical bone from background. Because computer cannot do it automatically as it is, it requires segmentation technique to distinct both areas.

Several segmentation techniques have been proposed. One that perform segmentation for cortical bone is through calculating line strength features using line detection technique. Like any other technique, the development of this technique is usually optimal on images with fixed size. Whereas actual medical image can vary in size. Hence the requirement of segmentation technique that can dynamically adapt to this condition.

This research proposes segmentation technique of cortical bone using dynamic multi-scale line detection. In general, the system is divided into 5 (five) stages. First stage is choosing Region

of Interest (ROI). Second stage is calculating parameter. Third stage is calculating line strength. Fourth stage is segmentation. And fifth stage is image refinement.

Experiments in this research are done on 30 grayscale images of dental panoramic radiograph. The results of experiment show that the system is capable of segmenting images with different sizes with standard deviation of F_1 score with value under 5% or 0.05. It also shows the average F_1 score is 0.890.

Keywords: Cortical Bone, Dental Panoramic Radiograph, Dynamic, Line Detection, Multi-scale

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah melimpahkan rahmat dan kasih sayang-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “**Segmentasi Cortical Bone pada Citra Dental Panoramic Radiograph Menggunakan Dynamic Multi-Scale Line Detection**”.

Buku ini disusun dengan harapan dapat memberikan manfaat dalam penelitian *dental panoramic radiograph* lebih lanjut. Selain itu, penulis berharap dapat memberikan kontribusi positif bagi Departemen Informatika ITS.

Proses perancangan, pengerjaan, dan penyusunan tugas akhir mendapat dapat masukan dari berbagai pihak. Oleh karena itu, dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Ibu Riskiyah dan Bapak Supriyadi, selaku orang tua dari penulis yang telah memberikan doa dan dukungan penuh kepada penulis.
2. Bapak Dr. Agus Zainal Arifin, S.Kom., M.Kom. dan Ibu Dini Adni Navastara, S.Kom., M.Sc. selaku dosen pembimbing penulis yang telah memberikan ide, nasihat, dan arahan sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik dan tepat waktu.
3. Abdul Rofik, Achmad Rifki Febri Andi, Achmad Sofyan Ansori selaku saudara penulis yang telah memberikan dukungan, motivasi, dan inspirasi baik secara langsung maupun tidak langsung.
4. Diana Suaidah yang unik dan selalu sabar menemani serta menasihati penulis di saat senang mau pun susah.
5. Teman-teman administrator Lab. KCV angkatan 2014 yang aneh, angkatan 2015 yang ceria, dan angkatan 2016 yang pendiam yang telah bersedia memfasilitasi, menemani, memotivasi, dan mewarnai hari-hari penulis.

6. Teman-teman satu bimbingan dengan Pak Agus dan Bu Dini, yaitu Irfan dan Adam yang secara langsung dan tidak langsung memotivasi penulis untuk terus bersemangat menyelesaikan tugas akhir ini.
7. Teman-teman penulis seperti Syauki yang dengan senang hati mau berbagi hobi dan Dzaky yang selalu memberikan informasi yang tidak dapat penulis perkirakan sebelumnya.
8. Teman-teman satu perjuangan tugas akhir di Lab. KCV yang tanpa henti memberikan hujan pertanyaan, baik itu akademik mau pun non akademik, dan yang tanpa henti mewarna musik yang penulis dengarkan dengan cerita-cerita yang ceria.
9. Teman-teman TC14 yang sering kali membuat penulis tersenyum dengan humornya.
10. Serta pihak lain yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa masih terdapat kesalahan dan kekurangan dalam buku tugas akhir ini. Penulis memohon maaf atas kesalahan, kelalaian, maupun kekurangan dalam penyusunan tugas akhir ini. Kritik dan saran yang membangun sangat dibutuhkan dan dapat disampaikan untuk pengembangan selanjutnya.

Surabaya, Mei 2018

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xix
DAFTAR GRAFIK.....	xxi
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Metodologi.....	3
1.7 Sistematika Laporan	5
BAB II Dasar Teori	7
2.1 <i>Dental Panoramic Radiograph</i>	7
2.1.1 Teori <i>Dental Panoramic Radiograph</i>	7
2.1.2 Prosedur Pengambilan <i>Dental Panoramic Radiograph</i>	8
2.2 Struktur Penyusun <i>Dental Panoramic Radiograph</i>	10
2.2.1 Tulang.....	10
2.2.1.1 <i>Cortical Bone</i>	11
2.2.1.2 <i>Trabecular Bone</i>	11
2.2.2 Penyakit Tulang.....	12
2.2.2.1 <i>Osteogenesis Imperfecta</i>	12
2.2.2.2 <i>Osteonecrosis</i>	12
2.2.2.3 <i>Osteoporosis</i>	13
2.3 <i>Region Of Interest (ROI)</i>	13
2.3.1 Pengertian <i>Region Of Interest (ROI)</i>	13
2.3.1.1 Eksplorasi Data.....	13
2.3.1.2 Meminimalkan Kesalahan.....	14
2.3.1.3 Membatasi <i>Testing</i>	14

2.3.2	<i>Region of Interest (ROI) yang Digunakan</i>	14
2.4	<i>Cubic Spline Interpolation</i>	15
2.5	<i>Bicubic Interpolation</i>	16
2.6	Sudut Orientasi	18
2.7	Garis Orientasi	18
2.8	<i>Euclidean Distance</i>	19
2.9	Ketebalan <i>Cortical Bone</i>	19
2.10	Normalisasi.....	20
2.11	<i>Line Strength</i>	20
2.12	<i>Multi-Scale Line Detection</i>	22
2.13	<i>Otsu's Thresholding</i>	23
2.14	Luas Citra Biner	25
2.15	F_{β} score.....	26
BAB III Analisis dan perancangan.....		29
3.1	Tahap Analisis	29
3.1.1	Spesifikasi Kebutuhan Sistem	29
3.1.2	Analisis Permasalahan.....	30
3.2	Tahap Perancangan	30
3.2.1	Perancangan Sistem.....	30
3.2.2	Perancangan Data	33
3.2.3	<i>Marking</i>	34
3.2.3.1	Menghapus Area.....	35
3.2.3.2	Penandaan Titik.....	36
3.2.3.3	Pemilihan ROI.....	37
3.2.4	<i>Pre-processing</i>	37
3.2.4.1	<i>Dental Panoramic Radiograph</i>	38
3.2.4.2	Lokasi ROI	39
3.2.4.3	Sudut Orientasi	39
3.2.4.4	Ketebalan <i>Cortical Bone</i>	40
3.2.5	<i>Line Strength</i>	40
3.2.5.1	Pencarian Sudut.....	41
3.2.5.2	Pembuatan <i>Moving Window</i>	42
3.2.5.3	Perhitungan <i>Line Strength</i>	43
3.2.5.4	Kombinasi <i>Line Strength</i>	44
3.2.6	<i>Segmentation</i>	45

3.2.7	<i>Post-processing</i>	46
BAB IV IMPLEMENTASI		47
4.1	Lingkungan Implementasi	47
4.1.1	Perangkat Keras	47
4.1.2	Perangkat Lunak	47
4.2	Implementasi Pembacaan Data	47
4.3	Implementasi <i>Marking</i>	49
4.3.1	Menghapus Area	49
4.3.2	Penandaan Titik	50
4.3.3	Pemilihan ROI	52
4.4	<i>Pre-processing</i>	54
4.4.1	Sudut Orientasi	54
4.4.2	Ketebalan <i>Cortical Bone</i>	56
4.5	<i>Line Strength</i>	56
4.5.1	Pencarian Sudut	57
4.5.2	Pembuatan <i>Moving Window</i>	59
4.5.3	Perhitungan <i>Line Strength</i>	61
4.5.4	Kombinasi <i>Line Strength</i>	63
4.6	<i>Segmentation</i>	63
4.7	Post-processing	64
BAB V UJI COBA DAN EVALUASI		67
5.1	Lingkungan Uji Coba	67
5.2	Data Uji Coba	67
5.3	Uji Coba	69
5.4	Uji Coba Sistem	70
5.4.1	<i>Marking</i>	70
5.4.1.1	Menghapus Area	70
5.4.2	<i>Pre-processing</i>	74
5.4.3	<i>Line Strength</i>	75
5.4.4	<i>Segmentation</i>	76
5.4.5	<i>Post-processing</i>	76
5.5	Uji Coba 1: Orientasi Garis	77
5.5.1	Metode Orientasi Nguyen	78
5.5.1.1	Hasil F_1 Score Terendah ROI Kiri	78
5.5.1.2	Hasil F_1 Score Terendah ROI Kanan	79

5.5.1.3	Hasil F_1 Score Tertinggi ROI Kiri.....	79
5.5.1.4	Hasil F_1 Score Tertinggi ROI Kanan.....	80
5.5.2	Metode Orientasi Usulan.....	81
5.5.2.1	Hasil F_1 Score Terendah ROI Kiri	81
5.5.2.2	Hasil F_1 Score Terendah ROI Kanan	83
5.5.2.3	Hasil F_1 Score Tertinggi ROI Kiri.....	83
5.5.2.4	Hasil F_1 Score Tertinggi ROI Kanan.....	84
5.5.3	Hasil Rata-rata F_1 score Metode Nguyen dan Usulan.	84
5.6	Uji Coba 2: Teknik Kombinasi.....	86
5.6.1	Penambahan Nilai <i>Grayscale</i> Metode Nguyen	87
5.6.2	Tanpa Penambahan Nilai <i>Grayscale</i> Metode Usulan..	90
5.7	Uji Coba 3: Teknik Normalisasi	96
5.7.1	Metode <i>Min-max</i> dengan F_1 score Terendah	97
5.7.2	Metode Usulan dengan F_1 score Terendah.....	98
5.7.3	Metode <i>Min-max</i> dan Usulan dengan F_1 score Tertinggi	100
5.8	Uji Coba 4: Lokasi dan Ukuran ROI	104
5.8.1	Segmentasi dengan F_1 score Terendah ROI Kiri	106
5.8.2	Segmentasi dengan F_1 score Terendah ROI Kanan ..	106
5.8.3	Segmentasi dengan Rata-rata F_1 score Terendah	107
5.8.4	Segmentasi dengan F_1 score Tertinggi ROI Kiri.....	108
5.8.5	Segmentasi dengan F_1 score Tertinggi ROI Kanan...	109
5.8.6	Segmentasi dengan Rata-rata F_1 score Tertinggi	110
5.8.7	Hasil Rata-rata F_1 score.....	111
5.9	Uji Coba 5: Perbandingan Efektivitas.....	112
5.9.1	Segmentasi dengan Nilai F_1 score Terendah pada Sistem Ramadhani	112
5.9.2	Segmentasi dengan Nilai F_1 score Tertinggi pada Sistem Ramadhani	115
5.9.3	Segmentasi dengan Nilai F_1 score Terendah pada Sistem Usulan.....	118
5.9.4	Segmentasi dengan Nilai F_1 score Tertinggi pada Sistem Usulan.....	121
5.9.5	Nilai Rata-rata F_1 score Hasil Segmentasi Sistem Ramadhani dan Sistem Usulan	124

5.10 Uji Coba 6: Perbandingan Efisiensi	127
BAB VI KESIMPULAN DAN SARAN	131
6.1 Kesimpulan	131
6.2 Saran	132
LAMPIRAN	133
DAFTAR PUSTAKA.....	141
BIODATA PENULIS.....	145

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1 Dental Panoramic Radiograph.....	7
Gambar 2.2 Posisi Pasien [10]	8
Gambar 2.3 Contoh CBCT [13]	9
Gambar 2.4 Posisi Pasien dari Atas [15].....	10
Gambar 2.5 Area Cortical Bone dan Trabecular Bone.....	11
Gambar 2.6 Contoh Are ROI	14
Gambar 2.7 Perbandingan Teknik Interpolasi [21]	15
Gambar 2.8 Perbandingan Performa Cubic Spline dan Newton Interpolation [21]	16
Gambar 2.9 Contoh Intensitas Grayscale	17
Gambar 2.10 Nearest Neighbor dan Linear Interpolation.....	17
Gambar 2.11 Contoh Pemetaan Cubic Spline	18
Gambar 2.12 Orientasi Line Operator [6]	21
Gambar 2.13 Penampakan Line Operator Berimpit dengan Pembuluh Darah [6].....	21
Gambar 2.14 Moving Window W dengan Level L [7]	23
Gambar 2.15 Ilustrasi Nilai Kebenaran pada Citra	26
Gambar 2.16 Confusion Matrix.....	27
Gambar 3.1 Ilustrasi Pengambilan Foto Digital	31
Gambar 3.2 Ilustrasi Mengubah Citra Fisik Menjadi Digital.....	31
Gambar 3.3 Diagram Alir Sistem.....	32
Gambar 3.4 Contoh Ground Truth	33
Gambar 3.5 Diagram Alir Proses Marking.....	35
Gambar 3.6 Diagram Alir Menghapus Area	36
Gambar 3.7 Diagram Alir Penanda Titik	37
Gambar 3.8 a) Titik Pilihan dan Interpolasi; b) Pemilihan Titik Batas ROI.....	38
Gambar 3.9 Penarikan Garis Titik Batas.....	39
Gambar 3.10 Perhitungan Batas Sudut Orientasi	40
Gambar 3.11 Ilustrasi Orientasi Garis pada Moving Window	41
Gambar 3.12 Keluaran Moving Window	43
Gambar 5.1 Dental Panoramic Radiograph.....	68
Gambar 5.2 ROI Terpilih (a) ROI Kiri (b) ROI Kanan.....	68
Gambar 5.3 Lokasi ROI pada Citra.....	68

Gambar 5.4 Ground Truth	69
Gambar 5.5 ROI Ground Truth (a) ROI Kiri (b) ROI Kanan.....	69
Gambar 5.6 Penandaan untuk Menghapus Area	71
Gambar 5.7 Hasil Penghapusan Area.....	71
Gambar 5.8 Hasil Penandaan Cortical Bone	72
Gambar 5.9 Hasil Penandaan Titik Area Cortical Bone.....	73
Gambar 5.10 Pemilihan ROI dari Area Batasan	73
Gambar 5.11 Penampakan Hasil Perhitungan Parameter	74
Gambar 5.12 Loading Bar Perhitungan Line Strength	75
Gambar 5.13 Window dari Nilai Line Strength	75
Gambar 5.14 Hasil Segmentasi dengan Otsu Thresholding	76
Gambar 5.15 Window Konfirmasi Post-processing	77
Gambar 5.16 Citra Hasil Post-processing	77
Gambar 5.17 Pemilihan Lokasi Empat ROI.....	104

DAFTAR GRAFIK

Grafik 5.1 Rata-rata F_1 score ROI Kiri (Uji Coba 1).....	85
Grafik 5.2 Rata-rata F_1 score ROI Kanan (Uji Coba 1).....	86
Grafik 5.3 Rata-rata F_1 score ROI Kiri (Uji Coba 2).....	95
Grafik 5.4 Rata-rata F_1 score ROI Kanan (Uji Coba 2).....	96
Grafik 5.5 Rata-rata F_1 score ROI Kiri (Uji Coba 3).....	102
Grafik 5.6 Rata-rata F_1 score ROI Kanan (Uji Coba 3).....	103
Grafik 5.7 Perbandingan F_1 score Sistem pada ROI Kiri	125
Grafik 5.8 Perbandingan F_1 score Sistem pada ROI Kanan	126
Grafik 5.9 Durasi Eksekusi pada ROI Kiri.....	129
Grafik 5.10 Durasi Eksekusi pada ROI Kanan.....	129

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 5.1 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Nguyen (Kiri) Skala 1,00.....	78
Tabel 5.2 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Nguyen (Kanan) Skala 1,00.....	79
Tabel 5.3 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Nguyen (Kiri) Skala 1,00.....	80
Tabel 5.4 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Nguyen (Kanan) Skala 1,00.....	81
Tabel 5.5 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Usulan (Kiri) Skala 1,00.....	82
Tabel 5.6 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Usulan (Kanan) Skala 1,00.....	82
Tabel 5.7 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Usulan (Kiri) Skala 1,00.....	83
Tabel 5.8 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Usulan (Kanan) Skala 1,00.....	84
Tabel 5.9 Rata-rata F_1 score Metode Nguyen dan Usulan.....	85
Tabel 5.10 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kiri Menggunakan Metode Nguyen (Skala 1,00).....	87
Tabel 5.11 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kiri Menggunakan Metode Nguyen (Skala 1,00).....	88
Tabel 5.12 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Nguyen pada ROI Kiri (Skala 1,00).....	88
Tabel 5.13 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kanan Menggunakan Metode Nguyen (Skala 1,00).....	89
Tabel 5.14 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kanan Menggunakan Metode Nguyen (Skala 1,00).....	89
Tabel 5.15 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Nguyen pada ROI Kanan (Skala 1,00).....	90
Tabel 5.16 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kiri Menggunakan Metode Usulan (Skala 1,00).....	91
Tabel 5.17 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kiri Menggunakan Metode Usulan (Skala 1,00).....	91

Tabel 5.18 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Usulan pada ROI Kiri (Skala 1,00).....	92
Tabel 5.19 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kanan Menggunakan Metode Usulan (Skala 1,00)	93
Tabel 5.20 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kanan Menggunakan Metode Usulan (Skala 1,00)	93
Tabel 5.21 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Usulan pada ROI Kanan (Skala 1,00).....	94
Tabel 5.22 Rata-rata F_1 score Seluruh Skala	95
Tabel 5.23 Rata-rata Nilai F_1 score Terendah Berdasarkan Normalisasi Min-max Skala 1,00	97
Tabel 5.24 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Min-max pada ROI Kiri Skala 1,00	97
Tabel 5.25 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Min-max pada ROI Kanan Skala 1,00	98
Tabel 5.26 Rata-rata Nilai F_1 score Terendah Berdasarkan Normalisasi Usulan Skala 1,00	98
Tabel 5.27 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Usulan pada ROI Kiri Skala 1,00	99
Tabel 5.28 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Usulan pada ROI Kanan Skala 1,00	99
Tabel 5.29 Rata-rata Nilai F_1 score Tertinggi Kedua Metode Normalisasi Skala 1,00	100
Tabel 5.30 Hasil Segmentasi dengan F_1 score Tertinggi Kedua Metode Normalisasi pada ROI Kiri Skala 1,00	101
Tabel 5.31 Hasil Segmentasi dengan F_1 score Tertinggi Ketiga Metode Normalisasi pada ROI Kanan Skala 1,00	101
Tabel 5.32 Rata-rata Nilai F_1 score Seluruh Data ROI Kiri	102
Tabel 5.33 Rata-rata F_1 score Seluruh Data ROI Kanan	103
Tabel 5.34 Hasil Segmentasi dengan Nilai F_1 score Terendah pada ROI Kiri (Skala 1,00).....	105

Tabel 5.35 Hasil Segmentasi dengan Nilai F_1 score Terendah pada ROI Kanan (Skala 1,00).....	106
Tabel 5.36 Hasil Segmentasi dengan Nilai Rata-rata F_1 score Terendah (Skala 1,00).....	107
Tabel 5.37 Hasil Segmentasi dengan Nilai F_1 score Tertinggi pada ROI Kiri (Skala 1,00).....	108
Tabel 5.38 Hasil Segmentasi dengan Nilai F_1 score Tertinggi pada ROI Kanan (Skala 1,00).....	109
Tabel 5.39 Hasil Segmentasi dengan Nilai Rata-rata F_1 score Tertinggi (Skala 1,00).....	110
Tabel 5.40 Rata-rata F_1 score Seluruh Citra.....	111
Tabel 5.41 Standard Deviasi F_1 score dari Seluruh Citra.....	112
Tabel 5.42 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Ramadhani pada ROI Kiri.....	113
Tabel 5.43 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Ramadhani pada ROI Kanan.....	114
Tabel 5.44 Nilai F_1 score Terendah dari Sistem Ramadhani.....	115
Tabel 5.45 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Ramadhani pada ROI Kiri.....	116
Tabel 5.46 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Ramadhani pada ROI Kanan.....	117
Tabel 5.47 Nilai F_1 score Tertinggi dari Sistem Ramadhani.....	118
Tabel 5.48 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Usulan pada ROI Kiri.....	119
Tabel 5.49 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Usulan pada ROI Kanan.....	120
Tabel 5.50 Nilai F_1 score Terendah dari Sistem Usulan.....	121
Tabel 5.51 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Usulan pada ROI Kiri.....	122
Tabel 5.52 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Usulan pada ROI Kanan.....	123
Tabel 5.53 Nilai F_1 score Tertinggi dari Sistem Usulan.....	124
Tabel 5.54 Perbandingan Rata-rata F_1 score Hasil Segmentasi Menggunakan Sistem Ramadhani dan Sistem Usulan.....	125

Tabel 5.55 Perbandingan Standard Deviasi F_1 score Sistem Ramadhani dan Sistem Usulan	127
Tabel 5.56 Waktu Eksekusi Sistem Ramadhani dan Sistem Usulan	128

DAFTAR KODE SUMBER

Kode Sumber 4.1 Pembaca Nama Citra	48
Kode Sumber 4.2 Pembacaan Data Citra	48
Kode Sumber 4.3 Pemilihan Area Hapus	49
Kode Sumber 4.4 Penghapusan Area yang Ditandai.....	50
Kode Sumber 4.5 Implementasi Marking Bagian Atas	51
Kode Sumber 4.6 Implementasi Marking Bagian Bawah	52
Kode Sumber 4.7 Pemilihan Batasan Area ROI	53
Kode Sumber 4.8 Pengambilan Lokasi ROI	53
Kode Sumber 4.9 Penentuan Titik-titik Batas	54
Kode Sumber 4.10 Perhitungan Sudut Batas Orientasi.....	55
Kode Sumber 4.11 Pemilihan Sudut θ_1 dan Sudut θ_2	56
Kode Sumber 4.12 Implementasi Fungsi Line Strength	57
Kode Sumber 4.13 Perhitungan Ketebalan Cortical Bone	57
Kode Sumber 4.14 Perhitungan Sudut Orientasi Garis	58
Kode Sumber 4.15 Implementasi Batasan Perulangan Orientasi	59
Kode Sumber 4.16 Inisialisasi Moving Window	59
Kode Sumber 4.17 Pembuatan Moving Window.....	60
Kode Sumber 4.18 Inisialisasi Data dan Pemlihan Pikel.....	61
Kode Sumber 4.19 Perhitungan Line Strength.....	62
Kode Sumber 4.20 Implementasi Kombinasi.....	63
Kode Sumber 4.21 Pemanggilan Fungsi Line Strength	63
Kode Sumber 4.22 Implementasi Segmentation	64
Kode Sumber 4.23 Implementasi Pilihan Pengguna	64
Kode Sumber 4.24 Implementasi Jika Area Maksimum.....	64
Kode Sumber 4.25 Implementasi Jika Pemilihan Manual	65

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Bab ini, dibahas beberapa hal yang menjadi dasar dalam pengerjaan tugas akhir ini. Bahasan tersebut meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi, dan sistematika laporan.

1.1 Latar Belakang

Tulang merupakan jaringan pada tubuh yang terus tumbuh. Tulang terdiri dari kolagen dan kalsium. Tulang dibedakan menjadi dua jenis, yaitu *cortical bone* dan *trabecular bone* [1]. Tulang juga dapat mengalami kerusakan akibat osteoporosis [2]. Osteoporosis merupakan penyakit tulang yang tidak menampilkan gejala kecuali dilakukan pemeriksaan atau sampai patah tulang terjadi [2]. Osteoporosis dapat diketahui dengan melihat kepadatan tulang seseorang.

Alat yang biasa digunakan untuk mengukur kepadatan tulang seseorang adalah DXA (*dual energy x-ray absorptiometry*). Namun, penelitian [3] mengungkapkan bahwa penggunaan DXA membutuhkan biaya yang mahal dan tidak semua rumah sakit tersedia. Oleh karena itu, sebagai alternatif digunakan *dental panoramic radiograph*, seperti diusulkan oleh penelitian [4].

Penelitian [4] mengusulkan penggunaan ketebalan *cortical bone* pada *dental panoramic radiograph* untuk mendeteksi kemungkinan seseorang mengidap osteoporosis. Pengukuran ketebalan *cortical bone* dilakukan secara otomatis oleh komputer. Namun, komputer tidak dapat membedakan area *cortical bone* dari area *background*. Untuk mendapatkan area *cortical bone*, diperlukan teknik segmentasi citra yang dapat memisahkan area *cortical bone* dari area *background*.

Penelitian [5] memaparkan beberapa teknik segmentasi yang kemudian dibagi menjadi tiga kelas, yaitu teknik segmentasi

berdasarkan struktur, teknik segmentasi berdasarkan nilai piksel, dan teknik segmentasi yang menggunakan keduanya. Penelitian [6] mengusulkan teknik segmentasi berdasarkan struktur menggunakan *line operator*. *Line operator* digunakan dengan anggapan bahwa objek yang diinginkan memiliki struktur garis.

Berdasarkan hasil penelitian [6], penelitian [7] mengusulkan teknik *multi-scale line detection*. *Multi-scale line detection* merupakan teknik yang digunakan untuk mengetahui apakah suatu intensitas piksel adalah intensitas piksel garis atau bukan, ditunjukkan dengan nilai *line strength*. Penelitian yang dilakukan oleh [7] menerapkan teknik *multi-scale line detection* untuk mendapatkan pembuluh darah pada citra dengan anggapan bahwa piksel pembuluh darah memiliki struktur garis. Dengan anggapan serupa, bahwa *cortical bone* memiliki struktur garis, penelitian [8] mengembangkan teknik segmentasi memanfaatkan *line strength operator* didasarkan pada penelitian [6].

Penelitian [8] dikembangkan untuk citra dengan ukuran tertentu. Padahal, citra pada penerapan medis dapat memiliki ukuran yang berbeda. Hal ini menyebabkan hasil yang diberikan oleh penelitian [8] tidak maksimal. Oleh karena itu, diperlukan metode yang dapat menyesuaikan dengan citra dengan kondisi ukuran berbeda.

Dengan asumsi bahwa objek *cortical bone* memiliki struktur garis, tugas akhir ini mengusulkan implementasi teknik *multi-scale line detection* yang secara dinamis dapat menyesuaikan dengan citra dengan ukuran berbeda untuk melakukan segmentasi *cortical bone* pada citra *dental panoramic radiograph*.

1.2 Rumusan Masalah

Berikut adalah rumusan masalah yang diangkat pada tugas akhir ini:

1. Bagaimana memilih *Region of Interest* (ROI) yang menyesuaikan ukuran citra?

2. Bagaimana menghitung nilai parameter yang optimal untuk perhitungan *line strength*?
3. Bagaimana menghitung *line strength* dengan menggunakan *dynamic multi-scale line detection*?
4. Bagaimana melakukan segmentasi *cortical bone* berdasarkan nilai *line strength* yang telah dihitung?

1.3 Batasan Masalah

Berikut adalah batasan masalah pada tugas akhir ini:

1. Citra masukan adalah citra 2D (dua dimensi) *grayscale* dari *dental panoramic radiograph*.
2. *Region of Interest* (ROI) dipilih secara manual.
3. Setiap ROI mengandung daerah *background* di bagian bawah dan di bagian atas daerah *cortical bone*.

1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah melakukan segmentasi *cortical bone* pada citra *dental panoramic radiograph* menggunakan *dynamic multi-scale line detection*.

1.5 Manfaat

Berikut adalah manfaat yang bisa diambil dari tugas akhir ini:

1. Membantu ahli medis untuk mendapatkan objek *cortical bone*.
2. Menyediakan metode yang dapat dikembangkan untuk perhitungan ketebalan *cortical bone*.

1.6 Metodologi

Berikut adalah tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini:

1. Studi literatur

Dilakukan studi terhadap beberapa topik penelitian yang sudah ada. Studi yang dilakukan berkaitan dengan pemahaman

dari algoritma dan sistem yang telah ada. Selain itu, dilakukan analisis untuk menemukan kemungkinan optimasi pada penelitian terkait. Berdasarkan studi tersebut, dipilih dua penelitian sebagai bahasan dasar untuk tugas akhir ini, yaitu penelitian [7] dan penelitian [8]. Masih dalam tahap ini, dilakukan pencarian beberapa referensi tambahan seperti *dental panoramic radiograph, line operator* [6], *Otsu thresholding* [9], dan beberapa referensi lainnya.

2. Analisis dan Desain Sistem

Analisis dilakukan dua tahap. Tahap pertama yaitu analisis terhadap penelitian acuan. Analisis ini dilakukan untuk mencari kemungkinan pengembangan selanjutnya. Dari hasil analisis, diketahui bahwa sistem yang dikembangkan oleh [8] tidak optimal menangani data dengan kondisi ukuran yang berbeda. Selain itu, algoritma *multi-scale line detection* yang dikembangkan oleh [7] sangat mungkin untuk diterapkan pada data citra *dental panoramic radiograph*. Berdasarkan hasil analisis tahap pertama, dilakukan analisis tahap kedua. Analisis tahap kedua berfokus pada perancangan sistem dinamis yang akan dikembangkan. Rancangan hasil analisis tahap kedua yaitu,

- pengguna melakukan pemilihan area yang dianggap tidak penting,
- pengguna memilih ROI di sekitar *cortical bone* dengan lokasi dan ukuran yang diinginkan,
- hasil dari ROI digunakan untuk melakukan proses kalkulasi parameter yang akan digunakan pada tahap kalkulasi *line strength* menggunakan teknik *multi-scale line detection*,
- dari *line strength* yang dihasilkan, dilakukan *thresholding* menggunakan *Otsu thresholding*,
- untuk memperbaiki citra, dilakukan pemilihan objek *cortical bone*,
- dihitung performa.

Hasil dari rancangan ini kemudian diimplementasikan pada bagian implementasi sistem.

3. Implementasi Sistem

Implementasi sistem dilakukan menggunakan MATLAB R2017a. Implementasi awal dilakukan pada fungsionalitas utama dari sistem, yaitu *multi-scale line detection*. Setelah itu, dilakukan implementasi fungsi tambahan seperti menu, konfirmasi pengguna, dan lain sebagainya (lihat bagian 4.1.2)

4. Uji Coba dan Evaluasi

Uji coba yang dilakukan pada tugas akhir ini dibagi menjadi dua, yaitu uji coba sistem dan uji coba metode. Uji coba sistem adalah untuk melihat performa sistem usulan dalam menjalankan satu siklus segmentasi. Uji coba metode adalah untuk mengetahui performa metode yang digunakan. Uji metode dibagi menjadi enam uji coba yaitu,

- **Uji Coba 1:** Orientasi garis
- **Uji Coba 2:** Teknik kombinasi
- **Uji Coba 3:** Teknik normalisasi
- **Uji Coba 4:** Lokasi dan ukuran ROI
- **Uji Coba 5:** Perbandingan efektivitas
- **Uji Coba 6:** Perbandingan efisiensi

1.7 Sistematika Laporan

Secara garis besar, buku tugas akhir ini dibagi menjadi enam bagian utama yaitu:

Ban I Pendahuluan

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

Bab II Dasar Teori

Bab ini berisi tentang penjelasan mengenai beberapa teori yang digunakan untuk menunjang pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan

Bab ini membahas tentang analisis yang dilakukan terhadap penelitian sebelumnya [8]. Dari hasil analisis tersebut, dilakukan perancangan terhadap sistem yang sedang dikembangkan.

Bab IV Implementasi

Bab ini membahas mengenai implementasi dari rancangan yang telah dibuat pada BAB III. Beberapa penjelasan mengenai kode yang digunakan juga disajikan.

Bab V Uji Coba dan Evaluasi

Bab ini membahas mengenai uji coba yang dibagi menjadi enam uji coba berdasarkan bagian 1.6. Evaluasi untuk setiap uji coba juga dilakukan untuk diambil kesimpulan.

Bab VI Kesimpulan dan Saran

Bab ini menyajikan kesimpulan yang dibuat berdasarkan hasil pada Bab V. Dari kesimpulan ini, disajikan pula saran untuk pengembangan sistem selanjutnya.

BAB II

DASAR TEORI

Bab ini menguraikan teori yang digunakan untuk menunjang pengerjaan tugas akhir ini.

2.1 *Dental Panoramic Radiograph*

Dental Panoramic Radiograph merupakan foto *x-ray* yang diambil untuk menangkap gambaran dari rahang manusia. Bagian yang ditangkap dalam citra adalah gigi, rahang atas, rahang bawah, dan area sekitarnya. **Gambar 2.1** merupakan salah satu contoh dari *dental panoramic radiograph*.

2.1.1 *Teori Dental Panoramic Radiograph*

Dental panoramic radiograph diambil berdasarkan prinsip *tomography* dan *slitradiography*. *Tomography* adalah teknik pemindaian bagian tubuh seseorang menggunakan pemindai *x-ray* yang bergerak untuk mendapatkan citra yang jelas dan rinci. *Slit-radiography* berkaitan dengan teknik untuk mendapatkan struktur pada citra dari perpindahan pemindai *slit beam*.



Gambar 2.1 Dental Panoramic Radiograph



Gambar 2.2 *Posisi Pasien* [10]

Referensi [10] mengungkapkan bahwa citra hasil prosedur sangat dipengaruhi oleh posisi pasien terhadap mesin. Oleh karena itu, dalam setiap prosedur terdapat instruktur yang akan memandu pasien pada posisi yang tepat. **Gambar 2.2** adalah contoh ilustrasi posisi dari alat dan posisi pasien. Berdasarkan citra hasil prosedur ini, dokter gigi dan ahli bedah mulut dapat melakukan diagnosis kondisi rahang pasien. Didasarkan dari informasi diagnosis ini, dokter dapat merencanakan perawatan terhadap pasien.

2.1.2 *Prosedur Pengambilan Dental Panoramic Radiograph*

Pengambilan *dental panoramic radiograph* dilakukan menggunakan alat pemindai *x-ray* atau alat *Cone Beam Computed Tomography* (CBCT). CBCT adalah tipe spesial dari mesin *x-ray* yang digunakan ketika alat *x-ray* biasa tidak cukup. Berdasarkan dari merk, satu alat CBCT memberikan fitur keluaran citra 2D saja, 3D saja, atau berdasarkan pilihan pengguna dari kedua opsi tersebut. **Gambar 2.3** merupakan citra tampilan dari salah satu merk CBCT [10].

Pasien yang akan melakukan prosedur pengambilan *dental panoramic radiograph* tidak membutuhkan persiapan khusus. Pasien biasanya diminta untuk melepas perhiasan yang dapat

mengganggu, seperti kacamata, anting, dan sebagainya. Khusus wanita, sangat dianjurkan untuk memberitahukan status kehamilan kepada dokter terkait. Pasien mengambil posisi duduk atau berbaring, bergantung dari jenis alat yang digunakan. Setelah itu, dokter akan membantu mengatur posisi pasien sesuai standard [11]. **Gambar 2.4** merupakan contoh posisi pasien terhadap alat dilihat dari atas. Pasien akan diminta untuk berdiam selama 20 hingga 40 detik [12].

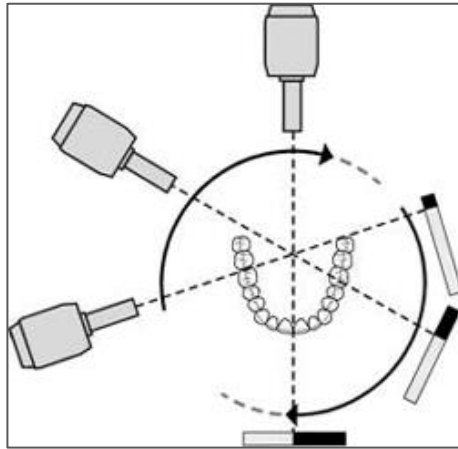


Gambar 2.3 Contoh CBCT [13]

Langkah selanjutnya adalah proses identifikasi dari *dental panoramic radiograph*. Hal yang dicatat biasanya nama pengguna, nomor pengambilan, dan tanggal dari prosedur pengambilan. Setelah itu, dokter terkait akan memberikan penjelasan terhadap citra hasil prosedur yang telah dilakukan [14]. **Gambar 2.1** merupakan salah satu contoh dari citra hasil prosedur yang telah dilakukan.

2.2 Struktur Penyusun *Dental Panoramic Radiograph*

Satu citra *dental panoramic radiograph* mengandung beberapa objek dengan bentuk yang bervariasi. Sebagian besar dari objek tersebut adalah tulang. Contoh tulang yang tampak adalah gigi, *trabecular bone*, dan *cortical bone*.



Gambar 2.4 Posisi Pasien dari Atas [15]

2.2.1 Tulang

Tulang merupakan jaringan yang hidup dan terus tumbuh. Tulang merupakan bagian tubuh yang sebagian besar terbuat dari kolagen dan kalsium. Kolagen merupakan protein yang lembut, sedangkan kalsium adalah mineral yang menguatkan. Kombinasi dari kolagen dan kalsium membuat tulang menjadi kuat dan namun tetap fleksibel dalam menerima beban. Tulang terdiri dari dua, yaitu *cortical bone* dan *trabecular bone* [1].

2.2.1.1 *Cortical Bone*

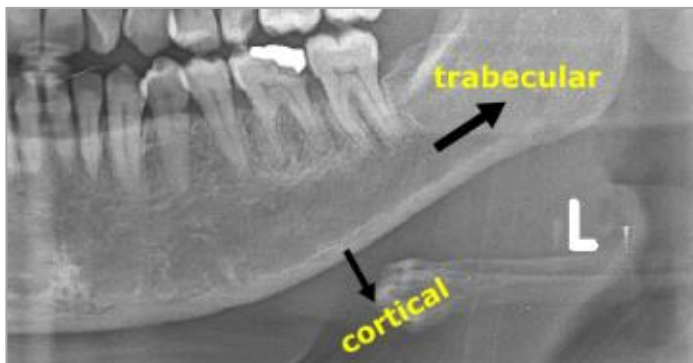
Cortical bone biasa disebut juga dengan *compact bone*. *Cortical bone* merupakan bagian paling luar dari tulang. *Cortical bone* memiliki struktur yang sangat rapat, keras, dan kaku. Sekitar 80% dari berat manusia adalah *cortical bone* [16].

Pada level mikroskopis, struktur susunan dari *cortical bone* sangat berbeda dengan *trabecular bone*. Beberapa fungsi dari *cortical bone* yaitu,

- menopang tubuh manusia,
- melindungi organ lain di bawahnya,
- menjadi tempat penyimpanan dari kalsium.

2.2.1.2 *Trabecular Bone*

Trabecular bone biasa juga disebut sebagai *cancellous bone*, merupakan bagian dari tulang yang bersifat seperti spons. Tidak seperti *cortical bone*, struktur dari *trabecular bone* tidak rapat, melainkan berongga dan lebih lembut. **Gambar 2.5** menunjukkan area *trabecular bone* pada citra *dental panoramic radiograph*. *Trabecular bone* tidak digunakan dalam tugas akhir ini [16].



Gambar 2.5 Area *Cortical Bone* dan *Trabecular Bone*

2.2.2 Penyakit Tulang

Tulang, seperti organ tubuh lainnya, dapat terganggu oleh penyakit. Beberapa penyakit tulang disebabkan oleh gen bawaan, pengaruh usia, bahkan kebiasaan setiap hari dari seseorang.

2.2.2.1 *Osteogenesis Imperfecta*

Osteogenesis imperfecta merupakan penyakit tulang yang dipengaruhi oleh bawaan gen. Penyakit ini menyebabkan tulang menjadi lemah dan mudah patah. Gejala dari penyakit ini juga berbeda berdasarkan dari jenisnya. Namun, ada beberapa umum yang biasa ditemui oleh penderita penyakit ini, yaitu,

- tulang yang berbentuk tidak wajar,
- tubuh kecil yang pendek,
- kulit yang mudah memar,
- sendi yang longgar,
- otot lemah, dan lain sebagainya.

Penanganan dari penyakit ini biasanya dilakukan untuk mencegah atau mengontrol gejalanya. Terkadang prosedur operasi dilakukan untuk membetulkan patah tulang, membantu kesalahan bentuk tulang, atau untuk menopang tulang belakang [17].

2.2.2.2 *Osteonecrosis*

Osteonecrosis adalah penyakit tulang yang disebabkan oleh hilangnya suplai darah ke tulang. Tanpa darah, jaringan tulang mati dan dapat menyebabkan tulang jadi hancur. Penyakit ini dapat menyerang siapa pun, namun umumnya menyerang orang yang berumur 30, 40, dan 50 tahun.

Osteonecrosis tidak menampakkan gejala pada fase awal. Namun, semakin parah penyakit ini, akan timbul rasa sakit ketika sendi mendapatkan beban. Ada saat sendi akan terasa sakit bahkan ketika seseorang sedang beristirahat [18].

2.2.2.3 Osteoporosis

Osteoporosis merupakan penyakit tulang yang menyebabkan tulang menjadi lemah dan mudah patah. Seseorang yang menderita *osteoporosis* umumnya akan mengalami patah tulang di bagian di area panggul, tulang belakang, dan pergelangan tangan.

Osteoporosis dapat terjadi pada pria dan wanita pada umur yang beragam, namun umumnya dapat terjadi pada wanita yang telah berusia lanjut. Meski pun kemungkinan terserang penyakit ini wanita sangat besar, tidak berarti pria tidak dapat mengalaminya. Faktanya, *osteoporosis* dapat terjadi pada pria dan wanita usia lanjut meski pun berbeda latar belakang.

Osteoporosis biasa disebut sebagai *silent disease*. Hal ini karena *osteoporosis* tidak menampilkan gejala sampai patah tulang terjadi. Sehingga diperlukan pemeriksaan rutin untuk mengetahui kemungkinan *osteoporosis* dari seseorang [2].

2.3 Region Of Interest (ROI)

2.3.1 Pengertian Region Of Interest (ROI)

Region of interest, biasa disingkat ROI, merupakan suatu area dalam suatu citra yang akan digunakan untuk suatu komputasi. ROI digunakan disebabkan karena tiga hal, yaitu eksplorasi data, meminimalkan kesalahan, dan membatasi *testing* [19].

2.3.1.1 Eksplorasi Data

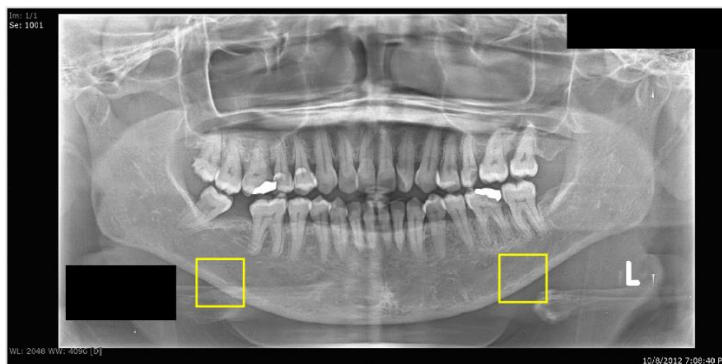
ROI memudahkan pengguna dalam melakukan eksplorasi data. Terkadang sulit melakukan eksplorasi atau menganalisis data mentah. Dari kesulitan tersebut, timbul permasalahan dalam hal pemodelan dari suatu sistem. Sehingga akan sulit menemukan pola dari suatu data untuk diambil informasi yang penting.

2.3.1.2 Meminimalkan Kesalahan

ROI dapat meminimalkan area analisis atau eksplorasi. Dari minimalnya area ini, tingkat keberagaman dari data juga dimungkinkan lebih kecil. Sehingga dapat memudahkan seseorang untuk melakukan analisis dan pemodelan sistem. Meski pun terjadi kesalahan dari suatu sistem, analisis untuk mencari dan membenahi kesalahan tersebut tergolong lebih mudah.

2.3.1.3 Membatasi *Testing*

Sangat penting untuk membatasi area *testing*. *Testing* diimplementasikan untuk mengetahui performa dari sistem yang telah dibuat. Selain itu, *testing* juga digunakan untuk mendapatkan informasi yang berharga dari suatu data. Dengan pembatasan area *testing*, pengguna dapat berfokus akan informasi yang ingin didapat. Sehingga apabila informasi yang didapat dianggap kurang, pengguna dapat dengan mudah mengidentifikasi *testing* seperti apa yang sebaiknya dilakukan.



Gambar 2.6 Contoh Are ROI

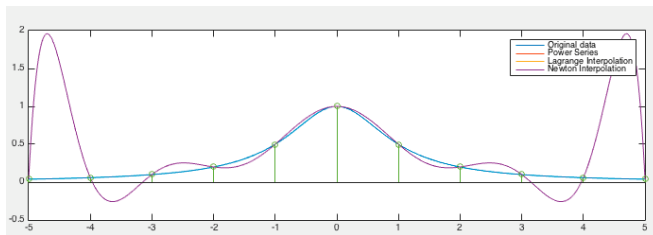
2.3.2 *Region of Interest* (ROI) yang Digunakan

Dari data *dental panoramic radiograph*, ROI yang digunakan adalah area di sekitar *cortical bone*. Area yang dibatasi oleh segi empat berwarna kuning pada **Gambar 2.6** merupakan salah satu

ROI pada *dental panoramic radiograph*. ROI yang digunakan dalam tugas akhir ini dibedakan menjadi dua berdasarkan uji coba yang sedang dilakukan. Pertama yaitu ROI dengan ukuran dan lokasi yang berbeda dengan skala citra yang ditentukan. Kedua yaitu ROI dengan ukuran yang tetap dengan lokasi yang berbeda. Bagian 3.2.2 menjelaskan lebih lanjut mengenai hal ini.

2.4 Cubic Spline Interpolation

Interpolasi merupakan teknik yang biasa digunakan untuk memperkirakan nilai x_c dan $v(x_c)$ yang berada di antara pola titik x_i dan $v(x_i)$ dengan ($i = 1, 2, \dots, n$). Berbeda dengan ekstrapolasi yang memperkirakan nilai $v(x_c)$ pada x_c di luar nilai x_i , interpolasi memastikan nilai $v(x_c)$ dengan x_c masih berada pada rentang nilai x_i [20]. Ada beberapa jenis interpolasi yang sudah dikembangkan, yaitu *Polynomial Interpolation*, *The Lagrange Interpolation*, *The Newton Polynomial Interpolation*, *Hermite Interpolation*, *Cubic Spline Interpolation*, *2-D Interpolation*. Teknik yang digunakan dalam tugas akhir ini adalah *Cubic Spline Interpolation* [20]. Gambar 2.7 menunjukkan performa teknik lain dalam memperkirakan titik untuk fungsi $y = f(x) = 1/(1+x^2)$.



Gambar 2.7 Perbandingan Teknik Interpolasi [21]

Cubic Spline Interpolation mengadopsi teknik dari *Polynomial Interpolation*. Namun terdapat perbedaan di mana kalau *Polynomial Interpolation* berusaha untuk menggambarkan kurva menggunakan seluruh titik acuan sedangkan *Cubic Spline Interpolation* hanya membutuhkan empat parameter tak tentu, lihat persamaan

(2.1), yaitu a_i, b_i, c_i, d_i . Untuk menemukan nilai keempat parameter tersebut agar nilai perkiraan $S(x)$ halus, maka persamaan (2.2), persamaan (2.3), persamaan (2.4), dan persamaan (2.5) harus terpenuhi [21].

$$C_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \quad (2.1)$$

$$C_i(x_i) = y_i \quad (2.2)$$

$$C_i(x_{i-1}) = y_{i-1} \quad (2.3)$$

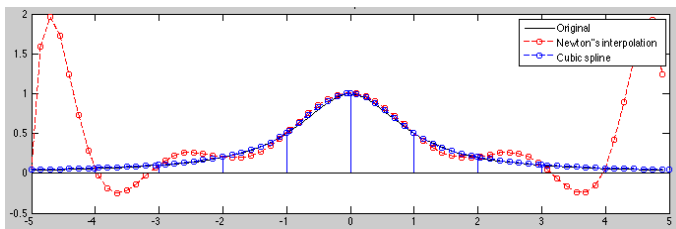
$$C'_i(x_i) = C'_{i+1}(x_i) \quad (2.4)$$

$$C''_i(x_i) = C''_{i+1}(x_i) \quad (2.5)$$

Gambar 2.8 menunjukkan perbandingan performa dari *Cubic Spline Interpolation* dan *Newton's Interpolation* pada nilai x untuk persamaan $y = f(x) = 1/(1+x^2)$.

2.5 Bicubic Interpolation

Image resize, seperti namanya, digunakan untuk mengubah ukuran citra dalam suatu proses. Beberapa teknik yang sudah dikembangkan yaitu *Nearest-neighbor Interpolation*, *Bilinear Interpolation*, dan *Bicubic Interpolation*. Pada tugas akhir ini digunakan *Bicubic Intepolation*.



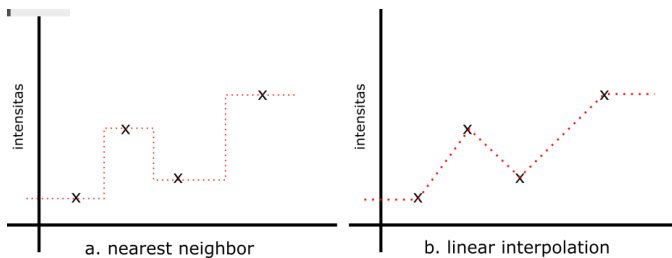
Gambar 2.8 Perbandingan Performa Cubic Spline dan Newton Interpolation [21]

Bicubic Interpolation adalah pengembangan dari *Cubic Spline Interpolation*. *Cubic Spline Interpolation* biasanya digunakan untuk data dimensi pertama sedangkan *Bicubic Interpolation* digunakan untuk data dua dimensi. *Bicubic Interpolation* biasa digunakan untuk melakukan perubahan skala pada citra.



Gambar 2.9 Contoh Intensitas Grayscale

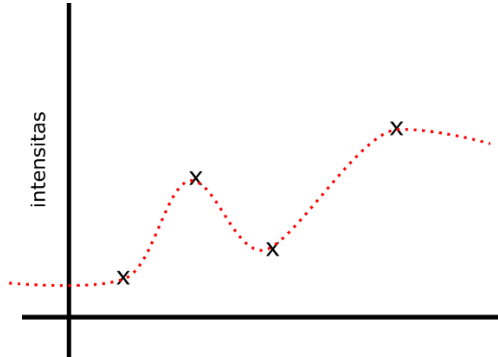
Bicubic Interpolation dipilih sebab hasil skala ulang citra mendekati hasil sebenarnya. Hal ini karena *Bicubic Interpolation* tidak melakukan duplikasi intensitas citra seperti *Nearest Neighbor Interpolation* (lihat **Gambar 2.10** a). Meski pun begitu, teknik ini tidak sesederhana *Linear Interpolation* (lihat **Gambar 2.10** b) dan tidak serumit *Polynomial Interpolation* yang membutuhkan waktu lebih besar. Untuk lebih jelas, **Gambar 2.9** merupakan contoh intensitas citra yang dipetakan pada **Gambar 2.11** dengan teknik *Bicubic Interpolation*.



Gambar 2.10 Nearest Neighbor dan Linear Intererpolation

Berbeda dengan *Cubic Spline Interpolation*, teknik *Bicubic Interpolation* menggunakan empat intensitas piksel tetangga.

Intensitas suatu piksel yang belum diketahui yang berada di antara empat piksel tetangga dapat ditentukan jika persamaan (2.1), persamaan (2.2), persamaan (2.3), persamaan (2.4), dan persamaan (2.5) dipenuhi.



Gambar 2.11 Contoh Pemetaan Cubic Spline

2.6 Sudut Orientasi

Salah satu parameter yang digunakan dalam kalkulasi *line strength* (lihat bagian 2.12 dan bagian 3.2.4.3) adalah sudut orientasi. Sudut orientasi θ dari dua titik dengan posisi (x_1, y_1) dan (x_2, y_2) dalam satu bidang dapat dihitung dengan menggunakan rumus trigonometri seperti pada persamaan (2.6).

$$\theta = \tan^{-1} \frac{y_2 - y_1}{x_2 - x_1} \quad (2.6)$$

2.7 Garis Orientasi

Selain sudut orientasi θ yang telah dijelaskan pada bagian 2.6, dalam proses perhitungan *line strength* (lihat bagian 2.12 dan bagian 3.2.5.2) akan dibuat garis orientasi untuk setiap level L . Dengan memanfaatkan sudut orientasi θ , garis orientasi pada satu *moving window* dibuat dengan persamaan (2.7). Dengan (x, y) adalah lokasi piksel yang dianggap garis dan $\frac{1-W}{2} \leq x \leq \frac{W-1}{2}$.

$$y = x \tan \theta \quad (2.7)$$

2.8 Euclidean Distance

Euclidean distance adalah suatu pengukuran jarak lurus antara dua titik pada bidang *Euclidean*. Pengukuran ini tidak hanya terbatas pada bidang dua dimensi, melainkan pada dimensi n yang lebih besar. Persamaan (2.8) merupakan bentuk umum dari *Euclidean distance* (dinotasikan D) pada dimensi n antara titik i dan titik j dengan semua titik berpasangan X [22].

$$D_{ij}^2 = \sum_{v=1}^n (X_{vi} - X_{vj})^2 \quad (2.8)$$

2.9 Ketebalan Cortical Bone

Ketebalan *cortical bone* merupakan salah satu parameter yang digunakan dalam proses kalkulasi *line strength* (lihat bagian 2.12 dan bagian 3.2.4). Ketebalan dari *cortical bone* dihitung berdasarkan titik penanda dari pengguna (lihat bagian 3.2.4), di sekitar area *cortical bone*, yang telah diinterpolasikan (lihat bagian 2.4). Titik-titik tersebut dibagi menjadi dua, yaitu titik atas *cortical bone* (x_{Ti} , y_{Ti}) dan titik bawah *cortical bone* (x_{Bi} , y_{Bi}) dengan $i = 1, 2, \dots, n$. Berikut algoritma yang digunakan:

- 1) Diambil titik (x_{Tc} , y_{Tc}) dan titik (x_{Bc} , y_{Bc}) dengan c berada pada nilai i .
- 2) Untuk titik (x_{Tc} , y_{Tc}), dihitung jarak ke semua titik (x_{Bi} , y_{Bi}), menggunakan *Euclidean distance* (dinotasikan D_{Tci}). Diambil jarak terkecil dari D_{Tci} yaitu D_{Tc} .
- 3) Untuk titik (x_{Bc} , y_{Bc}), dihitung jarak ke semua titik (x_{Ti} , y_{Ti}), menggunakan *Euclidean distance* (dinotasikan D_{Bci}). Diambil jarak terkecil dari D_{Bci} yaitu D_{Bc} .
- 4) Dari D_{Tc} dan D_{Bc} , diambil nilai paling kecil sebagai D_c dan dianggap sebagai ketebalan *cortical bone* pada titik c .
- 5) Proses a-d diulangi untuk semua nilai i .

- 6) Dari semua nilai ketebalan D_c , dihitung nilai ketebalan *cortical bone* (dinotasikan *Width*) dengan mengambil rata-rata.

2.10 Normalisasi

Normalisasi, atau biasa disebut standardisasi, merupakan teknik yang biasa digunakan untuk mengatur rentang suatu fitur pada *dataset* [23]. Teknik normalisasi biasanya digunakan untuk menjaga agar pengaruh antara suatu fitur dengan fitur yang lain dalam satu *dataset* dapat seimbang. Salah satu teknik normalisasi yang biasa digunakan biasa disebut teknik normalisasi *min-max* seperti pada persamaan (2.9). Dengan $I_{i,j}$ adalah intensitas baru hasil normalisasi pada piksel dengan lokasi (i, j) , $R_{i,j}$ adalah nilai *line strength* pada lokasi (i, j) , R_{min} adalah nilai *line strength* minimal dan R_{max} adalah nilai *line strength* maksimal.

$$I_{i,j} = \frac{R_{i,j} - R_{min}}{R_{max} - R_{min}} \times 255 \quad (2.9)$$

2.11 Line Strength

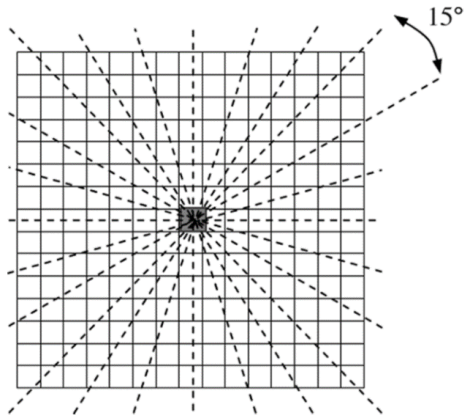
Penelitian oleh [6] menggunakan *line strength* untuk melakukan segmentasi terhadap pembuluh darah pada retina. *Line strength* pada suatu intensitas piksel $I_{i,j}$ pada citra didefinisikan dengan persamaan (2.10) dengan $L(i,j)$ adalah rata-rata maksimal intensitas piksel pada *line operator* dari semua orientasi dan $N(i,j)$ adalah rata-rata piksel pada *window* dengan pusat piksel pada $I_{i,j}$. Menggunakan *moving window* dengan ukuran 15 (lihat **Gambar 2.12**), dibuat 12 garis lurus dengan orientasi berbeda menggunakan sudut kelipatan 15° yang melalui piksel $I_{i,j}$ pada *moving window* (lihat **Gambar 2.12**).

$$S(i, j) = L(i, j) - N(i, j) \quad (2.10)$$

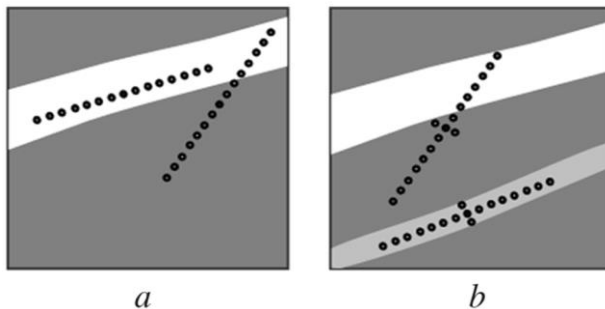
Nilai $S(i,j)$ besar apabila nilai $L(i,j)$ berimpit dengan pembuluh darah (lihat **Gambar 2.13**). Penelitian [8] menggunakan teknik ini

untuk mendeteksi struktur garis. Namun ada tiga permasalahan pada teknik ini yaitu,

- 1) cenderung menyatukan atau menyamakan dua pembuluh yang berdekatan,
- 2) menghasilkan ekstensi pada titik-titik *crossover*,
- 3) menghasilkan pendeteksian pembuluh yang salah pada area sekitar pembuluh.



Gambar 2.12 Orientasi Line Operator [6]



Gambar 2.13 Penampakan Line Operator Berimpit dengan Pembuluh Darah [6]

Berdasarkan penjelasan tersebut, penelitian ini kemudian dikembangkan oleh [7] yang dijelaskan pada bagian 2.12.

2.12 Multi-Scale Line Detection

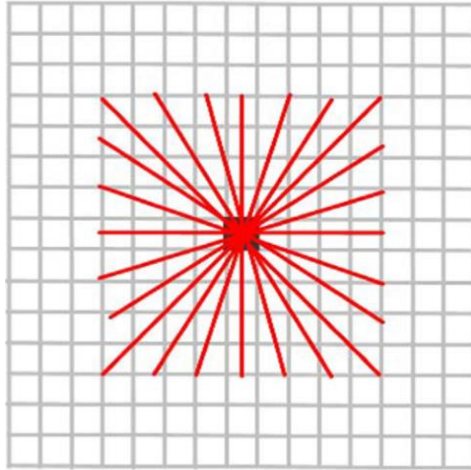
Penelitian oleh [7] mengembangkan teknik *line strength* oleh [6] berdasarkan permasalahan yang telah dijelaskan pada bagian 2.11. Dengan dasar yang sama seperti penelitian acuan, kalkulasi *line strength* memanfaatkan beberapa orientasi garis.

Terdapat sedikit perbedaan konsep pada penelitian ini yaitu skala untuk garis orientasi digunakan. Persamaan (2.11) digunakan untuk mencari *line strength* R_W^L suatu piksel I_{ij} pada *moving window* berukuran $W = 2 \times \text{Width} + 1$. L adalah skala dari orientasi garis dengan $L = 1, 3, \dots, W$. I_{max}^L adalah rata-rata maksimal dari semua garis orientasi dengan skala L (selanjutnya digunakan istilah level L untuk menghindari kerancuan) pada *moving window* W . I_{avg}^W adalah rata-rata intensitas piksel pada *moving window* W (lihat **Gambar 2.14**). **Gambar 2.14** adalah contoh *moving window* $W = 15$ dengan skala (garis merah) $L = 9$. Jumlah orientasi yang digunakan pada penelitian ini adalah 12, seperti tampak pada **Gambar 2.14**.

$$R_W^L = I_{max}^L - I_{avg}^W \quad (2.11)$$

Nilai *line strength* R_W^L hasil persamaan (2.11) akan bervariasi untuk satu level L dengan level lain. Oleh karena itu, diterapkan teknik standardisasi untuk nilai *line strength* pada setiap level L menggunakan persamaan (2.12). Hasil dari nilai *line strength* R tersebut kemudian dikombinasikan menggunakan persamaan (2.13) persamaan dengan n_L adalah jumlah level L garis orientasi yang digunakan pada *moving window* W dan I_{igc} adalah intensitas piksel dari citra. Penelitian [7] menambahkan I_{igc} dengan usulan bahwa nilai I_{igc} mengandung informasi patologi dan *optic disk* pada retina.

$$R' = \frac{R - R_{mean}}{R_{std}} \quad (2.12)$$



Gambar 2.14 Moving Window W dengan Level L [7]

$$R_{combine} = \frac{1}{n_L + 1} \left(\sum_L R_W^L + I_{lgc} \right) \quad (2.13)$$

2.13 Otsu's Thresholding

Thresholding adalah salah satu teknik membagi data menjadi dua berdasarkan suatu nilai pembagi yang disebut *threshold*. *Threshold* diharapkan dapat membagi citra menjadi objek dan *background* secara sempurna berdasarkan dari histogram citra itu sendiri. Dianggap sempurna apabila piksel objek dan *background* dikategorikan secara benar, yaitu 1 (satu) atau *True* untuk objek dan 0 (nol) atau *False* untuk *background*. Untuk mendapatkan hasil terbaik, nilai *threshold* harus memisahkan citra pada lembah terdalam dan tercuram dari histogram. Namun terdapat kesulitan

untuk menemukan nilai *threshold* ini sebab histogram citra pada umumnya tidak selalu berbentuk curam.

Penelitian oleh [9] mengusulkan teknik *thresholding* untuk mengatasi permasalahan ini, yang kemudian disebut *Otsu's thresholding*. *Otsu's thresholding* melakukan evaluasi untuk menemukan “goodness” dan secara otomatis mengambil *threshold* yang optimal tanpa tambahan informasi sebelumnya.

“Goodness” yang dimaksud menggunakan nilai variasi dari nilai masing-masing kelas. Semakin kecil variasi dari *intra*class dan semakin besar variasi dari *inter*class, maka nilai *threshold* yang dipilih mempunyai nilai “goodness” yang optimal. Untuk sebuah citra *grayscale* dengan intensitas $[1, 2, \dots, L]$, dihitung nilai variasi *intra*class σ_W^2 dan variasi *inter*class σ_B^2 dengan *threshold* k , dengan k merupakan salah satu nilai level intensitas *grayscale* pada citra, lihat persamaan (2.14) dan persamaan (2.15)

$$\sigma_W^2 = \omega_0 \sigma_0^2 + \omega_1 \sigma_1^2 \quad (2.14)$$

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2 \quad (2.15)$$

dengan ω_0 dan ω_1 masing-masing adalah kemungkinan kemunculan dari kelas *background* dan kelas objek, lihat persamaan (2.16) dan persamaan (2.17), dan μ_0 dan μ_1 adalah nilai rata-rata untuk kelas kelas terkait.

$$\omega_0 = \sum_{i=1}^k p_i \quad (2.16)$$

$$\omega_1 = \sum_{i=k+1}^L p_i \quad (2.17)$$

$$\mu_0 = \sum_{i=1}^k ip_i/\omega_0 \quad (2.18)$$

$$\mu_1 = \sum_{i=k+1}^L ip_i/\omega_1 \quad (2.19)$$

Nilai variasi dari objek σ_1^2 dan *background* σ_0^2 dihitung menggunakan persamaan (2.18) dan persamaan (2.19). Dibuktikan bahwa untuk variasi *intra*class minimal, maka variasi *inter*class maksimal [7] [24]. Oleh karena itu, implementasi akan lebih efisien jika digunakan nilai variasi *inter*class, lihat persamaan (2.15), saja untuk menemukan nilai *threshold* k yang optimal. Proses kalkulasi dilakukan untuk semua nilai k yang mungkin sehingga ditemukan nilai variasi *inter*class yang paling maksimal [24].

$$\sigma_0^2 = \sum_{i=1}^k (i - \mu_0)^2 p_i/\omega_0 \quad (2.20)$$

$$\sigma_1^2 = \sum_{i=k+1}^L (i - \mu_1)^2 p_i/\omega_1 \quad (2.21)$$

Keterangan:

- p_i : distribusi kemunculan intensitas *grayscale* i
- μ_0 : rata-rata kelas *background*
- μ_1 : rata-rata kelas objek

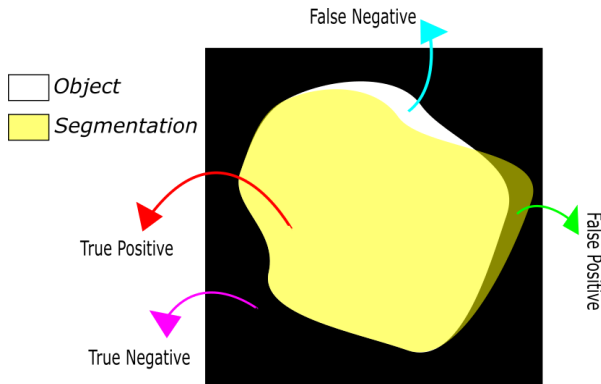
2.14 Luas Citra Biner

Kalkulasi luas dari objek dibutuhkan dalam proses segmentasi citra. Hal ini disebabkan karena hasil segmentasi suatu algoritma umumnya menghasilkan lebih dari satu objek pada lokasi terpisah dalam satu citra. Berbagai teknik kalkulasi luas citra yang telah dikembangkan. Dalam tugas akhir ini, kalkulasi luas objek citra biner menggunakan jumlah piksel yang termasuk dalam objek

tersebut. Jadi dalam satu citra hasil segmentasi, bisa terdapat lebih dari satu nilai luas objek bergantung dari jumlah objek terpisah dalam satu citra [25].

2.15 F_β score

Pengukuran performa suatu sistem sangat diperlukan untuk mengetahui sejauh mana kemampuan dari sistem tersebut. Dengan anggapan bahwa sistem pada tugas akhir ini melakukan klasifikasi piksel pada citra menjadi 0 (nol) *background* atau 1 (satu) objek, maka digunakan pengukuran berdasarkan *confusion matrix* (lihat **Gambar 2.15**).



Gambar 2.15 Ilustrasi Nilai Kebenaran pada Citra

Gambar 2.16 adalah salah satu contoh *confusion matrix* dengan *Actual* adalah jumlah data asli dengan kelas *Yes* atau *No* dan *Predict* adalah jumlah data yang diprediksikan sebagai kelas *Yes* atau *No*.

True Positive (TP) adalah keadaan ketika prediksi kelas suatu data benar dan kelas sebenarnya juga benar. *True Negative* (TN) adalah keadaan ketika prediksi kelas suatu data salah dan kelas sebenarnya juga salah. *False Positive* (FP) adalah keadaan ketika prediksi kelas suatu data benar namun kelas sebenarnya salah.

False Negative (FN) adalah keadaan ketika prediksi kelas suatu data salah namun kelas sebenarnya adalah benar.

Sistem dengan tugas utama klasifikasi biasanya menggunakan akurasi seperti pada persamaan (2.22). Akurasi valid digunakan sebagai dasar pengukuran apabila kelas data tidak terlalu timpang atau nilai TN tidak terlalu besar, jika dibandingkan dengan TP. Namun akurasi tidaklah cukup untuk mewakili performa suatu sistem apabila keadaan tersebut tidak terpenuhi.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.22)$$

		Predicted	
		Yes	No
Actual	Yes	True Positive	False Negative
	No	False Positive	True Negative

Gambar 2.16 *Confusion Matrix*

Akurasi tidak lagi valid digunakan apabila nilai TN jauh lebih besar dari nilai lain dalam *confusion matrix*. Keadaan lain yaitu ketika jumlah prediksi benar (TP atau FP atau kombinasi dari keduanya) adalah 0 (nol) sedangkan prediksi negatif (TN atau FN atau kombinasi keduanya) memiliki nilai. Kondisi ini disebut sebagai *accuracy paradox* [26]. Oleh karena itu, dibutuhkan pengukuran lain. Dalam tugas akhir ini, digunakan F_β score untuk menghitung performa sistem.

F_β score adalah salah satu pengukuran yang menggabungkan nilai *Precision* dan *Recall*. Persamaan (2.23) menunjukkan bentuk umum dari F_β score. *Precision* adalah perbandingan nilai kelas benar pada semua prediksi benar, lihat persamaan (2.24). *Recall* adalah perbandingan nilai prediksi benar dari semua kelas benar, lihat persamaan (2.25). β adalah nilai relatif kepentingan dari *Recall* dibandingkan *Precision* [27]. Semakin tinggi nilai F_β score, maka semakin baik performa suatu sistem.

$$F = \frac{(\beta^2 + 1.0) \times P \times R}{\beta^2 \times P + R} \quad (2.23)$$

$$P = \frac{TP}{TP + FP} \quad (2.24)$$

$$R = \frac{TP}{TP + FN} \quad (2.25)$$

Tugas akhir ini menganggap bahwa nilai *Precision* dan *Recall* setara. Oleh karena itu, digunakan $\beta = 1,0$ untuk kalkulasi nilai F_β score. Didasarkan dari persamaan (2.24) dan persamaan (2.25), persamaan umum F_β score dapat disederhanakan seperti pada persamaan (2.26).

$$F_1 = \frac{2 \times TP}{2TP + FP + FN} \quad (2.26)$$

BAB III

ANALISIS DAN PERANCANGAN

Bab analisis dan perancangan berisi analisis kebutuhan dan perancangan dari sistem pada tugas akhir ini.

3.1 Tahap Analisis

Tahap ini mendefinisikan kebutuhan yang harus dipenuhi oleh sistem untuk menyelesaikan permasalahan dan mencapai tujuan.

3.1.1 Spesifikasi Kebutuhan Sistem

Sistem pada tugas akhir ini dirancang untuk melakukan segmentasi *cortical bone* pada citra *dental panoramic radiograph* sistem ini diharapkan dapat menyesuaikan pada citra dengan ukuran yang berbeda. Berikut adalah proses yang diperlukan untuk membangun sistem ini:

1. Marking

Marking adalah tahap pengguna melakukan penandaan area luar sekitar *cortical bone* dimulai dengan menandai bagian atas dan diakhiri dengan menandai bagian bawah. Penandaan ini digunakan sebagai masukan untuk proses selanjutnya. Proses ini dibahas lebih lanjut pada bagian 3.2.3.

2. Pre-processing

Pre-processing untuk sistem pada tugas akhir ini bertujuan untuk mempersiapkan data sebagai parameter tahap berikutnya. Persiapan parameter didasarkan dari masukan yang telah diberikan pengguna pada tahap sebelumnya. Proses ini dibahas lebih lanjut pada bagian 3.2.4.

3. Line Strength

Perhitungan *line strength* diperlukan untuk mendeteksi piksel-piksel *cortical bone* pada ROI. Teknik yang digunakan adalah

multi-scale line detection [7] (lihat bagian 2.12) Hasil deteksi ini kemudian digunakan pada proses selanjutnya. Proses ini dibahas lebih lanjut pada bagian 3.2.4.

4. *Segmentation*

Segmentasi pada tugas akhir ini adalah mengubah hasil deteksi garis pada tahap sebelumnya menjadi citra biner. Citra biner berisi nilai 0 (nol) untuk *background* dan 1 (satu) untuk objek. Proses ini menggunakan algoritma *Otsu thresholding* [9] (lihat bagian 2.13). proses ini dibahas lebih lanjut pada bagian 3.2.4.

5. *Post-processing*

Gambar hasil *post-processing* mengandung *noise*. Diperlukan *post-processing* untuk mengatasi *noise* ini sehingga keluaran akhir dari sistem adalah citra hasil segmentasi. Proses ini dibahas lebih lanjut pada bagian 3.2.4.

3.1.2 Analisis Permasalahan

Permasalahan utama dari penelitian [8] adalah kemampuan sistem dalam menyesuaikan untuk citra dengan ukuran. Diperlukan sistem yang dapat menyelesaikan permasalahan ini dengan memperhatikan performa sistem yang stabil.

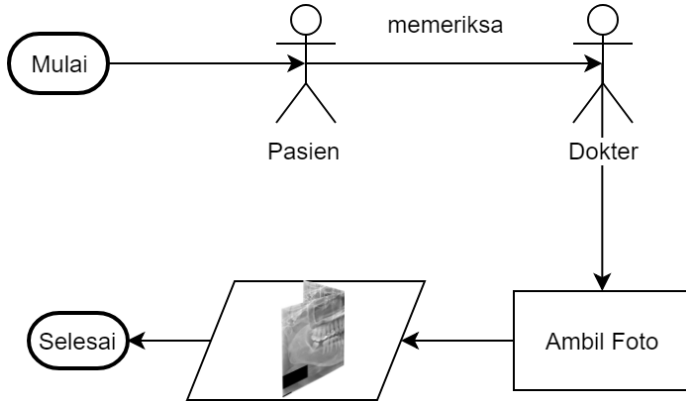
3.2 Tahap Perancangan

Tahap ini menjelaskan perancangan yang dilakukan untuk membangun sistem berdasarkan dari analisis yang telah dilakukan pada bagian 3.1. Tahap ini dibagi menjadi tiga bagian utama, yaitu perancangan sistem, perancangan data, dan perancangan proses.

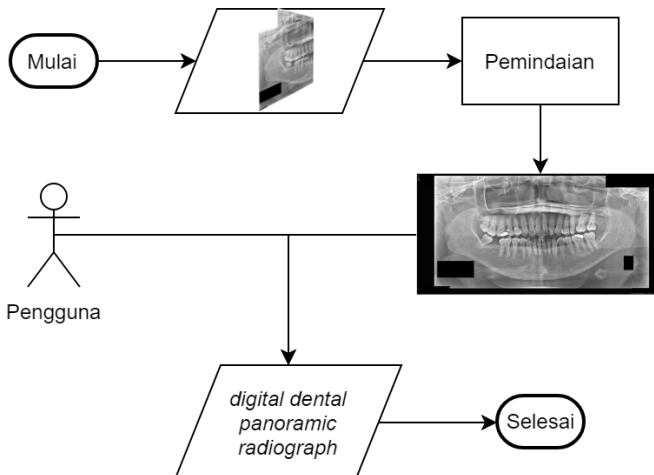
3.2.1 Perancangan Sistem

Perancangan sistem dilakukan untuk membuat referensi sistem secara keseluruhan. Referensi ini digunakan untuk membatasi proses yang diperlukan untuk menyelesaikan permasalahan dan mencapai tujuan.

Sistem dapat dijalankan dengan masukan adalah citra *dental panoramic radiograph*. Tentu citra tersebut harus dalam bentuk digital. Area yang dibutuhkan sesuai dengan deskripsi ROI yang telah dijelaskan pada bagian 2.3.2.



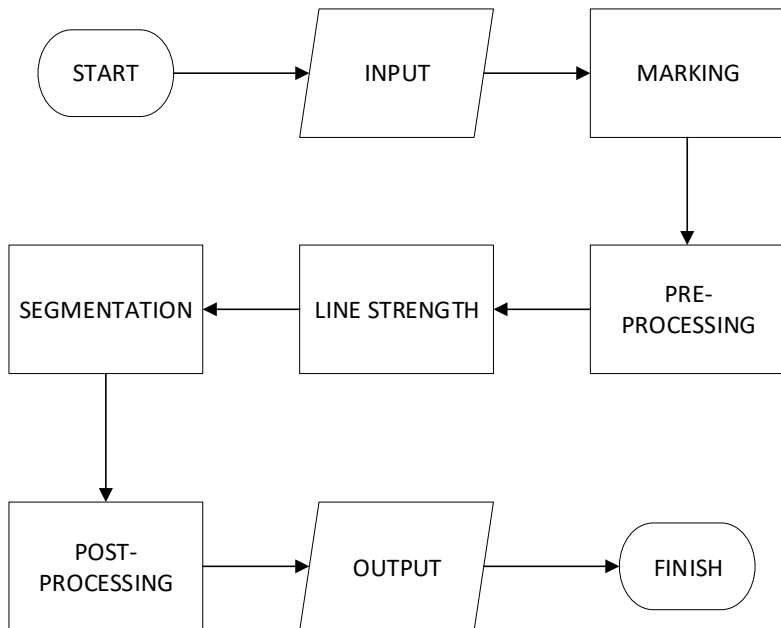
Gambar 3.1 Ilustrasi Pengambilan Foto Digital



Gambar 3.2 Ilustrasi Mengubah Citra Fisik Menjadi Digital

Citra *dental panoramic radiograph* bisa diperoleh dengan melakukan rontgen di rumah sakit atau laboratorium sekitar. Umumnya, citra yang didapat bukanlah citra digital. Oleh karena itu, diperlukan proses untuk mengubah citra yang didapat ke dalam format digital. **Gambar 3.1** dan **Gambar 3.2** mengilustrasikan tahapan yang bisa dilakukan untuk mendapatkan citra *dental panormaic radiograph* dalam format digital.

Berdasarkan dari spesifikasi yang telah dijabarkan pada bagian 3.1.1, dibuatlah rancangan fungsionalitas sistem secara keseluruhan seperti pada **Gambar 3.3**.



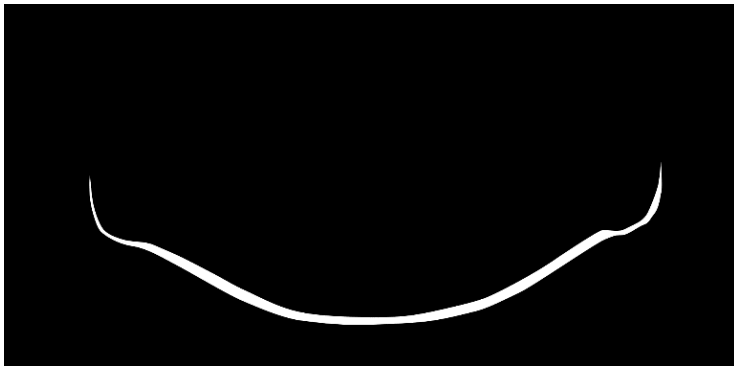
Gambar 3.3 Diagram Alir Sistem

Tampak pada **Gambar 3.3** bahwa terdapat 5 (lima) proses utama, yaitu *marking*, *pre-processing*, *line strength*, *segmentation*,

post-processing. Bagian 3.2 menjelaskan lebih lanjut masing-masing proses.

3.2.2 Perancangan Data

Perancangan data yang dilakukan tidak membutuhkan proses yang rumit. Data masukan (*input*) yang digunakan adalah *dental panoramic radiograph* yang sudah dalam format citra digital. **Gambar 2.1** adalah salah satu contoh citra masukan. Data *ground truth* adalah data yang digunakan sebagai dasar dari perhitungan performa sistem terhadap citra yang sedang diproses. **Gambar 3.4** adalah contoh dari data *ground truth*. Data keluaran (*output*) adalah citra digital biner, hitam dan putih dengan piksel putih adalah area *cortical bone* dan piksel hitam adalah area *background*.



Gambar 3.4 Contoh Ground Truth

Data masukan adalah *green channel* dari citra *dental panoramic radiograph* yang utuh. Sedangkan dalam prosesnya, pengguna akan memilih area tertentu sebagai ROI (lihat bagian 2.3.2). Area tersebut berada di sekitar *cortical bone* sehingga harus dipastikan bahwa bagian atas dan bagian bawah *cortical bone* masuk dalam ROI yang dipilih. Ukuran dari ROI didasarkan pada uji coba yang dilakukan (lihat bagian 5.3). Berikut adalah ukuran ROI yang digunakan berdasarkan uji coba:

- 1) **Uji Coba 1**, ukuran ROI adalah 128×128 dengan lokasi kanan dan kiri dari citra untuk skala 1,0. Skala yang digunakan adalah 0,25, 0,50, 0,75, 1,00, 1,25, 1,50, 1,75, 2,00. Untuk skala citra selain 1,0, ukuran ROI adalah nilai skala dari ukuran ROI pada skala 1,0. Contoh, untuk skala 0,25 maka ukuran ROI adalah 32×32 .
- 2) **Uji Coba 2**, ukuran ROI dan skala citra yang digunakan sama seperti pada **Uji Coba 1**.
- 3) **Uji Coba 3**, ukuran ROI dan skala citra yang digunakan sama seperti pada **Uji Coba 1**.
- 4) **Uji Coba 4**, ukuran ROI adalah sesuai dengan pilihan dari pengguna, baik untuk ROI kanan mau pun ROI kiri namun tetap diterapkan skala sesuai dengan **Uji Coba 1**.
- 5) **Uji Coba 5**, ukuran ROI dan skala citra yang digunakan sama seperti pada **Uji Coba 1**.
- 6) **Uji Coba 6**, ukuran ROI dan skala citra yang digunakan sama seperti pada **Uji Coba 1**.

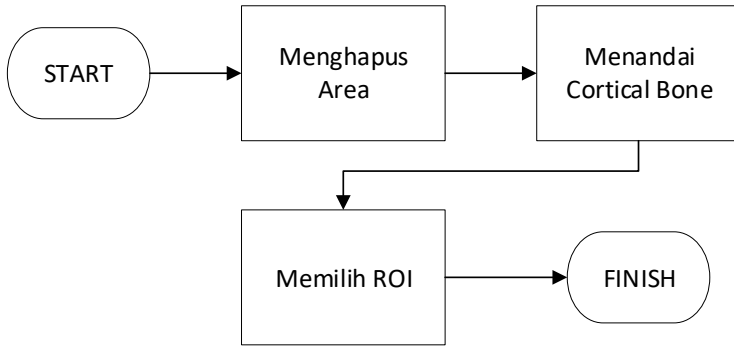
Data *ground truth* yang dibuat adalah citra hitam-putih dengan warna putih adalah area *cortical bone* dan area hitam adalah *background*. Performa dihitung berdasarkan dari citra ini (lihat bagian 2.15). Disebabkan karena parameter yang digunakan adalah lokasi dan ukuran ROI yang dimasukkan pengguna, maka area *ground truth* yang digunakan menyesuaikan dengan masukan dengan masukan pengguna.

Data keluaran (*output*) adalah citra biner dengan ukuran dan lokasi sesuai dengan ROI yang dipilih pengguna. Selain itu, citra, ROI, dan parameter lain secara otomatis disesuaikan dengan skala oleh sistem.

3.2.3 *Marking*

Tahap *marking* adalah tahap di mana pengguna memberikan penanda terhadap citra. Disebut penanda karena yang dilakukan oleh pengguna hanya menandai citra dan sistem hanya menangani proses ini. Penandaan ini menggunakan koordinat relatif. Proses

penandaan ini dibagi menjadi tiga tahap utama seperti tampak pada **Gambar 3.5**.



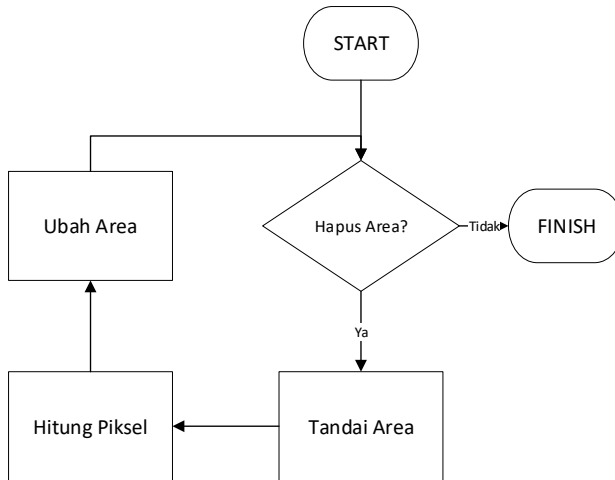
Gambar 3.5 Diagram Alir Proses Marking

3.2.3.1 Menghapus Area

Dikarenakan terdapat beberapa bagian pada citra yang sebenarnya tidak perlu, semisal label, maka sistem memberikan opsi kepada pengguna untuk menghapus beberapa area yang dianggap tidak perlu. Proses penghapusan ini dilakukan dengan mengubah nilai intensitas piksel *grayscale* pada area yang dipilih menjadi nilai rata-rata dari satu piksel di luar area yang dipilih.

Nilai piksel yang dipilih bukan 0 (nol) disebabkan karena akan terjadi *noise* disekitar area tersebut setelah proses perhitungan *multi-scale line detection* (lihat bagian 2.12). Hal ini terjadi apabila nilai area piksel untuk area yang berdekatan berbeda jauh. Area *cortical bone* adalah nilai intensitas piksel cerah, sehingga nilai 255 tidak digunakan untuk menggan-tikan nilai piksel. **Gambar 3.6** adalah diagram alir dari proses ini.

Proses ini dilakukan dengan asumsi bahwa area *cortical bone* yang dipilih tidak berada di sekitar area yang tidak penting. Keluaran dari proses ini adalah citra dengan piksel yang dianggap penting saja.



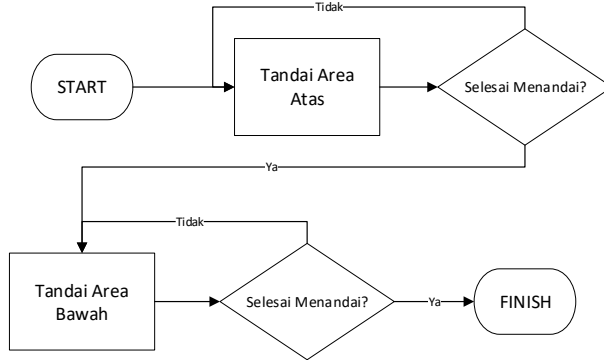
Gambar 3.6 Diagram Alir Menghapus Area

3.2.3.2 Penandaan Titik

Penandaan berikutnya adalah penandaan dengan titik pada bagian luar area di sekitar *cortical bone*. Titik yang ditandai oleh pengguna adalah titik pada area di sekitar *cortical bone* yang masih berada pada jarak 0,25 dari ketebalan *cortical bone* terdekat. Pengguna pertama menandai area luar bagian atas dari *cortical bone* dengan beberapa titik, dimulai dari kiri hingga kanan. Berikutnya pengguna memberitahukan kepada sistem bahwa area atas telah ditandai dengan mengirim perintah khusus. Setelah itu, pengguna mengulang hal yang sama pada area bawah *cortical bone*.

Langkah selanjutnya adalah pengguna memilih ROI dari area yang telah dibatasi oleh sistem. Ukuran dan lokasi dipilih sesuai dengan keinginan pengguna, dengan batasan bahwa area *cortical bone* (dari area bagian atas hingga area bagian bawah) berada di dalam ROI. Semua titik penanda disimpan dalam koordinat titik relatif. Dan titik penanda ini adalah keluaran dari tahap ini. **Gambar 3.7** adalah diagram alir dari proses ini.

Proses ini diimplementasikan berbeda pada skala selain 1,00. Penandaan ini tidak dilakukan oleh pengguna, namun dilakukan oleh komputer. Penjelasan lebih lanjut ada pada tahap selanjutnya.



Gambar 3.7 Diagram Alir Penanda Titik

3.2.3.3 Pemilihan ROI

Tahap ini dimulai dengan sistem memilih area untuk membatasi pengguna dalam memilih ROI. Area dibatasi dengan menggunakan titik yang sudah dipilih oleh pengguna. Dari titik tersebut, dilakukan interpolasi (lihat bagian 2.4) untuk bagian atas saja dan bagian bawah saja. Dari titik-titik atas dan titik-titik bawah area *cortical bone*, sistem memilih titik ujung paling kanan dan paling kiri dengan titik atas dan titik bawah masih memiliki pasangan pada kolom yang sama. Kemudian dibuat segi empat dengan menggunakan titik paling jauh sebagai batas. **Gambar 3.8** menunjukkan hasil dari proses ini. Titik-titik penanda ini adalah keluaran dari tahap ini.

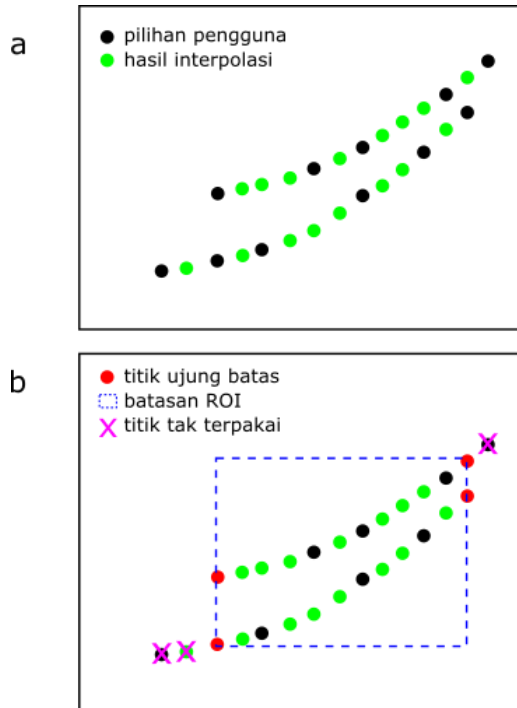
3.2.4 Pre-processing

Proses ini dilakukan untuk mendapatkan parameter yang dibutuhkan untuk kalkulasi *multi-scale line detection* (lihat bagian 2.12). Ada empat parameter yang dibutuhkan yaitu citra asli *dental panoramic radiograph*, lokasi ROI, sudut orientasi, dan ketebalan

dari *cortical bone*. Proses penentuan lokasi ROI, sudut orientasi, dan ketebalan *cortical bone* dilakukan menggunakan titik koordinat relatif yang telah dimasukkan oleh pengguna pada tahap sebelumnya.

3.2.4.1 *Dental Panoramic Radiograph*

Citra masukan *dental panoramic radiograph* adalah citra asli yang tidak mengalami potongan. Citra ini yang ditunjukkan ke pengguna untuk dilakukan proses penandaan titik-titik di sekitar area *cortical bone*. **Gambar 2.1** adalah contoh citra yang akan dijadikan masukan.



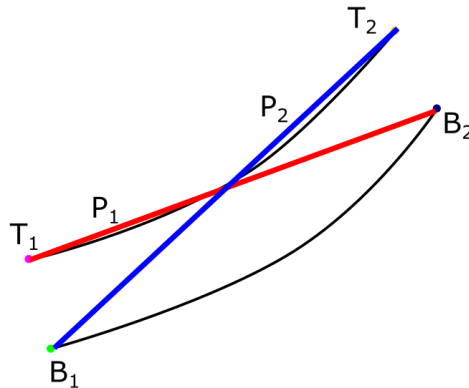
Gambar 3.8 a) *Titik Pilihan dan Interpolasi*; b) *Pemilihan Titik Batas ROI*

3.2.4.2 Lokasi ROI

Lokasi dari ROI didapat berdasarkan dari proses yang sudah dijelaskan pada bagian 3.2.3.3. Lokasi yang dimaksud adalah empat titik batas utama pada segi empat ROI. Dengan informasi ini, sistem dapat mengukur ukuran dari ROI.

3.2.4.3 Sudut Orientasi

Sudut orientasi digunakan untuk membatasi orientasi garis pada proses perhitungan *line strength* (lihat bagian 3.2.5). Digunakan dua sudut batas yaitu θ_1 dan θ_2 dengan $\theta_1 < \theta_2$. Dua sudut ini didapat dengan menarik dua garis lurus dari titik-titik batas seperti pada **Gambar 3.8 b**. Garis pertama dibentuk dengan menghubungkan batas kanan bagian bawah dengan batas kiri bagian atas. Garis kedua dibentuk dengan menghubungkan garis yang lain (lihat **Gambar 3.9**).

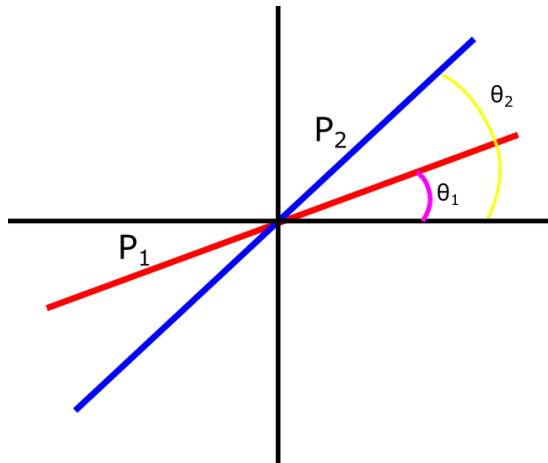


Gambar 3.9 Penarikan Garis Titik Batas

Perhitungan sudut dilakukan dengan menetapkan titik *origin* yaitu pada titik persimpangan antara dua garis ini. Kemudian dihitung sudut orientasi berdasarkan garis horizontal (lihat bagian 2.6). **Gambar 3.10** adalah contoh sudut yang terbentuk

3.2.4.4 Ketebalan *Cortical Bone*

Ketebalan *cortical bone* digunakan sebagai dasar ukuran dari *moving window* seperti dijelaskan pada bagian 2.9 dan bagian 2.12. Nilai ketebalan ini adalah jarak yang dihitung berdasarkan satuan piksel.



Gambar 3.10 Perhitungan Batas Sudut Orientasi

Berkaitan dengan uji coba seperti dijelaskan pada 5.3, nilai ketebalan adalah nilai skala dari ketebalan sebenarnya. Ketebalan sebenarnya adalah nilai ketebalan yang dihitung pada citra dengan skala 1,0 (satu koma nol).

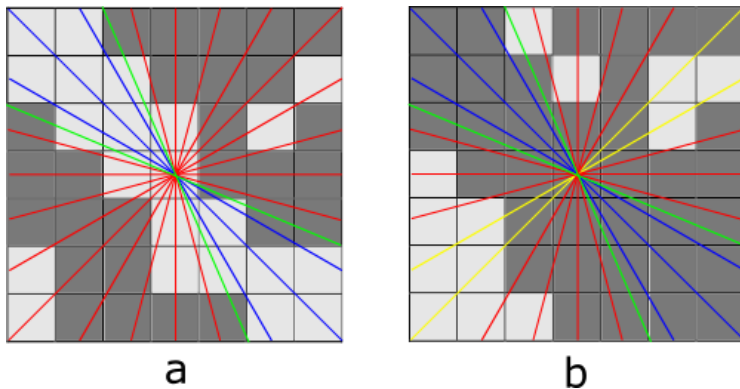
3.2.5 *Line Strength*

Line strength dihitung menggunakan algoritma *multi-scale line strength* (lihat bagian 2.12). Tahap perhitungan *line strength* pada tugas akhir ini dibagi menjadi empat tahap, yaitu pencarian sudut garis orientasi, pembuatan *moving window* untuk semua skala, perhitungan *line strength* untuk semua skala, dan kombinasi *line strength* dari semua level.

3.2.5.1 Pencarian Sudut

Dalam tugas akhir ini, parameter sudut orientasi θ_1 dan θ_2 digunakan sebagai pembatas orientasi. Jika dalam penelitian acuan oleh Nguyen [7] digunakan 12 sudut orientasi, namun dalam tugas akhir ini diusulkan untuk membatasi penggunaan garis orientasi berdasarkan sudut yang dianggap penting saja.

Gambar 3.11 adalah ilustrasi orientasi garis pada suatu *moving window*. Terdapat dua gambar *moving window*. **Gambar 3.11 a** adalah gambar *moving window* dengan area *cortical bone* sebagai pusat. **Gambar 3.11 b** adalah gambar *moving window* pada lokasi berbeda dalam satu citra dengan *background* sebagai pusat orientasi.



Gambar 3.11 Ilustrasi Orientasi Garis pada Moving Window

Gambar 3.11 juga memiliki garis-garis orientasi. Didasarkan pada **Gambar 3.11 a**, garis-garis tersebut diberi warna berbeda. Garis berwarna biru adalah garis dengan orientasi searah dengan orientasi *cortical bone* (area *cortical bone* adalah area lebih cerah) sehingga garis ini dianggap sebagai garis yang penting. Garis berwarna merah dan kuning adalah garis yang orientasinya tidak searah orientasi *cortical bone* dan dianggap sebagai garis yang tidak penting. Penggunaan orientasi dengan mengikutsertakan

garis ini dapat menimbulkan *noise*. **Gambar 3.11 b**, dapat menimbulkan *noise* dengan nilai *line strength* yang dihasilkan besar.

Didasarkan pernyataan pada paragraf sebelumnya, tugas akhir ini mengusulkan penghilangan orientasi garis yang tidak penting. Garis yang dianggap penting adalah garis dengan orientasi searah dengan orientasi *cortical bone*. Untuk mengetahui garis mana yang dianggap penting, maka digunakan θ_1 dan θ_2 sebagai batas. Perlu diingat bahwa $\theta_1 < \theta_2$ berdasarkan perhitungan sudut dengan arah berlawanan jarum jam.

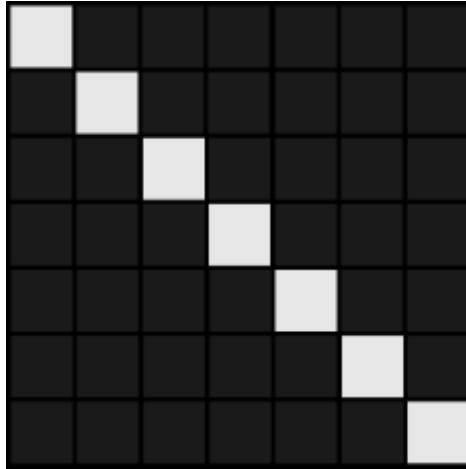
Garis berwarna hijau pada **Gambar 3.11** adalah contoh orientasi θ_1 dan θ_2 pada *moving window*. Dengan batasan orientasi ini, maka garis yang dianggap perlu adalah garis dengan warna biru. Dengan kata lain, garis berwarna merah dan kuning dianggap tidak perlu dan perlu dihilangkan. Dengan penghilangan garis-garis tidak penting tersebut, waktu dan sumber daya yang diperlukan untuk menghitung *line strength* dapat dikurangi.

Terdapat kasus tidak ditemukan sudut orientasi di antara θ_1 dan θ_2 yang bisa digunakan. Untuk mengatasi kasus tersebut, dilakukan proses pembagian sudut menjadi setengah dari jarak antar sudut sekarang. Proses ini menggunakan perulangan. Jika pada tahap pertama jumlah garis yang digunakan adalah 12 (dua belas) dengan sudut antar garis adalah 30° , maka perulangan berikutnya digunakan 24 garis dengan jarak sudut antar garis 15° . Perulangan terus dilakukan sampai jumlah garis yang berada di antara θ_1 dan θ_2 melebihi nilai yang ditentukan atau apabila perulangan terlalu banyak ditentukan oleh jumlah perulangan yang telah dilakukan.

3.2.5.2 Pembuatan *Moving Window*

Penelitian [8] membuat *moving window* pada setiap proses. Setiap proses yang dimaksud adalah proses perhitungan rata-rata intensitas piksel pada satu orientasi. Dalam perhitungan tersebut, terlebih dahulu diinisialisasi dengan membuat *moving window*

dengan satu orientasi tegak lurus. Lalu *moving window* tersebut dirotasi berdasarkan *orientasi* yang dibutuhkan. Langkah yang dilakukan efektif, namun sangat tidak efisien.



Gambar 3.12 Keluaran *Moving Window*

Didasarkan pada hal tersebut, pada tugas akhir ini dilakukan inisialisasi *moving window* untuk semua orientasi (lihat bagian 3.2.5.1) pada setiap level L . Dengan demikian, waktu komputasi dapat dikurangi, meski pun membutuhkan *memory* yang lebih untuk menyimpan hasil inisialisasi tersebut.

Pembuatan *moving window* menggunakan persamaan (2.7) seperti dijelaskan pada bagian 2.7. Keluaran dari proses ini adalah *window* biner dengan 0 (nol) adalah *background* dan 1 (satu) adalah garis. **Gambar 3.12** adalah contoh *moving window* keluaran dari proses ini.

3.2.5.3 Perhitungan *Line Strength*

Setelah dibuat cetakan untuk *moving window* untuk setiap orientasi pada setiap level L , dilakukan perhitungan *line strength*

menggunakan *multi-scale line detection*. Berikut adalah algoritma yang digunakan:

- 1) Untuk setiap piksel $I_{(i,j)}$ pada ROI, diambil piksel tetangga dengan ukuran *moving window* W dengan $I_{(i,j)}$ adalah titik pusat, sebut W_0 .
- 2) Untuk setiap level L pada setiap orientasi yang dipilih, diambil *moving window* W lalu dilakukan perkalian terhadap W_0 untuk menghasilkan W_1 .
- 3) Dihitung nilai rata-rata pada piksel W_1 yang tidak 0 (nol) lalu dibandingkan terhadap hasil rata-rata pada orientasi lain.
- 4) Diambil nilai maksimal lalu dihitung *line strength* menggunakan persamaan (2.11).

Keluaran dari proses ini adalah nilai *line strength* pada setiap level L yang kemudian akan dikombinasi pada bagian 3.2.5.4.

3.2.5.4 Kombinasi *Line Strength*

Keluaran dari bagian 3.2.5.3 adalah nilai *line strength* pada setiap level L . Dari nilai tersebut, dilakukan normalisasi pada setiap level L . Dari hasil normalisasi ini, kemudian dilakukan kombinasi untuk mendapatkan satu citra berisi *line strength* untuk setiap piksel, dengan ukuran dan lokasi sesuai dengan ROI.

Penelitian oleh Nguyen [7] mengusulkan penambahan nilai asli *grayscale* dengan menganggap bahwa nilai asli tersebut memberikan informasi tambahan untuk membedakan piksel pembuluh darah dan piksel lain. Hasil analisis dan hasil perhitungan *line strength R* pada citra menunjukkan bahwa *line strength R* dapat berupa nilai negatif, nol, atau pun positif.

Nilai *line strength R* menunjukkan seberapa kuat bahwa suatu piksel adalah piksel garis. Jika dilakukan penambahan nilai asli *grayscale*, maka nilai *line strength R* tidak lagi signifikan. Contoh, hasil perhitungan *line strength* suatu piksel adalah nilai negatif. Nilai *line strength R* negatif menunjukkan bahwa kemungkinan

piksel tersebut adalah piksel *background* (lihat **Uji Coba 3**). Apabila nilai asli *grayscale* tinggi, misal dikarenakan *noise*, maka jika ditambahkan nilai ini ke nilai *line strength* akan membuat hasil akhir nilai *line strength* tinggi. Nilai *line strength* R tinggi menunjukkan piksel tersebut adalah piksel *cortical bone*. Kasus ini akan meningkatkan *false positive* hasil segmentasi.

Berdasarkan pernyataan pada paragraf sebelumnya, tugas akhir ini mengusulkan teknik kombinasi tanpa menambahkan nilai intensitas asli piksel, seperti pada persamaan (3.27). Dilakukan pengujian untuk membandingkan metode kombinasi oleh Nguyen dan metode usulan pada tugas akhir ini (lihat **Uji Coba 2**).

$$R_{combine} = \frac{1}{n_L + 1} \left(\sum_L R_W^L \right) \quad (3.27)$$

3.2.6 Segmentation

Proses segmentasi yang dimaksud pada bagian ini adalah binerisasi citra *line strength*, yang merupakan keluaran bagian 3.2.5. Proses binerisasi adalah mengubah suatu citra menjadi 0 (nol) atau 1 (satu) berdasarkan metode. Metode yang digunakan pada tugas akhir ini adalah *Otsu thresholding* (lihat bagian 2.13).

Sebelum diambil nilai *threshold* optimal, dilakukan proses normalisasi terhadap nilai *line strength* untuk memetakan masing-masing nilai ke rentang nilai *grayscale*. Setelah dianalisis, tampak bahwa nilai *line strength* cenderung positif untuk piksel *cortical bone* dan cenderung negatif untuk piksel *background*. Oleh karena itu, pada tugas akhir ini diusulkan teknik normalisasi untuk memetakan nilai *line strength* ke rentang *grayscale* dengan memerhatikan kecenderungan nilai tersebut dalam bentuk persamaan (3.28) dengan $I_{i,j}$ adalah hasil normalisasi usulan dan $R_{i,j}$ adalah nilai *line strength* pada piksel dengan lokasi (i, j) .

$$I_{i,j} = \begin{cases} 0, R_{i,j} < 0 \\ R_{i,j} \times 255, 0 \leq R_{i,j} \leq 1 \\ 255, R_{i,j} > 1 \end{cases} \quad (3.28)$$

Persamaan (3.28) memetakan nilai *line strength* dengan memperhatikan kecenderungan setiap nilainya dengan tiga asumsi. Asumsi pertama nilai *line strength* kurang dari 0 (nol) adalah piksel *background*. Asumsi kedua nilai *line strength* lebih dari 1 (satu) adalah piksel *cortical bone*. Asumsi ketiga nilai *line strength* dalam rentang [0, 1] memiliki kemungkinan untuk menjadi *background* atau *cortical bone*. Untuk mengetahui kinerja dari teknik usulan, dilakukan pengujian yang dijelaskan pada **Uji Coba 3** (lihat 5.7). Sebagai pembandingan, dilakukan pengujian terhadap teknik normalisasi *min-max* seperti pada persamaan xxx. Dari hasil normalisasi ini, dilakukan pengambilan nilai *threshold* menggunakan *Otsu's thresholding* (lihat bagian 2.13). Setelah itu diterapkan binerisasi citra sebagai keluaran dari tahap ini.

3.2.7 *Post-processing*

Tahap *post-processing* adalah tahap untuk memperbaiki citra keluaran dari bagian 3.2.6. Hasil dari tahap sebelumnya masih mengandung *noise* yaitu munculnya area lain di luar area *cortical bone*. Oleh karena itu, perbaikan yang dilakukan adalah menghilangkan area tersebut. Dengan anggapan bahwa area *cortical bone* memiliki luas maksimum, dihitung luas untuk seluruh area yang ada pada citra (lihat bagian 2.14). Selanjutnya, area yang kurang dari nilai maksimum akan dihapus atau sesuai dengan pilihan dari pengguna. Keluaran dari tahap ini adalah segmentasi dari *cortical bone*.

BAB IV

IMPLEMENTASI

Bab ini menguraikan mengenai implementasi dari sistem berdasarkan rancangan pada BAB III. Selain itu, diuraikan juga implementasi dari setiap proses dalam sistem.

4.1 Lingkungan Implementasi

Lingkungan implementasi adalah lingkungan proses implementasi dilakukan. Lingkungan ini dibagi menjadi dua yaitu lingkungan perangkat keras dan lingkungan perangkat lunak.

4.1.1 Perangkat Keras

Lingkungan implementasi yang digunakan pada tugas akhir ini adalah sebuah *personal computer* (PC). PC yang digunakan adalah Dell Inc. 07KY25 dengan dua layar monitor. *Central processing unit* (CPU) yang digunakan adalah Intel Core i5 7400 @3.00 GHz dengan empat *core*. *Random access memory* (RAM) yang digunakan adalah Micron Technology DDR4 8 GBytes.

4.1.2 Perangkat Lunak

Perangkat lunak yang digunakan untuk implementasi adalah MATLAB R2017a 64-bit. Kode yang ada pada tugas akhir ini dibuat menggunakan perangkat lunak tersebut. Selain itu, untuk mendukung pengerjaan tugas akhir ini, digunakan perangkat lunak lain. Untuk membuat *ground truth*, digunakan Inkscape 0.92. Untuk dokumentasi, digunakan Microsoft Word 2016, Microsoft PowerPoint 2016, dan Microsoft Excel 2016.

4.2 Implementasi Pembacaan Data

Pembacaan data dilakukan secara dinamis. Data yang ingin diproses diletakkan di suatu direktori oleh pengguna. Dengan menuliskan lokasi direktori tersebut pada kode sumber, sistem secara otomatis akan membaca semua dokumen citra pada

direktori tersebut. Perlu dituliskan pula tipe data dari citra yang digunakan.

1	dataset = 'dataset';
2	gt = 'gt';
3	output = 'output';
4	img_dir = dir(dataset);
5	max_r = length(img_dir);
6	ctr = 1;
7	img_names = struct;
8	for i=1:max_r
9	x = img_dir(i).name;
10	y = strsplit(x, '.');
11	if strcmp(y{2}, 'bmp')
12	img_names(ctr).file = x;
13	ctr = ctr + 1;
14	end
15	clear x;
16	clear y;
17	end

Kode Sumber 4.1 Pembaca Nama Citra

1	min_r = 1;
2	max_r = length(img_names);
3	for i=min_r:max_r
4	max_img = max_img + 1;
5	img_adr= strcat(dataset, '/', img_names(i).file);
6	I = imread(img_adr);
7	I_green = I(:, :, 2);

Kode Sumber 4.2 Pembacaan Data Citra

Kode Sumber 4.1 melakukan pembacaan nama setiap citra dari direktori yang sudah ditentukan oleh pengguna. Nama ini kemudian digabung dengan direktori untuk dibaca data citra yang ditentukan seperti pada **Kode Sumber 4.2**.

4.3 Implementasi *Marking*

4.3.1 Menghapus Area

Implementasi pertama adari tahap *marking* adalah menghapus area yang dianggap tidak penting (lihat bagian 3.2.3.1). Area ini dipilih oleh pengguna.

1	<code>while true</code>
2	<code>while true</code>
3	<code>imshow(I_black);</code>
4	<code>choice = questdlg('Do you want to</code> <code>remove a region?',...</code>
5	<code>'Confirmation', 'YES', 'NO', 'NO');</code>
6	<code>if strcmp(choice, 'NO') break;</code>
7	<code>end</code>
8	<code>questdlg('Drag and drop on the</code> <code>area to remove, double click</code> <code>if done?',...</code>
9	<code>'Confirmation', 'OK', 'OK');</code>
10	<code>tp0 = size(I_black)/2;</code>
11	<code>tp1 = size(I_black)/4;</code>
12	<code>rect = imrect(gca, [tp0(2),</code> <code>tp0(1), tp1(2), tp1(1)]);</code>
13	<code>rect.wait();</code>
14	<code>rect0 = rect.getPosition();</code>
15	<code>rect.delete();</code>
16	<code>r_min = int32(rect0(2));</code>
17	<code>r_max = int32(rect0(2)+rect0(4));</code>
18	<code>c_min = int32(rect0(1));</code>
19	<code>c_max = int32(rect0(1)+rect0(3));</code>

Kode Sumber 4.3 Pemilihan Area Hapus

Kode Sumber 4.3 adalah implementasi untuk menandai area yang akan dihapus oleh pengguna. **Kode Sumber 4.4** mengganti area yang sudah ditandai dengan rata-rata piksel tetangga terdekat (lihat bagian 3.2.3.1).

1	try
2	I_black(r_min:r_max,c_min:c_max) = ...
3	uint8(zeros(r_max-r_min+1, c_max-c_min+1));
4	c = L;
5	while c > 0
6	try
7	tp = I_black(r_min-c:r_max+c, c_min-c:c_max+c);break;
8	catch c = c - 1;
9	end
10	end
11	[ir,ic] = size(tp);
12	tp(c+1:ir-c,c+1:ic-c) = uint8(zeros(ir-2*c,ic-2*c));
13	c = uint8(sum(sum(tp))/ sum(sum(tp>0)));
14	I_clean(r_min:r_max,c_min:c_max) = ...
15	uint8(ones(r_max-r_min+1, c_max-c_min+1))*c;
16	catch
17	ch = questdlg(strcat('Area selected is outside the boundary. ',... 'Try again?'), 'Confirmation', 'YES', 'NO', 'NO');
19	if strcmp(ch, 'NO')break;
20	end
21	end

Kode Sumber 4.4 Penghapusan Area yang Ditandai

4.3.2 Penandaan Titik

Kode Sumber 4.5 adalah implementasi penandaan area bagian atas *cortical bone* dengan titik-titik oleh pengguna. Penandaan ini berada di dalam iterasi tak hingga sebagai pengaman semisal terjadi kesalahan masukan dari pengguna.

1	<code>questdlg('Mark top boundary of cortical bone!', 'Attention', 'OK', 'OK');</code>
2	<code>while true</code>
3	<code> try</code>
4	<code> [pt_c1_top, pt_r1_top] = getpts;</code>
5	<code> pt_c2_top = round(min(pt_c1_top)+ 1:1:max(pt_c1_top)-1);</code>
6	<code> pt_r2_top = round(interp1(pt_c1_top, pt_r1_top, pt_c2_top, 'spline')));</code>
7	<code> pt_coor_top = [pt_c2_top; pt_r2_top]';</code>
8	<code> catch</code>
9	<code> ch = questdlg(strcat('Points chosen get conflict interpolation. ', ...</code>
10	<code> 'Try again?'), 'Confirmation', 'YES', 'NO', 'NO');</code>
11	<code> if strcmp(ch, 'NO')</code>
12	<code> return;</code>
13	<code> end</code>
14	<code> end</code>
15	<code> break;</code>
16	<code>end</code>

Kode Sumber 4.5 Implementasi Marking Bagian Atas

Kode Sumber 4.6 adalah implementasi penandaan area bagian bawah *cortical bone* dengan titik-titik oleh pengguna. Sama dengan **Kode Sumber 4.5**, penandaan ini berada di dalam iterasi sebagai pengaman semisal pengguna melakukan kesalahan penandaan.

Pada **Kode Sumber 4.5** (baris 5-7) dan **Kode Sumber 4.6** (baris 5-7) terdapat implementasi interpolasi yang akan digunakan pada tahap 4.3.3. Perlu diperhatikan bahwa urutan dari implementasi adalah bagian atas terlebih dahulu kemudian bagian bawah area *cortical bone*.

1	<code>questdlg('Mark bottom boundary of cortical bone!', 'Attention', 'OK', 'OK');</code>
2	<code>while true</code>
3	<code> try</code>
4	<code> [pt_c1_down, pt_r1_down] = getpts;</code>
5	<code> pt_c2_down = round(min(pt_c1_down+1):1:max(pt_c1_down)-1);</code>
6	<code> pt_r2_down= round(interp1(pt_c1_down, pt_r1_down, pt_c2_down, 'spline'));</code>
7	<code> pt_coor_down = [pt_c2_down; pt_r2_down]';</code>
8	<code> catch</code>
9	<code> ch = questdlg(strcat('Points chosen get conflict interpolation. ', ...</code>
10	<code> 'Try again?'), 'Confirmation', 'YES', 'NO', 'NO');</code>
11	<code> if strcmp(ch, 'NO')</code>
12	<code> return;</code>
13	<code> end</code>
14	<code> end</code>
15	<code> break;</code>
16	<code>end</code>

Kode Sumber 4.6 Implementasi Marking Bagian Bawah

4.3.3 Pemilihan ROI

Implementasi pemilihan ROI dibagi menjadi dua tahap, yaitu pemilihan batasan ROI oleh sistem lalu pemilihan ROI oleh pengguna berdasarkan batasan tersebut. Seperti dijelaskan pada bagian 3.2.3.3, batasan dari ROI yang dapat dipilih oleh pengguna adalah hasil interpolasi titik pada bagian 4.3.2. **Kode Sumber 4.7** adalah implementasi dari pemilihan batasan area ROI yang dapat dipilih oleh pengguna. Tampak dari kode sumber bahwa batasan area sangat dipengaruhi oleh penandaan pengguna pada tahap sebelumnya.

1	<code>x_s_top = min(pt_c2_top);</code>
2	<code>x_l_top = max(pt_c2_top);</code>
3	<code>y_s_top = min(pt_r2_top);</code>
4	<code>y_l_top = y_s_top;</code>
5	<code>x_s_down = min(pt_c2_down);</code>
6	<code>x_l_down = max(pt_c2_down);</code>
7	<code>y_s_down = max(pt_r2_down);</code>
8	<code>y_l_down = y_s_down;</code>
9	<code>c_min = max(x_s_top, x_s_down);</code>
10	<code>c_max = min(x_l_top, x_l_down);</code>
11	<code>r_min = y_l_top;</code>
12	<code>r_max = y_l_down;</code>

Kode Sumber 4.7 Pemilihan Batasan Area ROI

1	<code>I_limit=I_clean(r_min:r_max,c_min:c_max);</code>
2	<code>f1 = figure('Name', 'Limited Image');</code>
3	<code>imshow(I_limit);</code>
4	<code>questdlg('Drag and drop to mark ROI of cortical bone, double click if done!', 'Attention', 'OK', 'OK');</code>
5	<code>tp0 = size(I_limit)/2;</code>
6	<code>tp1 = size(I_limit)/4;</code>
7	<code>rect = imrect(gca, [tp0(2), tp0(1), tp1(2), tp1(1)]);</code>
8	<code>rect.wait();</code>
9	<code>rect1 = rect.getPosition();</code>
10	<code>rect.delete();</code>
11	<code>r_min = int32(r_min + rect1(2));</code>
12	<code>r_max = int32(r_min + rect1(4));</code>
13	<code>c_min = int32(c_min + rect1(1));</code>
14	<code>c_max = int32(c_min + rect1(3));</code>

Kode Sumber 4.8 Pengambilan Lokasi ROI

Kode Sumber 4.8 adalah implementasi untuk mendapatkan area ROI yang dipilih oleh pengguna. Keluaran dari tahap ini adalah lokasi ROI yang telah dipilih.

4.4 *Pre-processing*

Implementasi *pre-processing* adalah untuk mendapatkan parameter yang dibutuhkan untuk perhitungan *line strength*. Parameter yang dibutuhkan telah dijelaskan pada bagian 3.2.4. Dari parameter yang telah diuraikan, 2 (dua) parameter yang masih belum masuk dalam implementasi adalah sudut orientasi dan ketebalan *cortical bone*.

4.4.1 Sudut Orientasi

Perhitungan sudut batas orientasi dibagi menjadi 3 (tiga) tahap, yaitu penentuan titik-titik batas, pembuatan citra kosong, dan pengukuran sudut (lihat bagian 2.6 dan bagian 3.2.4.3).

1	<code>coor_selected_top = pt_coor_top(pt_coor_top(:,1)>c_min,:);</code>
2	<code>coor_selected_top = coor_selected_top(coor_selected_top(:,1)<c_max,:);</code>
3	<code>coor_selected_down = pt_coor_down(pt_coor_down(:,1)>c_min,:);</code>
4	<code>coor_selected_down = coor_selected_down(coor_selected_down(:,1)<c_max,:);</code>
5	<code>tp = coor_selected_top(:,1)==c_min;</code>
6	<code>left_coor_top = coor_selected_top(tp,:);</code>
7	<code>tp = coor_selected_top(:,1)==c_max;</code>
8	<code>right_coor_top = coor_selected_top(tp,:);</code>
9	<code>tp = coor_selected_down(:,1)==c_min;</code>
10	<code>left_coor_down = coor_selected_down(tp,:);</code>
11	<code>tp = coor_selected_down(:,1)==c_max;</code>
12	<code>right_coor_down=coor_selected_down(tp,:);</code>

Kode Sumber 4.9 Penentuan Titik-titik Batas

Kode Sumber 4.9 adalah implementasi penentuan titik batas untuk perhitungan sudut batas orientasi.

1	<code>point_panel = uint8(zeros(size(I_clean)));</code>
2	<code>point_panel(left_coor_top(2), left_coor_top(1)) = 1;</code>
3	<code>point_panel(right_coor_down(2), right_coor_down(1)) = 1;</code>
4	<code>degree_one = regionprops(point_panel, 'orientation');</code>
5	<code>if degree_one.Orientation < 0</code>
6	<code>degree_one.Orientation = degree_one.Orientation + 180;</code>
7	<code>End</code>
8	<code>point_panel = uint8(zeros(size(I_clean)));</code>
9	<code>point_panel(right_coor_top(2), right_coor_top(1)) = 1;</code>
10	<code>point_panel(left_coor_down(2), left_coor_down(1)) = 1;</code>
11	<code>degree_two = regionprops(point_panel, 'orientation');</code>
12	<code>if degree_two.Orientation < 0</code>
13	<code>degree_two.Orientation = degree_two.Orientation + 180;</code>
14	<code>End</code>
15	<code>point_panel = uint8(zeros(size(I_clean)));</code>
16	<code>point_panel(left_coor_top(2), left_coor_top(1)) = 1;</code>
17	<code>point_panel(right_coor_down(2), right_coor_down(1)) = 1;</code>
18	<code>degree_one = regionprops(point_panel, 'orientation');</code>
19	<code>if degree_one.Orientation < 0</code>
20	<code>degree_one.Orientation = degree_one.Orientation + 180;</code>
21	<code>End</code>

Kode Sumber 4.10 Perhitungan Sudut Batas Orientasi

Kode Sumber 4.10 adalah perhitungan sudut batas orientasi berdasarkan titik batas yang telah ditentukan (lihat **Kode Sumber 4.9**). Keluaran dari tahap ini adalah 2 (dua) sudut batas yang kemudian dinotasikan θ_1 dan θ_2 dengan $\theta_1 < \theta_2$. Pemilihan nilai sudut untuk mengisi nilai θ_1 dan θ_2 dilakukan oleh **Kode Sumber 4.11**.

1	if degree_one.Orientation < degree_two.Orientation
2	minimum_degree = degree_one.Orientation;
3	maximum_degree = degree_two.Orientation;
4	else
5	minimum_degree = degree_two.Orientation;
6	maximum_degree = degree_one.Orientation;
7	end

Kode Sumber 4.11 Pemilihan Sudut θ_1 dan Sudut θ_2

4.4.2 Ketebalan Cortical Bone

Implementasi dari perhitungan ketebalan *cortical bone* menggunakan algoritma seperti dijelaskan pada bagian 2.9 dan bagian 3.2.4.4. **Kode Sumber 4.13** adalah implementasi dari perhitungan ketebalan *cortical width*.

4.5 Line Strength

Perhitungan *line strength* dibagi menjadi 4 (empat) bagian utama (lihat bagian 3.2.5) yaitu pencarian sudut, pembuatan *moving window*, perhitungan *line strength*, dan kombinasi *line strength*.

Implementasi *line strength* ditulis dalam suatu fungsi dinamakan MultiScaleLineStrength yang diimplementasikan pada bagian selanjutnya.

1	function [R, Line_Scale, Orientation] = MultiScaleLineStrength(I, LOC, Width, OR)
---	---

Kode Sumber 4.12 Implementasi Fungsi Line Strength

4.5.1 Pencarian Sudut

Pencarian sudut dilakukan menggunakan perulangan dengan batasan setidaknya 2 (dua) garis berada di antara θ_1 dan θ_2 (lihat bagian 3.2.5.1). Batasan untuk perulangan semisal tidak ditemukan sudut yang sesuai adalah 1024 (seribu dua puluh empat). Untuk alasan fleksibilitas dan implementasi terhadap lingkungan implementasi perangkat lunak, maka nilai keluaran dari proses ini adalah daftar koefisien π sehingga hasil perkalian koefisien tersebut dengan π adalah sudut di antara θ_1 dan θ_2 . **Kode Sumber 4.14** adalah implementasi dari tahap ini.

1	if length(coor_selected_down) < length(coor_selected_top)
2	l_coor = length(coor_selected_down);
3	else
4	l_coor = length(coor_selected_top);
5	end
6	list_width = zeros([1,l_coor]);
7	for j = 1:l_coor
8	st = coor_selected_top(j,:);
9	t = (coor_selected_down-st).^2;
10	w1 = sum(t,2).^5;
11	st = coor_selected_down(j,:);
12	t = (coor_selected_top-st).^2;
13	w2 = sum(t,2).^5;
14	list_width(j) = min([w1;w2]);
15	end
16	cortical_width = mean(list_width);
17	Width = int32(cortical_width);

Kode Sumber 4.13 Perhitungan Ketebalan Cortical Bone

1	limit = 1024;
2	L = 12;
3	while true
4	list_degree_all = (1:L)/L;
5	if or (and (and (OR.minimum_degree>0, OR.minimum_degree<90)),...
6	and (OR.maximum_degree>0, OR.maximum_degree<90)),...
7	and (and (OR.minimum_degree>90, OR.minimum_degree<180)),...
8	and (OR.maximum_degree>90, OR.maximum_degree<180))
9	list_count = list_degree_all > OR.minimum_degree/180 & ...
10	else
11	if and (OR.minimum_degree*
12	OR.maximum_degree < 0,...
13	OR.maximum_degree>0)
14	or_left=180+OR.minimum_degree;
15	or_right = OR.maximum_degree;
16	else
17	or_left = OR.maximum_degree;
18	or_right = OR.minimum_degree;
19	end
20	list_count = list_degree_all > 0 & list_degree_all < or_right/180 ...
21	list_degree_all > or_left/180 &
22	list_degree_all < 180/180;
23	end

Kode Sumber 4.14 Perhitungan Sudut Orientasi Garis

Implementasi lanjutan untuk **Kode Sumber 4.14** terdapat pada **Kode Sumber 4.15**. **Kode Sumber 4.15** adalah batasan agar perulangan tidak berjalan terus menerus. Tampak bahwa apabila sudut yang sesuai tidak ditemukan sampai batasan jumlah

perulangan, maka seluruh komputasi akan dihentikan. Kejadian ini tidak akan dihasilkan keluaran *line strength* yang diinginkan.

4.5.2 Pembuatan *Moving Window*

Pembuatan *moving window* didasarkan dari jumlah orientasi garis keluaran dari bagian 4.5.1. Implementasi dari tahap ini dibagi menjadi dua, yaitu inisialisasi data dan pembuatan *moving window*.

1	<code>if sum(list_count) > 1</code>
2	<code>list_degree_selected = list_degree_all(list_count);</code>
3	<code>break;</code>
4	<code>end</code>
5	<code>clear list_count list_degree</code>
6	<code>if L > limit</code>
7	<code>error('Exceeded limit of loop.');</code>
8	<code>End L = L * 2;</code>
9	<code>End</code>

Kode Sumber 4.15 Implementasi Batasan Perulangan Orientasi

1	<code>window_size = int32(Width)*2 + 1;</code>
2	<code>r = int32(LOC.r_max - LOC.r_min + 1);</code>
3	<code>c = int32(LOC.c_max - LOC.c_min + 1);</code>
4	<code>[I_r, I_c] = size(I);</code>
5	<code>line_scale = struct;</code>
6	<code>for i = 1:window_size</code>
7	<code>if mod(i, 2) == 0</code>
8	<code>continue;</code>
9	<code>end</code>
10	<code>line_degree = struct;</code>
11	<code>for j = 1:length(list_degree_selected)</code>
12	<code>window = zeros(i,i);</code>
13	<code>xc = (i+1)/2;</code>
14	<code>yc = xc;</code>

Kode Sumber 4.16 Inisialisasi *Moving Window*

Implementasi untuk inisialisasi data ada pada **Kode Sumber 4.16**. Tampak pada kode sumber 2 (dua) perulangan. Perulangan pertama adalah untuk level L yang berbeda. Perulangan ke dua adalah untuk orientasi yang berbeda.

1	for k = xc-i:i-xc
2	if and(list_degree_selected(j) == .5, k==0)
3	for l = yc-i:i-yc
4	window(int32(k+xc),
5	int32(l+yc)) = 1;
6	end
7	else
8	l = round(k * tan(
9	list_degree_selected(j) * pi));
10	if or(or(k+xc>i, k+xc<1),
11	or(l+yc>i, l+yc<1))
12	continue
13	end
14	window(int32(k+xc),
15	int32(l+yc)) = 1;
16	end
17	end
18	line_degree(j).window = fliplr(window');
19	line_scale(i).degree = line_degree;
20	end
21	line_scale(i).image = zeros(r,c);
22	end

Kode Sumber 4.17 Pembuatan Moving Window

Implementasi dari pembuatan *moving window* tampak pada **Kode Sumber 4.17**. Kode sumber ini adalah lanjutan dari **Kode Sumber 4.16**. Tampak satu perulangan pada kode sumber ini.

Perulangan tersebut untuk menandai piksel yang merupakan piksel garis (lihat bagian 2.7).

4.5.3 Perhitungan *Line Strength*

Implementasi dari perhitungan *line strength* dibagi menjadi 3 (tiga), yaitu inisialisasi data, pemilihan piksel, dan perhitungan *line strength*. **Kode Sumber 4.18** adalah implementasi dari 2 (dua) tahap pertama.

1	<code>I_reference = uint8(zeros(I_r+(2*Width), I_c+(2*Width)));</code>
2	<code>I_reference(Width+1:Width+I_r, Width+1:Width+I_c) = I;</code>
3	<code>input = I_reference(int32((LOC.r_min+Width)-Width):...</code>
4	<code>int32((LOC.r_max+Width)+Width),...</code>
5	<code>int32((LOC.c_min+Width)-Width):...</code>
6	<code>int32((LOC.c_max+Width)+Width));</code>
7	<code>for i = Width + 1:Width + r</code>
8	<code>for j = Width + 1:Width + c</code>
9	<code>x = i; y = j;</code>
10	<code>window_w = zeros(window_size,window_size);</code>
11	<code>window_w = window_w + double(input(x-Width:x+Width, y-Width:y+Width));</code>
12	<code>i_w_avg = reshape(window_w,1,[]);</code>
13	<code>i_w_avg = mean(i_w_avg);</code>

Kode Sumber 4.18 Inisialisasi Data dan Pemilihan Piksel

Implementasi dari tahap ketiga terdapat pada Kode Sumber 4.19. Tampak dua perulangan. Perulangan pertama adalah perulangan untuk menunjukkan level L yang akan dihitung. Perulangan kedua adalah perulangan untuk orientasi yang telah terpilih.

Tampak dari **Kode Sumber 4.18** dan **Kode Sumber 4.19** bahwa jumlah perulangan yang digunakan dalam operasi perhitu-

ngan *line strength* adalah 4 (empat). Hal ini menyebabkan komputasi membutuhkan waktu yang lebih lama untuk ukuran ROI yang lebih besar. Bagian 5.10 membahas tentang hal ini.

1	for k = 1:window_size
2	if mod(k, 2) == 0
3	continue;
4	end
5	e = int32((k+1)/2-1);
6	window_l = zeros(k,k);
7	window_w = window_l + window_w(Width+1-e:Width+1+e, Width+1-e:Width+1+e);
8	i_l_max = zeros(1,L);
9	for l = 1:length(list_degree_selected)
10	temp_mul = window_l .* line_scale(k) .degree(l) .window;
11	
12	
13	i_l = reshape(temp_mul, 1, []);
14	i_l = sum(i_l)/sum(sum(line_scale(k) .degree(l) .window));
15	i_l_max(l) = i_l;
16	end
17	i_l_max = max(i_l_max);
18	r_l_w = i_l_max - i_w_avg;
19	line_scale(k) .image(x-Width,y-Width) = r_l_w;
20	
21	end
22	end
23	end

Kode Sumber 4.19 Perhitungan Line Strength

4.5.4 Kombinasi *Line Strength*

Tahap terakhir adalah kombinasi *line strength*. **Kode Sumber 4.20** adalah implementasi dari kombinasi *line strength*.

1	<code>r_combined = zeros(r,c);</code>
2	<code>for i = 1:window_size</code>
3	<code> if mod(i,2) == 0</code>
4	<code> continue;</code>
5	<code> end</code>
6	<code> t_img = reshape(line_scale(i</code> <code>).image, 1, []);</code>
7	<code> r_avg = mean(t_img);</code>
8	<code> r_std = std(t_img);</code>
9	<code> r_standard = (line_scale(i</code> <code>).image-r_avg)/r_std;</code>
10	<code> line_scale(i).standard = r_standard;</code>
11	<code> r_combined = r_combined + r_standard;</code>
12	<code>End</code>
13	<code>R = r_combined /</code> <code> ((double(window_size)+1)/2+1);</code>
14	<code>Line_Scale = line_scale;</code>
15	<code>Orientation = list_degree_selected;</code>

Kode Sumber 4.20 Implementasi Kombinasi

Tampak pada kode sumber baris 13-15 adalah keluaran dari perhitungan *line strength* pada bagian 4.5.

1	<code>[R, Line_Scale, Orientation] =</code> <code> MultiScaleLineStrength(I_clean,</code> <code> LOC, Width, OR);</code>
---	--

Kode Sumber 4.21 Pemanggilan Fungsi *Line Strength*

4.6 *Segmentation*

Implementasi tahap *segmentation* dibagi menjadi 3 (tiga), yaitu normalisasi *line strength*, pencarian *threshold*, dan binerisasi citra

(lihat bagian 3.2.6). Implementasi dari ketiga tahap tampak pada **Kode Sumber 4.22** pada masing-masing baris.

1	<code>R_none = uint8(255*R.*(R>0));</code>
2	<code>threshold = graythresh(R_none);</code>
3	<code>BW = imbinarize(R, threshold);</code>

Kode Sumber 4.22 Implementasi Segmentation

4.7 Post-processing

Tahap *post-processing* dibagi menjadi 3 (tiga) tahap, yaitu pilihan pengguna, perhitungan luas, dan penghapusan area. **Kode Sumber 4.23** adalah implementasi untuk menanyakan pilihan pengguna untuk memilih area maksimum atau memilih secara manual area yang akan dipertahankan.

1	<code>choice = questdlg(strcat('Choose if the 'machine will get the ',... 'region of cortical bone using Maximum area or Manual choosing.'),... 'Confirmation', 'Maximum', 'Manual', 'Manual');</code>
2	
3	

Kode Sumber 4.23 Implementasi Pilihan Pengguna

Kode Sumber 4.24 adalah implementasi yang akan dijalankan jika pengguna memilih area maksimum.

1	<code>if strcmp(choice, 'Maximum') st = regionprops('table', BW);</code>
2	<code>index_area_max = find(st.Area == max(st.Area));</code>
3	<code>cortical_region = bwareafilt(BW,...</code>
4	<code>[st.Area(index_area_max), st.Area(index_area_max)];</code>

Kode Sumber 4.24 Implementasi Jika Area Maksimum

Kode Sumber 4.25 adalah implementasi yang akan dijalankan jika pengguna ingin memilih secara manual area yang dianggap *cortical bone*.

1	else
2	questdlg(strcat('Choose a point of cortical bone ',... 'on the image following!'), 'Attention', 'OK', 'OK');
3	BW_labeled = bwlabel(BW);
4	f7 = figure;
5	while true
6	imshow(BW);
7	[xi,yi] = getpts;
8	c = int32(xi(1));
9	max_r = int32(yi(1));
10	if BW(max_r,c)
11	break;
12	end
13	questdlg(strcat('Chosen coordinate is not an object. ',... 'Please try again!'), 'OK', 'OK');
14	End
15	close(f7);
16	cortical_region = BW_labeled== BW_labeled(max_r,c);
17	end
18	else
19	
20	

Kode Sumber 4.25 Implementasi Jika Pemilihan Manual

Implementasi penghapusan area yang dianggap tidak penting ada pada **Kode Sumber 4.24** baris 3 dan baris 4 dan pada **Kode Sumber 4.25** baris 18.

[Halaman ini sengaja dikosongkan]

BAB V

UJI COBA DAN EVALUASI

Bab ini membahas mengenai hasil uji coba sistem yang telah dikembangkan. Uji coba dilakukan untuk mengetahui performa sistem pada tugas akhir ini.

5.1 Lingkungan Uji Coba

Lingkungan uji coba pada tugas akhir ini berbeda dengan lingkungan implementasi sistem dalam hal lingkungan perangkat keras (lihat bagian 4.1). PC yang digunakan pada uji coba adalah Dell Inc. 0Y7WYT dengan CPU Intel Core i7 6700 @3.4 GHz dengan 4 (empat) *core*. RAM yang digunakan adalah Corsair DDR4 24 GBytes.

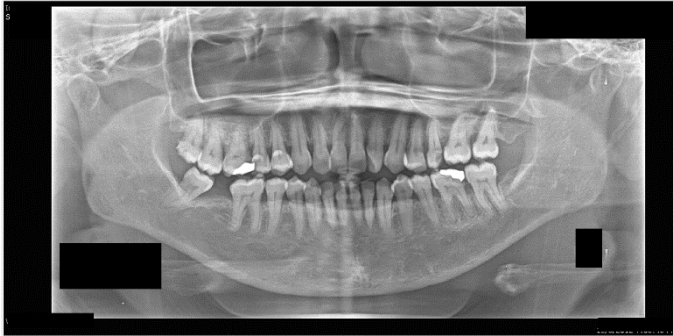
5.2 Data Uji Coba

Data uji coba yang digunakan pada tugas akhir ini adalah 30 (tiga puluh) citra *dental panoramic radiograph*. **Gambar 5.1** adalah contoh citra yang digunakan. ROI yang digunakan pada setiap citra adalah satu citra ROI pada bagian kanan dan satu citra ROI pada bagian kiri dengan ukuran masing-masing 128×128 . Khusus **Uji Coba 3**, ROI yang diambil adalah dua citra ROI di bagian kanan dan dua citra ROI di bagian kiri citra dengan satu sama lain memiliki ukuran dan lokasi berbeda. **Gambar 5.2** adalah contoh ROI dari citra asli. **Gambar 5.3** adalah lokasi ROI pada citra.

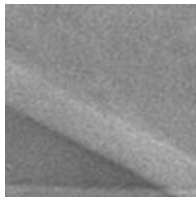
Berdasarkan ROI yang dipilih, dilakukan perhitungan performa sistem. Performa dari hasil segmentasi dihitung berdasarkan citra *ground truth* sebagai acuan. **Gambar 5.4** adalah contoh *ground truth* citra dan **Gambar 5.5** adalah contoh ROI dari *ground truth* berdasarkan dari ukuran dan lokasi ROI citra asli yang dipilih.

Citra masukan pada tugas akhir ini diskala ulang seperti telah dijelaskan pada bagian 3.2.2. Citra asli, ROI citra asli, *ground*

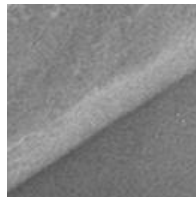
truth, dan ROI dari *ground truth* juga akan diubah menyesuaikan dengan skala yang telah ditetapkan.



Gambar 5.1 Dental Panoramic Radiograph

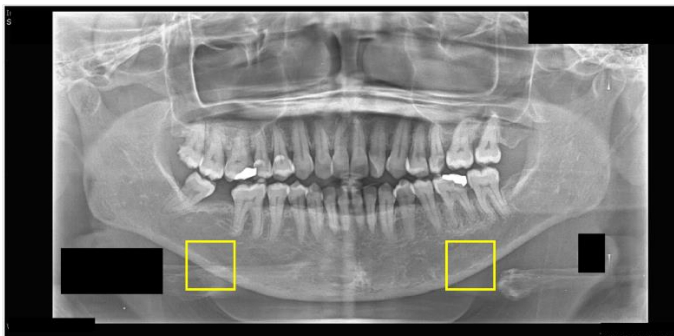


(a)

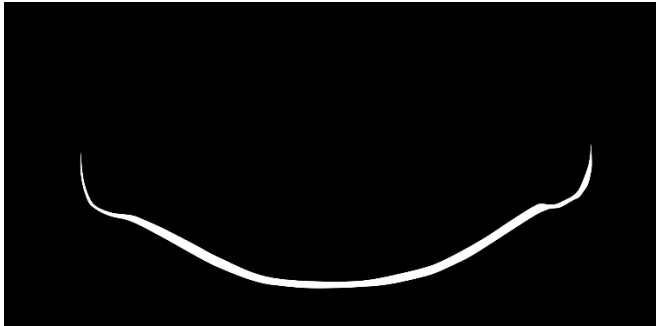


(b)

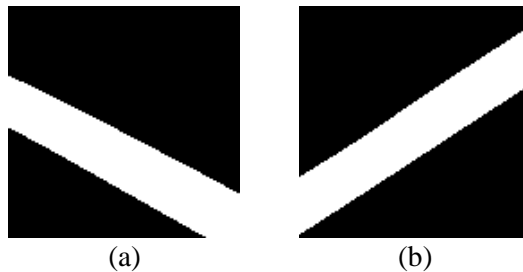
Gambar 5.2 ROI Terpilih (a) ROI Kiri (b) ROI Kanan



Gambar 5.3 Lokasi ROI pada Citra



Gambar 5.4 Ground Truth



Gambar 5.5 ROI Ground Truth (a) ROI Kiri (b) ROI Kanan

5.3 Uji Coba

Uji coba pada tugas akhir ini dibagi menjadi dua, yaitu uji coba fungsionalitas dan uji coba metode. Uji coba fungsionalitas dilakukan untuk mengetahui performa sistem dalam menjalankan satu siklus segmentasi. Uji coba metode dilakukan untuk mengetahui performa metode yang digunakan pada sistem. Terdapat enam uji coba metode yang dilakukan pada tugas akhir ini yaitu,

- 1) **Uji Coba 1:** Orientasi garis. Uji coba penggunaan orientasi garis terpilih pada operator *moving window* (lihat bagian 3.2.5),
- 2) **Uji Coba 2:** Teknik kombinasi. Uji coba penambahan nilai asli *grayscale* pada tahap kombinasi *line strength R* (lihat bagian 2.12),

- 3) **Uji Coba 3:** Teknik normalisasi. Uji coba metode normalisasi pada nilai *line strength R* (lihat bagian 2.10),
- 4) **Uji Coba 4:** Lokasi dan ukuran ROI. Uji coba performa sistem untuk ROI dengan lokasi dan ukuran berbeda,
- 5) **Uji Coba 5:** Perbandingan efektivitas. Uji coba perbandingan efektivitas sistem pada tugas akhir ini dengan sistem penelitian sebelumnya [8], dan
- 6) **Uji Coba 6:** Perbandingan efisiensi. Uji coba perbandingan efisiensi sistem pada tugas akhir ini dengan sistem pada penelitian sebelumnya [8].

Performa untuk **Uji Coba 1** hingga **Uji Coba 5** dihitung menggunakan F_β score dengan $\beta=1,0$ (lihat bagian 2.15), selanjutnya disebut F_1 score. Khusus untuk **Uji Coba 6**, perhitungan efisiensi adalah dalam satuan detik. Dan untuk semua uji coba, ukuran citra yang digunakan diskala ulang dengan nilai 0,25, 0,50, 0,75, 1,00, 1,25, 1,50, 1,75, dan 2,00 dari ukuran citra asli. Dengan demikian, akan dihasilkan performa dan waktu eksekusi pada citra pada skala citra berbeda.

5.4 Uji Coba Sistem

Seperti yang dijelaskan pada bagian 5.3, uji coba ini dilakukan untuk mengetahui performa sistem dalam menyelesaikan satu siklus operasi segmentasi. Sesuai dengan penjelasan sebelumnya, terdapat lima proses utama yang dikerjakan oleh sistem.

5.4.1 Marking

Penandaan dibagi menjadi dua, yaitu penandaan untuk menghilangkan area yang dianggap tidak penting dan penandaan untuk disekitar area *cortical bone*.

5.4.1.1 Menghapus Area

Gambar 5.6 adalah citra penanda untuk area yang dianggap tidak penting. Contoh area ini adalah label *L*. **Gambar 5.7** adalah gambar hasil penghapusan area ini (perlu diperhatikan label citra).



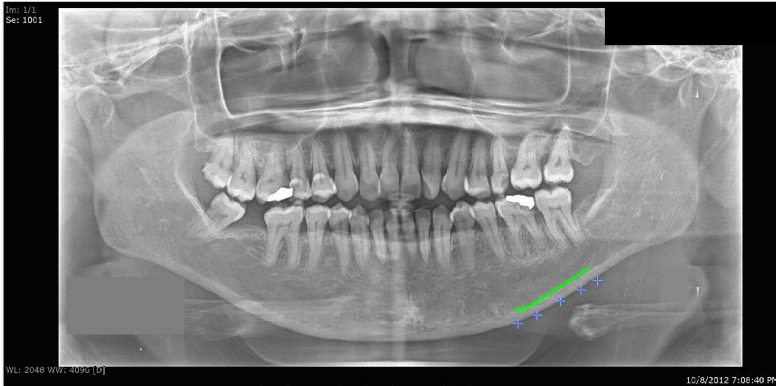
Gambar 5.6 Penandaan untuk Menghapus Area



Gambar 5.7 Hasil Penghapusan Area

Seperti dijelaskan pada bagian 3.2.3, area yang ditandai untuk dihilangkan kemudian setiap pikselnya diganti dengan piksel rata-rata satu piksel di sekitar area tersebut. Tampak pada **Gambar 5.7** bahwa citra hasil penandaan berwarna keabuan. Hal ini dikarenakan area sekitar berwarna keabuan. Jadi untuk area yang dikelilingi oleh piksel berwarna hitam, maka area tersebut akan diganti dengan piksel gelap atau berwarna hitam sesuai dengan rata-rata area sekitar.

Tahap kedua dari penandaan ini adalah menandai area sekitar *cortical bone*. **Gambar 5.8** adalah gambar hasil penandaan ini. Tampak pada gambar titik-titik penanda oleh pengguna.



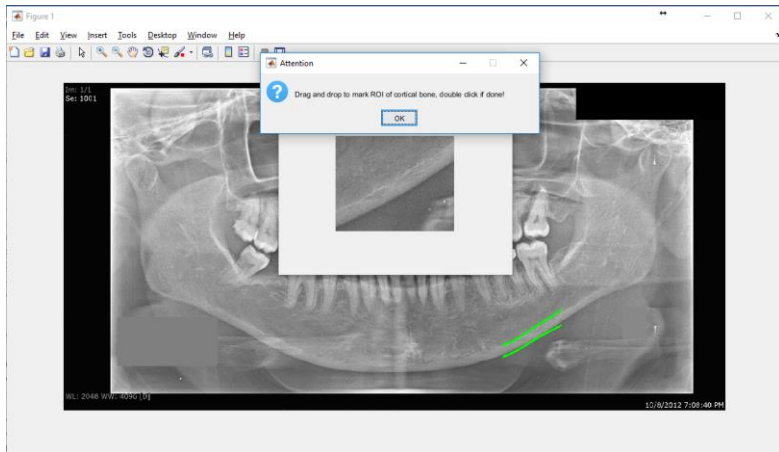
Gambar 5.8 Hasil Penandaan Cortical Bone

Seperti yang dijelaskan, terdapat dua tahap penandaan titik-titik di sekitar *cortical bone*. Tahap pertama adalah menandai area bagian luar sebelah atas dari *cortical bone*. Ketika dikirimkan perintah khusus, bahwa penandaan bagian atas selesai, akan ditampakan garis berwarna hijau (lihat **Gambar 5.8**) hasil interpolasi titik-titik tersebut.

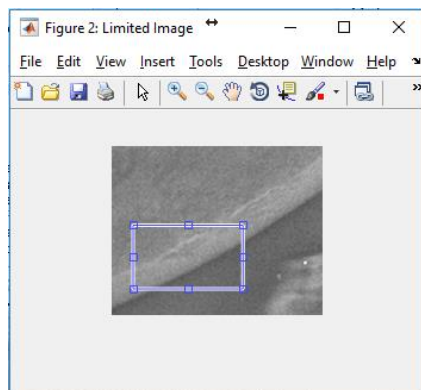
Setelah dilakukan penandaan pada bagian atas, pengguna kemudian melanjutkan menandai pada bagian bawah. Sama seperti pada penandaan bagian atas, dilakukan interpolasi pada titik-titik hasil penandaan. **Gambar 5.9** menunjukkan contoh hasil keluaran setelah kedua tahap penandaan titik-titik dilakukan.

Terdapat tiga *window* keluaran pada **Gambar 5.9**. *Window* paling bawah (paling belakang) adalah *window* keluaran citra hasil tahap penandaan titik. Terdapat garis hasil interpolasi yang telah dipotong (lihat bagian 3.2.4). *Window* di tengah adalah area batasan yang dibuat oleh sistem berdasarkan penanda titik. Ukuran dari citra di dalam *window* adalah sesuai dengan batas atas dan bawah

serta batas kanan dan batas kiri dari titik penanda. Perlu dicatat bahwa di dalam *window* ini terdapat area *cortical bone*. *Window* paling atas menginstruksikan kepada pengguna untuk memilih ROI pada *window* bagian tengah dengan ukuran sesuai kebutuhan. Perlu dicatat bahwa pemilihan ROI harus sesuai dengan batasan yang telah ditetapkan pada bagian 1.3.



Gambar 5.9 Hasil Penandaan Titik Area Cortical Bone

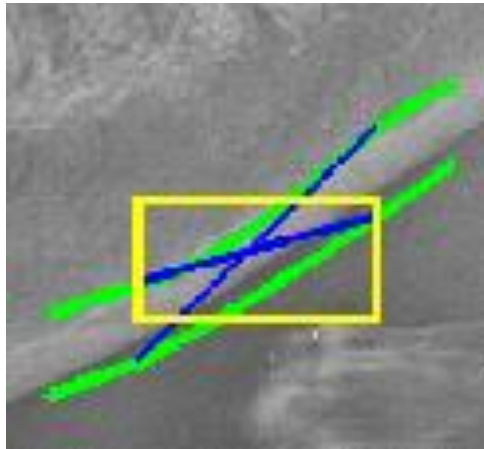


Gambar 5.10 Pemilihan ROI dari Area Batasan

Gambar 5.10 adalah pemilihan area ROI oleh pengguna. Ukuran dan rasio dari ROI mengikuti kebutuhan dari pengguna. Hasil dari tahap ini adalah lokasi dan ukuran ROI serta titik-titik penanda yang dipilih oleh pengguna untuk kemudian digunakan pada tahap berikutnya.

5.4.2 *Pre-processing*

Berdasarkan keluaran dari tahap sebelumnya, yaitu lokasi dan ukuran ROI serta titik-titik penanda, dilakukan perhitungan parameter. Parameter yang dihitung adalah sudut batas orientasi dan ketebalan sementara dari *cortical bone* (lihat bagian 3.2.4.3 dan bagian 3.2.4.4).

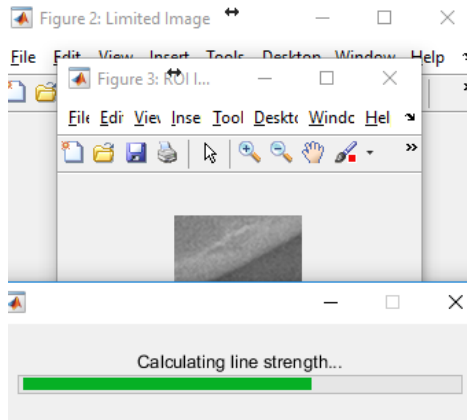


Gambar 5.11 Penampakan Hasil Perhitungan Parameter

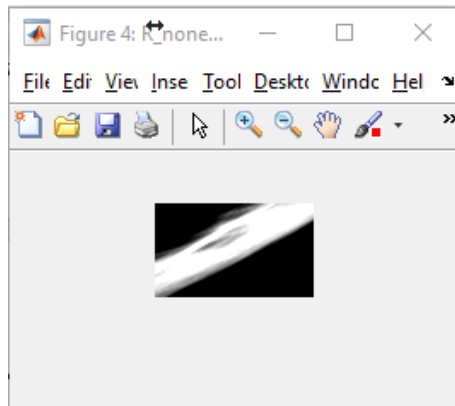
Gambar 5.11 adalah tampilan dari parameter yang telah dihitung. Garis berwarna hijau adalah hasil interpolasi dari titik-titik masukan dari pengguna. Garis berwarna biru adalah garis batas orientasi. Dan garis berwarna kuning adalah area ROI yang dipilih oleh pengguna. Parameter ini yang akan digunakan pada tahap perhitungan *line strength*.

5.4.3 *Line Strength*

Berdasarkan parameter yang telah dihitung pada proses sebelumnya, dilakukan perhitungan *line strength*. Pada proses perhitungan ini, akan muncul sebuah *window* yang berisi *loading bar* yang memberitahukan proses yang sedang dikerjakan dalam proses perhitungan *line strength* seperti pada **Gambar 5.12**.



Gambar 5.12 *Loading Bar Perhitungan Line Strength*

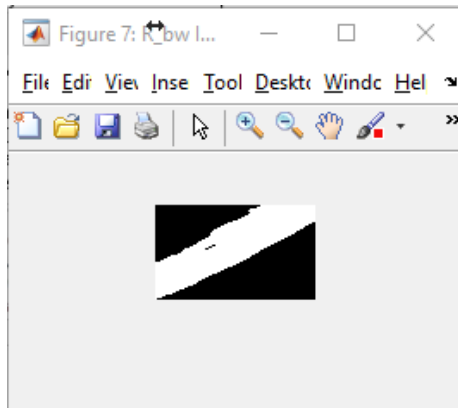


Gambar 5.13 *Window dari Nilai Line Strength*

Apabila telah selesai, sistem akan menghilangkan *loading bar* dan akan menampilkan nilai *line strength* yang telah dipetakan ke rentang nilai *grayscale* seperti pada **Gambar 5.13**.

5.4.4 *Segmentation*

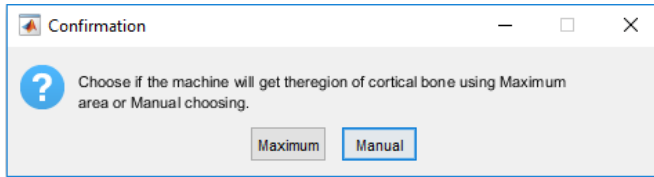
Hasil dari perhitungan *line strength* kemudian dinormalkan seperti pada **Gambar 5.13** yang kemudian disegmentasi dengan menggunakan *Otsu's thresholding*. Hasil dari *thresholding* ini tampak pada **Gambar 5.14**.



Gambar 5.14 Hasil Segmentasi dengan *Otsu Thresholding*

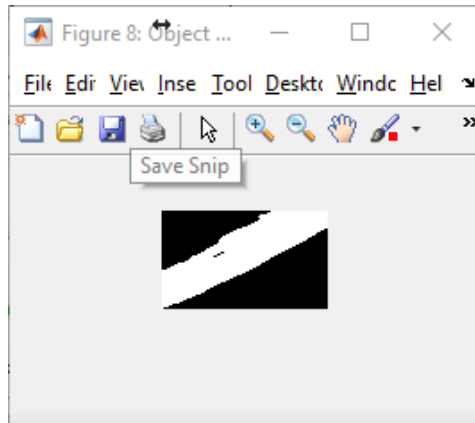
5.4.5 *Post-processing*

Proses ini dilakukan setelah proses segmentasi dilakukan. Tahap ini diawali dengan keluarnya *window* konfirmasi menanyakan *post-processing* yang akan digunakan oleh pengguna. Terdapat dua pilihan, yaitu *Maximum*, sistem memilih area paling maksimal yang dianggap sebagai *cortical bone*, dan *Manual*, pengguna memilih satu titik pada area yang dianggap *cortical bone*. Pilihan ini tampak pada **Gambar 5.15**.



Gambar 5.15 Window Konfirmasi Post-processing

Hasil dari proses ini tampak pada **Gambar 5.16**. Tampak bahwa tidak terlihat perubahan yang signifikan dari **Gambar 5.14** dan **Gambar 5.16** (hasil dari tahap ini). Hal ini karena terdapat sedikit *noise* dari hasil segmentasi yang dihilangkan pada proses ini.



Gambar 5.16 Citra Hasil Post-processing

5.5 Uji Coba 1: Orientasi Garis

Uji coba ini dilakukan untuk mengetahui pengaruh antara penggunaan semua orientasi garis, seperti usulan oleh Nguyen, dan orientasi garis terpilih usulan tugas akhir ini pada operator *moving window* (lihat bagian 3.2.5). Hipotesis pada uji coba ini adalah metode usulan memberikan nilai F_1 score lebih tinggi dibandingkan penggunaan semua orientasi seperti diusulkan oleh usulan Nguyen [7].

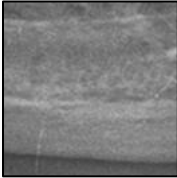


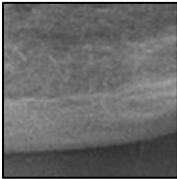


5.5.1 Metode Orientasi Nguyen

Metode usulan Nguyen menggunakan semua orientasi yaitu 12 orientasi garis. Penggunaan metode ini efektif pada citra *retinal blood vessel* sebab orientasi garis dari pembuluh darah beragam. Hasil terbaik dan terburuk, untuk ROI kiri dan ROI kanan, dari uji coba ini dipaparkan pada bagian berikutnya. Selain itu, dipaparkan pula hasil dari penggunaan metode usulan pada citra dan ROI yang sama sebagai pembandingan.

5.5.1.1 Hasil F_1 Score Terendah ROI Kiri

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score terendah berdasarkan ROI kiri untuk metode Nguyen. Pada citra yang sama, ROI bagian kanan juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode usulan sebagai pembandingan.

Tabel 5.1 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Nguyen (Kiri) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

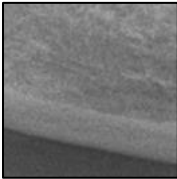


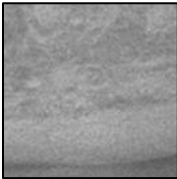


Tabel 5.1 memaparkan hasil segmentasi dengan nilai F_1 score terendah pada ROI kiri dengan menggunakan metode Nguyen.

5.5.1.2 Hasil F_1 Score Terendah ROI Kanan

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score terendah berdasarkan ROI kanan untuk metode Nguyen. Pada citra yang sama, ROI bagian kiri juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode usulan sebagai pembandingan.

Tabel 5.2 memaparkan hasil segmentasi dengan nilai F_1 score terendah pada ROI kiri dengan menggunakan metode Nguyen.

Tabel 5.2 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Nguyen (Kanan) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

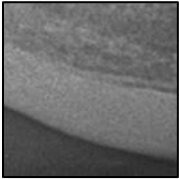


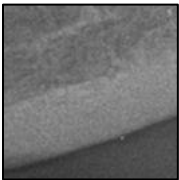


5.5.1.3 Hasil F_1 Score Tertinggi ROI Kiri

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score tertinggi berdasarkan ROI kiri untuk metode Nguyen. Pada citra yang sama, ROI bagian kanan juga dipaparkan. Selain itu, dipapar-

kan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode usulan sebagai pembandingan.

Tabel 5.3 memaparkan hasil segmentasi dengan nilai F_1 score tertinggi pada ROI kiri dengan menggunakan metode Nguyen.

Tabel 5.3 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Nguyen (Kiri) Skala 1,00

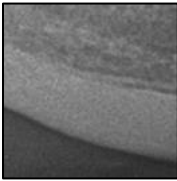


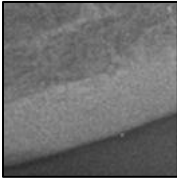


Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

5.5.1.4 Hasil F_1 Score Tertinggi ROI Kanan

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score tertinggi berdasarkan ROI kanan untuk metode Nguyen. Pada citra yang sama, ROI bagian kiri juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode usulan sebagai pembandingan.

Tabel 5.4 memaparkan hasil segmentasi dengan nilai F_1 score tertinggi pada ROI kanan dengan menggunakan metode Nguyen.

Tabel 5.4 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Nguyen (Kanan) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

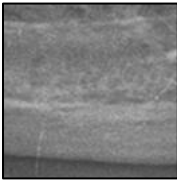


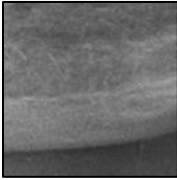


5.5.2 Metode Orientasi Usulan

Metode usulan menggunakan orientasi terpilih berdasarkan sudut batas orientasi θ_1 dan θ_2 (lihat bagian 3.2.5). Hasil terbaik dan terburuk, untuk ROI kiri dan ROI kanan, dari uji coba ini dipaparkan pada bagian berikutnya. Selain itu, dipaparkan pula hasil dari penggunaan metode yang diusulkan oleh Nguyen pada citra dan ROI yang sama sebagai pembandingan.

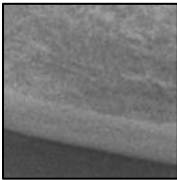


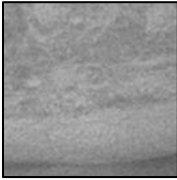


5.5.2.1 Hasil F_1 Score Terendah ROI Kiri

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score terendah berdasarkan ROI kiri untuk metode usulan. Pada citra yang sama, ROI bagian kanan juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode Nguyen sebagai pembandingan. **Tabel 5.5** memaparkan hasil segmentasi dengan nilai F_1 score terendah pada ROI kiri dengan menggunakan metode usulan.

Tabel 5.5 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Usulan (Kiri) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.6 Hasil Segmentasi dengan F_1 score Terendah Menggunakan Metode Usulan (Kanan) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

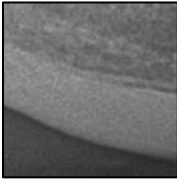


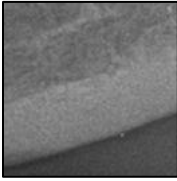


5.5.2.2 Hasil F_1 Score Terendah ROI Kanan

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score terendah berdasarkan ROI kanan untuk metode usulan. Pada citra yang sama, ROI bagian kiri juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode Nguyen sebagai pembanding. **Tabel 5.6** memaparkan hasil segmentasi dengan nilai F_1 score terendah pada ROI kanan dengan menggunakan metode usulan.

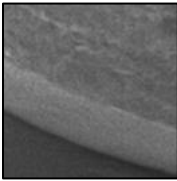


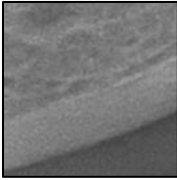


5.5.2.3 Hasil F_1 Score Tertinggi ROI Kiri

Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score tertinggi berdasarkan ROI kiri untuk metode usulan. Pada citra yang sama, ROI bagian kanan juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode Nguyen sebagai pembanding. **Tabel 5.7** memaparkan hasil segmentasi dengan nilai F_1 score tertinggi pada ROI kiri dengan menggunakan metode usulan.

Tabel 5.7 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Usulan (Kiri) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.8 Hasil Segmentasi dengan F_1 score Tertinggi Menggunakan Metode Usulan (Kanan) Skala 1,00

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

5.5.2.4 Hasil F_1 Score Tertinggi ROI Kanan

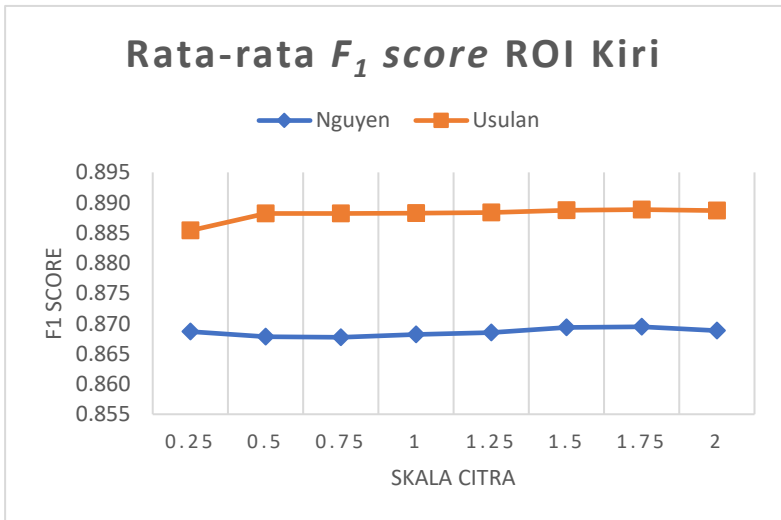
Bagian ini memaparkan hasil segmentasi dengan nilai F_1 score tertinggi berdasarkan ROI kanan untuk metode usulan. Pada citra yang sama, ROI bagian kiri juga dipaparkan. Selain itu, dipaparkan pula hasil segmentasi pada citra dan ROI yang sama menggunakan metode Nguyen sebagai pembandingan. **Tabel 5.8** memaparkan hasil segmentasi dengan nilai F_1 score tertinggi pada ROI kanan dengan menggunakan metode usulan.

5.5.3 Hasil Rata-rata F_1 score Metode Nguyen dan Usulan

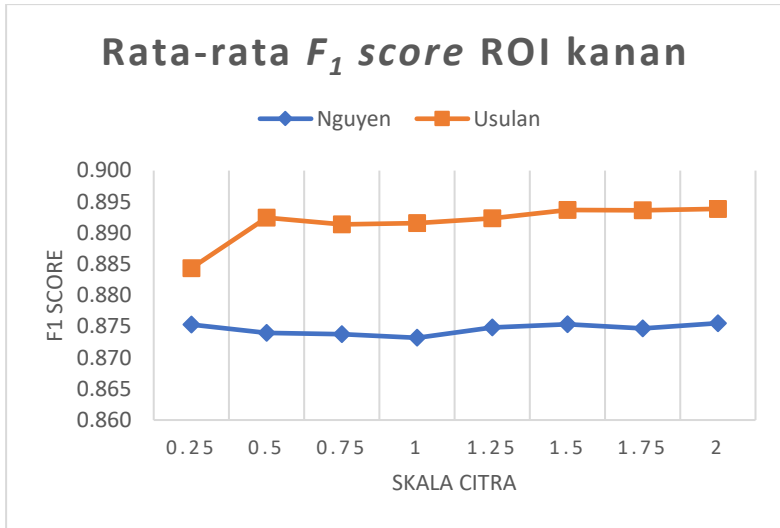
Tabel 5.9 menunjukkan rata-rata F_1 score dari penggunaan garis orientasi dengan metode Nguyen dan metod usulan. Tampak bahwa untuk ROI yang sama pada setiap skala, penggunaan garis orientasi dengan metode usulan memberikan F_1 score lebih baik (lihat **Grafik 5.1** dan **Grafik 5.2**) dibandingkan penggunaan metode Nguyen. Hal ini sesuai dengan hipotesis yang telah ditentukan.

Tabel 5.9 Rata-rata F_1 score Metode Nguyen dan Usulan

Skala Citra	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
0,25	0,869	0,885	0,875	0,884
0,50	0,868	0,888	0,874	0,892
0,75	0,868	0,888	0,874	0,891
1,00	0,868	0,888	0,873	0,892
1,25	0,869	0,888	0,875	0,892
1,50	0,869	0,889	0,875	0,894
1,75	0,869	0,889	0,875	0,894
2,00	0,869	0,889	0,876	0,894
Rata-rata	0,869	0,888	0,875	0,892



Grafik 5.1 Rata-rata F_1 score ROI Kiri (Uji Coba 1)



Grafik 5.2 Rata-rata F_1 score ROI Kanan (Uji Coba 1)

5.6 Uji Coba 2: Teknik Kombinasi

Uji coba dilakukan untuk mengetahui pengaruh penambahan nilai asli *grayscale* pada tahap kombinasi perhitungan *line strength R*. Hipotesis pada uji coba ini adalah penambahan nilai asli *grayscale* pada tahap kombinasi membuat nilai *line strength R* tidak signifikan. Dengan kata lain, F_1 score hasil segmentasi akan lebih rendah jika ditambahkan nilai asli *grayscale* pada kombinasi nilai *line strength R* (lihat bagian 2.12).

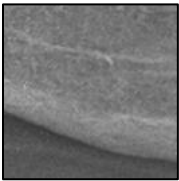
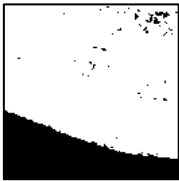

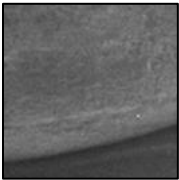
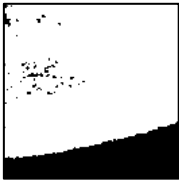

Data dan ROI pada uji coba telah dijelaskan pada bagian 3.2.2. Digunakan dua metode pada uji coba ini, yaitu metode penambahan intensitas asli *grayscale* sesuai usulan Nguyen pada persamaan (2.13) dan metode usulan tanpa penambahan intensitas asli *grayscale* sesuai persamaan (3.27).

5.6.1 Penambahan Nilai *Grayscale* Metode Nguyen

Penambahan nilai asli *grayscale* dilakukan seperti pada persamaan (2.13). Persamaan tersebut sesuai dengan persamaan yang diusulkan pada penelitian [7]. Bagian ini menampilkan hasil dari implementasi persamaan tersebut.

Tabel 5.10 memaparkan hasil dengan F_1 score terendah untuk ROI kiri dari seluruh citra menggunakan metode Nguyen. Disertakan pula hasil untuk ROI bagian kanan menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode usulan.

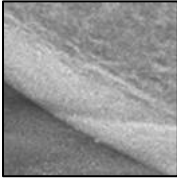


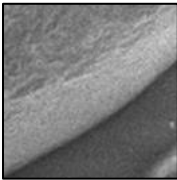


Tabel 5.10 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kiri Menggunakan Metode Nguyen (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.11 memaparkan hasil dengan F_1 score tertinggi untuk ROI kiri dari seluruh citra menggunakan metode Nguyen. Disertakan pula hasil untuk ROI bagian kanan menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi

pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode usulan.

Tabel 5.11 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kiri Menggunakan Metode Nguyen (Skala 1,00)

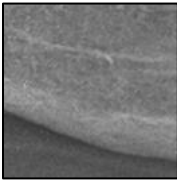
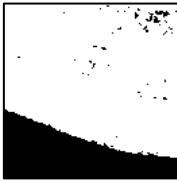

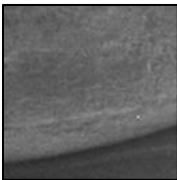
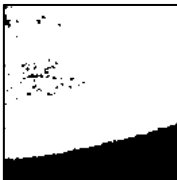

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.12 memaparkan nilai F_1 score sistem dalam melakukan segmentasi sesuai dengan hasil yang telah dipaparkan pada **Tabel 5.10** dan pada **Tabel 5.11**. Nilai F_1 score pada tabel adalah nilai terburuk dan terbaik dari seluruh citra untuk ROI bagian kiri menggunakan Nguyen. Selain itu, dipaparkan pula nilai F_1 score menggunakan metode usulan sebagai pembandingan.

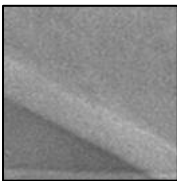
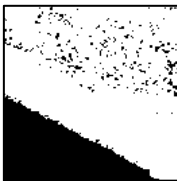

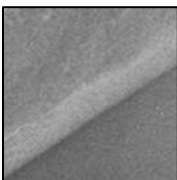


Tabel 5.12 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Nguyen pada ROI Kiri (Skala 1,00)

F_1 score	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
Terburuk	0,410	0,851	0,368	0,801
Terbaik	0,824	0,853	0,699	0,887

Tabel 5.13 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kanan Menggunakan Metode Nguyen (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.14 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kanan Menggunakan Metode Nguyen (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.13 memaparkan hasil dengan F_1 score terendah untuk ROI kanan dari seluruh citra menggunakan metode Nguyen. Disertakan pula hasil untuk ROI bagian kiri menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode usulan.

Tabel 5.14 memaparkan hasil dengan F_1 score tertinggi untuk ROI kanan dari seluruh citra menggunakan metode Nguyen. Disertakan pula hasil untuk ROI bagian kiri menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode usulan.

Tabel 5.15 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Nguyen pada ROI Kanan (Skala 1,00)

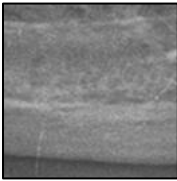
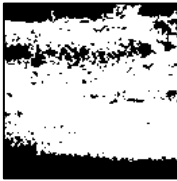

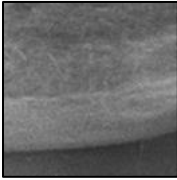


F_1 score	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
Terburuk	0,4103	0,851	0,368	0,801
Terbaik	0,454	0,894	0,704	0,893

Tabel 5.15 memaparkan nilai F_1 score sistem dalam melakukan segmentasi sesuai dengan hasil yang telah dipaparkan pada **Tabel 5.13** dan pada **Tabel 5.14**. Nilai F_1 score pada tabel adalah nilai terendah dan tertinggi dari seluruh citra untuk ROI bagian kanan menggunakan Nguyen. Selain itu, dipaparkan pula hasil menggunakan usulan sebagai pembandingan.

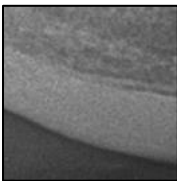
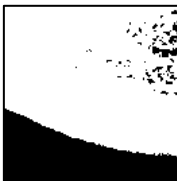

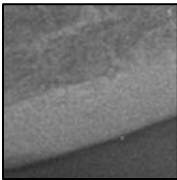


5.6.2 Tanpa Penambahan Nilai Grayscale Metode Usulan

Sebagai pembandingan untuk kasus pada bagian 5.6.1, persamaan (3.27) digunakan. Persamaan tersebut adalah usulan pada tugas akhir ini berdasarkan usulan oleh Nguyen [7]. Bagian ini memaparkan hasil penggunaan metode usulan. Sebagai pembandingan, metode usulan oleh Nguyen juga disertakan.

Tabel 5.16 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kiri Menggunakan Metode Usulan (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.17 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kiri Menggunakan Metode Usulan (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.16 memaparkan hasil dengan F_1 score terendah untuk ROI kiri dari seluruh citra menggunakan metode usulan. Disertakan pula hasil untuk ROI bagian kanan menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode Nguyen.

Tabel 5.17 memaparkan hasil dengan F_1 score tertinggi untuk ROI kiri dari seluruh citra menggunakan metode usulan. Disertakan pula hasil untuk ROI bagian kanan menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode Nguyen.

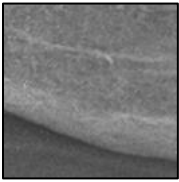
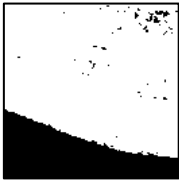

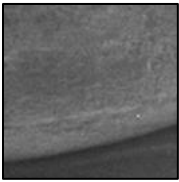
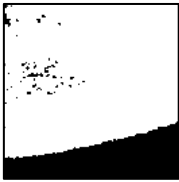

Tabel 5.18 memaparkan nilai F_1 score dari sistem dalam melakukan segmentasi sesuai dengan hasil yang telah dipaparkan pada **Tabel 5.16** dan pada **Tabel 5.17**. Nilai F_1 score pada tabel adalah nilai terburuk dan terbaik dari seluruh citra untuk ROI kiri menggunakan usulan. Selain itu, dipaparkan pula hasil menggunakan Nguyen sebagai pembandingan.

Tabel 5.18 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Usulan pada ROI Kiri (Skala 1,00)

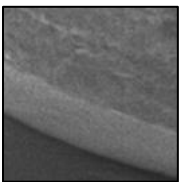
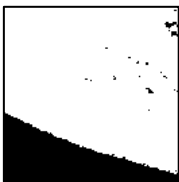

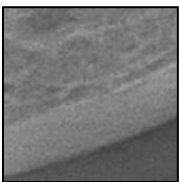


F_1 score	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
Terburuk	0,550	0,768	0,580	0,883
Terbaik	0,635	0,945	0,624	0,933

Tabel 5.19 memaparkan hasil dengan F_1 score terendah untuk ROI kanan dari seluruh citra menggunakan metode usulan. Disertakan pula hasil untuk ROI bagian kiri menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode Nguyen.

Tabel 5.19 Hasil Segmentasi dengan F_1 score Terendah pada ROI Kanan Menggunakan Metode Usulan (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.20 Hasil Segmentasi dengan F_1 score Tertinggi pada ROI Kanan Menggunakan Metode Usulan (Skala 1,00)

Bagian	ROI	Nguyen	Usulan
Kiri			
Kanan			

Tabel 5.20 memaparkan hasil dengan F_1 score tertinggi untuk ROI kanan dari seluruh citra menggunakan metode usulan. Disertakan pula hasil untuk ROI bagian kiri menggunakan metode yang sama. Selain itu, sebagai pembandingan, disertakan hasil segmentasi pada ROI yang sama, kanan dan kiri, pada citra yang sama menggunakan metode Nguyen.

Tabel 5.21 memaparkan nilai F_1 score dari sistem dalam melakukan segmentasi sesuai dengan hasil yang telah dipaparkan pada **Tabel 5.19** dan pada **Tabel 5.19**. Nilai F_1 score pada tabel adalah nilai terburuk dan terbaik dari seluruh citra untuk ROI bagian kanan menggunakan metode usulan. Selain itu, dipaparkan pula hasil menggunakan metode Nguyen sebagai pembandingan.

Tabel 5.21 Nilai F_1 score Terendah dan Tertinggi Berdasarkan Metode Usulan pada ROI Kanan (Skala 1,00)

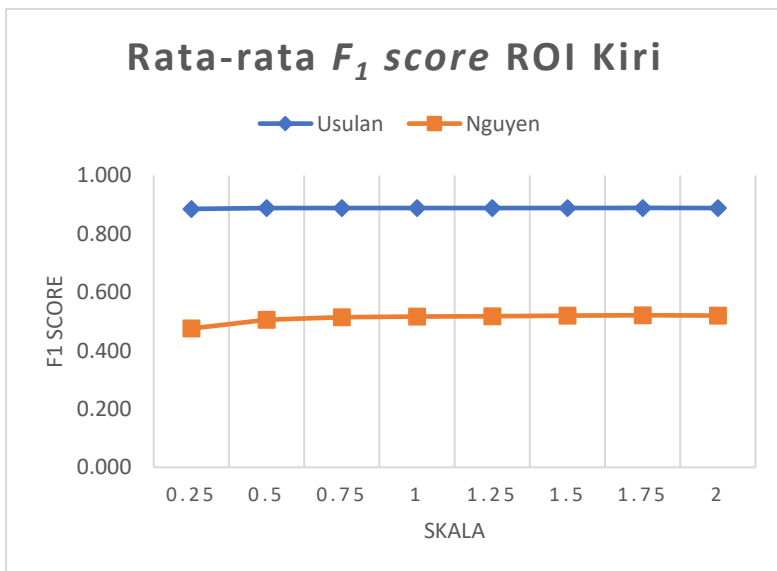
F_1 score	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
Terburuk	0,4103	0,851	0,368	0,801
Terbaik	0,501	0,924	0,544	0,935

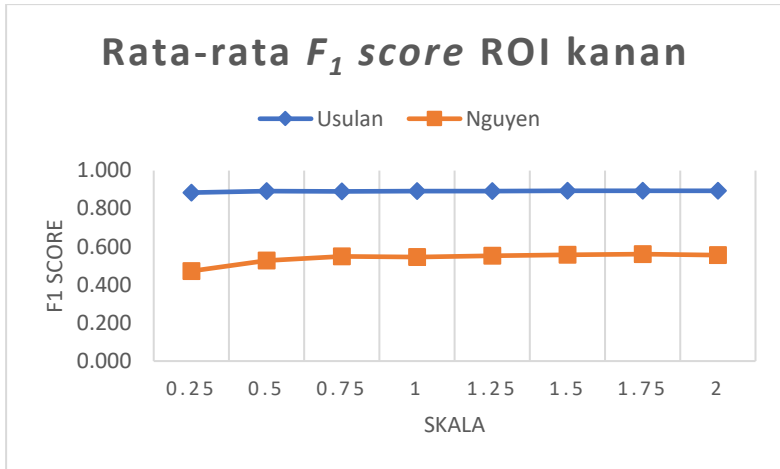
Tabel 5.22 memaparkan rata-rata nilai F_1 score hasil segmentasi pada seluruh citra pada semua skala. Dipaparkan pula hasil ini dalam bentuk grafik (lihat **Grafik 5.3** dan **Grafik 5.4**). Berdasarkan dari hasil yang telah dipaparkan, tampak bahwa penggunaan metode usulan memberikan nilai F_1 score yang lebih baik dibandingkan dengan metode yang diusulkan oleh Nguyen. Dan hal ini sesuai dengan hipotesis yang telah ditetapkan.

Berdasarkan pernyataan pada paragraf sebelumnya, dapat disimpulkan bahwa penambahan intensitas asli *grayscale* pada tahap kombinasi membuat nilai *line strength* menjadi tidak signifikan. Hal ini ditunjukkan oleh nilai F_1 score yang lebih kecil dibandingkan tanpa penambahan.

Tabel 5.22 Rata-rata F_1 score Seluruh Skala

Skala	ROI Kiri		ROI Kanan	
	Nguyen	Usulan	Nguyen	Usulan
0,25	0,476	0,885	0,472	0,884
0,50	0,505	0,888	0,528	0,892
0,75	0,514	0,888	0,549	0,891
1,00	0,516	0,888	0,556	0,891
1,25	0,517	0,888	0,557	0,894
1,50	0,520	0,889	0,557	0,894
1,75	0,521	0,889	0,561	0,894
2,00	0,520	0,889	0,557	0,894
Rata-rata	0,511	0,888	0,540	0,892

Grafik 5.3 Rata-rata F_1 score ROI Kiri (Uji Coba 2)



Grafik 5.4 Rata-rata F_1 score ROI Kanan (Uji Coba 2)

5.7 Uji Coba 3: Teknik Normalisasi

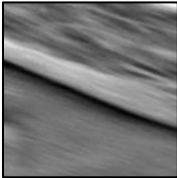
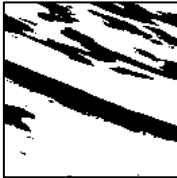
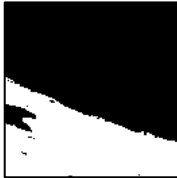
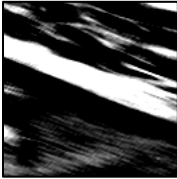


Uji coba yang dilakukan adalah untuk menentukan teknik normalisasi yang memetakan nilai *line strength* ke rentang *grayscale* dengan nilai F_1 score tertinggi. Hipotesis untuk percobaan ini adalah penggunaan teknik normalisasi yang memetakan nilai *line strength* ke rentang nilai *grayscale* dengan memerhatikan kecenderungan masing-masing nilainya memberikan nilai F_1 score lebih tinggi dibandingkan tanpa memerhatikan kecenderungan ini. Kecenderungan ini dijelaskan pada bagian 2.10.

Sebelum ditentukan segmentasi menggunakan *Otsu's thresholding* (lihat bagian 3.2.6), nilai *line strength* perlu dipetakan terlebih dahulu ke rentang *grayscale* dengan teknik normalisasi. Pada uji coba ini, digunakan dua teknik normalisasi, yaitu normalisasi *min-max* sesuai dengan persamaan xxx dan normalisasi usulan yang usulan pada tugas akhir ini sesuai dengan persamaan xxx.

Tabel 5.23 Rata-rata Nilai F_1 score Terendah Berdasarkan Normalisasi Min-max Skala 1,00

Normalisasi	<i>Min-max</i>	Usulan
F_1 score	0,438	0,849

Tabel 5.24 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Min-max pada ROI Kiri Skala 1,00

Metode	<i>Line Strength</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan			

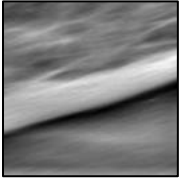


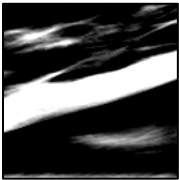
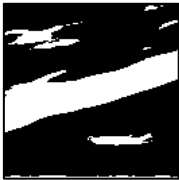

5.7.1 Metode *Min-max* dengan F_1 score Terendah

Tabel 5.23 memaparkan nilai F_1 score terendah menggunakan normalisasi *min-max*. Dipaparkan pula nilai F_1 score hasil normalisasi pada citra dan ROI yang sama menggunakan normalisasi usulan sebagai pembanding.

Tabel 5.24 dan **Tabel 5.25** memaparkan hasil perhitungan *line strength R* pada citra dan ROI sesuai dengan **Tabel 5.23** untuk ROI bagian kiri dan kanan yang dinormalkan menggunakan kedua metode normalisasi (kolom *Line Strength*). Dipaparkan pula hasil binerisasi dan hasil segmentasi untuk setiap metode normalisasi

seperti tampak pada kolom *Black & White* dan *Segmented* berturut-turut.

Tabel 5.25 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Min-max pada ROI Kanan Skala 1,00

Metode	<i>Line Strength</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan			

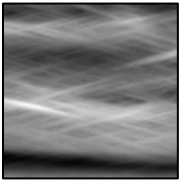


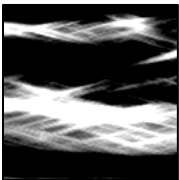


5.7.2 Metode Usulan dengan F_1 score Terendah

Tabel 5.26 memaparkan nilai F_1 score terendah menggunakan teknik normalisasi usulan. Dipaparkan pula nilai F_1 score hasil normalisasi pada citra dan ROI yang sama menggunakan teknik normalisasi *min-max* sebagai pembandingan.

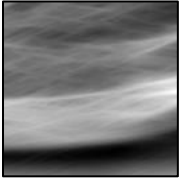


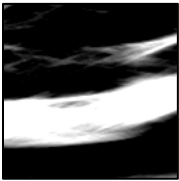


Tabel 5.26 Rata-rata Nilai F_1 score Terendah Berdasarkan Normalisasi Usulan Skala 1,00

Normalisasi	<i>Min-max</i>	Usulan
F_1 score	0,838	0,826

Tabel 5.27 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Usulan pada ROI Kiri Skala 1,00

Metode	<i>Line Strength</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan			

Tabel 5.28 Hasil Segmentasi dengan F_1 score Terendah Berdasarkan Normalisasi Usulan pada ROI Kanan Skala 1,00

Metode	<i>Line Strength</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan			

Tabel 5.27 dan **Tabel 5.28** memaparkan hasil perhitungan *line strength R* pada citra dan ROI sesuai dengan **Tabel 5.26** untuk ROI bagian kiri dan kanan yang dinormalkan menggunakan kedua metode normalisasi (kolom *Line Strength*). Dipaparkan pula binerisasi dan hasil segmentasi untuk setiap metode normalisasi seperti tampak pada kolom *Black & White* dan *Segmented* berturut-turut.

5.7.3 Metode *Min-max* dan Usulan dengan F_1 score Tertinggi

Bagian ini memaparkan hasil segmentasi dari masing-masing metode normalisasi yang menghasilkan F_1 score tertinggi. Pemaparan ini digabung menjadi satu disebabkan karena secara kebetulan, hasil terbaik dari kedua teknik berada pada citra dan ROI yang sama.

Tabel 5.29 memaparkan nilai F_1 score terbaik hasil masing-masing metode normalisasi. Hasil ini dipaparkan pada satu tabel karena secara kebetulan nilai F_1 score terbaik dari masing-masing metode terdapat pada citra dan ROI yang sama.

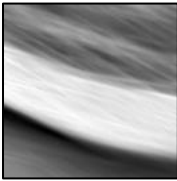





Tabel 5.29 Rata-rata Nilai F_1 score Tertinggi Kedua Metode Normalisasi Skala 1,00

Normalisasi	<i>Min-max</i>	Usulan
F_1 score	0,936	0,939

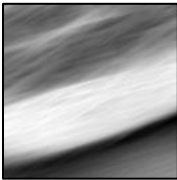


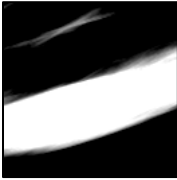


Tabel 5.30 dan **Tabel 5.31** memaparkan hasil perhitungan *line strength R* pada citra dan ROI sesuai dengan **Tabel 5.29** untuk ROI bagian kiri dan kanan yang dinormalkan menggunakan kedua metode normalisasi (kolom *Line Strength*). Dipaparkan pula binerisasi dan hasil segmentasi untuk setiap metode normalisasi seperti tampak pada kolom *Black & White* dan *Segmented* berturut-turut.

Tabel 5.32 adalah rata-rata F_1 score seluruh data untuk setiap metode yang digunakan pada ROI bagian kiri dari citra. Dipaparkan nilai F_1 score hasil segmentasi pada masing-masing skala. Selain itu, rata-rata dari setiap metode untuk seluruh skala dipaparkan.

Tabel 5.30 Hasil Segmentasi dengan F_1 score Tertinggi Kedua Metode Normalisasi pada ROI Kiri Skala 1,00

Metode	<i>R</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan			

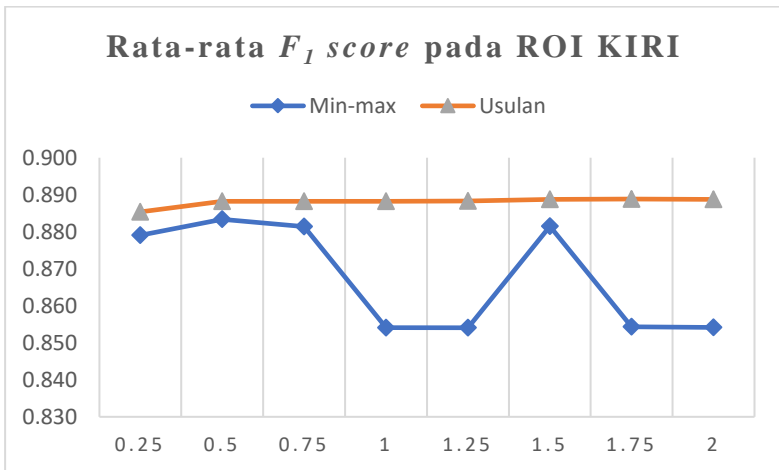
Tabel 5.31 Hasil Segmentasi dengan F_1 score Tertinggi Ketiga Metode Normalisasi pada ROI Kanan Skala 1,00

Metode	<i>R</i>	<i>Black & White</i>	<i>Segmented</i>
<i>Min-max</i>			
Usulan 2			

Tabel 5.32 Rata-rata Nilai F_1 score Seluruh Data ROI Kiri

Skala	Min-max	Usulan
0,25	0,879	0,885
0,50	0,883	0,888
0,75	0,881	0,888
1,00	0,854	0,888
1,25	0,854	0,888
1,50	0,881	0,889
1,75	0,854	0,889
2,00	0,854	0,889
Rata-rata	0,868	0,888

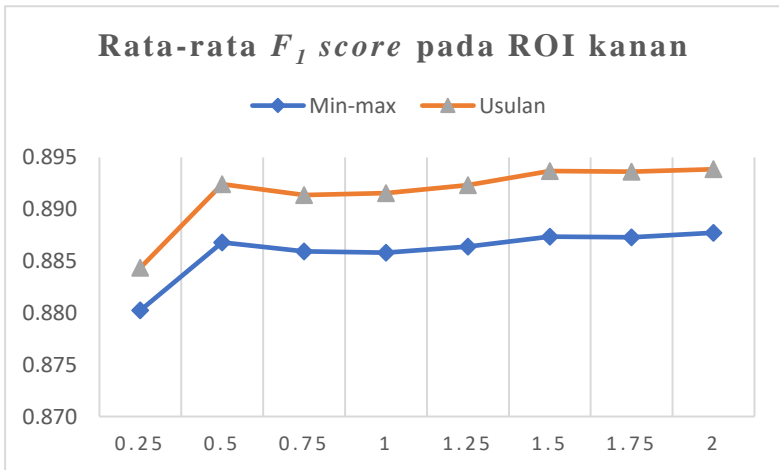
Grafik 5.5 menunjukkan perbandingan nilai F_1 score kedua metode untuk ROI pada bagian kiri dari semua citra pada semua skala. Tampak bahwa metode usulan memberikan nilai F_1 score lebih tinggi dibandingkan metode lain.



Grafik 5.5 Rata-rata F_1 score ROI Kiri (Uji Coba 3)

Tabel 5.33 Rata-rata F_1 score Seluruh Data ROI Kanan

Skala	Min-max	Usulan
0,25	0,879	0,885
0,50	0,883	0,888
0,75	0,881	0,888
1,00	0,854	0,888
1,25	0,854	0,888
1,50	0,881	0,889
1,75	0,854	0,889
2,00	0,854	0,889
Rata-rata	0,886	0,892



Grafik 5.6 Rata-rata F_1 score ROI Kanan (Uji Coba 3)

Tabel 5.33 adalah rata-rata F_1 score seluruh data untuk setiap metode yang digunakan pada ROI bagian kanan dari citra. Dipaparkan nilai F_1 score hasil segmentasi pada masing-masing skala.

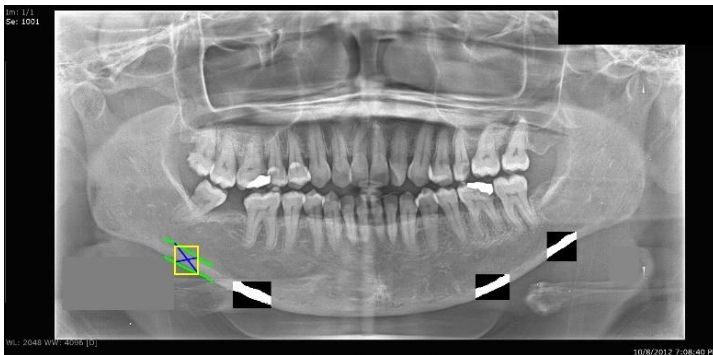
Selain itu, rata-rata dari setiap metode untuk seluruh skala dipaparkan.

Grafik 5.6 menunjukkan perbandingan nilai F_1 score ketiga metode untuk ROI pada bagian kanan dari semua citra pada semua skala. Tampak bahwa metode usulan memberikan nilai F_1 score lebih tinggi dibandingkan metode lain. Hal ini sesuai dengan hipotesis yang telah ditetapkan.

Berdasarkan hasil pada **Tabel 5.32**, **Tabel 5.33**, dan hipotesis yang telah ditetapkan, dapat disimpulkan bahwa penggunaan pemetaan nilai *line strength* dengan memerhatikan kecenderungan masing-masing nilai memberikan nilai F_1 score dibandingkan normalisasi tanpa memerhatikan kecenderungan ini.

5.8 Uji Coba 4: Lokasi dan Ukuran ROI

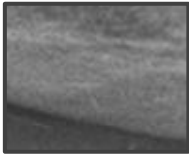

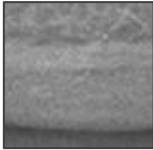

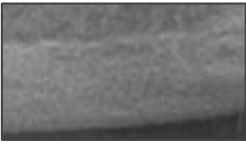

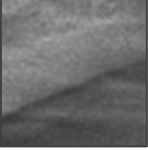

Uji coba yang dilakukan adalah untuk mengetahui F_1 score sistem terhadap kondisi ukuran ROI yang berbeda. Hipotesis pada percobaan ini adalah sistem memberikan F_1 score yang stabil untuk ukuran dan lokasi ROI yang berbeda. Dalam tugas akhir ini, dikatakan stabil apabila nilai standard deviasi F_1 score sistem berada di bawah 5% atau 0,05. Hal ini menyesuaikan dengan penelitian [28] dan [29].



Gambar 5.17 Pemilihan Lokasi Empat ROI

ROI yang digunakan pada uji coba ini adalah ROI dari semua citra dengan setiap citra diambil empat ROI berbeda ukuran (dan rasio) dan lokasi. Lokasi ROI yaitu dua di bagian kanan dan dua di bagian kiri. **Gambar 5.17** adalah contoh pemilihan lokasi dan ukuran (dan rasio) ROI pada suatu citra. Pada bagian berikutnya, akan dipaparkan hasil segmentasi untuk citra dengan skala 1,00 saja.

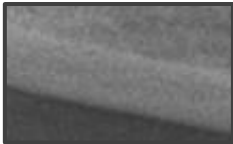





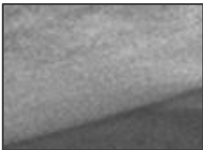

Tabel 5.34 Hasil Segmentasi dengan Nilai F_1 score Terendah pada ROI Kiri (Skala 1,00)

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.1 Segmentasi dengan F_1 score Terendah ROI Kiri

Bagian ini memaparkan hasil segmentasi citra dengan nilai F_1 score terendah pada ROI kiri yang dilakukan oleh sistem. **Tabel 5.34** adalah hasil segmentasi dengan nilai rata-rata F_1 score terendah ROI pada bagian kiri citra. Dipaparkan pula hasil segmentasi pada ROI bagian kanan dari citra yang sama.

Tabel 5.35 Hasil Segmentasi dengan Nilai F_1 score Terendah pada ROI Kanan (Skala 1,00)

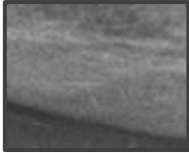

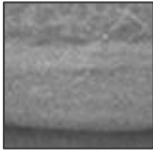

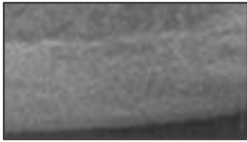

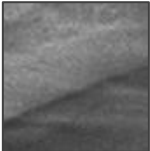

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.2 Segmentasi dengan F_1 score Terendah ROI Kanan

Bagian ini memaparkan hasil segmentasi citra dengan nilai F_1 score terendah pada ROI kanan yang dilakukan oleh sistem. **Tabel 5.35** adalah hasil segmentasi dengan nilai rata-rata F_1 score

terendah ROI pada bagian kanan citra. Dipaparkan pula hasil segmentasi pada ROI bagian kiri dari citra yang sama.



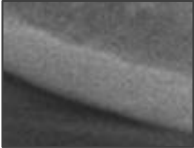

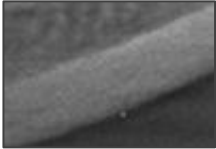

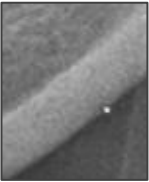

Tabel 5.36 Hasil Segmentasi dengan Nilai Rata-rata F_1 score Terendah (Skala 1,00)

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.3 Segmentasi dengan Rata-rata F_1 score Terendah

Bagian ini memaparkan hasil segmentasi citra dengan nilai rata-rata F_1 score terendah yang dilakukan oleh sistem. **Tabel 5.36** adalah hasil segmentasi dengan nilai rata-rata F_1 score terendah.

Tabel 5.37 Hasil Segmentasi dengan Nilai F_1 score Tertinggi pada ROI Kiri (Skala 1,00)

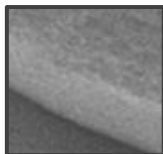



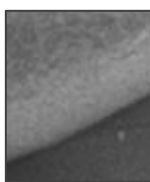

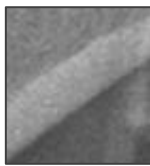

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.4 Segmentasi dengan F_1 score Tertinggi ROI Kiri

Bagian ini memaparkan hasil segmentasi citra dengan nilai F_1 score tertinggi pada ROI kiri yang dilakukan oleh sistem. **Tabel 5.37** adalah hasil segmentasi dengan nilai rata-rata F_1 score

terendah ROI pada bagian kiri citra. Dipaparkan pula hasil segmentasi pada ROI bagian kanan dari citra yang sama.

Tabel 5.38 Hasil Segmentasi dengan Nilai F_1 score Tertinggi pada ROI Kanan (Skala 1,00)

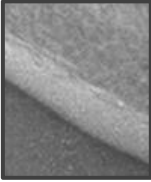

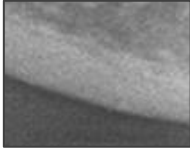

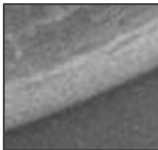

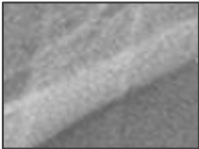

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.5 Segmentasi dengan F_1 score Tertinggi ROI Kanan

Bagian ini memaparkan hasil segmentasi citra dengan nilai F_1 score tertinggi pada ROI kanan yang dilakukan oleh sistem. **Tabel 5.38** adalah hasil segmentasi dengan nilai rata-rata F_1 score tertinggi.

gi ROI pada bagian kanan citra. Dipaparkan pula hasil segmentasi pada ROI bagian kiri dari citra yang sama.

Tabel 5.39 Hasil Segmentasi dengan Nilai Rata-rata F_1 score Tertinggi (Skala 1,00)

Lokasi ROI	ROI Awal	Hasil Segmentasi
Kiri-Kiri		
Kiri-Tengah		
Kanan-Tengah		
Kanan-Kanan		

5.8.6 Segmentasi dengan Rata-rata F_1 score Tertinggi

Bagian ini memaparkan hasil segmentasi citra dengan nilai rata-rata F_1 score tertinggi yang dilakukan oleh sistem. **Tabel 5.39** adalah hasil segmentasi dengan nilai rata-rata F_1 score terendah.

5.8.7 Hasil Rata-rata F_1 score

Telah dipaparkan F_1 score tertinggi dan terendah pada bagian sebelumnya. Bagian ini memaparkan nilai rata-rata F_1 score dari hasil segmentasi untuk setiap citra pada semua skala yang telah ditentukan (lihat bagian 3.2.2).

Tabel 5.40 Rata-rata F_1 score Seluruh Citra

Skala Citra	ROI Kiri	ROI Kanan	Rata-rata
0,25	0,847	0,852	0,850
0,50	0,849	0,858	0,853
0,75	0,850	0,858	0,854
1,00	0,850	0,858	0,854
1,25	0,850	0,859	0,854
1,50	0,851	0,859	0,855
1,75	0,851	0,859	0,855
2,00	0,851	0,859	0,855
Rata-rata	0,858	0,850	0,854

Tabel 5.40 memaparkan rata-rata F_1 score seluruh citra hasil segmentasi sesuai dengan hasil yang dipaparkan pada bagian 5.8. Dipaparkan pula F_1 score hasil segmentasi citra pada skala yang berbeda.

Tabel 5.41 memaparkan standard deviasi dari nilai F_1 score sistem untuk semua citra pada semua skala. Tampak nilai standard deviasi untuk seluruh citra pada setiap skala yang ditentukan berada di bawah nilai 5% atau 0,05. Dengan demikian, untuk citra dan skala yang telah ditentukan, sistem dikatakan stabil dan sesuai dengan hipotesis yang telah ditentukan. Dapat disimpulkan bahwa sistem mampu melakukan segmentasi pada citra dengan skala yang berbeda dengan nilai rata-rata nilai F_1 score 0,854 dan standard deviasi F_1 score di bawah 5% atau 0,05.

Tabel 5.41 Standard Deviasi F_1 score dari Seluruh Citra

Skala Citra	ROI Kiri	ROI Kanan	Rata-rata
0,25	0,038	0,039	0,039
0,50	0,037	0,038	0,038
0,75	0,037	0,037	0,037
1,00	0,038	0,037	0,037
1,25	0,038	0,037	0,037
1,50	0,038	0,037	0,037
1,75	0,038	0,037	0,037
2,00	0,038	0,037	0,037
Rata-rata	0,0375	0,0377	0,038

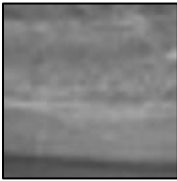


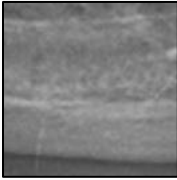


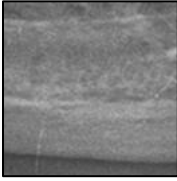


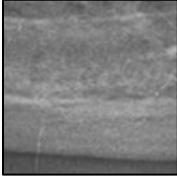


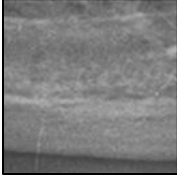


5.9 Uji Coba 5: Perbandingan Efektivitas

Uji coba yang dilakukan adalah membandingkan nilai F_1 score antara sistem yang diusulkan pada tugas akhir ini dengan sistem pada penelitian oleh Ramadhani [8] pada citra dengan ukuran skala berbeda. Hipotesis dari uji coba ini adalah sistem pada tugas akhir ini lebih stabil dibandingkan dengan sistem pada penelitian sebelumnya. Dikatakan lebih stabil apabila nilai standard deviasi nilai F_1 score sistem pada tugas akhir ini lebih rendah dibandingkan dengan standard deviasi penelitian sebelumnya.

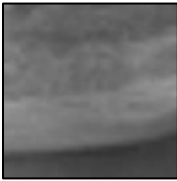


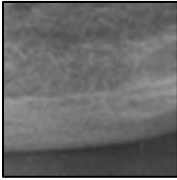


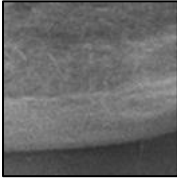


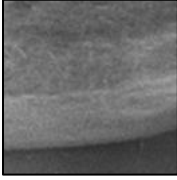


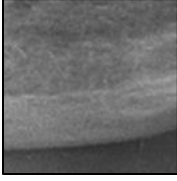


5.9.1 Segmentasi dengan Nilai F_1 score Terendah pada Sistem Ramadhani

Bagian ini memaparkan hasil segmentasi citra dengan rata-rata F_1 score terendah yang dihasilkan oleh sistem Ramadhani pada semua skala. Selain itu, dipaparkan pula hasil segmentasi citra menggunakan sistem yang diusulkan pada tugas akhir ini sebagai pembanding. Hasil yang dipaparkan pada bagian ini diperbesar atau diperkecil agar memiliki ukuran yang sama dengan ROI untuk skala 1,00.

Tabel 5.42 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Ramadhani pada ROI Kiri

Skala	<i>ROI Awal</i>	<i>Ramadhani</i>	<i>Usulan</i>
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.43 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Ramadhani pada ROI Kanan

Skala	ROI Awal	Ramadhani	Usulan
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.42 dan **Tabel 5.43** memaparkan hasil segmentasi dengan nilai F_1 score terendah dari sistem Ramadhani dilihat dari ROI kiri saja dan ROI kanan saja secara berturut-turut. Tampak bahwa hasil segmentasi dari sistem Ramadhani cenderung menghasilkan kesalahan prediksi untuk skala selain 1,00. Sedangkan untuk hasil segmentasi pada sistem yang diusulkan cenderung lebih stabil.

Tabel 5.44 Nilai F_1 score Terendah dari Sistem Ramadhani

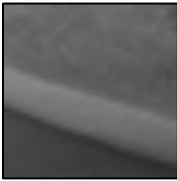
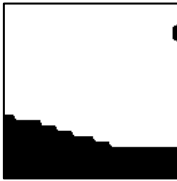

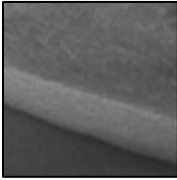


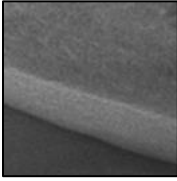


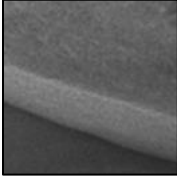


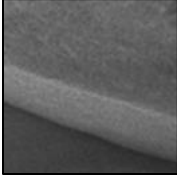


Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,548	0,641	0,767	0,886
0,50	0,828	0,818	0,752	0,886
0,75	0,886	0,848	0,773	0,882
1,00	0,869	0,882	0,786	0,883
1,25	0,604	0,887	0,769	0,884
1,50	0,507	0,756	0,769	0,885
1,75	0,325	0,675	0,768	0,883
2,00	0,306	0,613	0,758	0,883

Tabel 5.44 memaparkan nilai F_1 score hasil segmentasi dari suatu citra yang memberikan nilai F_1 score terendah berdasarkan ROI kiri dan ROI kanan.

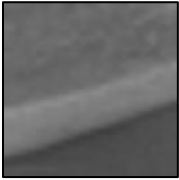
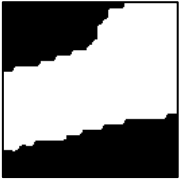

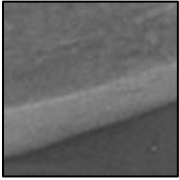


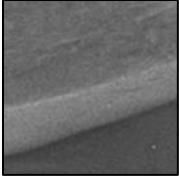


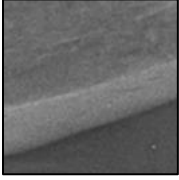


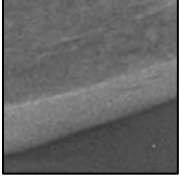


5.9.2 Segmentasi dengan Nilai F_1 score Tertinggi pada Sistem Ramadhani

Bagian ini memaparkan hasil segmentasi citra dengan rata-rata F_1 score tertinggi yang dihasilkan oleh sistem Ramadhani pada semua skala. Selain itu, dipaparkan pula hasil segmentasi citra menggunakan sistem yang diusulkan pada tugas akhir ini sebagai pembandingan.

Tabel 5.45 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Ramadhani pada ROI Kiri

Skala	<i>ROI Awal</i>	<i>Ramadhani</i>	<i>Usulan</i>
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.46 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Ramadhani pada ROI Kanan

Skala	<i>ROI Awal</i>	<i>Ramadhani</i>	<i>Usulan</i>
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.45 dan **Tabel 5.46** memaparkan hasil segmentasi dengan nilai F_1 score tertinggi dari sistem Ramadhani dilihat dari ROI kiri saja dan ROI kanan saja secara berturut-turut. Tampak bahwa hasil segmentasi dari sistem Ramadhani cenderung menghasilkan kesalahan prediksi untuk skala selain 1,00. Sedangkan untuk hasil segmentasi pada sistem yang diusulkan cenderung lebih stabil.

Tabel 5.47 Nilai F_1 score Tertinggi dari Sistem Ramadhani

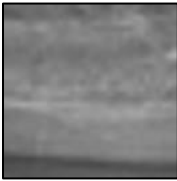


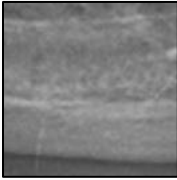


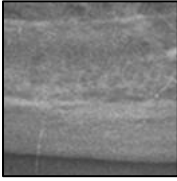


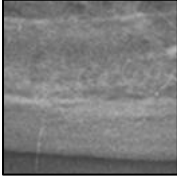


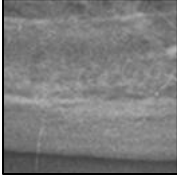


Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,405	0,583	0,892	0,895
0,50	0,842	0,856	0,887	0,911
0,75	0,943	0,909	0,883	0,911
1,00	0,924	0,912	0,884	0,910
1,25	0,929	0,938	0,885	0,911
1,50	0,956	0,957	0,885	0,911
1,75	0,955	0,928	0,885	0,911
2,00	0,838	0,810	0,885	0,911

Tabel 5.47 memaparkan nilai F_1 score hasil segmentasi dari suatu citra yang memberikan nilai F_1 score tertinggi berdasarkan ROI kiri dan ROI kanan.

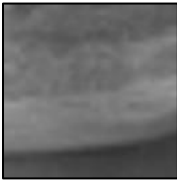


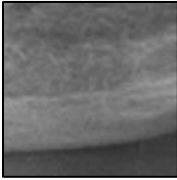


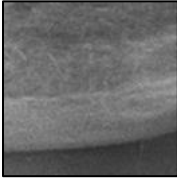


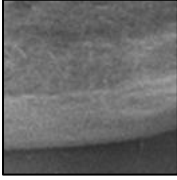


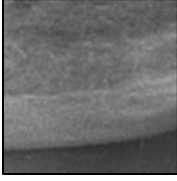


5.9.3 Segmentasi dengan Nilai F_1 score Terendah pada Sistem Usulan

Bagian ini memaparkan hasil segmentasi citra dengan rata-rata F_1 score terendah yang dihasilkan oleh sistem usulan pada semua skala. Dipaparkan pula hasil segmentasi citra menggunakan sistem oleh Ramadhani sebagai pembandingan. Hasil yang dipaparkan pada bagian ini diperbesar atau diperkecil agar memiliki ukuran yang sama dengan ROI untuk skala 1,00.

Tabel 5.48 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Usulan pada ROI Kiri

Skala	ROI Awal	Ramadhani	Usulan
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.49 Hasil Segmentasi dengan nilai F_1 score Terendah oleh Sistem Usulan pada ROI Kanan

Skala	ROI Awal	Ramadhani	Usulan
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.48 dan **Tabel 5.49** memaparkan hasil segmentasi dengan nilai F_1 score terendah dari sistem usulan dilihat dari ROI kiri saja dan ROI kanan saja secara berturut-turut. Tampak bahwa hasil segmentasi dari sistem Ramadhani cenderung menghasilkan kesalahan prediksi untuk skala selain 1,00. Sedangkan untuk hasil segmentasi pada sistem yang diusulkan cenderung lebih stabil.

Tabel 5.50 Nilai F_1 score Terendah dari Sistem Usulan

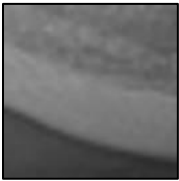
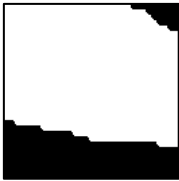

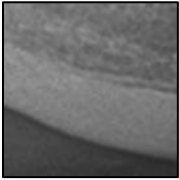


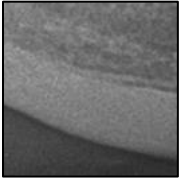


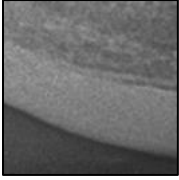


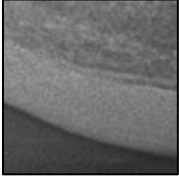


Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,548	0,641	0,767	0,886
0,50	0,828	0,818	0,752	0,886
0,75	0,886	0,848	0,773	0,882
1,00	0,869	0,882	0,786	0,883
1,25	0,604	0,887	0,769	0,884
1,50	0,507	0,756	0,769	0,885
1,75	0,325	0,675	0,768	0,883
2,00	0,306	0,613	0,758	0,883

Tabel 5.50 memaparkan nilai F_1 score hasil segmentasi dari suatu citra yang memberikan nilai F_1 score terendah berdasarkan ROI kiri dan ROI kanan.

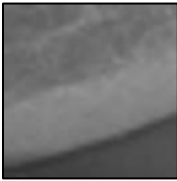


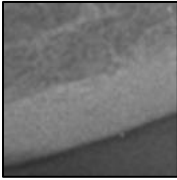


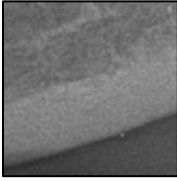


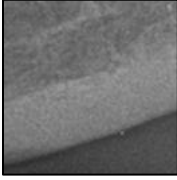


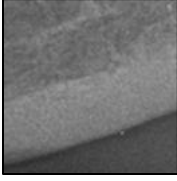


5.9.4 Segmentasi dengan Nilai F_1 score Tertinggi pada Sistem Usulan

Bagian ini memaparkan hasil segmentasi citra dengan rata-rata F_1 score tertinggi yang dihasilkan oleh sistem usulan pada semua skala. Dipaparkan pula hasil segmentasi citra menggunakan sistem oleh Ramadhani sebagai pembandingan. Hasil yang dipaparkan pada bagian ini diperbesar atau diperkecil agar memiliki ukuran yang sama dengan ROI untuk skala 1,00.

Tabel 5.51 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Usulan pada ROI Kiri

Skala	ROI Awal	Ramadhani	Usulan
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.52 Hasil Segmentasi dengan nilai F_1 score Tertinggi oleh Sistem Usulan pada ROI Kanan

Skala	<i>ROI Awal</i>	<i>Ramadhani</i>	<i>Usulan</i>
0,25			
0,50			
1,00			
1,75			
2,00			

Tabel 5.51 dan **Tabel 5.52** memaparkan hasil segmentasi dengan nilai F_1 score tertinggi dari sistem usulan dilihat dari ROI kiri saja dan ROI kanan saja secara berturut-turut. Tampak bahwa hasil segmentasi dari sistem Ramadhani cenderung menghasilkan kesalahan prediksi untuk skala selain 1,00. Sedangkan untuk hasil segmentasi pada sistem yang diusulkan cenderung lebih stabil.

Tabel 5.53 Nilai F_1 score Tertinggi dari Sistem Usulan

Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,599	0,652	0,948	0,925
0,50	0,961	0,882	0,947	0,932
0,75	0,927	0,940	0,947	0,932
1,00	0,895	0,935	0,945	0,933
1,25	0,622	0,624	0,945	0,934
1,50	0,527	0,539	0,945	0,933
1,75	0,462	0,479	0,946	0,933
2,00	0,419	0,436	0,946	0,933

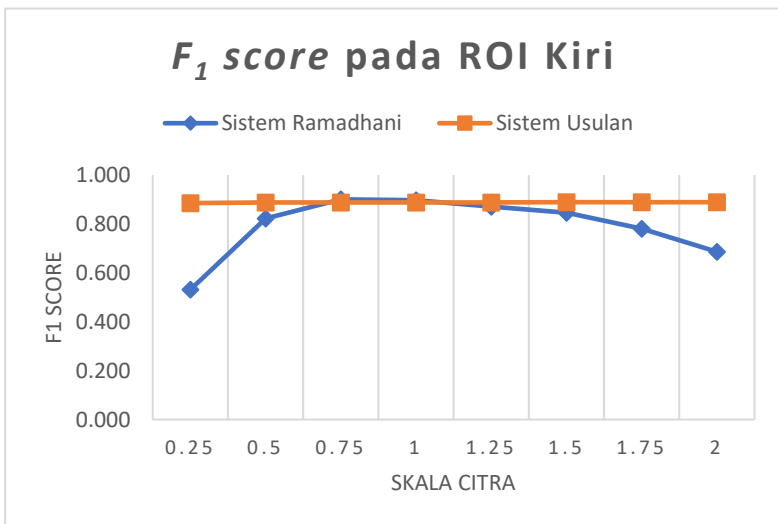
Tabel 5.53 memaparkan nilai F_1 score hasil segmentasi dari suatu citra yang memberikan nilai F_1 score terendah berdasarkan ROI kiri dan ROI kanan.

5.9.5 Nilai Rata-rata F_1 score Hasil Segmentasi Sistem Ramadhani dan Sistem Usulan

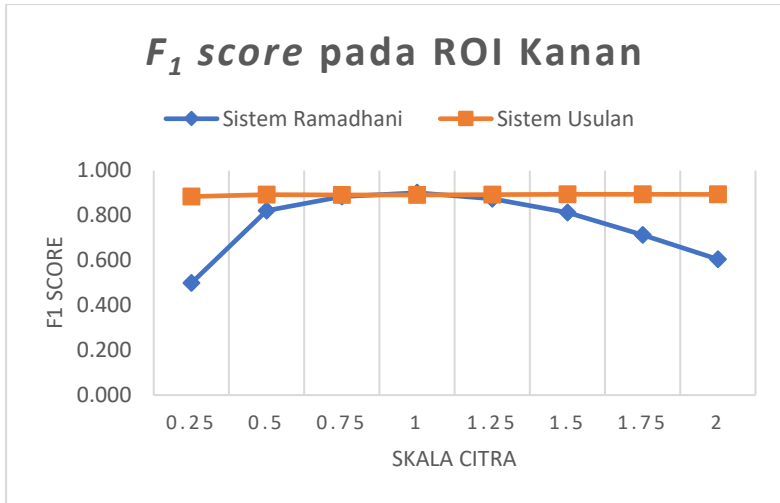
Bagian ini memaparkan perbandingan F_1 score hasil segmentasi yang dilakukan menggunakan sistem oleh Ramadhani dan sistem usulan. Nilai F_1 score yang dipaparkan adalah nilai rata-rata dari masing-masing sistem untuk semua citra pada setiap skala citra. Selain itu, dipaparkan pula standard deviasi dari F_1 score hasil segmentasi menggunakan kedua sistem untuk seluruh citra pada masing-masing skala.

Tabel 5.54 Perbandingan Rata-rata F_1 score Hasil Segmentasi Menggunakan Sistem Ramadhani dan Sistem Usulan

Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,531	0,499	0,885	0,884
0,50	0,822	0,821	0,888	0,892
0,75	0,901	0,883	0,888	0,891
1,00	0,897	0,901	0,888	0,891
1,25	0,871	0,874	0,888	0,892
1,50	0,846	0,813	0,888	0,894
1,75	0,780	0,714	0,889	0,894
2,00	0,687	0,605	0,889	0,894
Rata-rata	0,792	0,764	0,888	0,892



Grafik 5.7 Perbandingan F_1 score Sistem pada ROI Kiri



Grafik 5.8 Perbandingan F_1 score Sistem pada ROI Kanan

Tampak pada **Tabel 5.54**, **Grafik 5.7**, dan **Grafik 5.8** bahwa sistem oleh Ramadhani memberikan nilai F_1 score lebih tinggi pada skala citra 1,00 namun terjadi penurunan F_1 score pada skala yang lain. Hal ini disebabkan sistem oleh Ramadhani dikembangkan khusus pada citra dengan skala 1,00. Sehingga parameter yang digunakan pun khusus untuk citra dengan skala 1,00.

Berbeda dengan sistem oleh Ramadhani, sistem yang diusulkan pada tugas akhir ini memberikan hasil yang lebih stabil. Hal ini dikarenakan sistem dikembangkan agar dapat menyesuaikan citra dengan ukuran skala berbeda. Kestabilan dari kedua sistem ini ditunjukkan oleh standard deviasi F_1 score pada **Tabel 5.55**.

Tabel 5.55 menunjukkan perbandingan standard deviasi F_1 score dari kedua dalam menangani semua citra pada setiap ukuran skala citra yang telah ditentukan. Tampak bahwa beberapa nilai standard deviasi F_1 score dari sistem oleh Ramadhani berada di bawah 5% atau 0,05 namun beberapa yang lain berada di atas 5% atau 0,05. Sedangkan untuk sistem usulan, semua nilai standard

deviasi F_1 score berada di bawah 5% atau 0,05. Oleh karena itu, sistem usulan dapat dikatakan stabil untuk semua skala citra, sesuai dengan hipotesis pada bagian 5.8.

Selain berdasarkan hipotesis pada bagian 5.8, tampak bahwa untuk semua ROI pada semua skala, sistem usulan memberikan nilai standard deviasi lebih rendah dibandingkan sistem oleh Ramadhani. Hal ini sesuai dengan hipotesis yang telah ditentukan pada bagian 5.9.

Tabel 5.55 Perbandingan Standard Deviasi F_1 score Sistem Ramadhani dan Sistem Usulan

Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
0,25	0,086	0,080	0,040	0,033
0,50	0,077	0,060	0,041	0,029
0,75	0,038	0,040	0,039	0,031
1,00	0,030	0,029	0,040	0,032
1,25	0,079	0,071	0,040	0,030
1,50	0,110	0,133	0,040	0,028
1,75	0,152	0,176	0,040	0,028
2,00	0,157	0,206	0,040	0,028
Rata-rata	0,091	0,099	0,040	0,030

5.10 Uji Coba 6: Perbandingan Efisiensi

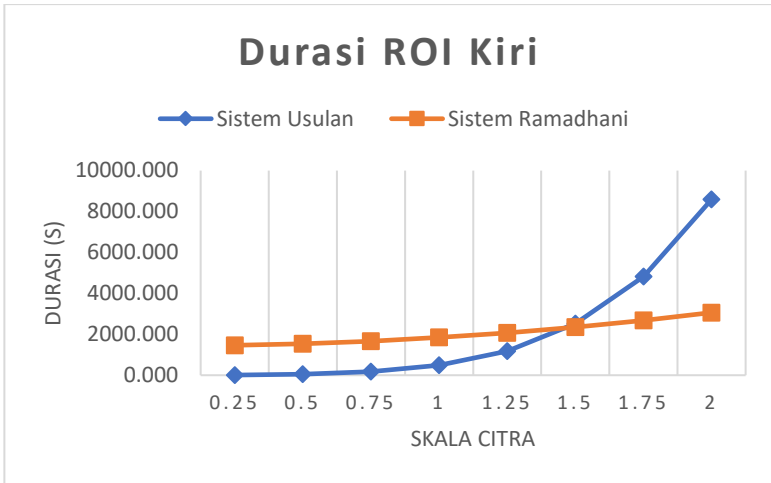
Uji coba yang dilakukan adalah membandingkan efisiensi waktu eksekusi dari sistem oleh Ramadhani dan sistem usulan. Uji coba ini bertujuan untuk melihat seberapa efisien sistem pada usulan. Hipotesis pada uji coba ini adalah sistem usulan memerlukan waktu sedikit dibandingkan sistem Ramadhani (lihat bagian 3.2.2).

Tabel 5.56 memaparkan waktu eksekusi yang dibutuhkan kedua sistem dalam melakukan segmentasi ROI pada semua citra untuk setiap skala yang telah ditentukan. Tampak bahwa sistem oleh usulan membutuhkan waktu lebih sedikit dibandingkan sistem oleh Ramadhani untuk citra dengan skala di bawah 1,50. Namun, untuk citra dengan skala lebih besar, sistem usulan membutuhkan waktu lebih lama dibanding dengan sistem oleh Ramadhani.

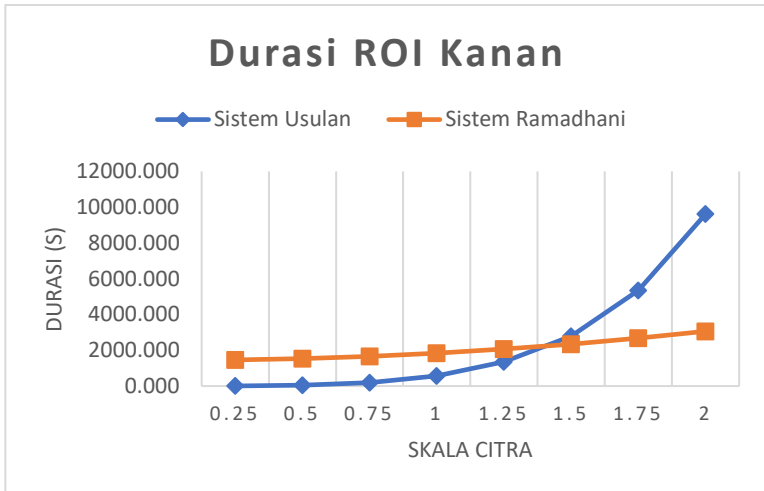
Tabel 5.56 Waktu Eksekusi Sistem Ramadhani dan Sistem Usulan

Skala Citra	Sistem Ramadhani		Sistem Usulan	
	ROI Kiri (detik)	ROI Kanan (detik)	ROI Kiri (detik)	ROI Kanan (detik)
0,25	1465	1461	8	6
0,50	1539	1538	46	48
0,75	1668	1669	174	197
1,00	1842	1843	495	571
1,25	2072	2074	1171	1361
1,50	2352	2349	2520	2792
1,75	2674	2677	4818	5339
2,00	3061	3062	8588	9615
Total	16674	16674	17819	19931

Sistem oleh Ramadhani membutuhkan waktu lebih lama pada skala di bawah 1,50 karena sistem ini melakukan proses *training*. Sedangkan sistem usulan tidak memerlukan proses *training*. Hal inilah yang menyebabkan sistem usulan lebih cepat. Namun sistem usulan membutuhkan waktu lebih lama untuk skala citra di atas 1,25 karena kompleksitas perhitungan *line strength* yang lebih besar. Lihat **Grafik 5.9** dan **Grafik 5.10**.



Grafik 5.9 Durasi Eksekusi pada ROI Kiri



Grafik 5.10 Durasi Eksekusi pada ROI Kanan

Berdasarkan **Tabel 5.56** dan analisis pada paragraf sebelumnya, sistem usulan membutuhkan waktu lebih sedikit dibandingkan

sistem oleh Ramadhani dalam menangani citra dengan skala di bawah 1,50 dan hal ini sesuai dengan hipotesis yang telah ditentukan. Namun sistem oleh Ramadhani membutuhkan waktu lebih sedikit dibandingkan sistem usulan untuk citra dengan skala di atas 1,25 dan hal ini tidak sesuai dengan hipotesis yang telah ditentukan.

Sistem usulan memang tidak lebih cepat untuk citra dengan ukuran besar. Namun untuk ukuran citra yang lebih kecil, sistem ini jauh lebih cepat hingga 200 kali dibandingkan dengan sistem oleh Ramadhani (dilihat dari skala 0,25). Selain itu, sistem usulan memberikan hasil yang lebih stabil dibandingkan dengan sistem oleh Ramadhani, ditunjukkan pada **Uji Coba 4** dan **Uji Coba 5**. Berdasarkan hasil dari kedua uji coba tersebut, pengguna dapat menggunakan sistem usulan dengan memanfaatkan skala ulang dari citra masukan. Jadi pengguna melakukan penandaan citra dan pemilihan ROI pada ukuran asli (atau ukuran sesuai kebutuhan) lalu mengecilkan citra tersebut sebelum proses segmentasi. Setelah didapat citra hasil segmentasi, citra tersebut kemudian diperbesar kembali ke ukuran semula. Penggunaan sistem usulan dengan cara ini dapat mempercepat proses segmentasi hingga 200 kali dibandingkan dengan sistem oleh Ramadhani dengan tetap mempertahankan F_1 score segmentasi.

BAB VI

KESIMPULAN DAN SARAN

Bab ini memaparkan kesimpulan yang diambil berdasarkan hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang telah ditentukan. Selain itu, dipaparkan pula saran untuk pengembangan penelitian lebih lanjut.

6.1 Kesimpulan

Berikut adalah kesimpulan yang diambil berdasarkan uji coba yang telah dilakukan:

1. Penandaan menggunakan koordinat relatif dalam pemilihan *Region Of Interest* (ROI) dapat dimanfaatkan untuk menghitung parameter segmentasi yang dapat menyesuaikan dengan ukuran citra yang berbeda.
2. Penggunaan orientasi garis terpilih seperti pada usulan memberikan rata-rata nilai F_1 score, yaitu 0,890, lebih tinggi dibandingkan dengan menggunakan semua orientasi, yaitu 0,872.
3. Penambahan intensitas citra masukan pada proses tahap kombinasi membuat nilai *line strength R* tidak signifikan sehingga rata-rata nilai F_1 score, yaitu 0,523, lebih rendah dibandingkan tanpa penambahan intensitas citra, yaitu 0,890.
4. Teknik normalisasi yang memetakan *line strength* ke intensitas *grayscale* dengan memerhatikan kecenderungan nilainya, seperti pada usulan, terbukti menghasilkan rata-rata nilai F_1 score, yaitu 0,890, lebih tinggi dibandingkan teknik normalisasi tanpa memerhatikan kecenderungan ini, yaitu 0,877.
5. Sistem usulan terbukti mampu melakukan segmentasi pada citra dengan ukuran, lokasi, dan rasio beragam dengan nilai rata-rata F_1 score yaitu 0,854.

6. Sistem usulan mampu secara melakukan segmentasi terhadap citra dengan ukuran berbeda dengan nilai rata-rata F_1 score, yaitu 0,890, lebih tinggi dibandingkan sistem pada penelitian sebelumnya, yaitu 0,778.
7. Sistem usulan lebih stabil melakukan segmentasi terhadap citra dengan ukuran berbeda ditunjukkan dengan nilai rata-rata dari standard deviasi F_1 score, yaitu 0,035, lebih kecil dibandingkan sistem sebelumnya, yaitu 0,095.
8. Dengan memanfaatkan skala ulang dan sifat dinamisnya, sistem usulan mampu melakukan segmentasi citra lebih cepat hingga 200 kali dibandingkan dengan sistem pada penelitian sebelumnya dengan tetap mempertahankan nilai F_1 score segmentasi.

6.2 Saran

Berikut adalah saran dari penulis untuk pengembangan sistem selanjutnya.

1. Diperlukan algoritma pada tahap *pre-processing* yang dapat meningkatkan kontras antara *background* dan *cortical bone* secara dinamis menyesuaikan dengan keadaan citra.
2. Diperlukan optimasi algoritma pada proses perhitungan *line strength* menggunakan *multi-scale line detection* yang lebih efisien dan efektif menyesuaikan dengan sudut orientasi *cortical bone* pada ROI.
3. Diperlukan algoritma pada tahap *post-processing* untuk memperbaiki citra hasil segmentasi secara signifikan yang menyesuaikan dengan keadaan ROI yang telah dipilih.

LAMPIRAN

A. Rata-rata F_1 score Segmentasi Uji Coba 1 – Bagian 1/2

Citra ke-	Nguyen		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
1	0,891	0,875	0,893	0,892
2	0,929	0,878	0,940	0,889
3	0,817	0,832	0,850	0,824
4	0,861	0,856	0,879	0,889
5	0,891	0,886	0,906	0,903
6	0,888	0,843	0,907	0,883
7	0,870	0,835	0,884	0,844
8	0,918	0,879	0,934	0,898
9	0,888	0,905	0,903	0,918
10	0,805	0,854	0,826	0,869
11	0,883	0,877	0,900	0,893
12	0,868	0,876	0,897	0,904
13	0,881	0,910	0,891	0,926
14	0,786	0,895	0,816	0,898
15	0,880	0,852	0,907	0,872

B. Rata-rata F_1 score Segmentasi Uji Coba 1 – Bagian 2/2

Citra ke-	Nguyen		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kiri
16	0,915	0,902	0,935	0,915
17	0,896	0,840	0,913	0,859
18	0,889	0,915	0,900	0,917
19	0,888	0,899	0,909	0,913
20	0,910	0,768	0,926	0,809
21	0,904	0,924	0,923	0,937
22	0,880	0,903	0,903	0,903
23	0,870	0,894	0,888	0,916
24	0,867	0,843	0,883	0,886
25	0,663	0,841	0,767	0,884
26	0,834	0,894	0,841	0,915
27	0,871	0,891	0,886	0,909
28	0,940	0,926	0,946	0,932
29	0,837	0,861	0,834	0,864
30	0,839	0,884	0,853	0,890

C. Rata-rata F_1 score Segmentasi Uji Coba 2 – Bagian 1/2

Citra ke-	Nguyen		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
1	0,452	0,684	0,893	0,892
2	0,430	0,502	0,940	0,889
3	0,404	0,360	0,850	0,824
4	0,481	0,517	0,879	0,889
5	0,634	0,522	0,906	0,903
6	0,475	0,510	0,907	0,883
7	0,612	0,562	0,884	0,844
8	0,503	0,591	0,934	0,898
9	0,537	0,573	0,903	0,918
10	0,419	0,479	0,826	0,869
11	0,456	0,506	0,900	0,893
12	0,516	0,553	0,897	0,904
13	0,433	0,546	0,891	0,926
14	0,475	0,523	0,816	0,898
15	0,416	0,437	0,907	0,872

D. Rata-rata F_1 score Segmentasi Uji coba 2 – Bagian 2/2

Citra ke-	Nguyen		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
16	0,475	0,634	0,935	0,915
17	0,521	0,501	0,913	0,859
18	0,486	0,513	0,900	0,917
19	0,449	0,511	0,909	0,913
20	0,433	0,466	0,926	0,809
21	0,500	0,536	0,923	0,937
22	0,591	0,589	0,903	0,903
23	0,547	0,618	0,888	0,916
24	0,615	0,530	0,883	0,886
25	0,544	0,576	0,767	0,884
26	0,505	0,596	0,841	0,915
27	0,462	0,511	0,886	0,909
28	0,636	0,615	0,946	0,932
29	0,509	0,458	0,834	0,864
30	0,820	0,698	0,853	0,890

E. Rata-rata F_1 score Segmentasi Uji Coba 3 – Bagian 1/2

Citra ke-	<i>Min-max</i>		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
1	0,903	0,889	0,893	0,892
2	0,941	0,888	0,940	0,889
3	0,851	0,821	0,850	0,824
4	0,879	0,892	0,879	0,889
5	0,909	0,903	0,906	0,903
6	0,910	0,886	0,907	0,883
7	0,901	0,823	0,884	0,844
8	0,937	0,909	0,934	0,898
9	0,906	0,922	0,903	0,918
10	0,413	0,875	0,826	0,869
11	0,897	0,895	0,900	0,893
12	0,903	0,915	0,897	0,904
13	0,891	0,927	0,891	0,926
14	0,847	0,898	0,816	0,898
15	0,900	0,876	0,907	0,872

F. Rata-rata F_1 score Segmentasi Uji Coba 3 – Bagian 2/2

Citra ke-	<i>Min-max</i>		Usulan	
	ROI Kiri	ROI Kanan	ROI Kiri	ROI Kanan
1	0,937	0,920	0,935	0,915
2	0,923	0,859	0,913	0,859
3	0,899	0,917	0,900	0,917
4	0,920	0,909	0,909	0,913
5	0,888	0,638	0,926	0,809
6	0,924	0,913	0,923	0,937
7	0,927	0,905	0,903	0,903
8	0,892	0,910	0,888	0,916
9	0,883	0,880	0,883	0,886
10	0,781	0,897	0,767	0,884
11	0,825	0,920	0,841	0,915
12	0,882	0,910	0,886	0,909
13	0,944	0,927	0,946	0,932
14	0,549	0,864	0,834	0,864
15	0,872	0,891	0,853	0,890

G. Rata-rata F_1 score Segmentasi Uji Coba 4 – Bagian 1/2

Citra ke-	Kiri-kiri	Kiri-tengah	Kanan-tengah	Kanan-kanan
1	0,827	0,818	0,886	0,865
2	0,832	0,891	0,849	0,891
3	0,836	0,808	0,813	0,842
4	0,900	0,842	0,814	0,815
5	0,874	0,809	0,899	0,820
6	0,807	0,847	0,848	0,800
7	0,890	0,880	0,838	0,922
8	0,912	0,898	0,873	0,886
9	0,837	0,868	0,875	0,883
10	0,761	0,769	0,886	0,828
11	0,896	0,833	0,909	0,909
12	0,930	0,844	0,886	0,888
13	0,879	0,852	0,875	0,901
14	0,811	0,807	0,807	0,903
15	0,838	0,802	0,880	0,870

H. Rata-rata F_1 score Segmentasi Uji Coba 5 – Bagian 2/2

Citra ke-	Kiri-kiri	Kiri-tengah	Kanan-tengah	Kanan-kanan
16	0,850	0,881	0,855	0,878
17	0,892	0,898	0,834	0,808
18	0,894	0,823	0,850	0,883
19	0,851	0,827	0,886	0,876
20	0,913	0,711	0,735	0,718
21	0,907	0,847	0,885	0,897
22	0,898	0,846	0,873	0,895
23	0,872	0,918	0,909	0,903
24	0,824	0,831	0,800	0,899
25	0,706	0,757	0,787	0,767
26	0,882	0,812	0,884	0,828
27	0,905	0,795	0,860	0,880
28	0,937	0,827	0,836	0,838
29	0,838	0,913	0,857	0,830
30	0,866	0,867	0,910	0,854

DAFTAR PUSTAKA

- [1] J. Y. Rho, R. B. Ashman and C. H. Turner, "Young's Modulus of Trabecular and Cortical Bone Material: Ultrasonic and Microtensile Measurements," *J. Biomechanics*, vol. 26, no. 2, pp. 111-119, May 1993.
- [2] J. A. Kanis, L. J. Melton, C. Christiansen, C. C. Johnston and N. Khaltayev, "The Diagnosis of Osteoporosis," *Journal of Bone and Mineral Research*, vol. 9, no. 8, pp. 1137-1141, 28 February 1994.
- [3] A. Z. Arifin, A. Asano, A. Taguchi, T. Nakamoto, A. Yuniarti, L. R. Dewi and H. Studiawan, "Line Strength Measurement for Trabecular Bone Analysis of Mandible on Dental Panoramic Radiographs," *IWAIT*, 2010.
- [4] A. Z. Arifin, A. Asano, A. Taguchi, T. Nakamoto, M. Ohtsuka, M. Tsuda, Y. Kudo and K. Tanimoto, "Computer-aided System for Measuring the Mandibular Cortical Width on Dental Panoramic Radiographs in Identifying Postmenopausal Women with Low Bone Mineral Density," *Osteoporos Int*, pp. 753-759, 2006.
- [5] E. Anjna, E. R. Kaur and , "Review of Image Segmentation Technique," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 4, 2017.
- [6] E. Ricci and R. Perfetti, "Retinal Blood Vessel Segmentation Using Line Operators and Support Vector Classification," *IEEE Transactions on Medical Imaging*, vol. 26, no. 10, pp. 1357-1365, 2007.
- [7] U. T. Nguyen, A. Bhuiyan, L. A. Park and K. Ramamohanarao, "An Effective Retinal Blood Vessel

- Segmentation Method using Multi-scale Line Detection," *Elsevier Ltd.*, no. 46, pp. 703-715, 2012.
- [8] F. A. Ramadhani, A. Z. Arifin and W. N. Khotimah, "Segmentasi Cortical Bone Citra Dental Panoramic Radiograph menggunakan Multi-Scale Line Operator dan Supervised Classifier," in *Publikasi Online Mahasiswa ITS (POMITS)*, Surabaya, 2015.
- [9] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS*, vol. 9, no. 1, pp. 62-66, 1979.
- [10] RadiologyInfo.org, "Panoramic Dental X-ray," 25 February 2018. [Online]. Available: <https://www.radiologyinfo.org/en/info.cfm?pg=panoramic-xray>. [Accessed 9 May 2018].
- [11] W. C. Scarfe and G. F. Williamson, "Patient Positioning and Exposure," *dentalcare.com*, 2018. [Online]. Available: <https://www.dentalcare.com/en-us/professional-education/ce-courses/ce71/patient-positioning-and-exposure>. [Accessed 11 May 2018].
- [12] RadiologyInfo.org, "Dental Cone Beam CT," 2017. [Online]. Available: <https://www.radiologyinfo.org/en/info.cfm?pg=dentalconect>. [Accessed 11 May 2018].
- [13] Chemira Medpharma, "Dental Panoramic Tomography X ray Machine," 31 Desember 2017. [Online]. Available: <http://chemiramemed.com/product/dental-panoramic-tomography-x-ray-machine/>. [Accessed 12 May 2018].
- [14] W. C. Scarfe and G. F. Williamson, "Image Projection," *dentalcare.com*, 2018. [Online]. Available: <https://www.dentalcare.com/en-us/professional-education/ce-courses/ce71/image-projection>. [Accessed 11 May 2018].

- [15] W. C. Scarfe and G. F. Williamson, "Slit Radiography," dentalcare.com, 2018. [Online]. Available: <https://www.dentalcare.com/en-us/professional-education/ce-courses/ce71/slit-radiography>. [Accessed 12 May 2018].
- [16] A. P. Singh, "Cortical Bone and Cancellous Bone," 2017. [Online]. Available: <http://boneandspine.com/cortical-bone-and-cancellous-bone/>. [Accessed 11 May 2018].
- [17] National Institutes of Health (NIH), "Osteogenesis Imperfecta," 2014. [Online]. Available: <https://www.niams.nih.gov/health-topics/osteogenesis-imperfecta>. [Accessed 11 May 2018].
- [18] National Institutes of Health (NIH), "Osteonecrosis," 2015. [Online]. Available: <https://www.niams.nih.gov/health-topics/osteonecrosis>. [Accessed 11 May 2018].
- [19] R. A. Poldrack, "Region of Interest Analysis for fMRI," *Social Cognitive and Affective Neuroscience*, vol. 2, no. 1, pp. 67-70, 2007.
- [20] R. Wang, "Interpolation and Extrapolation," 2018. [Online]. Available: <http://fourier.eng.hmc.edu/e176/lectures/ch7/node1.html>. [Accessed 12 May 2018].
- [21] R. Wang, "Cubic Spline Interpolation," 2018. [Online]. Available: <http://fourier.eng.hmc.edu/e176/lectures/ch7/node6.html>. [Accessed 12 May 2018].
- [22] B. Halabisky, A. D. Gordon and H. e. a. , "Euclidean Distance In 'n'-Dimensional Space," 1999. [Online]. Available: https://hlab.stanford.edu/brian/cluster_analysis.html. [Accessed 12 May 2018].

- [23] Scikit Learn, "Preprocessing Data," 2017. [Online]. Available: <http://scikit-learn.org/stable/modules/preprocessing.html>. [Accessed 12 May 2018].
- [24] A. Greensted, "Otsu Thresholding," 2010. [Online]. Available: <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>. [Accessed 12 May 2018].
- [25] MathWorks, "regionprops," 2018. [Online]. Available: <https://www.mathworks.com/help/images/ref/regionprops.html>. [Accessed 12 May 2018].
- [26] J. Brownlee, "Classification Accuracy is Not Enough: More Performance Measures You Can Use," 2014. [Online]. Available: <https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/>. [Accessed 112 May 2018].
- [27] N. Chinchor, "MUC-4 Evaluation Metrics," in *Proceedings of the 4th Conference on Message Understanding*, McLean, 1992.
- [28] Y. Seppenwoolde, R. I. Berbeco, S. Nishioka, H. Shirato and B. Heijmen, "Accuracy of Tumor Motion Compensation Algorithm from a Robotic Respiratory Tracking System: A Simulation Study," *Medical Physics*, vol. 34, no. 7, pp. 2774-2784, 2007.
- [29] G. R. Norman, J. A. Sloan and K. W. Wyrwich, "Interpretation of Changes in Health-related Quality of Life: The Remarkable Universality of Half a Standard Deviation," *Medical Care*, vol. 41, no. 5, pp. 582-592, 2003.

BIODATA PENULIS



Mohammad Anwar Hidayat, atau biasa dipanggil Anwar, lahir di Jember pada 14 Mei 1995. Penulis menempuh pendidikan mulai dari TK Ar-Rahim (2000-2002), SDN Kemuning Lor III (2002-2008), SMP Negeri 4 Jember (2008-2011), SMA Negeri 1 Jember (2011-2014) hingga saat ini di Institut Teknologi Sepuluh Nopember, Surabaya (2014-2018) di Departemen Informatika, Fakultas Teknologi Informasi dan Komunikasi angkatan 2014.

Selama menjadi mahasiswa di Departemen Informatika, penulis aktif sebagai asisten dosen mengajar di beberapa mata kuliah. Penulis juga aktif sebagai peneliti sekaligus administrator Lab. Komputasi Cerdas dan Visi (KCV) sejak tahun 2015. Selain dalam bidang akademik, penulis juga aktif bergerak di organisasi ekstra kampus seperti Himpunan Mahasiswa Teknik Computer-Informatika (HMTTC) sebagai staf Departemen Pengembangan Profesi (2015-2016) dan Dewan Perwakilan Mahasiswa (DPM) ITS sebagai staf ahli Bidang Kontrol (2015-2016).

Bidang minat penulis adalah Komputasi Cerdas dan Visi (KCV) dengan pendalaman pada *Natural Language Understanding* dan *Computer Vision*. Selain disibukkan dengan penelitian bidang minat tersebut, penulis juga disibukkan dengan hobi membaca novel, bermain *game*, dan *coding*. Penulis dapat dihubungi melalui email anwarhidayat14@gmail.com.