

BAB V

KESIMPULAN DAN SARAN

V.1 Kesimpulan

Dengan mengembangkan aplikasi *Edge Detection System (EDS)* ini penulis dapat menarik beberapa kesimpulan penting sebagai berikut:

1. Telah berhasil dikembangkan suatu perangkat lunak yang memiliki tujuan untuk melakukan operasi deteksi tepi citra digital dengan menggunakan transformasi *wavelet*, operator *Sobel*, *Prewitt* dan operator *Compass*.
2. Melalui aplikasi deteksi tepi ini dapat dilakukan pendeteksian tepi dari suatu citra dengan menggunakan empat metode yang berbeda.
3. Hasil dari detail tepi citra yang didapat tergantung dari metode yang digunakan dan besarnya parameter *threshold*. Deteksi tepi dengan menggunakan operator *Sobel*, *Prewitt*, *Compass* dan *wavelet1* memberikan hasil tepi yang baik secara visual. Deteksi tepi dengan menggunakan operator *wavelet2* kurang baik untuk diamati secara visual.
4. Gangguan berupa *noise* pada citra menyebabkan berkurangnya kuantitas detail tepi yang didapat karena hilangnya beberapa informasi utama dalam citra.
5. Untuk menghasilkan citra terdeteksi dengan kualitas yang baik, hal-hal yang perlu diperhatikan antara lain:
 - a. Pemberian nilai *threshold* yang tepat. Untuk metode *Sobel*, *Prewitt*, *Compass* diberikan

nilai threshold 75. Untuk metode wavelet2 diberikan nilai threshold 25.

- b. Pemberian efek pelembutan (*smoothing*) pada citra asli untuk mengurangi gangguan berupa *noise*.
- c. Citra asli memiliki tingkat kecerahan (*brightness*) yang baik.
- d. Intensitas cahaya citra asli tidak terlalu lembut/tajam.

V.2 Saran

Aplikasi *Edge Detection System (EDS)* ini dapat digolongkan sebagai sistem yang masih belum sempurna. Berbagai macam pengembangan masih dapat dilakukan untuk mencapai kesempurnaan dalam memberikan tepi suatu citra dengan kualitas yang baik. Berikut ini akan diberikan beberapa saran yang dapat digunakan untuk pengembangan sistem, yaitu:

1. Penambahan metode deteksi tepi yang lain untuk melakukan pendeteksian tepi.
2. Penambahan fungsi-fungsi preprocessing citra untuk memperbaiki kualitas citra terhadap gangguan sebelum dilakukan operasi deteksi tepi.

DAFTAR PUSTAKA

- Abdullah, Riyad Mubarak, 2003, *Pemampatan Data dengan Menggunakan Transformasi Gelombang-Singkat (Wavelet Transform)*, Program Pasca Sarjana Universitas Yogyakarta.
- Ary Dwi Suta Atmaja, I Made, *Kompresi Citra Dengan Menggunakan Algoritma Wavelet Haar 2-D dan Penyandian Dengan Algoritma RLE (RUN LENGTH ENCODING)*, Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta, 2005.
- Basuki, Ahmad, F.Palandi, Jozua dan Fatchurrochman, 2005, *Pengolahan Citra Digital menggunakan Visual Basic*, Graha Ilmu, Yogyakarta.
- Jain, Anil K, *Fundamentals of Digital Image Processing*, Prentice Hall of India, 1995.
- McCane, Brendan, 2001, *Edge Detection*, University of Otago, Dunedin, New Zealand.
- Mimawaty S, Fransiska, *Aplikasi Deteksi Tepi Citra Menggunakan Operator Gradien, Kompas, Laplacian dan Detektor Canny*, Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta, 2004.
- Munir, Rinaldi, 2004, *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*, Informatika, Bandung.
- Xiaoping, Guan dan Zequn, Guan, *Detection of High Resolution Remote Sensing Imagery Using Wavelet*, 2001, School of Information Engineering in Remote Sensing, Wuhan University.

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

EDGE DETECTION SYSTEM

(EDS)


Dipersiapkan oleh:

I NYOMAN WIRA SUTHA KUSUMA

(01 07 03117/TF)

Program Studi Teknik Informatika – Universitas Atma Jaya

Jalan Babarsari 44, Yogyakarta 52181

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		SKPL- EDS		1/35
	Universitas Atmajaya Yogyakarta	Revisi		<i>Tgl:19 Agustus 2006</i>

DAFTAR PERUBAHAN

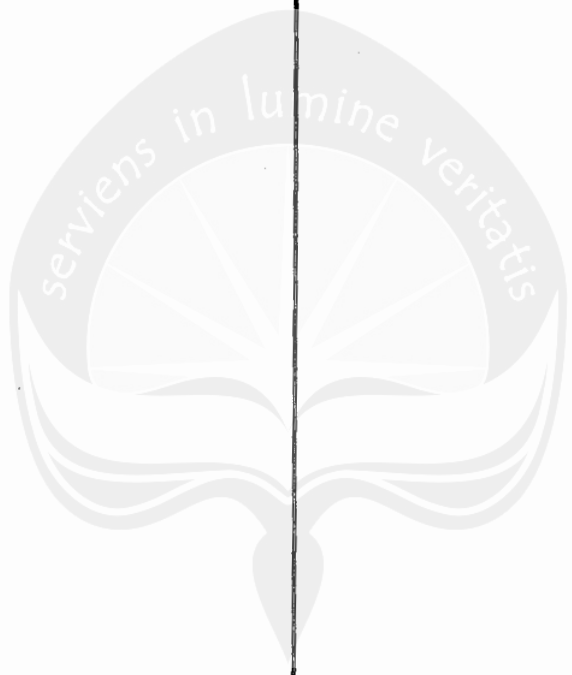
Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								



Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



Daftar Isi

1	Pendahuluan	7
1.1	Tujuan.....	7
1.2	Lingkup Masalah.....	7
1.3	Definisi, Akronim dan Singkatan.....	7
1.4	Referensi	8
1.5	Deskripsi umum (Overview).....	8
2	Deskripsi Keseluruhan.....	9
2.1	Perspektif produk.....	9
2.2	Fungsi Produk.....	9
2.3	Karakteristik Pengguna.....	10
2.4	Batasan-batasan.....	10
3	Deskripsi Rinci Kebutuhan	10
3.1	Kebutuhan antarmuka eksternal	10
3.1.1	Antarmuka pemakai	11
3.1.2	Antarmuka perangkat keras.....	11
3.1.3	Antarmuka perangkat lunak.....	11
3.2	Kebutuhan fungsionalitas	11
3.2.1	Aliran informasi.....	12
3.2.1.1	DFD Level 0.....	12
3.2.1.2	DFD Level 1.....	13
3.2.1.3	DFD Level 2 Proses Image Processing	14
3.2.1.4	DFD Level 2 Proses 4.1 (Wavelet1)	17
3.2.1.5	DFD Level 2 Proses 4.2 (Wavelet2).....	20
3.2.1.6	DFD Level 2 Proses 4.3 (Sobel)	22
3.2.1.7	DFD Level 2 Proses 4.4 (Prewitt).....	23
3.2.1.8	DFD Level 2 Proses 4.5 (Compass)	25
3.2.1.9	DFD Level 2 Proses 4.6 (Noise).....	26
3.2.1.10	DFD Level 2 Proses 4.7 (Smoothing)	27
3.2.1.11	DFD Level 2 Proses 4.8 (Low Pass Filter).....	28
3.2.1.12	DFD Level 2 Proses 4.9 (High Pass Filter).....	29
3.2.1.13	DFD Level 2 Proses 4.10 (Brightness)	30
3.2.1.14	DFD Level 2 Proses 4.11 (Histogram Equalization)	31
3.2.1.15	DFD Level 2 Proses 4.12 (MSE)	33
3.3	Kebutuhan performansi.....	34
3.4	Atribut sistem perangkat lunak.....	34
3.5	Kebutuhan lain.....	35
3.6	Entity Relationship Diagram	36

Daftar Gambar

Gambar 1 DFD Level 0.....	12
Gambar 2 DFD Level 1.....	14
Gambar 3 DFD Level 2 Proses Image Processing.....	17
Gambar 4 DFD Level 2 Proses Wavelet1.....	19
Gambar 5 DFD Level 2 Proses Wavelet2.....	21
Gambar 6 DFD Level 2 Proses <i>Sobel</i>	23
Gambar 7 DFD Level 2 Proses <i>Prewitt</i>	24
Gambar 8 DFD Level 2 Proses <i>Compass</i>	26
Gambar 9 DFD Level 2 Proses Noise.....	27
Gambar 10 DFD Level 2 Proses Smoothing.....	28
Gambar 11 DFD Level 2 Proses Low Pass Filter.....	29
Gambar 12 DFD Level 2 Proses High Pass Filter.....	29
Gambar 13 DFD Level 2 Proses Brightness.....	31
Gambar 14 DFD Level 2 Proses Histogram Equalization.....	32
Gambar 15 DFD Level 2 Proses MSE.....	34

1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan dokumentasi kebutuhan perangkat lunak yang akan dikembangkan. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan untuk perancangan dan pengembangan perangkat lunak.

1.2 Lingkup Masalah

Edge Detection System (EDS) merupakan perangkat lunak yang dikembangkan untuk membantu dalam pembacaan tepi dari objek didalam citra. Pengguna akan membuka sebuah file citra yang akan dideteksi tepinya, citra akan ditampilkan dalam bentuk citra asli dan citra hasil deteksi. Sistem ini akan menyajikan informasi mengenai citra asli.

1.3 Definisi, Akronim dan Singkatan

1. SKPL : Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggrisnya sering juga disebut sebagai SRS (*Software Requirements Specification*), merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.
2. EDS : *Edge Detection System*. Merupakan versi 1.0 perangkat lunak yang dikembangkan untuk melakukan pendeteksian tepi citra.
3. SKPL.EDS.Kxxx adalah kode yang merepresentasikan kebutuhan (*requirements*) pada EDS, dengan EDS

merupakan kode perangkat lunak, EDS.K adalah kode fase, dan xxx adalah digit/nomor kebutuhan (*requirement*).

4. DFD : *Data Flow Diagram*, merupakan model diagram yang mempresentasikan aliran proses dalam sistem perangkat lunak ini.

5. ERD : *Entity Relation Diagram*, merupakan model diagram yang digunakan untuk mempresentasikan hubungan antar entitas pengguna dari perangkat lunak ini.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak ini adalah:

1. GL01, *Template Spesifikasi Kebutuhan Perangkat Lunak*, Jurusan Teknik Informatika-ITB

1.5 Deskripsi umum (Overview)

Dokumen SKPL ini dibagi menjadi tiga bagian utama. Bagian pertama berisikan penjelasan tentang SKPL yang mencakup tujuan pembuatan dokumen, lingkup masalah yang diselesaikan oleh perangkat lunak, referensi, definisi kata, dan deskripsi umum.

Bagian kedua berisikan penjelasan secara umum mengenai perangkat lunak yang akan dikembangkan meliputi perspektif produk, fungsi produk, karakteristik pengguna, batasan-batasan dan asumsi serta ketergantungan yang diambil dalam mengembangkan perangkat lunak.

Bagian ketiga berisi tentang kebutuhan perangkat lunak secara lebih rinci, diantaranya kebutuhan Program Studi Teknik Informatika UAJY SKPL-EDS

antarmuka eksternal perangkat lunak, kebutuhan fungsionalitasnya, kebutuhan performansinya, batasan perancangannya, atribut sistem perangkat lunak serta kebutuhan-kebutuhan lainnya.

2 Deskripsi Keseluruhan

2.1 Perspektif produk

EDS merupakan perangkat lunak yang dikembangkan dengan tujuan melakukan pembacaan tepi dalam citra digital. Perangkat Lunak ini dikembangkan dengan menggunakan bahasa pemrograman *Microsoft Visual Basic 6.0*. Pengguna berinteraksi dengan EDS melalui tampilan grafis yang mudah dimengerti. EDS dapat beroperasi dalam lingkungan sistem operasi *Windows*. Masukan dari perangkat lunak ini adalah citra dalam format *JPEG, JPG* dan *BMP*.

2.2 Fungsi Produk

Perangkat lunak ini berguna untuk membaca tepi dalam citra. Fungsi-fungsi yang terdapat dalam perangkat lunak ini antara lain:

1. Pemilihan menu [SKPL-EDS.K-0001].

Menu dalam EDS akan memberikan pilihan operasi yang dapat dilakukan pada perangkat lunak, seperti membuka file citra, menyimpan file citra hasil, melakukan operasi deteksi tepi, menampilkan menu help dan pembuat sistem dan melakukan operasi *preprocessing* citra. Setelah pengguna memilih operasi yang diinginkan maka sistem akan menampilkan hasil operasinya.

2. Memilih dan Menyimpan citra [SKPL-EDS.K-0002]

Pengguna akan memilih citra dalam format *JPEG*, *JPG* atau *BMP* yang ingin dideteksi tepinya, kemudian hasil dari deteksi dapat disimpan sebagai citra dengan format *BMP*.

3. Memilih Proses terhadap citra [**SKPL-EDS.K-0003**]

Pengguna memilih proses yang dapat dilakukan terhadap citra. Proses yang dapat dilakukan adalah pendeteksian tepi citra, menambahkan noise, memberi efek *smoothing*, penajaman citra dan perbaikan menambah terang/gelap citra.

4. Melihat bantuan [**SKPL-EDS.K-0004**]

Bantuan yang disajikan berupa informasi mengenai perangkat lunak EDS versi 1.0 dan petunjuk penggunaan aplikasi.

5. Melihat info pembuat [**SKPL-EDS.K-0005**]

Info pembuat menyajikan informasi tentang pembuat perangkat lunak EDS versi 1.0.

2.3 Karakteristik Pengguna

Pengguna perangkat lunak ini adalah pengguna yang ingin melakukan pendeteksian tepi citra.

2.4 Batasan-batasan

Batasan masalah dalam mengembangkan aplikasi *Edge Detection System* ini adalah sebagai berikut:.

- a. Perangkat lunak yang dibangun digunakan untuk membantu pengguna untuk membaca tepi dari citra dan melakukan *preprocessing* citra.
- b. Sistem akan memberikan informasi tepi dari citra dan informasi lebar dan tinggi dari citra dan waktu komputasi proses.

c. Bahasa pemrograman yang digunakan adalah *Microsoft Visual Basic 6.0*

d. Aplikasi sistem akan diuji coba dan jalankan pada komputer.

3 Deskripsi Rinci Kebutuhan

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antarmuka eksternal pada *EDS (Edge Detection System)* meliputi kebutuhan antarmuka pemakai, kebutuhan antarmuka perangkat keras, kebutuhan antarmuka perangkat lunak.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan perangkat lunak *EDS* dengan antarmuka grafis berbasis *Windows*. Masukan dari perangkat lunak ini adalah pilihan menu yang dipilih melalui tombol mouse untuk menginputkan citra. Keluaran dari perangkat lunak *EDS* berupa citra yang telah diproses dan waktu komputasinya.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang dibutuhkan *EDS* adalah:

- a. PC
- b. Monitor
- c. RAM 128 MB
- d. Keyboard
- e. Mouse

3.1.3 Antarmuka perangkat lunak

Antarmuka perangkat lunak yang dibutuhkan *EDS* adalah:

a. Sistem Operasi : *Windows XP*.

b. Antarmuka : *Microsoft Visual Basic 6.0*

3.2 Kebutuhan fungsionalitas

3.2.1 Aliran Informasi

3.2.1.1 DFD Level 0

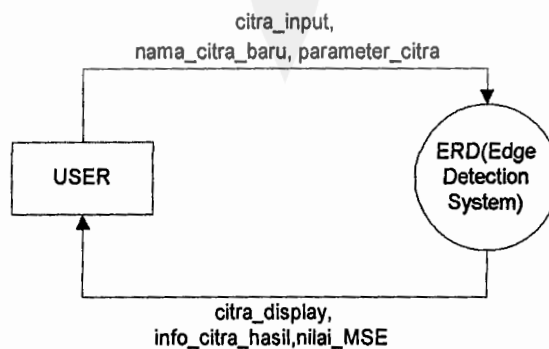
3.2.1.1.1 Entitas

Entitas Luar yang terlibat dalam penggunaan perangkat lunak EDS ini adalah semua pengguna yang ingin melakukan pendeteksian tepi pada citra.

3.2.1.1.2 Proses

Proses dalam EDS ini adalah pengguna memberikan inputan ke pada sistem berupa *citra_input* dalam format *JPEG*, *JPG* atau *BMP*, *nama_citra_baru* dan *parameter_citra*.

3.2.1.1.3 Topologi



Gambar 1 DFD Level 0

3.2.1.2 DFD level 1

3.2.1.2.1 Entitas

Entitas data pada level 1 ini dibuat berdasarkan entitas data pada diagram konteks *Edge Detection System (EDS)*.

3.2.1.2.2 Proses

Proses ini merupakan penjabaran dari DFD Level 0, terdapat empat proses utama yang terjadi dalam DFD level 1 *Edge Detection System (EDS)* yaitu:

1. Proses Open File

Dalam proses ini dilakukan pengambilan *citra_input* yang berekstensi *JPEG*, *JPG* atau *BMP*, lalu menampilkan *info_citra* di jendela *display*.

2. Proses Simpan File

Pada proses ini dilakukan penyimpanan citra hasil deteksi atau citra hasil *preprocessing* dalam format *BMP* dengan pemberian *nama_citra_baru*.

3. Proses Image Processing

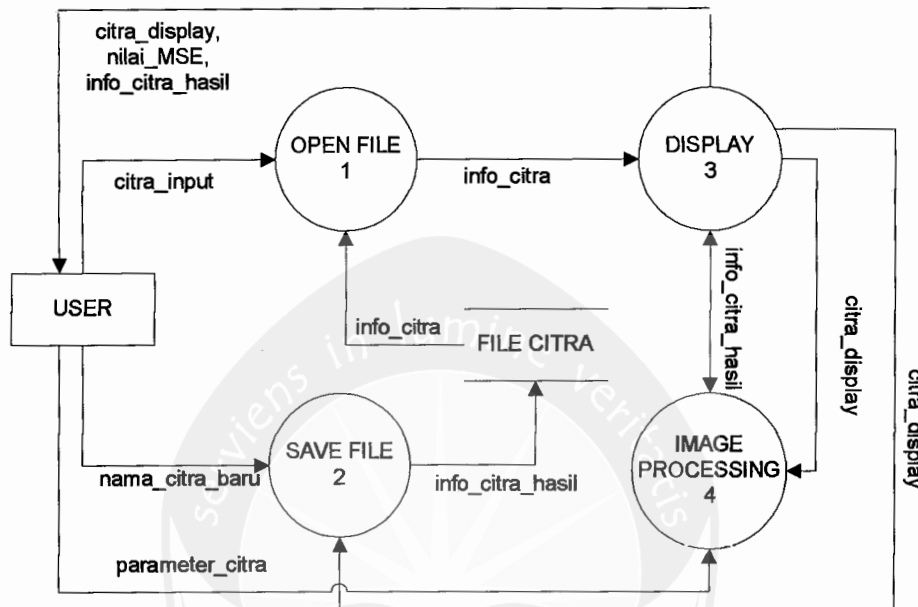
Proses ini memerlukan inputan *citra_display* dalam format *JPEG*, *JPG* atau *BMP*. Citra tersebut kemudian akan dibaca nilai tiap pikselnya sehingga kemudian bisa dilakukan proses pendeteksian tepi atau *preprocessing* citra untuk mendapatkan *info_citra_hasil*.

4. Proses Display

Pada proses ini dilakukan penampilan *citra_display* pada monitor dengan membaca informasi *info_citra* yang diperoleh dari proses

open file ataupun `info_citra_hasil` dari proses *image processing*.

3.2.1.2.3 Topologi



Gambar 2 DFD Level 1

3.2.1.3 DFD level 2 proses *IMAGE PROCESSING*

3.2.1.3.1 Entitas

Entitas data pada level 2 ini dibuat berdasarkan entitas data pada DFD level 1 *Edge Detection System (EDS)*.

3.2.1.3.2 Proses

Pada proses ini akan dilakukan pemilihan menu untuk menentukan operator deteksi tepi yang ingin dilakukan terhadap citra dan hasil dari deteksi tepi yang berupa citra yang telah terdeteksi tepinya akan ditampilkan sebagai `info_citra_hasil`. Selain untuk

melakukan pendeteksian tepi citra juga terdapat menu untuk melakukan operasi *preprocessing* citra. Pilihan menunya antara lain:

1. Wavelet1

Pemilihan menu ini memerintahkan sistem untuk melakukan deteksi tepi citra dengan menggunakan transformasi *wavelet Haar 2-D* dan menginputkan parameter_citra berupa nilai_threshold, menampilkannya sebagai info_citra_hasil.

2. Wavelet2

Pemilihan menu ini memerintahkan sistem untuk melakukan deteksi tepi citra dengan menggunakan transformasi *wavelet Haar 2-D* dan menampilkannya sebagai info_citra_hasil.

3. Sobel

Pemilihan menu ini memerintahkan sistem untuk melakukan deteksi tepi citra dengan menggunakan operator *Sobel* dan menginputkan parameter_citra berupa nilai_threshold, menampilkannya sebagai info_citra_hasil.

4. Prewitt1

Pemilihan menu ini memerintahkan sistem untuk melakukan deteksi tepi citra dengan menggunakan operator *Prewitt1* dan menginputkan parameter_citra berupa nilai_threshold, menampilkannya sebagai info_citra_hasil.

5. Compass

Pemilihan menu ini memerintahkan sistem untuk melakukan deteksi tepi citra dengan menggunakan operator *Compass* dan menginputkan parameter_citra berupa nilai_threshold, menampilkannya sebagai info_citra_hasil.

6. Noise

Pemilihan menu ini memerintahkan sistem untuk menambahkan gangguan (*noise*) pada citra. *Noise* akan diberikan secara *random* berdasarkan parameter citra *nilai_noise* yang diinputkan.

7. Smoothing

Pemilihan menu ini memerintahkan sistem untuk memberikan efek penghalusan pada citra.

8. Low Pass Filter

Pemilihan menu ini memerintahkan sistem untuk memberikan efek pelembutan pada citra berdasarkan pilihan kernel konvolusi.

9. High Pass Filter

Pemilihan menu ini memerintahkan sistem untuk memberikan efek penajaman pada citra berdasarkan pilihan kernel konvolusi.

10. Brightness

Pemilihan menu ini memerintahkan sistem untuk memberikan efek mengatur kecerahan pada citra berdasarkan *nilai_brightness* yang diinputkan.

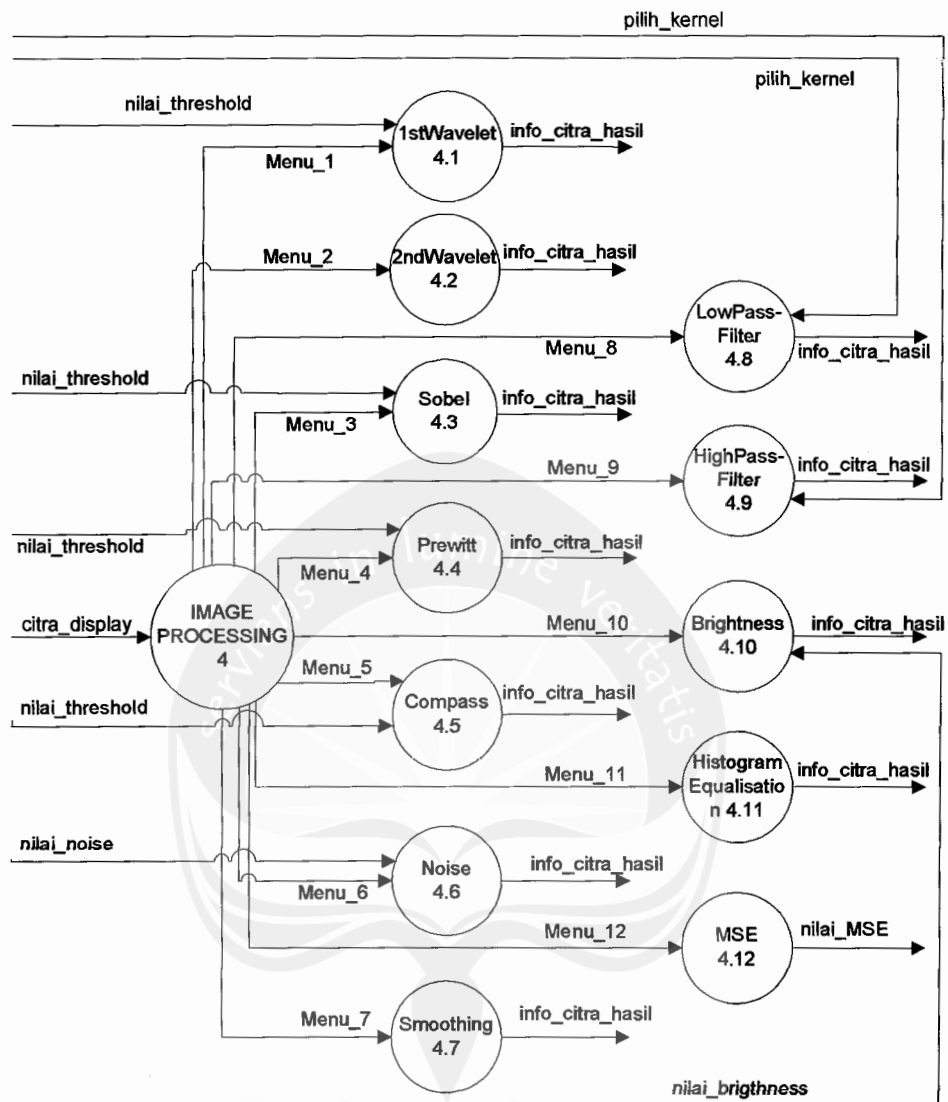
11. Histogram Equalisation

Pemilihan menu ini memerintahkan sistem untuk memperoleh penyebaran histogram yang merata pada citra.

12. MSE

Pemilihan menu ini memerintahkan sistem untuk menampilkan form MSE yang digunakan untuk menghitung nilai MSE.

3.2.1.3.3 Topologi



Gambar 3 DFD Level 2 Proses Image Processing

3.2.1.4 DFD level 2 proses 4.1 (Wavelet1)

3.2.1.4.1 Entitas

Entitas data pada level 2 proses 4.1 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

citra dengan menggunakan transformasi *wavelet* metode pertama. Proses-proses yang terjadi antara lain:

1. Baca matriks citra

Pada proses ini akan dilakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. Greyscale

Pada proses ini citra akan diubah menjadi citra *greyscale* atau citra aras keabuan. Dimana nilai untuk *Red*, *Green* dan *Blue*-nya adalah sama.

3. Smoothing

Pada proses ini akan dilakukan penghalusan citra untuk mengurangi gangguan sinyal (*noise*) yang tedapat pada citra.

4. Transformasi wavelet

Pada proses ini akan dilakukan operasi dekomposisi *wavelet* dengan menggunakan algoritma *wavelet Haar 2-D* dengan menggunakan metode tak standar. Pertama akan dilakukan transformasi 1-dimensi untuk setiap baris dan dilanjutkan dengan melakukan alihragam 1-dimensi untuk setiap kolom.

5. Multiscale Edge Detection

Pada proses ini akan dicari nilai gradient dari detail vertikal dan detail horizontalnya yang kemudian dibandingkan dengan suatu nilai ambang.

6. Konvolusi kernel wavelet

Pada proses ini matriks rerata akan dikonvolusikan dengan kernel wavelet dan mencari nilai kekuatan gradienya dan pemberian nilai *threshold*.

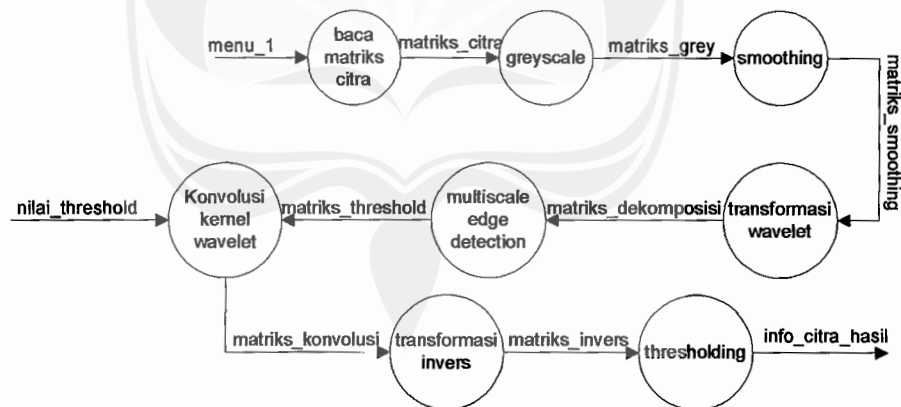
7. Transformasi invers wavelet

Pada proses ini akan dilakukan transformasi balik atau invers terhadap citra hasil dekomposisi. Pertama akan dilakukan transformasi balik terhadap kolom dan dilanjutkan dengan transformasi balik terhadap baris.

8. Thresholding

Pada proses ini nilai dari piksel akan dibatasi dengan suatu nilai ambang tertentu.

3.2.1.4.3 Topologi



Gambar 4 DFD Level 2 Proses wavelet1

3.2.1.5 DFD level 2 proses 4.2 (Wavelet2)

3.2.1.5.1 Entitas

Entitas data pada level 2 proses 4.2 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.5.2 Proses

Pada proses 4.2 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam pendeteksian tepi citra dengan menggunakan transformasi wavelet metode kedua. Proses-proses yang terjadi antara lain:

1. Baca matriks citra

Pada proses ini akan dilakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. Greyscale

Pada proses ini citra akan diubah menjadi citra *greyscale* atau citra aras keabuan. Dimana nilai untuk *Red*, *Green* dan *Blue*-nya adalah sama.

3. Smoothing

Pada proses ini akan dilakukan penghalusan citra untuk mengurangi gangguan sinyal (*noise*) yang tedapat pada citra.

4. Transformasi wavelet

Pada proses ini akan dilakukan operasi dekomposisi wavelet dengan menggunakan algoritma wavelet *Haar 2-D* dengan menggunakan metode tak standar. Pertama akan dilakukan

transformasi 1-dimensi untuk setiap baris dan dilanjutkan dengan melakukan alihragam 1-dimensi untuk setiap kolom.

5. Multiscale Edge Detection

Pada proses ini akan dicari nilai gradient dari detail vertikal dan detail horizontalnya yang kemudian dibandingkan dengan suatu nilai ambang.

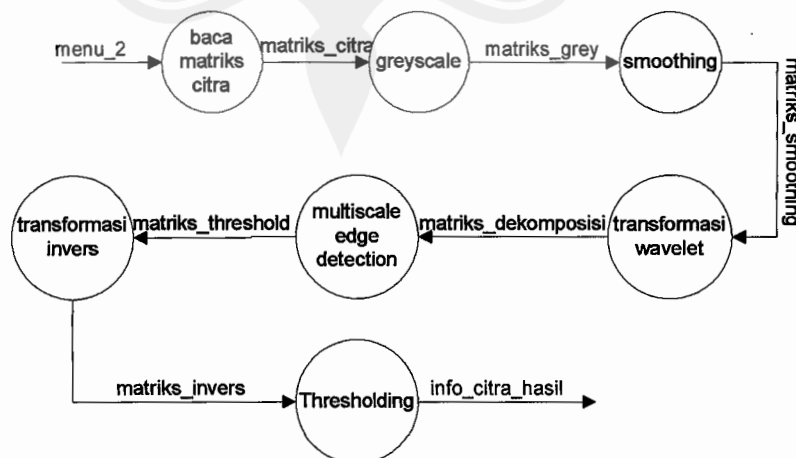
6. Transformasi invers wavelet

Pada proses ini akan dilakukan transformasi balik atau invers terhadap citra hasil dekomposisi. Pertama akan dilakukan transformasi balik terhadap kolom dan dilanjutkan dengan transformasi balik terhadap baris.

7. Thresholding

Pada proses ini nilai dari piksel akan dibatasi dengan suatu nilai ambang tertentu.

3.2.1.5.3 Topologi



Gambar 5 DFD Level 2 Proses wavelet2

3.2.1.6 DFD level 2 proses 4.3 (Sobel)

3.2.1.6.1 Entitas

Entitas data pada level 2 proses 4.3 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.6.2 Proses

Pada proses 4.3 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam pendeteksian tepi citra dengan menggunakan operator *Sobel*. Proses-proses yang terjadi antara lain:

1. Baca matriks citra

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. Greyscale

Pada proses ini citra akan diubah menjadi citra *greyscale* atau citra aras keabuan. Dimana nilai untuk *Red*, *Green* dan *Blue*-nya adalah sama.

3. konvolusi

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi G_x dan G_y untuk mendapatkan nilai dari gradien dalam arah sumbu x dan sumbu y .

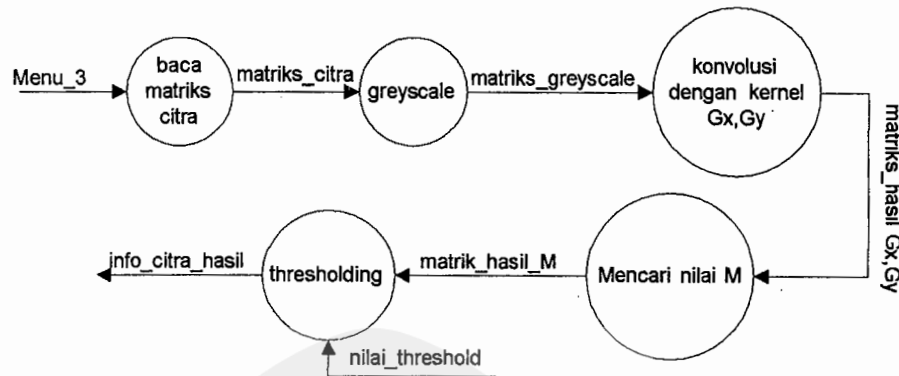
4. Mencari nilai M

Pada proses ini kita menghitung kekuatan tepi (M) yang merupakan magnitudo dari gradien fungsi G_x dan G_y .

5. Thresholding

Pada proses ini akan diberikan suatu nilai ambang terhadap nilai M untuk mendapatkan `info_citra_hasil` berupa citra yang telah terdeteksi tepinya.

3.2.1.6.3 Topologi



Gambar 6 DFD Level 2 Proses Sobel

3.2.1.7 DFD level 2 proses 4.4 (Prewitt1)

3.2.1.7.1 Entitas

Entitas data pada level 2 proses 4.4 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.7.2 Proses

Pada proses 4.4 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam pendeteksian tepi citra dengan menggunakan operator *Prewitt1*. Proses-proses yang terjadi antara lain:

1. Baca matriks citra

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya

yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. Greyscale

Pada proses ini citra akan diubah menjadi citra greyscale atau citra aras keabuan. Dimana nilai untuk Red, Green dan Blue-nya adalah sama.

3. konvolusi

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi G_x dan G_y untuk mendapatkan nilai dari gradien dalam arah sumbu x dan sumbu y.

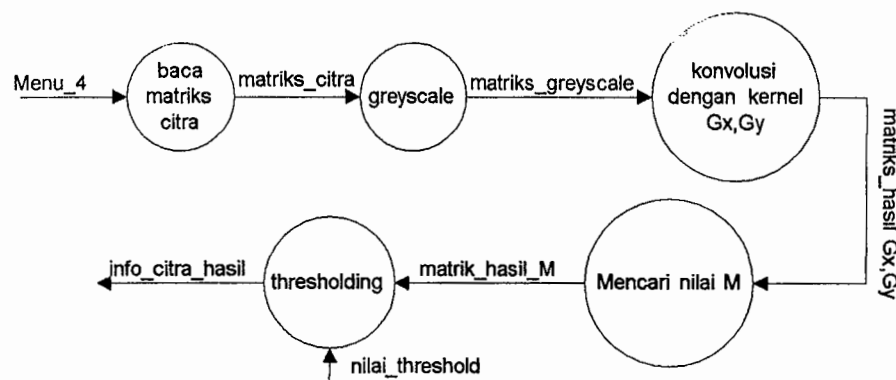
4. Mencari nilai M

Pada proses ini kita menghitung kekuatan tepi (M) yang merupakan magnitudo dari gradien fungsi G_x dan G_y .

5. Thresholding

Pada proses ini akan diberikan suatu nilai ambang terhadap nilai M untuk mendapatkan info_citra_hasil berupa citra yang telah terdeteksi tepinya.

3.2.1.7.3 Topologi



Gambar 7 DFD Level 2 Proses Prewitt1

3.2.1.8 DFD level 2 proses 4.5 (*Compass*)

3.2.1.8.1 Entitas

Entitas data pada level 2 proses 4.5 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.8.2 Proses

Pada proses 4.5 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam pendeteksian tepi citra dengan menggunakan operator *Compass*. Proses-proses yang terjadi antara lain:

1. *Baca matriks citra*

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. *Greyscale*

Pada proses ini citra akan diubah menjadi citra *greyscale* atau citra aras keabuan. Dimana nilai untuk *Red*, *Green* dan *Blue*-nya adalah sama.

3. *konvolusi*

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi *Compass* untuk mendapatkan nilai dari gradien dari 8 arah kernel konvolusi.

4. *Mencari nilai M*

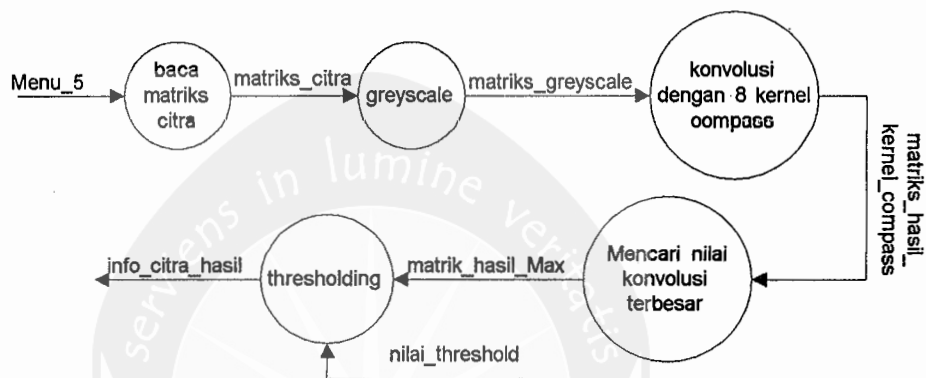
Pada proses ini kita menghitung kekuatan tepi (M) yang merupakan magnitude dari gradien 8 arah kernel konvolusi. Nilai M yang digunakan

adalah nilai M yang terbesar dari tiap titik dari 8 arah kernel konvolusi.

5. *Thresholding*

Pada proses ini akan diberikan suatu nilai ambang terhadap nilai M untuk mendapatkan info_citra_hasil berupa citra yang telah terdeteksi tepinya.

3.2.1.8.3 Topologi



Gambar 8 DFD Level 2 Proses *Compass*

3.2.1.9 DFD level 2 proses 4.6(Noise)

3.2.1.9.1 Entitas

Entitas data pada level 2 proses 4.6 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.9.2 Proses

Pada proses 4.6 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam pemberian gangguan pada citra. Proses-proses yang terjadi antara lain:

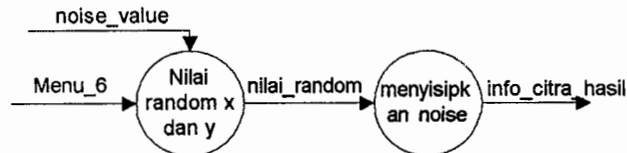
1. *Nilai random x dan y*

Pada proses ini akan dilakukan penentuan kordinat gangguan secara acak.

2. Menyisipkan noise

Pada proses ini kita menyisipkan gangguan berdasarkan kordinat secara acak unutm mendapatkan info_citra_hasil.

3.2.1.9.3 Topologi



Gambar 9 DFD Level 2 Proses Noise

3.2.1.10 DFD level 2 proses 4.7 (Smoothing)

3.2.1.10.1 Entitas

Entitas data pada level 2 proses 4.7 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.10.2 Proses

Pada proses 4.7 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam penghalusan citra. Proses-proses yang terjadi antara lain:

1. Baca matriks citra

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpiksel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

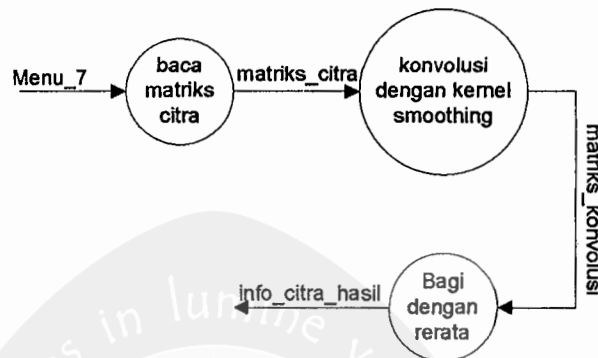
2. konvolusi

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi *smoothing*.

3. Bagi dengan rerata

Pada proses ini nilai dari konvolusi matriks akan dibagi dengan suatu rerata sesuai dengan ukuran kernel *smoothing* yang digunakan.

3.2.1.10.3 Topologi



Gambar 10 DFD Level 2 Proses *Smoothing*

3.2.1.11 DFD level 2 proses 4.8 (*Low Pass Filter*)

3.2.1.11.1 Entitas

Entitas data pada level 2 proses 4.8 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.11.2 Proses

Pada proses 4.8 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam proses pelembutan citra. Proses-proses yang terjadi antara lain:

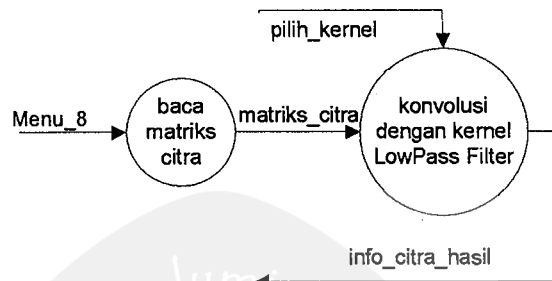
1. *Baca matriks citra*

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu *matriks_citra*.

2. konvolusi dengan kernel Low Pass Filter

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi *Low Pass Filter*.

3.2.1.11.3 Topologi



Gambar 11 DFD Level 2 Proses *Low Pass Filter*

3.2.1.12 DFD level 2 proses 4.9 (*High Pass Filter*)

3.2.1.12.1 Entitas

Entitas data pada level 2 proses 4.9 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.12.2 Proses

Pada proses 4.9 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam proses penajaman citra. Proses-proses yang terjadi antara lain:

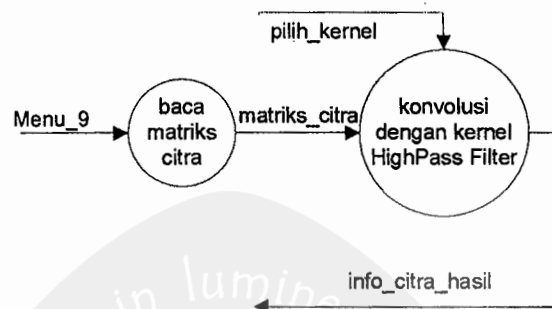
1. *Baca matriks citra*

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu *matriks_citra*.

2. konvolusi dengan kernel High Pass Filter

Pada proses ini akan dilakukan konvolusi matriks citra dengan matriks kernel konvolusi Low Pass Filter.

3.2.1.12.3 Topologi



Gambar 12 DFD Level 2 Proses High Pass Filter

3.2.1.13 DFD level 2 proses 4.10 (Brightness)

3.2.1.13.1 Entitas

Entitas data pada level 2 proses 4.10 ini dibuat berdasarkan entitas data pada DFD level 2 Edge Detection System (EDS).

3.2.1.13.2 Proses

Pada proses 4.10 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam proses pengaturan kecerahan citra. Proses-proses yang terjadi antara lain:

1. Baca Matriks Citra

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-

nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

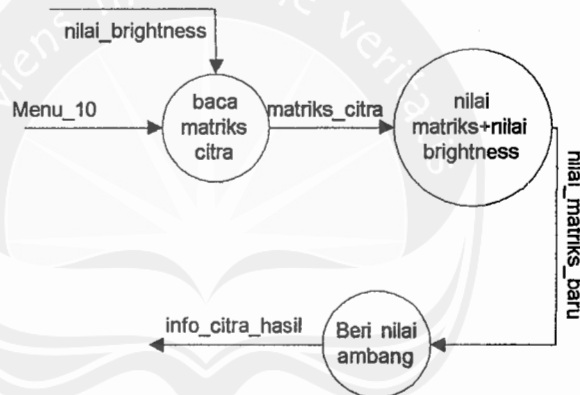
2. Nilai matriks+nilai brightness

Pada proses ini nilai brightness yang diinputkan oleh user akan ditambahkan dengan nilai RGB matriks_citra per piksel.

3. Beri nilai ambang

Pada proses ini nilai dari konvolusi matriks akan dibatasi nilainya supaya tidak kurang dari 0 dan lebih dari 255.

3.2.1.13.3 Topologi



Gambar 13 DFD Level 2 Proses *Brightness*

3.2.1.14 DFD level 2 proses 4.11 (*Histogram Equalisation*)

3.2.1.14.1 Entitas

Entitas data pada level 2 proses 4.11 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.14.2 Proses

Pada proses 4.11 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam proses penyamaan

nilai histogram citra. Proses-proses yang terjadi antara lain:

1. Baca Matriks Citra

Proses ini akan melakukan pembacaan matriks dari citra yang akan dikenakan operasi. Citra akan dibaca perpixel dan mencari nilai RGB-nya yang kemudian nilainya disimpan dalam suatu matriks_citra.

2. Hitung jumlah histogram

Pada proses ini akan dilakukan penghitungan jumlah histogram tiap warna Red, Green dan Blue.

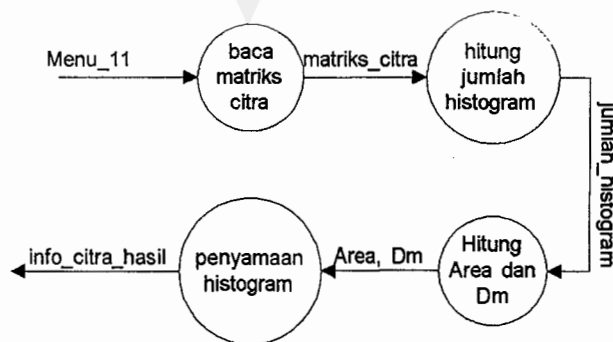
3. Hitung nilai Area dan Dm

Pada proses ini akan dicari nilai Area yang merupakan jumlah piksel dalam citra dan menghitung nilai Dm yang merupakan jumlah aras keabuan.

4. Penyamaan histogram

Pada proses ini akan dilakukan proses perhitungan penyamaan nilai histogram.

3.2.1.14.3 Topologi



Gambar 14 DFD Level 2 Proses Histogram Equalisation

3.2.1.15 DFD level 2 proses 4.12 (MSE)

3.2.1.15.1 Entitas

Entitas data pada level 2 proses 4.12 ini dibuat berdasarkan entitas data pada DFD level 2 *Edge Detection System (EDS)*.

3.2.1.15.2 Proses

Pada proses 4.12 akan dijelaskan dengan lebih rinci proses-proses yang terjadi dalam proses menghitung nilai MSE. Proses-proses yang terjadi antara lain:

1. Baca matrik citral

Proses ini akan melakukan pembacaan matriks dari citra. Citra akan dibaca perpiksel dan mencari nilainya yang kemudian nilainya disimpan dalam suatu matriks_citra.

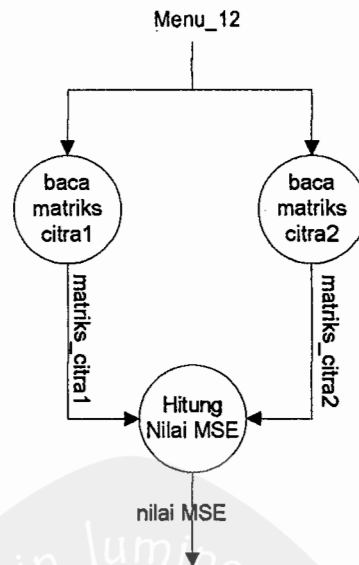
2. Baca matrik citra2

Proses ini akan melakukan pembacaan matriks dari citra yang mengalami gangguan. Citra akan dibaca perpiksel dan mencari nilainya yang kemudian nilainya disimpan dalam suatu matriks_citra.

3. Hitung nilai MSE

Pada proses ini akan dilakukan perhitungan untuk mencari nilai MSE dengan rumus.

3.2.1.15.3 Topologi



Gambar 15 DFD Level 2 Proses MSE

3.3. Kebutuhan Performansi

Kebutuhan performansi yang dibutuhkan untuk pemakaian *Edge Detection System (EDS)* ini antara lain adalah:

- Komputer untuk user.
- Spesifikasi komputer yang digunakan diharapkan memiliki spesifikasi yang tinggi sehingga akan mempercepat proses komputasi.

3.4. Atribut Sistem Perangkat Lunak

Atribut yang digunakan dalam pembuatan sistem perangkat lunak *Edge Detection System (EDS)* ini antara lain:

- Sistem sewaktu-waktu bisa diperbaharui sesuai dengan tingkat kebutuhan.
- Menggunakan sistem operasi *Windows XP*.
- Menggunakan *Microsoft Visual Basic 6.0*.

3.5. Kebutuhan Lain

Kebutuhan lain dari sistem ini adalah pengguna yang mengerti dalam mengoperasikan komputer dan membutuhkan sedikit latihan untuk membantu menjalankan sistem pendeteksian tepi ini sehingga mampu memanfaatkan sistem semaksimal mungkin.

3.6. Entity Relationship Diagram (ERD)

Pada perancangan perangkat lunak ini tidak disertakan dengan ERD karena perangkat lunak ini tidak melakukan pengolahan terhadap entitas-entitas yang terkait di dalam basis data.



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

EDGE DETECTION SYSTEM

(EDS)




Dipersiapkan oleh:

I NYOMAN WIRA SUTHA KUSUMA

(01 07 03117/TF)

Program Studi Teknik Informatika – Universitas Atma Jaya

Jalan Babarsari 44, Yogyakarta 52181

	Program Studi Teknik Informatika Universitas Atmajaya Yogyakarta	Nomor Dokumen		Halaman
		<i>DPPL-EDS</i>		1/24
		Revisi		<i>Tgl: 19 Agustus 2006</i>

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G

Daftar Halaman Perubahan

--	--	--	--

Daftar Isi

1	Pendahuluan	6
1.1	Tujuan.....	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim dan Singkatan.....	6
2	Referensi	7
3	Deskripsi Dekomposisi	8
4	Deskripsi Antarmuka.....	9
4.1	Antarmuka Form Splash.....	9
4.1.1	Rancangan Antarnuka	9
4.1.2	Rancangan Prosedural/Fungsional	9
4.2	Antarmuka Form Utama.....	10
4.2.1	Rancangan Antarnuka	10
4.2.2	Rancangan Prosedural/Fungsional	10
4.2.2.1	Deskripsi Tombol Open dan Menu Open	12
4.2.2.2	Deskripsi Tombol Save dan Menu Save.....	12
4.2.2.3	Deskripsi Tombol Wavelet1.....	13
4.2.2.4	Deskripsi Tombol Wavelet2.....	14
4.2.2.5	Deskripsi Tombol Sobel.....	16
4.2.2.6	Deskripsi Tombol Prewitt.....	17
4.2.2.7	Deskripsi Tombol Compass	17
4.2.2.8	Deskripsi Tombol Noise.....	18
4.2.2.9	Deskripsi Tombol Smoothing.....	18
4.2.2.10	Deskripsi Tombol Low Pass Filter	19
4.2.2.11	Deskripsi Tombol High Pass Filter	19
4.2.2.12	Deskripsi Tombol Brightness	20
4.2.2.13	Deskripsi Tombol Histogram Equalization.....	20
4.2.2.14	Deskripsi Tombol MSE	21
4.2.2.15	Deskripsi Tombol Help and Support.....	21
4.2.2.16	Deskripsi Tombol About.....	21
4.2.2.11	Deskripsi Tombol Exit.....	21
4.3	Antarmuka Form MSE.....	22
4.3.1	Rancangan Antarnuka	22
4.3.2	Rancangan Prosedural/Fungsional	22
4.3.2.1	Deskripsi Tombol Open	23
4.3.2.2	Deskripsi Tombol MSE	23
4.3.2.3	Deskripsi Tombol Exit.....	24
4.4	Antarmuka Form About.....	24
4.4.1	Rancangan Antarnuka	24
4.4.2	Rancangan Prosedural/Fungsional	24

Daftar Gambar

Gambar 1 Deskripsi Dekomposisi EDS.....	8
Gambar 2 Rancangan Form Splash.....	9
Gambar 3 Rancangan Form Utama.....	10
Gambar 4 Rancangan Form MSE.....	22
Gambar 5 Rancangan Form About.....	24



1. Pendahuluan

1.1 Tujuan

Dokumen ini merupakan Deskripsi Perancangan Perangkat Lunak (DPPL) atau *Software Design Description* (SSD) dari EDS (*Edge Detection System*). Dibuat dengan tujuan untuk mendokumentasikan perancangan dari perangkat lunak yang akan dikembangkan. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan pada tahap implementasi perangkat lunak EDS selanjutnya.

1.2 Lingkup Masalah

Perangkat lunak EDS ini merupakan sistem yang akan melakukan pengolahan citra yaitu melakukan pendeteksian tepi pada citra dan. Fungsi utama dari perangkat lunak ini adalah melakukan pendeteksian tepi suatu citra dengan menggunakan transformasi *wavelet haar 2-D*, *Operator Sobel*, *Prewitt*, *Compass* dan fungsi tambahannya untuk melakukan *preprocessing* citra dan menghitung nilai *MSE*.

1.3 Definisi, Akronim dan Singkatan

- SKPL : Spesifikasi Kebutuhan Perangkat Lunak, biasa juga disebut sebagai *Software Requirements Specifications* (SRS) yang merupakan spesifikasi dari produk perangkat lunak yang akan dikembangkan.
- DPPL : Deskripsi Pengembangan Perangkat Lunak, biasa disebut *Software Design Description* (SSD), yang merupakan deskripsi dari perangkat lunak yang dikembangkan.

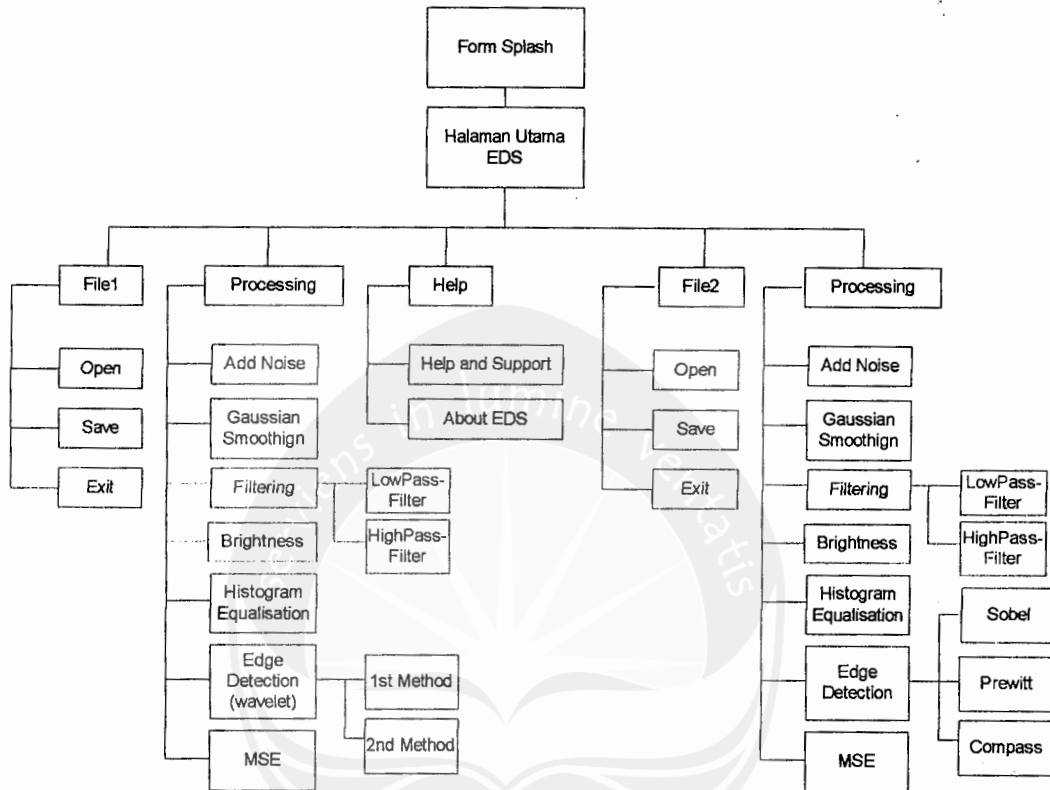
- *EDS: Edge Detection System*, merupakan sistem yang dibangun untuk mendeteksi tepi suatu citra.
- *DFD* atau *Data Flow Diagram* : menggambarkan diagram aliran proses yang terjadi.

2. Referensi

- Rinaldi Munir, 2004, *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*, Informatika Bandung, Bandung.
- GL01, *Template Spesifikasi Kebutuhan Perangkat Lunak*, Jurusan Teknik Informatika, ITB.
- GL02, *Template Deskripsi Perancangan Perangkat Lunak*, Jurusan Teknik Informatika, ITB.

3. Deskripsi Dekomposisi

3.1 Dekomposisi Modul

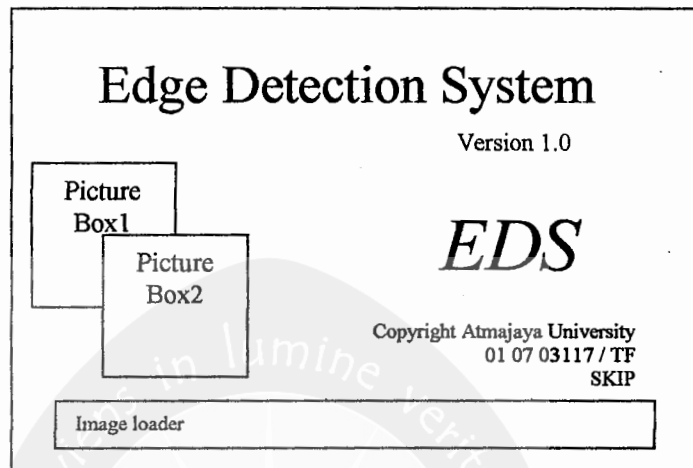


Gambar 1 Dekripsi Dekomposisi EDS

4. Deskripsi Antarmuka

4.1 Antarmuka Form Splash

4.1.1 Rancangan Antarmuka



Gambar 2 Rancangan Form splash

4.1.2 Rancangan Prosedural/Fungsional

Form awal dari sistem EDS adalah *form Splash*. *Form Splash* akan muncul saat sistem dijalankan yang merupakan *loading* untuk menampilkan *form* utama sistem. Setelah *loading* selesai maka akan ditampilkan *form* main yang merupakan *form* utama dari sistem dan *form splash* akan hilang.

4.2 Antarmuka Form Utama

4.2.1 Rancangan Antarmuka

Processing			Help	File2	Processing	
Open	Add Noise		Help and Support	Open	Add Noise	
Save	Smoothing		About EDS	Save	Smoothing	Low Pass Filter
Exit	Filtering		Low Pass Filter	Exit	Filtering	High Pass Filter
	Brightness		High Pass Filter		Brightness	
	Hist.Equal				Hist.Equal	Sobel
	Wavelet	1st Method			Edge Detection	Prewitt
	MSE	2nd Method			MSE	Comapss
Save1	Open1	Undo1		Save2	Open2	Undo2
Edge Detection Using Wavelet Transform				Edge Detection Using Sobel and Prewitt Detector		
PictureBox1				PictureBox2		
ProgressBar1				ProgressBar2		
Image Resolution 1		Image Resolution 2		Date Display		Time Display

Gambar 3 Rancangan Form Utama

4.2.2 Rancangan Prosedural/Fungsional

Form utama merupakan tempat dimana semua proses deteksi tepi dan *preprocessing* citra akan dilakukan. Pada Form ini terdapat 2 *PictureBox* yang digunakan untuk menampilkan citra asli dan citra yang telah terdeteksi tepinya. *PictureBox1* akan digunakan untuk melakukan pendeteksian tepi citra dengan menggunakan transformasi *wavelet*, sedangkan *PictureBox2* digunakan untuk menampilkan citra yang akan dideteksi tepinya dengan menggunakan operator *Sobel*, *Prewitt* dan *Compass*.

Pada Form ini terdapat menu pilihan untuk membuka citra yang akan diproses dan menyimpan citra yang telah terdeteksi. Menu dibagi menjadi 2, yaitu menu untuk proses deteksi tepi citra dengan menggunakan transformasi *wavelet* dan menu untuk proses deteksi tepi citra dengan menggunakan operator *Sobel*, *Prewitt* dan *Compass*. Pada masing-masing menu terdapat pilihan menu untuk membuka dan menyimpan file citra, menambah *noise*, efek *smoothing*, *filtering*, memperbaiki kecerahan citra, penyamaan histogram, pendeteksian tepi citra dan penghitungan nilai *MSE*. Untuk melakukan proses, selain dengan menggunakan pilihan menu, untuk menampilkan dan menyimpan citra dapat dilakukann dengan menggunakan tombol *Open* dan *Save*.

Citra hasil deteksi tepi dapat disimpan ke dalam media penyimpanan dengan menggunakan pilihan menu *Save* ataupun tombol *Save*. Selain tombol *Open* dan *Save* pada form Utama juga terdapat tombol *Undo* pada masing-masing *PictureBox* yang digunakan untuk membatalkan proses pendeteksian. Selama pendeteksian tepi citra berjalan, *ProgressBar* akan menunjukkan berapa persen proses telah berlangsung. Pada *StatusBar* akan ditampilkan info resolusi dari citra pada *PictureBox1* dan *PictureBox2*, waktu komputasi, tanggal dan jam. Pada sistem ini juga terdapat menu tambahan untuk menampilkan *help and support*, *form about* dan untuk menghitung nilai *MSE*.

Help and Support akan memberikan informasi bagaimana menjalankan sistem dengan benar. Selain bantuan untuk menjalankan sistem, pada *Help and Support* juga diberikan dasar teori *wavelet*, *Sobel*, *Prewitt* dan *Compass*. Sedangkan *form About* akan memberikan informasi tentang perancang sistem. Untuk melakukan penghitungan

nilai *MSE* akan dilakukan pada form *MSE*. Bila pilihan menu *MSE* dipilih maka form main akan hilang dan form *MSE* ditampilkan.

4.2.2.1 Deskripsi Tombol *Open* dan Menu *Open*

Merupakan tombol dan pilihan menu untuk mencari citra hasil dari suatu media penyimpanan dan menampilkannya pada *PictureBox*. Proses dimulai dengan menekan tombol *Open* atau memilih menu *Open*. Dialog Box akan ditampilkan untuk mencari citra asli dan menampilkannya. Resolusi dari citra akan ditampilkan pada Status Bar.

Secara Prosedural :

```
On_Klik Open
Tampilkan Dialog box Open
Input citra yang akan dideteksi tepinya
If fname <> "" Then
    PictureBox ← LoadPicture(fname)
    StatusBar ← Image_Resolution
Endif
```

4.2.2.2 Deskripsi Tombol *Save* dan Menu *Save*

Merupakan tombol dan pilihan menu untuk menyimpan citra hasil deteksi. Proses dimulai dengan menekan tombol *Save* atau memilih menu *Save*. Dialog box akan ditampilkan untuk menyimpan citra hasil ke dalam media penyimpanan.

Secara Prosedural :

```
On_Klik Save
Tampilkan Dialog box Save
Input filename citra terdeteksi
If fname <> "" Then savePicture
```

4.2.2.3 Deskripsi Tombol *Wavelet1*

Merupakan pilihan menu untuk melakukan pendeteksian tepi citra dengan menggunakan transformasi wavelet. Citra yang telah tertampil pada *PictureBox1* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya dan disimpan ke dalam suatu matriks temp. Citra yang akan dideteksi diubah menjadi citra *greyscale* dengan membagi nilai RGBnya dengan 3. Citra yang telah menjadi citra *greyscale* dikenakan operasi konvolusi dengan mengalikan matriks citra *greyscale* dengan suatu kernel konvolusi. Matriks citra yang telah tersmoothing akan ditransformasi dengan transformasi wavelet. Disini dekomposisi yang digunakan adalah dekomposisi takstandar dengan mengkombinasikan pasangan transformasi baris dan kolom secara bergantian. Pertama akan dilakukan alihragam 1-dimensi pada baris dan kemudian dilanjutkan pada kolom. Proses akan dilakukan 1 level. Nilai dari detail vertikal dan detail horizontal dari hasil transformasi akan dicari nilai gradientnya dan membandingkannya dengan suatu nilai ambang. Nilai rerata akan dikenakan proses konvolusi dengan kernel wavelet dan diberi nilai ambang. Selanjutnya matriks dekomposisi hasil transform akan dikenakan operasi transformasi invers wavelet. Pertama akan dilakukan alihragam invers 1-dimensi pada kolom dan kemudian dilanjutkan pada baris. Proses dilakukan 1 level. Matriks hasil dari invers akan diberi suatu nilai threshold untuk mendapatkan info citra hasil yang berupa citra terdeteksi.

Secara Prosedural :

On_Klik Wavelet

Input (citra_bmp,citra_jpg,citra_jpeg)

Baca matriks_citra

```

Ubah citra_asli ke citra_grayscale
inisialisasi kernel_smoothing
smoothing citra_grayscale
for i←1 to langkah do
    copy matriks_smoothing ke matriks_temp
    transform setiap baris citra
    copy matriks_smoothing ke matriks_temp
    transform setiap kolom citra
    ambil nilai detail horizontal dan vertikal
    copy matriks_dekomposisi ke matriks_temp_horizontal
    ambil nilai detail horizontal
    copy matriks_dekomposisi ke matriks_temp_vertikal
    ambil nilai detail vertikal
    jumlahkan nilai detail_horizontal dan detail_vertikal
    Bandingkan nilai detail dengan ambang
Endfor
Konvolusikan rerata dengan kernel wavelet
for i←1 to langkah do
    copy matriks_threshold ke matriks_temp
    transform invers setiap kolom citra
    copy matriks_threshold ke matriks_temp
    transform invers setiap baris citra
endfor
bandingkan dengan nilai ambang
Output (info_citra_hasil)

```

4.2.2.4 Deskripsi Tombol Wavelet2

Merupakan pilihan menu untuk melakukan pendeteksian tepi citra dengan menggunakan transformasi wavelet. Citra yang telah tertampil pada *PictureBox1* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya dan disimpan ke dalam suatu matriks temp. Citra yang akan dideteksi diubah menjadi citra *greyscale* dengan membagi nilai RGBnya dengan 3. Citra yang telah menjadi citra *greyscale* dikenakan operasi konvolusi dengan mengalikan matriks citra *greyscale* dengan suatu kernel konvolusi. Matriks citra yang telah tersmoothing

akan ditransformasi dengan transformasi wavelet. Disini dekomposisi yang digunakan adalah dekomposisi takstandar dengan mengkombinasikan pasangan transformasi baris dan kolom secara bergantian. Pertama akan dilakukan alihragam 1-dimensi pada baris dan kemudian dilanjutkan pada kolom. Proses akan dilakukan 1 level. Nilai dari detail vertikal dan detail horizontal dari hasil transformasi akan dicari nilai gradientnya dan membandingkannya dengan suatu nilai ambang. Selanjutnya matriks dekomposisi hasil transform akan dikenakan operasi transformasi invers wavelet. Pertama akan dilakukan alihragam invers 1-dimensi pada kolom dan kemudian dilanjutkan pada baris. Proses dilakukan 1 level. Matriks hasil dari invers akan diberi suatu nilai threshold untuk mendapatkan info citra hasil yang berupa citra terdeteksi.

Secara Prosedural :

On_Klik Wavelet

Input (citra_bmp,citra_jpg,citra_jpeg)

Baca matriks_citra

Ubah citra_asli ke citra_grayscale

inisialisasi kernel_smoothing

smoothing citra_grayscale

for i=1 to langkah do

 copy matriks_smoothing ke matriks_temp

 transform setiap baris citra

 copy matriks_smoothing ke matriks_temp

 transform setiap kolom citra

 ambil nilai detail horizontal dan vertikal

 copy matriks_dekomposisi ke matriks_temp_horizontal

 ambil nilai detail horizontal

 copy matriks_dekomposisi ke matriks_temp_vertikal

 ambil nilai detail vertikal

 jumlahkan nilai detail_horizontal dan detail_vertikal

 Bandingkan nilai detail dengan ambang

endfor

```

for i=1 to langkah do
  copy matriks_threshold ke matriks_temp
  transform invers setiap kolom citra
  copy matriks_threshold ke matriks_temp
  transform invers setiap baris citra
endfor
bandingkan dengan nilai ambang
Output (info_citra_hasil)

```

4.2.2.5 Deskripsi Tombol *Sobel*

Merupakan pilihan menu untuk melakukan pendeteksian tepi citra dengan menggunakan detektor *Sobel*. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya dan disimpan ke dalam suatu matriks temp. Citra yang akan dideteksi diubah menjadi citra *greyscale* dengan membagi nilai RGBnya dengan 3. Matriks *greyscale* dari citra akan dikalikan dengan suatu kernel konvolusi pada arah *Gx* dan *Gy*. Nilai dari konvolusi tersebut akan dicari nilai gradientnya dan nilainya dibandingkan dengan suatu nilai ambang untuk mendapatkan citra hasil deteksi.

Secara Prosedural :

```

On_Klik Sobel
Input (citra_bmp,citra_jpg,citra_jpeg)
Input (nilai_threshold)
Baca matriks_citra
Ubah citra_asli ke citra_grayscale
inisialisasi kernel Sx
Konvolusi Gx
inisialisasi kernel Sy
Konvolusi Gy
hitung nilai gradient
bandingkan dengan nilai ambang
Output (info_citra_hasil)

```

4.2.2.6 Deskripsi Tombol *Prewitt*

Merupakan pilihan menu untuk melakukan pendeteksian tepi citra dengan menggunakan detektor *Prewitt*. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya dan disimpan ke dalam suatu matriks temp. Citra yang akan dideteksi diubah menjadi citra *greyscale* dengan membagi nilai RGBnya dengan 3. Matriks *greyscale* dari citra akan dikalikan dengan suatu kernel konvolusi pada arah Gx dan Gy. Nilai dari konvolusi tersebut akan dicari nilai gradientnya dan nilainya dibandingkan dengan suatu nilai ambang untuk mendapatkan citra hasil deteksi.

Secara Prosedural :

On_Klik *Prewitt*

Input (*citra_bmp, citra_jpg, citra_jpeg*)

Input (*nilai_threshold*)

Baca *matriks_citra*

Ubah *citra_asli* ke *citra_grayscale*

inisialisasi kernel *Sx*

Konvolusi *Gx*

inisialisasi kernel *Sy*

Konvolusi *Gy*

hitung nilai *gradient*

bandingkan dengan nilai ambang

Output (*info_citra_hasil*)

4.2.2.7 Deskripsi Tombol *Compass*

Merupakan pilihan menu untuk melakukan pendeteksian tepi citra dengan menggunakan detektor *Compass*. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya dan disimpan ke dalam suatu matriks temp. Citra yang akan dideteksi diubah menjadi citra *greyscale*

dengan membagi nilai RGBnya dengan 3. Matriks *greyscale* dari citra akan dikalikan dengan kernel konvolusi *Compass* dari berbagai arah di dalam citra. Dari 8 nilai kernel konvolusi akan dicari nilai maksimalnya yang akan ditetapkan sebagai arah tepi.

Secara Prosedural :

On_Klik Compass

Input (citra_bmp,citra_jpg,citra_jpeg)

Input (nilai_threshold)

Baca matriks_citra

Ubah citra_asli ke citra_grayscale

inisialisasi 8 kernel Compass

konvolusikan matriks_citra dengan 8 kernel

cari nilai maksimalnya

bandingkan dengan nilai ambang

Output (info_citra_hasil)

4.2.2.8 Deskripsi Tombol Noise

Merupakan pilihan menu untuk menambahkan gangguan pada citra. Frame noise akan ditampilkan untuk menginputkan nilai noise. Nilai random x dan y akan ditentukan yang merupakan koordinat noise pada citra. Kemudian noise akan disisipkan sesuai dengan koordinat x dan y.

Secara Prosedural :

On_Klik Noise

Tampilkan frame noise

Input (nilai_noise)

On_Klik Ok

Cari nilai random x dan y

Sisipkan nilai noise ke citra

Output (info_citra_hasil)

4.2.2.9 Deskripsi Tombol Smoothing

Merupakan pilihan menu untuk melakukan penghalusan pada citra. Citra yang telah tertampil pada *PictureBox2*

akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya. Nilai RGB citra kemudian dikalikan dengan kernel konvolusi *gaussian smoothing* dan hasilnya dibagi dengan rerata.

Secara Prosedural :

On_Klik Smoothing

Baca matriks_citra

inisialisasi kernel smoothing

konvolusikan matriks_citra dengan kernel smoothing

bagi hasil konvolusi dengan rerata

Output (*info_citra_hasil*)

4.2.2.10 Deskripsi Tombol Low Pass Filter

Merupakan pilihan menu untuk melakukan pelembutan pada citra. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya. Nilai RGB citra kemudian dikalikan dengan kernel konvolusi *Low Pass Filter*.

Secara Prosedural :

On_Klik Low Pass Filter

Input(*pilih_kernel*)

Baca matriks_citra

inisialisasi kernel low pass filter

konvolusikan matriks_citra dengan kernel low_pass_filter

Output (*info_citra_hasil*)

4.2.2.11 Deskripsi Tombol High Pass Filter

Merupakan pilihan menu untuk melakukan penajaman pada citra. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya. Nilai RGB citra kemudian dikalikan dengan kernel konvolusi *High Pass Filter*.

Secara Prosedural :

On_Klik HighPass Filter

Input(*pilih_kernel*)

Baca matriks_citra
inisialisasi kernel high pass filter
konvolusikan matriks_citra dengan kernel high_pass _filter
Output (info_citra_hasil)

4.2.2.12 Deskripsi Tombol *Brightness*

Merupakan pilihan menu untuk mengubah kecerahan citra. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya. Nilai RGB citra kemudian dijumlahkankan dengan nilai *brightness* yang diinputkan. Nilai dari penjumlahan dibatasi tidak lebih kecil dari 0 dan tidak lebih besar dari 255.

Secara Prosedural :

On_Klik Brightness
Input(nilai_brightness)
Baca matriks_citra
Jumlahkan nilai matriks_citra dengan nilai brightness
Batasi dengan nilai ambang atas dan bawah
Output (info_citra_hasil)

4.2.2.13 Deskripsi Tombol *Histogram Equalisation*

Merupakan pilihan menu untuk panyamaan histogram citra. Citra yang telah tertampil pada *PictureBox2* akan dibaca matriksnya untuk mendapatkan nilai dari RGBnya. Kemudian akan dilakukan perhitungan nilai histogram berdasarkan tiap warna. Setelah jumlah piksel dan jumlah aras keabuan dalam citra didapat, dilakukan perhitungan untuk medapat penyamaan histogram.

Secara Prosedural :

On_Klik Brightness
Baca matriks_citra
Hitung jumlah histogram
Hitung jumlah piksel dan aras keabuan

Lakukan perhitungan penyamaan histogram

Output (info_citra_hasil)

4.2.2.14 Deskripsi Tombol MSE

Merupakan pilihan menu untuk menampilkan form MSE.

Secara Prosedural :

On_Klik MSE

Tampilkan form MSE

Sembunyikan form Main

4.2.2.15 Deskripsi Tombol Help and Support

Merupakan pilihan menu untuk menampilkan form Help and Support.

Secara Prosedural :

On_Klik Help and Support

Tampilkan help and support

4.2.2.16 Deskripsi Tombol About

Merupakan pilihan menu untuk menampilkan form About.

Secara Prosedural :

On_Klik About EDS

Tampilkan form About

4.2.2.17 Deskripsi Tombol Exit

Merupakan pilihan menu untuk keluar dari perangkat lunak EDS.

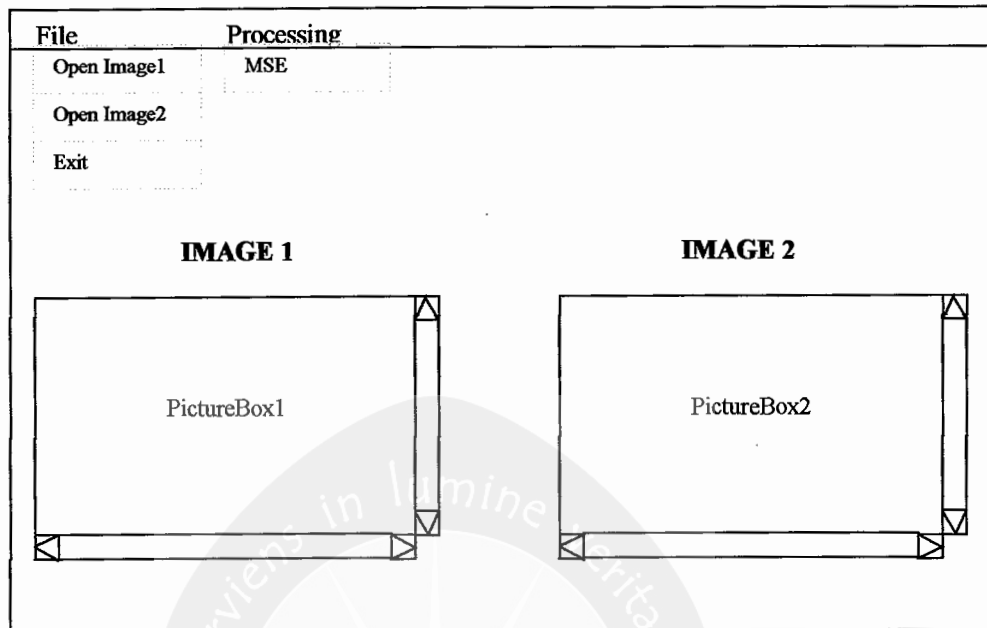
Secara Prosedural :

On_Klik Exit

Keluar sistem

4.3 Antarmuka Form MSE

4.3.1 Rancangan Antarmuka



Gambar 4 Rancangan Form MSE

4.3.2 Rancangan Prosedural/Fungsional

Form MSE akan tampil bila pilihan menu MSE pada menu form main dipilih. Saat form MSE ditampilkan secara otomatis form Main akan disembunyikan. Pada form ini terdapat 2 *PictureBox* yang digunakan untuk menampilkan citra yang akan dihitung nilai MSE-nya. *PictureBox1* digunakan untuk menampilkan citra tanpa gangguan yang telah terdeteksi, *PictureBox2* digunakan untuk menampilkan citra dengan gangguan yang telah terdeteksi. Pada form ini terdapat 4 tombol, yaitu Open Image1, Open Image2, Exit dan MSE. Tombol Open Image1 digunakan untuk menampilkan citra kedalam *PictureBox1*. Tombol Open Image2 digunakan untuk menampilkan citra kedalam *PictureBox2*. Tombol MSE digunakan untuk menghitung nilai MSE yang tertampil di *PictureBox1* dan

PictureBox2. Tombol *Exit* digunakan untuk keluar dari form *MSE* dan kembali ke form *Main*.

4.3.2.1 Deskripsi Tombol *Open*

Merupakan pilihan menu untuk mencari citra hasil deteksi dari suatu media penyimpanan dan menampilkannya pada *PictureBox*. Proses dimulai dengan memilih menu *Open Image1/Open Image2*. Dialog Box akan ditampilkan untuk mencari citra terdeteksi dan menampilkannya.

Secara Prosedural :

```
On_Klik Open
Tampilkan Dialog box Open
Input citra terdeteksi
If fname <> "" Then
    PictureBox ← LoadPicture(fname)
Endif
```

4.3.2.2 Deskripsi Tombol *MSE*

Merupakan pilihan menu untuk menghitung nilai *MSE* dari citra terdeteksi yang ditampilkan pada *PictureBox1* dan *PictureBox2*. Proses dimulai dengan membaca nilai dari matriks_citra pada *PictureBox1* dan matriks_citra pada *PictureBox2* kemudian dilakukan perhitungan nilai

MSE dengan rumus
$$\sqrt{\frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M \left(\int ij - \int ij \right)^2}$$

Dimana *N* adalah lebar citra, *M* adalah tinggi citra.

$\int ij$ adalah citra pada *PictureBox1* sedangkan $\int ij$ adalah citra pada *PictureBox2*.

Secara Prosedural :

```
On_Klik MSE
Baca matriks_citra PictureBox1 dan PictureBox2
Hitung nilai MSE
Tampilkan nilai MSE
```

Output (nilai_MSE)

4.3.2.3 Deskripsi Tombol Exit

Merupakan pilihan menu untuk keluar dari form MSE dan kembali ke form Main.

Secara Prosedural :

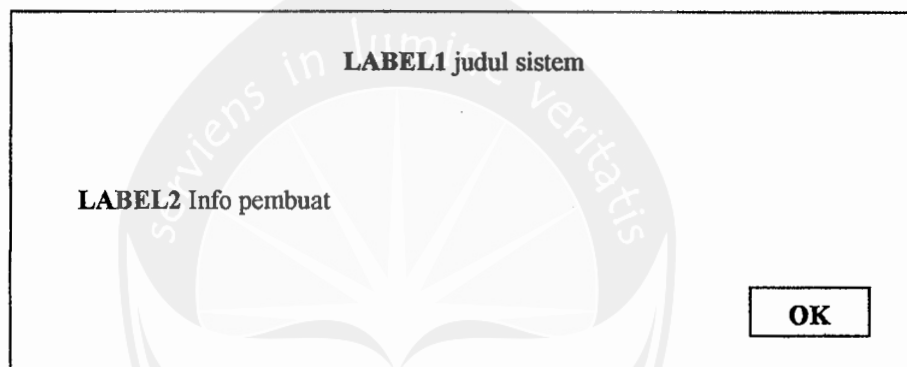
On_Klik Exit

Sembunyikan form MSE

Tampilkan form Main

4.4 Antarmuka Form About

4.4.1 Rancangan Antarmuka



The image shows a rectangular window representing the 'About' form. At the top center, there is a label 'LABEL1 judul sistem'. In the middle-left area, there is a label 'LABEL2 Info pembuat'. In the bottom-right corner, there is a rectangular button labeled 'OK'. A large, faint watermark of a university logo is visible in the background of the window.

Gambar 5 Rancangan Form About

4.4.2 Rancangan Prosedural/Fungsional

Form *about* akan tampil jika pada form menu utama dipilih menu *About EDS*. Pada form *about* terdapat label judul dan label info pembuat sistem. Pada form ini terdapat tombol OK, yang apabila ditekan akan membawa kembali ke form utama.

Secara Prosedural :

On_Klik OK

Tampilkan form utama

PDHUPL

**PERENCANAAN, DESKRIPSI, DAN HASIL
UJI PERANGKAT LUNAK**

**EDGE DETECTION SYSTEM
(EDS)**


Dipersiapkan oleh:

**I Nyoman Wira Sutha Kusuma
(3117/TF)**

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

Jl. Babarsari 43, Jogjakarta 50281

	Program Studi Teknik Informatika FTI - UAJY	Nomor Dokumen		Halaman
		<i>PDHUPL-EDS</i>		1/ 19
		Revisi	-	<i>Tgl:19 Agustus 2006</i>

DAFTAR PERUBAHAN

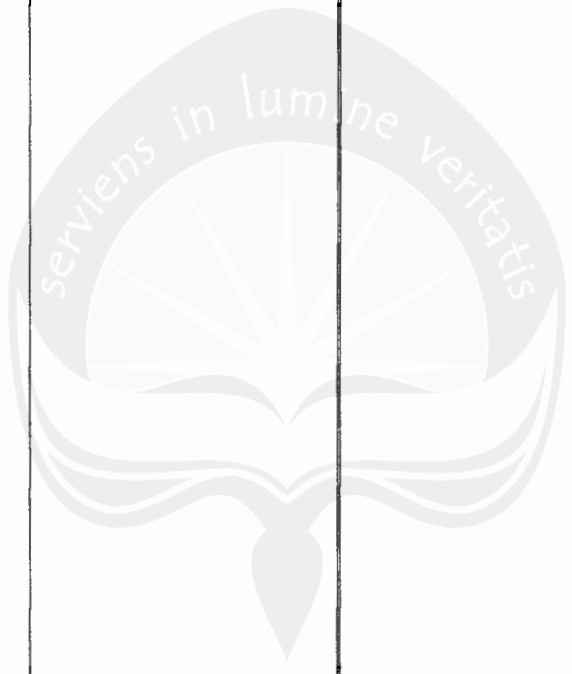
Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								



Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



Daftar Isi

1.	Pendahuluan	8
1.1	Tujuan Pembuatan Dokumen	8
1.2	Deskripsi Umum Sistem	8
1.3	Deskripsi Dokumen (Ikhtisar)	10
1.4	Definisi dan Singkatan	10
1.5	Dokumen Referensi	10
2	Lingkungan Pengujian Perangkat Lunak	10
2.1	Perangkat Lunak Pengujian	10
2.2	Perangkat Keras Pengujian	10
2.3	Material Pengujian	10
2.4	Sumber Daya Manusia	11
2.5	Prosedur Umum Pengujian	11
2.5.1	Persiapan Perangkat Keras	11
2.5.2	Persiapan Perangkat Lunak	11
2.5.3	Pelaksanaan	11
2.5.4	Pelaporan Hasil	11
3	Identifikasi dan Rencana Pengujian(acuan skplnya)	11
4	Deskripsi dan Hasil Uji	13
4.1	Identifikasi Butir Pengujian Proses Open File - PDHUPL-EDS-01	13
4.2	Identifikasi Butir Pengujian Proses Display - PDHUPL-EDS-02	13
4.3	Identifikasi Kelas Pengujian Proses Pengolahan Citra .	13
4.3.1	Identifikasi Butir Pengujian Pilih Menu - PDHUPL-EDS-03-01	13
4.3.2	Identifikasi Butir Pengujian Proses Invert - PDHUPL-EDS-03-02	14
4.3.3	Identifikasi Butir Pengujian Proses Grayscale - PDHUPL-EDS-03-03	14
4.3.4	Identifikasi Butir Pengujian Proses Brightness - PDHUPL-EDS-03-04	15
4.3.5	Identifikasi Butir Pengujian Proses Contrast - PDHUPL-EDS-03-05	15
4.3.6	Identifikasi Butir Pengujian Proses Emboss - PDHUPL-EDS-03-06	15
4.3.7	Identifikasi Butir Pengujian Proses Zoom - PDHUPL-EDS-03-07	15
4.4	Identifikasi Butir Pengujian Antarmuka Simpan Citra - EDS-04	15
4.5	Identifikasi Kelas Pengujian Antarmuka About - PDHUPL-EDS-05	15

Daftar Gambar



Daftar Tabel

	hal
Tabel 1. Definisi	11
Tabel 2. Identifikasi Pengujian	13
Tabel 3. Deskripsi dan Hasil Pengujian	16



Pendahuluan

1.1 Tujuan Pembuatan Dokumen

Dokumen PDHUPL-EDS ini adalah dokumen yang berisi perencanaan, deskripsi dan hasil pengujian perangkat lunak yang spesifikasi terdapat pada dokumen SKPL-EDS. Dokumen PDHUPL-EDS ini dibuat untuk jurusan Teknik Informatika Universitas Atma Jaya Yogyakarta. Selanjutnya dokumen PDHUPL-EDS ini dipergunakan sebagai bahan panduan untuk melakukan pengujian terhadap perangkat lunak Aplikasi Emboss pada Pengolahan Citra.

PDHUPL-EDS ini juga akan digunakan untuk menguji keseluruhan perangkat lunak Aplikasi *Edge Detection System* pada Pengolahan Citra.

1.2 Deskripsi Umum Sistem

Aplikasi *Edge Detection System* pada Pengolahan Citra merupakan sebuah aplikasi yang akan memberikan kebutuhan untuk memanipulasi citra khususnya pendeteksian tepi citra digital. Aplikasi ini mencakup beberapa kebutuhan dalam memanipulasi citra, seperti:

- Proses Open file.

Dalam proses ini dilakukan pengambilan citra dalam format *JPG, JPEG, BMP*.

- Proses Display.

Dalam proses ini dilakukan penampilan citra pada layar monitor.

- Proses Pengolahan Citra.

Proses pengolahan citra dalam perangkat lunak ini merupakan proses memanipulasi citra sehingga dihasilkan citra yang diinginkan. Proses manipulasi citra bisa

dilakukan sesuai dengan fungsi-fungsi yang terdapat pada perangkat lunak.

- Proses Hitung MSE.

Pada proses ini dilakukan penghitungan nilai MSE.

- Proses Simpan Citra.

Pada proses ini dilakukan penyimpanan citra hasil hanya dalam format BMP.

1.3 Deskripsi Dokumen (Ikhtisar)

Dokumen PDHUPL-EDS ini mempunyai sistematika penulisan sebagai berikut:

Bagian 1. Pendahuluan

1.1. Tujuan Pembuatan Dokumen

1.2. Deskripsi Umum Sistem

1.3. Deskripsi Dokumen atau Ikhtisar

1.4. Definisi dan Singkatan

1.5. Dokumen Referensi

Bagian 2. Lingkungan Pengujian Perangkat Lunak

2.1. Perangkat Lunak Pengujian

2.2. Perangkat Keras Pengujian

2.3. Material Pengujian

2.4. Sumber Daya Manusia

2.5. Prosedur Umum Pengujian

2.5.1. Persiapan Perangkat Keras

2.5.2. Persiapan Perangkat Lunak

2.5.3. Pelaksanaan

2.5.4. Pelaporan Hasil

Bagian 3. Identifikasi dan Rencana Pengujian

Bagian 4. Deskripsi dan Hasil uji

1.4 Definisi dan Singkatan

Tabel 1. Definisi

Kata Kunci atau Frase	Definisi
SKPL-EDS	Dokumen yang berisi spesifikasi dari perangkat lunak yang dikembangkan
PDHUPL-EDS	Dokumen yang berisi tentang perencanaan, deskripsi dan hasil uji perangkat lunak.

1.5 Dokumen Referensi

1. PDHUPL EIS.
2. *Standard IEEE nomor ANSI/IEEE Std 1058.1-1987 (Reaffirmend 1993).*
3. *Software Engineering, 1997, Roger S. Pressman, Mc Graw-Hill International Edition.*

2 Lingkungan Pengujian Perangkat Lunak

2.1 Perangkat Lunak Pengujian

Perangkat lunak Pengujian berupa:

1. *Windows XP* dari *Microsoft* sebagai sistem operasi
2. *Microsoft Visual Basic 6.0* sebagai tools perancangan sistem

2.2 Perangkat Keras Pengujian

1. Komputer PC dengan spesifikasi cukup untuk menjalankan *Microsoft Visual Basic 6.0* diatas sistem operasi *Microsoft Windows XP*.

2.3 Material Pengujian

Material untuk pengujian ini yaitu contoh hasil keluaran berupa citra hasil manipulasi.

2.4 Sumber Daya Manusia

Sumber daya pengujian ini berupa:

1. Tester (Pengguna penguji)
2. Dosen Penguji

2.5 Prosedur Umum Pengujian

2.5.1 Persiapan Perangkat Keras

Persiapan perangkat keras berupa : Persiapan program *Microsoft Visual Basic 6.0*. Keyboard standar windows, mouse dan monitor disiapkan. Perangkat Keras beserta spesifikasinya berupa:

1. VIA Samuel 2 796 MHz
2. RAM 128 MB

2.5.2. Persiapan Perangkat Lunak

1. Perangkat Lunak EDS disiapkan dalam harddisk computer yang akan digunakan.
2. Siapkan listing modul apa saja yang akan diuji.

2.5.3. Pelaksanaan

Pelaksanaan pengujian akan dilaksanakan dalam dua tahap, yaitu pengujian unit (modul-modul kecil) dan pengujian sistem secara keseluruhan.

2.5.4. Pelaporan Hasil

Hasil pengujian akan diserahkan kepada dosen penguji yang sekaligus sebagai dosen pembimbing.

3 Identifikasi dan Rencana Pengujian(acuan skplnya)

Tabel 2. Identifikasi Pengujian

Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian	Jadwal
		SKPL	PDHUPL			
Pengujian proses	Pengujian open file	SKPL_01	PDHUPL-EDS-01	Pengujian Unit	Black Box	25/07/2006

open file	citra.					
Pengujian proses display	Pengujian proses Display.	SKPL_02	PDHUPL-EDS-02	Pengujian Unit	Black Box	25/07/2006
Pengujian proses pengolahan citra	Pengujian Proses pilih menu	SKPL_03_01	PDHUPL-EDS-03-01	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Add Noise	SKPL_03_02	PDHUPL-EDS-03-02	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Gaussian Smoothing	SKPL_03_03	PDHUPL-EDS-03-03	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Lowpass-Filter	SKPL_03_04	PDHUPL-EDS-03-04	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Highpass-Filter	SKPL_03_05	PDHUPL-EDS-03-05	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Brightness	SKPL_03_06	PDHUPL-EDS-03-06	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Histogram Equalization	SKPL_03_07	PDHUPL-EDS-03-07	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses Edge Detection	SKPL_03_08	PDHUPL-EDS-03-08	Pengujian Unit	Black Box	25/07/2006
Pengujian Proses simpan citra	Pengujian Simpan citra	SKPL_04	PDHUPL-EDS-04	Pengujian Unit	Black Box	25/07/2006
	Pengujian Proses	SKPL_04_01	PDHUPL-EDS-04-01	Pengujian Unit	Black Box	25/07/2006

	menghitung nilai MSE		01			
Proses MSE	Pengujian Proses MSE	SKPL_05	PDHUPL-EDS-05	Pengujian Unit	Black Box	25/07/2006
Proses Help and Support	Pengujian Proses Help and Support	SKPL_06	PDHUPL-EDS-06	Pengujian Unit	Black Box	25/07/2006
Proses About	Pengujian Proses About	SKPL_07	PDHUPL-EDS-07	Pengujian Unit	Black Box	25/07/2006

4 Deskripsi dan Hasil Uji

4.1 Identifikasi butir Pengujian Proses Open File - PDHUPL-EDS-01

Kelas Pengujian antarmuka Open File yaitu kelas pengujian untuk proses membuka file citra. Pada proses ini dilakukan pengambilan citra dalam format *JPEG*, *JPG*, dan *BMP*.

4.2 Identifikasi butir Pengujian Proses Display - PDHUPL-EDS-02

Kelas Pengujian antarmuka display adalah kelas pengujian yang berhubungan dengan penampilan citra pada layar monitor sebelum dan sesudah manipulasi citra.

4.3 Identifikasi Kelas Pengujian Proses Pengolahan Citra

Kelas Pengujian antarmuka proses pengolahan citra adalah kelas pengujian yang meliputi pengujian-pengujian yang berhubungan dengan manipulasi citra.

4.3.1 Identifikasi Butir Pengujian Pilih Menu - PDHUPL-EDS-03-01

Butir ini memilih beberapa menu yang telah disediakan aplikasi untuk pengolahan citra.

4.3.2 Identifikasi Butir Pengujian Proses Add Noise - PDHUPL-EDS-03-02

Butir pengujian ini menguji *display* citra untuk menambahkan *random noise* pada citra.

4.3.3 Identifikasi Butir Pengujian Proses Gaussian Smoothing - PDHUPL-EDS-03-03

Butir pengujian ini menguji *display* citra untuk memberikan efek penghalusan.

4.3.4 Identifikasi Butir Pengujian Proses Lowpass-Filter - PDHUPL-EDS-03-04

Butir pengujian ini menguji *display* citra untuk melakukan pelembutan citra dengan menggunakan *lowpass filter*. Input berupa pilihan kernel *lowpass filter*.

4.3.5 Identifikasi Butir Pengujian Highpass-Filter - PDHUPL-EDS-03-05

Butir pengujian ini menguji *display* citra untuk melakukan penajaman citra dengan menggunakan *highpass filter*. Input berupa pilihan kernel *highpass filter*.

4.3.6 Identifikasi Butir Pengujian Proses Brightness - PDHUPL-EDS-03-06

Butir pengujian ini menguji *display* citra untuk mengatur kecerahan dari citra dengan memberi suatu konstanta peubah kecerahan citra dengan rentang nilai -255 sampai dengan 255 yang diinputkan melalui *textbox*.

4.3.7 Identifikasi Butir Pengujian Proses Histogram Equalization - PDHUPL-EDS-03-07

Butir pengujian ini menguji *display* citra untuk melakukan penyamaan nilai histogram.

4.3.8 Identifikasi Butir Pengujian Proses Edge Detection - PDHUPL-EDS-03-08

Butir pengujian ini menguji display citra untuk melakukan pendeteksian tepi citra display. Ada beberapa metode yang digunakan pada aplikasi ini, yaitu dengan menggunakan operator sobel, prewitt, compass dan dengan menggunakan metode wavelet.

4.4 Identifikasi Butir Pengujian Antarmuka Simpan Citra - PDHUPL-EDS-04

Butir pengujian ini akan menguji proses penyimpanan citra ke dalam media penyimpan hanya dalam format BMP.

4.5 Identifikasi Butir Pengujian Antarmuka MSE - PDHUPL-EDS-05

Menampilkan form perhitungan nilai MSE.

4.5.1 Identifikasi Butir Pengujian menghitung nilai MSE - PDHUPL-EDS-05

Butir pengujian ini akan menguji proses penghitungan nilai MSE.

4.6 Identifikasi Butir Pengujian Antarmuka Help and Support - PDHUPL-EDS-06

Menampilkan informasi cara menjalankan sistem dan dasar teori dari pendeteksian tepi.

4.7 Identifikasi Butir Pengujian Antarmuka About - PDHUPL-EDS-07

Menampilkan informasi dari pembuat aplikasi.

Tabel berikut adalah hasil pengujian dari setiap menu

Tabel 3. Deskripsi dan Hasil Pengujian

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yg diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PDHUPL-EDS-01	Pengujian Open file citra	Klik menu File1/File2-Open atau klik icon Open image pada toolbar	Klik file citra yang akan dimanipulasi melalui OpenFileDialog	Citra terdisplay pada area manipulasi	Citra terdisplay pada area manipulasi	Citra terdisplay	Handal
PDHUPL-EDS-02	Pengujian proses display	-	Citra	Citra terdisplay	Citra terdisplay	Citra terdisplay	Handal
PDHUPL-EDS-03-01	Pengujian Pilih Menu	Klik menu-menu pada main menu	-	Menu berfungsi dengan baik	Menu berfungsi dengan baik	Menu berfungsi dengan baik	Handal
PDHUPL-EDS-03-02	Pengujian Proses Add Noise	Klik menu Processing-Add Noise	-	Citra display diberi noise	Display : citra yang mengandung noise	Display : citra yang mengandung noise	Handal
PDHUPL-EDS-03-03	Pengujian Proses Gaussian Smoothing	Klik menu Processing-Gaussian Smoothing	-	Citra display di-smoothing	Display : citra yang di-smoothing	Display : citra yang di-smoothing	Handal

PDHUPL-EDS-03-04	Penguji an Proses Lowpass-Filer	Klik menu Processing- Filtering -Lowpass Filter	Input matriks kernel dengan memilih kernel- lowpass filter	Citra display menjadi lebih lembut	Display : perubahan kelembutan citra display	Display : perubahan kelembutan citra display	Handal
PDHUPL-EDS-03-05	Penguji an Proses Highpass-Filer	Klik menu Processing- Filtering -Highpass Filter	Input matriks kernel dengan memilih kernel highpass filter	Citra display menjadi lebih tajam	Display : perubahan ketajaman citra display	Display : perubahan kelembutan citra display	Handal
PDHUPL-EDS-03-06	Penguji an Proses brightness	Klik menu Processing- brightness	Input parameter dengan rentang nilai dari - 255 sampai 255	Berubahnya kecerahan citra display dengan memasukkan nilai parameter	Display : perubahan kecerahan citra display	Display : perubahan kecerahan citra display	Handal
PDHUPL-EDS-03-07	Penguji an Proses histogram equalization	Klik menu Processing- hist.Equalization	-	Citra terdisplay memiliki kecerahan yang sama	Display : Citra yang ter- histogram equalization	Display : Citra yang ter- histogram equalization	Handal

PDHUPL-EDS-03-08	Penguji an Proses Edge Detection operator atau klik menu Edge Detection (wavelet) kemudian pilih salah satu metode	Klik menu Edge Detection kemudian pilih operator atau klik menu Edge Detection (wavelet) kemudian pilih salah satu metode	Input parameter threshold > 0	Citra terdeteksi tepi	Display : Citra terdeteksi tepi	Display : Citra terdeteksi tepi	Handal
PDHUPL-EDS-04	Penguji an proses simpan citra	Klik menu File-Save atau klik icon Save image, muncul SaveDialogBox	Nama citra	Citra tersimpan setelah dimanipulasi	Citra tersimpan setelah dimanipulasi	Citra tersimpan setelah dimanipulasi	Handal
PDHUPL-EDS-05	Penguji an antarmuka MSE	Klik menu Processing-MSE	-	Muncul antarmuka MSE	Citra tersimpan setelah dimanipulasi	Citra tersimpan setelah dimanipulasi	Handal
PDHUPL-EDS-05-01	Penguji an perhitungan MSE	Klik menu Processing-MSE	-	Nilai MSE tertampil	Nilai MSE tertampil	Nilai MSE tertampil	Handal

PDHUPL-EDS-06	Penguji an antarmuka Hekp and Support	Klik menu Help-Help and Support	-	Muncul antarmuka Help and Support	Display : antarmuka Help and Support	Display : antarmuka Help and Support	Handal
PDHUPL-EDS-07	Penguji an antarmuka ka About	Klik menu Help-About	-	Muncul antarmuka About	Display : antarmuka About	Display : antarmuka About	Handal

