

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pembahasan dari bab-bab sebelumnya beserta dengan lampiran yang telah dibuat, maka dapat ditarik kesimpulan dari pembangunan aplikasi *contrast stretching* citra X-Ray menggunakan *Logarithmic Image Processing*, yaitu :

1. Perangkat lunak COSTREXPAPIC "Contrast Stretching X-Ray Picture" ini telah berhasil dibangun dengan tujuan untuk meningkatkan kontras dan kejelasan citra X-Ray.
2. Metode *Logarithmic Image Processing* (LIP) dapat mengefektifkan peningkatan detail dalam area citra X-Ray yang sangat gelap atau sangat terang, sehingga dapat digunakan untuk meningkatkan citra yang tidak terlihat (*underexposed*) atau citra yang terlalu terang (*overexposed*).
3. Dengan pemilihan nilai parameter yang sesuai, LIP dapat menghasilkan kontras yang lebih baik daripada metode *Linear Contrast Stretching*, *Histogram Equalization*, maupun *Lee's Algorithm*.
4. Faktor-faktor yang mempengaruhi keberhasilan pengontrasan citra dengan menggunakan *Logarithmic Image Processing* (LIP) ini antara lain :
 - a. Citra yang akan dikontraskan. Citra dengan kontras yang sudah bagus biasanya tidak akan berubah apabila dilakukan pengontrasan.

- b. Parameter input yang dimasukkan oleh user. Nilai input sangat menentukan keberhasilan dalam melakukan pengontrasan. Nilai input yang sesuai, akan menghasilkan kontras yang bagus.

5.2 Saran

Beberapa saran yang dapat ditarik dari proses analisis hingga pembuatan dokumentasi Tugas Akhir ini dan untuk pengembangan lebih lanjut pembangunan aplikasi *contrast stretching* menggunakan *Logarithmic Image Processing* adalah sebagai berikut :

1. Metode pengontrasan dalam perangkat lunak ini menggunakan nilai rata-rata suatu jendela/matriks 3×3 (paling efektif), untuk pengembangan selanjutnya dapat menggunakan matriks yang lebih besar yang dapat menghasilkan gambar dengan tepi yang tajam/jelas yang mungkin diinginkan untuk aplikasi deteksi tepi.
2. Pada pengembangan perangkat lunak ini dapat dilakukan penambahan fungsi untuk melakukan pencetakan (*print out*) hasil pengontrasan yang sudah dilakukan.
3. Perangkat lunak ini dikhususkan untuk pengontrasan citra *grayscale*, sehingga dapat dikembangkan lebih lanjut untuk pengontrasan citra warna (*true color*).
4. Untuk citra ber-*noise*, *noise* dapat ditekan dengan teknik *Wavelet De-noising* atau *Weiner Filter*. Jadi citra dijadikan *de-noised* menggunakan salah satu teknik penghilang *noise* diatas, kemudian ditingkatkan dengan metode *Logarithmic Image Processing* (LIP).

DAFTAR PUSTAKA

- Ahmad, Usman, 2005, *Pengolahan Citra Digital & Teknik Pemrogramannya*, Graha Ilmu, Yogyakarta.
- Akhadi, Mukhlis, 2001, *Napak Tilas 106 Tahun Perjalanan Sinar-X : 8 Nopember 1895 - 8 Nopember 2001*, Badan Tenaga Nuklir Nasional.
- Albahari, Ben, Drayton, Meter, Merril, Brad, 2001, *C# Essentials*, O'reilly, U.S.A.
- Budiharto, Widodo, Sukmadi, Dodi Yogi, 2004, *Aplikasi e-Commerce Menggunakan Visual C# .NET*, Andi, Yogyakarta.
- Dwiandiyanta, B Yudi, 2004, *Bahan Kuliah Pengolahan Citra*, Teknik Informatika FTI UAJY, Yogyakarta.
- Jain, Anil K, 1995, *Fundamentals of Digital Image Processing*, Prentice Hall, India.
- Jainudin, 2006, *Belajar Sendiri .NET dengan Visual C# 2005*, Andi, Yogyakarta.
- Munir, Rinaldi, 2004, *Pengolahan Citra Digital dengan Pendekatan Algoritmia*, Informatika, Bandung.
- Nugroho, Adi, 2005, *Rational Rose untuk Pemodelan Berorientasi Objek*, Informatika, Bandung.

Kimmel, Paul, 2002, *Advanced C# Programming*, McGraw-Hill/Osborne, U.S.A.

Schildt, Herbert, 2002, *The Complete Reference C#*, McGraw-Hill, U.S.A.

Troelsen, Andrew, 2001, *C# and The .NET Platform*, Apress, Berkeley.

Tarsito, 2005, *Review : Komputasi Tomografi dan Aplikasi dalam Proses Industri*, U.S.A.

Weisenbeck, Ben, Wong, Oiki, 2004, *Logarithmic Image Processing*, Department of Electrical and Computer Engineering.

Weisenbeck, Benjamin J, Wong, Oiki, 2004, *Image Enhancement Using Logarithmic Image Processing (LIP) Technique*, Department of Electrical and Computer Engineering.



LAMP IRAN

SKPL

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

**CONTRAST STRETCHING X-RAY PICTURE
(COSTREXRAPIC)**

Disusun oleh:


Ningrum Pratiwi

02 07 03474

Program Studi Teknik Informatika

Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Universitas Atma Jaya Yogyakarta	Nomor Dokumen		Halaman
		<i>SKPL- COSTREXRAPIC</i>		1/42
		Revisi	<i>B</i>	08/12/2006

DAFTAR PERUBAHAN

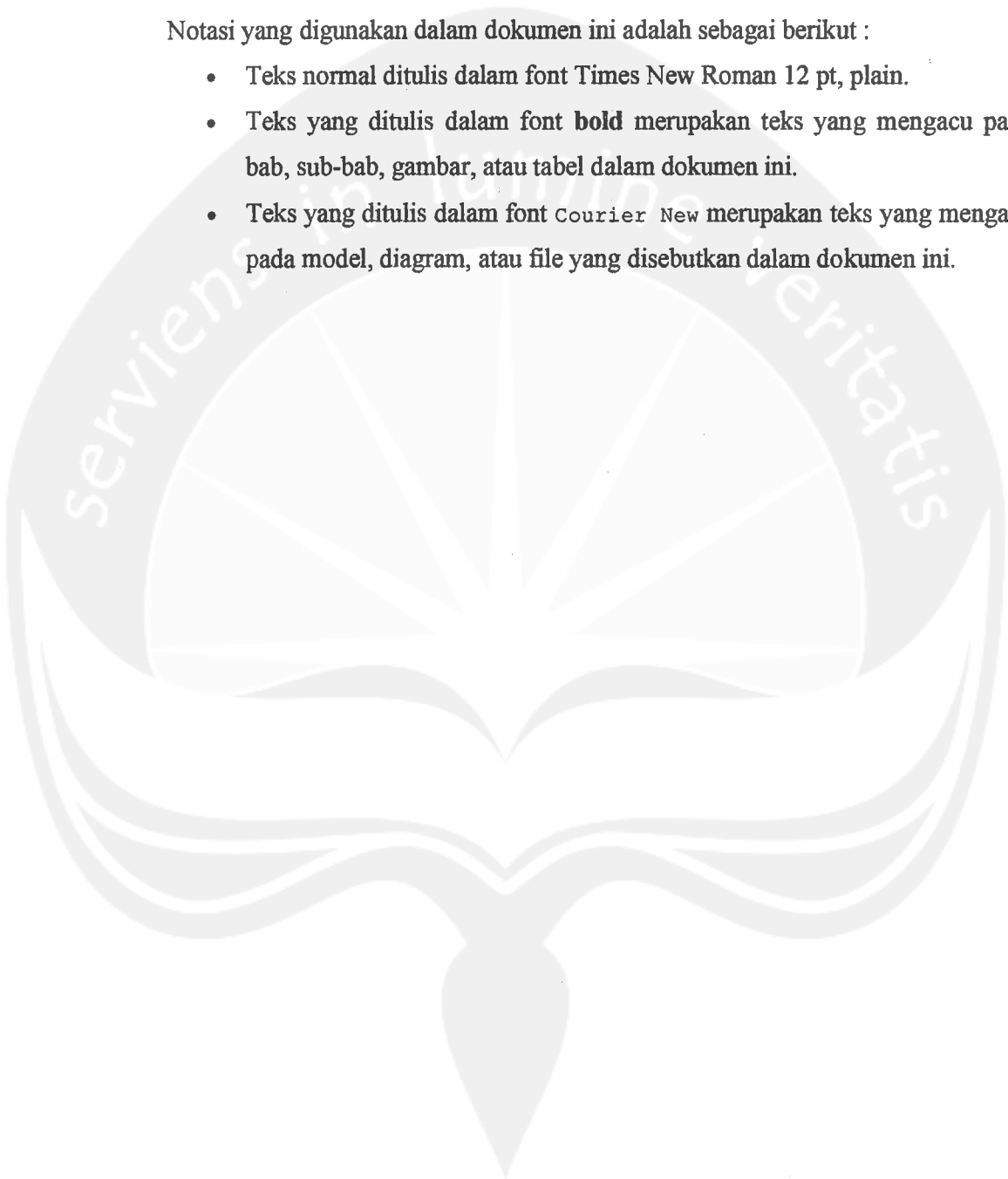
Revisi	Deskripsi
A	<ul style="list-style-type: none"> ▪ Menambah Basic Path pada Spesifikasi Use Case. ▪ Mengubah Analysis Collaboration Diagram.
B	<ul style="list-style-type: none"> ▪ Menambah Use Case View, Basic Path View, Collaboration Diagram View.
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F
Ditulis oleh	NP						
Diperiksa oleh	BYD KSN						
Disetujui oleh	BYD KSN						

NOTASI DOKUMEN

Notasi yang digunakan dalam dokumen ini adalah sebagai berikut :

- Teks normal ditulis dalam font Times New Roman 12 pt, plain.
- Teks yang ditulis dalam font **bold** merupakan teks yang mengacu pada bab, sub-bab, gambar, atau tabel dalam dokumen ini.
- Teks yang ditulis dalam font `Courier New` merupakan teks yang mengacu pada model, diagram, atau file yang disebutkan dalam dokumen ini.



DAFTAR ISI

1	Pendahuluan	8
1.1	Tujuan	8
1.2	Lingkup Masalah	8
1.3	Definisi Istilah dan Singkatan	8
1.4	Referensi	9
1.5	Deskripsi Umum (Overview)	9
2	Deskripsi Global Perangkat Lunak	10
2.1	Perspektif Produk	10
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	12
2.4	Batasan-batasan	13
2.5	Kebutuhan Fungsionalitas Perangkat Lunak	13
2.5.1	Use Case : OpenOriginalPicture	14
2.5.2	Use Case : LinearContrastStretching	14
2.5.3	Use Case : HistogramEqualization	14
2.5.4	Use Case : Lee'sAlgorithm	14
2.5.5	Use Case : LogarithmicImageProcessing	14
2.5.6	Use Case : View	15
2.5.7	Use Case : SavePicture	15
2.5.8	Use Case : ZoomIn	15
2.5.9	Use Case : ZoomOut	15
2.5.10	Use Case : AutoFit	15
2.6	Karakteristik Aktor	16
2.6.1	User	16
3	Deskripsi Rinci Kebutuhan	16
3.1	Spesifikasi Kebutuhan Fungsionalitas	16
3.1.1	Spesifikasi Use Case : OpenOriginalPicture	16
3.1.2	Spesifikasi Use Case : LinearContrastStretching	17
3.1.3	Spesifikasi Use Case : HistogramEqualization	19
3.1.4	Spesifikasi Use Case : Lee'sAlgorithm	20
3.1.5	Spesifikasi Use Case : LogarithmicImageProcessing	22
3.1.6	Spesifikasi Use Case : View	25
3.1.7	Spesifikasi Use Case : SavePicture	26
3.1.8	Spesifikasi Use Case : ZoomIn	28
3.1.9	Spesifikasi Use Case : ZoomOut	30
3.1.10	Spesifikasi Use Case : AutoFit	31
3.2	Spesifikasi Kebutuhan Non-Fungsionalitas	33
3.2.1	Kebutuhan Antarmuka Eksternal	33
3.2.2	Antarmuka Pemakai	33
3.2.3	Antarmuka Perangkat Keras	33
3.2.4	Antarmuka Perangkat Lunak	33
4	Realisasi Use Case	34
4.1	Interaction Diagram	34
4.1.1	Analysis Collaboration Diagram : Use Case OpenOriginalPicture	34
4.1.2	Analysis Collaboration Diagram : Use Case LinearContrastStretching	35
4.1.3	Analysis Collaboration Diagram : Use Case HistogramEqualization	36
4.1.4	Analysis Collaboration Diagram : Use Case Lee'sAlgorithm	37
4.1.5	Analysis Collaboration Diagram : Use Case LogarithmicImageProcessing	38

4.1.6	Analysis Collaboration Diagram : Use Case View	39
4.1.7	Analysis Collaboration Diagram : Use Case SavePicture	39
4.1.8	Analysis Collaboration Diagram : Use Case ZoomIn	40
4.1.9	Analysis Collaboration Diagram : Use Case ZoomOut.....	40
4.1.10	Analysis Collaboration Diagram : Use Case AutoFit	40
Apendiks A : Daftar Istilah dan Singkatan		41



DAFTAR GAMBAR

<i>Gambar 2.1 Use Case Diagram</i>	13
<i>Gambar 4.1 Analysis Collaboration Diagram : Use Case OpenOriginalPicture</i>	34
<i>Gambar 4.2 Analysis Collaboration Diagram : Use Case LinearContrastStretching</i>	35
<i>Gambar 4.3 Analysis Collaboration Diagram : Use Case HistogramEqualization</i>	36
<i>Gambar 4.4 Analysis Collaboration Diagram : Use Case Lee'sAlgorithm</i>	37
<i>Gambar 4.5 Analysis Collaboration Diagram : Use Case LogarithmicImageProcessing</i>	38
<i>Gambar 4.6 Analysis Collaboration Diagram : Use Case View</i>	39
<i>Gambar 4.7 Analysis Collaboration Diagram : Use Case SavePicture</i>	39
<i>Gambar 4.8 Analysis Collaboration Diagram : Use Case ZoomIn</i>	40
<i>Gambar 4.9 Analysis Collaboration Diagram : Use Case ZoomOut</i>	40
<i>Gambar 4.10 Analysis Collaboration Diagram : Use Case AutoFit</i>	40

DAFTAR TABEL

<i>Tabel 3.1 Spesifikasi Use Case : OpenOriginalPicture</i>	<i>16</i>
<i>Tabel 3.2 Spesifikasi Use Case : LinearContrastStretching.....</i>	<i>17</i>
<i>Tabel 3.3 Spesifikasi Use Case : HistogramEqualization</i>	<i>19</i>
<i>Tabel 3.4 Spesifikasi Use Case : Lee'sAlgorithm</i>	<i>20</i>
<i>Tabel 3.5 Spesifikasi Use Case : LogarithmicImageProcessing</i>	<i>22</i>
<i>Tabel 3.6 Spesifikasi Use Case : View</i>	<i>25</i>
<i>Tabel 3.7 Spesifikasi Use Case : SavePicture.....</i>	<i>26</i>
<i>Tabel 3.8 Spesifikasi Use Case : ZoomIn</i>	<i>28</i>
<i>Tabel 3.9 Spesifikasi Use Case : ZoomOut.....</i>	<i>30</i>
<i>Tabel 3.10 Spesifikasi Use Case : AutoFit.....</i>	<i>31</i>

1 Pendahuluan

1.1 Tujuan

Dokumen ini berisi penjelasan tentang Spesifikasi Kebutuhan Perangkat Lunak (SKPL). Dokumen SKPL ini merupakan dokumen spesifikasi kebutuhan perangkat lunak untuk *CO*ntrast *STRE*tching *X-RAY* *PICTure* (*COSTREXRAPIC*) versi 1.0 yang akan dibangun. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis untuk pengembangan perangkat lunak. Dalam SKPL ini akan dijelaskan mengenai kebutuhan-kebutuhan yang harus tersedia agar perangkat lunak yang diharapkan dapat terwujud.

1.2 Lingkup Masalah

Perangkat lunak *COSTREXRAPIC* dikembangkan dengan tujuan untuk :

1. Menangani proses perentangan kontras citra dengan menggunakan *Linear Contrast Stretching*.
2. Menangani proses perentangan kontras citra dengan menggunakan *Histogram Equalization*.
3. Menangani proses perentangan kontras citra dengan menggunakan *Lee's Algorithm*.
4. Menangani proses perentangan kontras citra dengan menggunakan *Logarithmic Image Processing*.

1.3 Definisi Istilah dan Singkatan

Definisi, istilah dan singkatan yang digunakan dalam dokumen ini mengacu pada **Apendiks A : Daftar Istilah dan Singkatan**.

1.4 Referensi

Referensi yang digunakan dalam pembuatan dokumen ini adalah :

1. Nugroho, Adi. *Rational Rose untuk Pemodelan Berorientasi Objek*. Informatika. 2005.
2. Ivar Jacobson, Grady Booch, James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley. 1998.
3. Martin Fowler, Kendall Scott. *UML Distilled – Second Edition*. Addison Wesley. 1999.
4. Stephen R Schach. *An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process*. Mc Graw-Hill. 2004.
5. Wendy Boggs, Michael Boggs. *Mastering UML With Rational Rose*. 2002.

1.5 Deskripsi Umum (Overview)

Dokumen SKPL ini dibagi menjadi empat bab. Bab pertama adalah **Pendahuluan**, yang berisi tentang deskripsi dokumen. Bab kedua adalah **Deskripsi Global Perangkat Lunak**, yang berisi penjelasan secara umum mengenai sistem yang akan dikembangkan meliputi fungsi-fungsi dari sistem, karakteristik pengguna, batasan dan asumsi yang diambil dalam pengembangan perangkat lunak. Bab ketiga adalah **Deskripsi Rinci Kebutuhan**, yang berisi penjelasan tentang kebutuhan sistem yang akan dikembangkan secara lebih rinci. Bab keempat adalah **Realisasi Use Case**, yang berisi realisasi use case dalam tahap analisis (konseptual), yang akan digunakan sebagai dasar realisasi use case pada tahap desain.

2 Deskripsi Global Perangkat Lunak

2.1 Perspektif Produk

COSTREXRAPIC adalah sebuah sistem informasi berbasis aplikasi *windows* yang dibangun untuk membantu dunia kedokteran dalam meningkatkan kualitas citra hasil *scanning* dengan sinar X (citra rontgen). Caranya dengan memperbaiki kontras citra sehingga menghasilkan kualitas citra yang lebih baik dari citra awal. Peningkatan kualitas citra ini akan menggunakan metode *Logarithmic Image Processing* dan sebagai pembandingnya adalah metode *Linear Contrast Stretching*, metode *Histogram Equalization* dan metode *Lee's Algorithm*. Sistem dapat mengatasi kelemahan-kelemahan hasil rontgen citra awal sehingga dapat lebih mudah diterjemahkan baik oleh dokter maupun orang awam.

Pengguna sistem *COSTREXRAPIC* adalah *User*. Pengguna berinteraksi dengan sistem yang ditampilkan di layar komputer dengan pilihan menu dalam form induk. Secara garis besar proses diawali dengan adanya *request* form induk dari user. Untuk melakukan proses pengontrasan citra, user terlebih dahulu membuka citra asli baru kemudian melakukan proses pengontrasan citra. Pengontrasan citra terbagi menjadi 4 metode, yaitu dengan metode *Linear Contrast Stretching*, metode *Histogram Equalization*, metode *Lee's Algorithm*, dan metode *Logarithmic Image Processing*.

Disamping menu-menu yang digunakan khusus untuk pengontrasan citra, dibuat juga menu pilihan yang lain yaitu *Save* yang digunakan untuk melakukan penyimpanan file citra, *Exit* untuk keluar dari sistem, *Zoom In* untuk melakukan perbesaran citra, *Zoom Out* untuk melakukan pengecilan citra, *Auto Fit* untuk melakukan penyesuaian citra dengan *PictureBox*, *Cascade* untuk display form secara *cascade* (menumpuk), *Tile Vertical* untuk display form secara vertikal, *Tile Horizontal* untuk display form secara horisontal. *View* digunakan untuk melihat perbandingan hasil pengontrasan keempat metode pengontrasan. Disamping itu untuk menutup

form yang terbuka dapat dilakukan dengan memilih *Close* (untuk satu form yang akan ditutup) atau *Close All Document* (untuk semua form yang berada pada form induk). Untuk mengetahui petunjuk cara menjalankan perangkat lunak dapat memilih *Help*, dan untuk mengetahui tentang versi atau seputar perangkat lunak dapat melihat *About*.

Perangkat lunak *COSTREXRAPIC* akan dibangun dengan menggunakan bahasa pemrograman *C# .Net*. Pengguna akan berinteraksi dengan perangkat lunak melalui antarmuka berbasis GUI (*Graphical User Interface*), sehingga piranti masukan yang digunakan adalah *mouse* dan *keyboard*.

2.2 Fungsi Produk

Fungsi produk perangkat lunak *COSTREXRAPIC* berdasarkan pengguna sistem adalah sebagai berikut :

1. Fungsi *Open Original Picture (SKPL-COSTREXRAPIC-01)* adalah fungsi yang digunakan oleh pengguna untuk membuka gambar asli sebelum proses peningkatan kontras citra.
2. Fungsi *Linear Contrast Stretching (SKPL-COSTREXRAPIC-02)* adalah fungsi yang digunakan untuk meningkatkan kontras citra dengan merentangkan kisaran nilai intensitas yang sempit menjadi kisaran intensitas yang diinginkan $[0,255]$.
3. Fungsi *Histogram Equalization (SKPL-COSTREXRAPIC-03)* adalah fungsi yang digunakan untuk meningkatkan kontras citra dengan melibatkan fungsi alih non linear dan tak monoton untuk memetakan intensitas piksel.
4. Fungsi *Lee's Algorithm (SKPL-COSTREXRAPIC-A-04)* adalah fungsi yang digunakan untuk meningkatkan kontras citra yang diusulkan oleh Lee, dimana algoritma Lee ini merupakan dasar pengontrasan citra metode *Logarithmic Image Processing*.

5. Fungsi *Logarithmic Image Processing* (SKPL-COSTREXRAPIC-05) adalah fungsi yang digunakan untuk meningkatkan kontras citra yang didasarkan pada suatu struktur matematika untuk pengolahan gambar yang logaritmis.
6. Fungsi *View* (SKPL-COSTREXRAPIC-06) adalah fungsi yang digunakan untuk melihat perbandingan hasil pengontrasan dengan keempat metode pengontrasan, yaitu *Linear Contrast Stretching*, *Histogram Equalization*, *Lee's Algorithm*, dan *Logarithmic Image Processing*.
7. Fungsi *Save Picture* (SKPL-COSTREXRAPIC-07) adalah fungsi yang digunakan oleh pengguna untuk menyimpan gambar hasil peningkatan kontras citra.
8. Fungsi *Zoom In* (SKPL-COSTREXRAPIC-08) adalah fungsi yang digunakan untuk memperbesar ukuran gambar.
9. Fungsi *Zoom Out* (SKPL-COSTREXRAPIC-09) adalah fungsi yang digunakan untuk memperkecil ukuran gambar.
10. Fungsi *Auto Fit* (SKPL-COSTREXRAPIC-10) adalah fungsi yang digunakan untuk menyesuaikan ukuran gambar dengan PictureBox.

2.3 Karakteristik Pengguna

Pengguna adalah orang yang akan menggunakan perangkat lunak *COSTREXRAPIC*. Dimana dalam hal ini adalah User, yang bertindak sebagai aktor dalam use case diagram. Secara umum karakteristik pengguna adalah sebagai berikut :

1. Memahami pengoperasian komputer secara aktif
2. Memahami sistem komputer tempat perangkat lunak dijalankan
3. Mengerti dan memahami perangkat lunak yang digunakan

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak *COSTREXRAPIC* tersebut adalah :

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak *COSTREXRAPIC*.

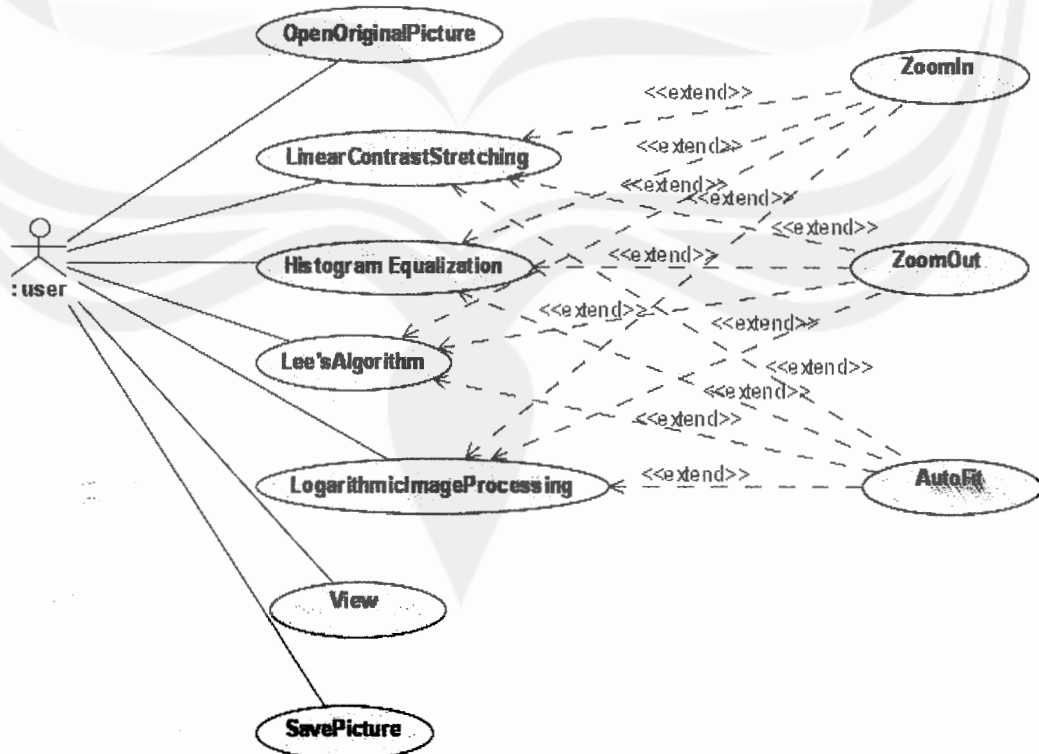
2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

3. Kebutuhan keandalan

Pengembangan perangkat lunak ini dibatasi pada kemudahan penggunaan dan kecepatan dalam proses pengolahannya.

2.5 Kebutuhan Fungsionalitas Perangkat Lunak



Gambar 2.1 Use Case Diagram

2.5.1 Use Case : OpenOriginalPicture

Use case ini digunakan oleh aktor untuk membuka citra asli (citra awal) sebelum terjadi pengontrasan citra.

Lihat : Spesifikasi Use Case : **OpenOriginalPicture**

2.5.2 Use Case : LinearContrastStretching

Use case ini digunakan oleh aktor untuk melakukan proses perentangan kontras citra dengan menggunakan metode *Linear Contrast Stretching*.

Lihat : Spesifikasi Use Case : **LinearContrastStretching**

2.5.3 Use Case : HistogramEqualization

Use case ini digunakan oleh aktor untuk melakukan proses perentangan kontras citra dengan menggunakan metode *Histogram Equalization*.

Lihat : Spesifikasi Use Case : **HistogramEqualization**

2.5.4 Use Case : Lee'sAlgorithm

Use case ini digunakan oleh aktor untuk melakukan proses perentangan kontras citra dengan menggunakan *Lee's Algorithm*.

Lihat : Spesifikasi Use Case : **Lee'sAlgorithm**

2.5.5 Use Case : LogarithmicImageProcessing

Use case ini digunakan oleh aktor untuk melakukan proses perentangan kontras citra dengan menggunakan metode *Logarithmic Image Processing*.

Lihat : Spesifikasi Use Case : **LogarithmicImageProcessing**

2.5.6 Use Case : View

Use case ini digunakan oleh aktor untuk melihat perbandingan gambar, histogram, maupun parameter input setelah proses pengontrasan untuk keempat metode, yaitu *Linear Contrast Stretching*, *Histogram Equalization*, *Lee's Algorithm*, dan *Logarithmic Image Processing*.

Lihat : **Spesifikasi Use Case : View**

2.5.7 Use Case : SavePicture

Use case ini digunakan oleh aktor untuk melakukan penyimpanan citra hasil perentangan kontras citra baik dengan metode *Linear Contrast Stretching*, *Histogram Equalization*, maupun *Logarithmic Image Processing*.

Lihat : **Spesifikasi Use Case : SavePicture**

2.5.8 Use Case : ZoomIn

Use case ini digunakan oleh aktor untuk memperbesar ukuran gambar.

Lihat : **Spesifikasi Use Case : ZoomIn**

2.5.9 Use Case : ZoomOut

Use case ini digunakan oleh aktor untuk memperkecil ukuran gambar.

Lihat : **Spesifikasi Use Case : ZoomOut**

2.5.10 Use Case : AutoFit

Use case ini digunakan oleh aktor untuk menyesuaikan ukuran gambar dengan ukuran *picture box*, agar semua bagian gambar dapat terlihat di layar monitor.

Lihat : **Spesifikasi Use Case : AutoFit**

2.6 Karakteristik Aktor

Aktor adalah representasi pemakai sistem. Aktor yang dimaksudkan dalam use case diagram adalah User, dalam hal ini adalah dokter atau radiolog yang nantinya akan memakai perangkat lunak *COSTREXRAPIC*.

2.6.1 User

User harus memenuhi hal-hal berikut :

- Mengerti tentang metode-metode yang digunakan dalam perentangan kontras citra untuk meningkatkan kualitas citra.
- Memahami cara melihat detail citra rontgen (citra *X-Ray*) sehingga diharapkan dapat mengetahui perbedaan citra setelah pemrosesan.
- Mengetahui file-file citra awal yang digunakan dalam proses perentangan kontras citra.

3 Deskripsi Rinci Kebutuhan

3.1 Spesifikasi Kebutuhan Fungsionalitas

3.1.1 Spesifikasi Use Case : OpenOriginalPicture

Tabel 3.1 Spesifikasi Use Case : OpenOriginalPicture

Use Case ID	UC-COSTREXRAPIC-01
Use Case Name	OpenOriginalPicture
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk melakukan pengelolaan pembukaan file gambar awal sebelum proses contrast stretching.
Preconditions	Aktor sudah berhasil memasuki sistem.
Basic Path	<ol style="list-style-type: none">1. Sistem menampilkan main menu atau form induk.2. Aktor memilih menu File dan sub menu Open.3. Sistem akan membuka Dialog Box Open File Original dimana terdapat direktori file-file yang

	<p>akan dibuka.</p> <p>4. Aktor memilih file gambar yang akan dikontraskan.</p> <p>5. Sistem me-load gambar terpilih ke PictureBox pada form Original Picture.</p>
Alternative Paths	-
Postconditions	File Original Picture akan terbuka sehingga aktor dapat melanjutkan pemrosesan.
Exception Paths	<p>1. Pembatalan open file gambar (setelah Tabel 3.1 Basic Path 3)</p> <p>a. Sistem akan kembali ke form utama.</p> <p>b. Kembali ke Tabel 3.1 Basic Path 3.</p>
Extends	-
Includes	-

3.1.2 Spesifikasi Use Case : LinearContrastStretching

Tabel 3.2 Spesifikasi Use Case : LinearContrastStretching

Use Case ID	UC-COSTREXPIC-02
Use Case Name	LinearContrastStretching
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk mengkontraskan gambar dengan metode Linear Contrast Stretching.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan dan aktor sudah memilih menu Processing pada main form.
Basic Path	1. Sistem akan menampilkan beberapa sub menu dalam menu Processing.

	<ol style="list-style-type: none"> 2. Aktor memilih sub menu Linear Contrast Stretching. 3. Sistem me-load gambar original ke PictureBox pada form Linear Contrast Stretching. 4. Sistem menyesuaikan ukuran gambar original dengan ukuran PictureBox pada form Linear Contrast Stretching. 5. Sistem melakukan center gambar orginal pada form Linear Contrast Stretching. 6. Aktor menekan Button Histogram untuk melihat histogram citra sebelum dikontraskan. 7. Sistem memproses gambar original sehingga menghasilkan gambar histogram. 8. Aktor menginputkan nilai a, b, Va, dan Vb pada TextBox yang ada pada form Linear Contrast Stretching. 9. Aktor menekan Button Process untuk melakukan pengontrasan gambar. 10. Sistem akan memproses gambar original untuk mendapatkan piksel asli. 11. Sistem akan memproses input dari aktor dan piksel asli gambar yang sudah didapat, kemudian dimasukkan dalam rumus Linear Contrast Stretching sehingga didapat piksel hasil. 12. Sistem melakukan pengecekan piksel hasil sesuai dengan syarat metode Linear Contrast Stretching. 13. Sistem akan melakukan set pixel hasil yang didapat, ke dalam gambar original yang akan dikontraskan. 14. Aktor menekan Button Histogram untuk melihat histogram citra yang telah dikontraskan.
--	---

	15. Sistem memproses gambar yang telah dikontraskan sehingga menghasilkan gambar histogram.
Alternative Paths	-
Postconditions	Kontras gambar original yang telah terbuka sebelumnya pada PictureBox berubah sesuai dengan metode pengontrasan yang dipilih yaitu Linear Contrast Stretching dan histogram citra asli serta citra hasil pengontrasan dapat terlihat perbedaannya.
Exception Paths	-
Extends	-
Includes	-

3.1.3 Spesifikasi Use Case : HistogramEqualization

Tabel 3.3 Spesifikasi Use Case : HistogramEqualization

Use Case ID	UC-COSTREXPIC-03
Use Case Name	HistogramEqualization
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk melakukan pengontrasan gambar dengan menggunakan metode Histogram Equalization.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan dan aktor sudah memilih menu Processing pada main form.
Basic Path	1. Sistem akan menampilkan beberapa sub menu dalam menu Processing. 2. Aktor memilih sub menu Histogram Equalization. 3. Sistem me-load gambar original ke PictureBox

	<p>pada form Histogram Equalization.</p> <ol style="list-style-type: none"> 4. Sistem menyesuaikan ukuran gambar original dengan ukuran PictureBox pada form Histogram Equalization. 5. Sistem melakukan center gambar original pada form Histogram Equalization. 6. Aktor menekan Button Histogram. 7. Sistem akan memproses gambar original untuk mendapatkan piksel asli. 8. Sistem akan memproses histogram gambar original. 9. Aktor menekan Button Equalize. 10. Sistem melakukan pengontrasan gambar original sesuai dengan metode Histogram Equalization dan melakukan penyebaran histogram.
Alternative Paths	-
Postconditions	Kontras gambar original yang telah terbuka sebelumnya berubah kontras dan penyebaran histogramnya sesuai dengan metode yang dipilih yaitu Histogram Equalization.
Exception Paths	-
Extends	-
Includes	-

3.1.4 Spesifikasi Use Case : Lee'sAlgorithm

Tabel 3.4 Spesifikasi Use Case : Lee'sAlgorithm

Use Case ID	UC-COSTREXRAPIC-04
Use Case Name	Lee'sAlgorithm
Use Case Type	Essential
Priority	High

Actors	User
Description	Use case ini digunakan oleh aktor untuk melakukan pengontrasan gambar dengan menggunakan metode Lee's Algorithm.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan dan aktor sudah memilih menu Processing.
Basic Path	<ol style="list-style-type: none"> 1. Sistem akan menampilkan beberapa sub menu dalam menu Processing. 2. Aktor memilih sub menu Lee's Algorithm. 3. Sistem me-load gambar original ke PictureBox pada form Lee's Algorithm. 4. Sistem menyesuaikan ukuran gambar original dengan ukuran PictureBox pada form Lee's Algorithm. 5. Sistem melakukan center gambar original pada form Lee's Algorithm. 6. Aktor menekan Button Histogram untuk melihat gambar histogram sebelum gambar original di kontraskan. 7. Sistem memproses gambar original sehingga menghasilkan gambar histogram. 8. Aktor menginputkan nilai α dan β pada TextBox. 9. Aktor menekan Button Process untuk melakukan pengontrasan gambar. 10. Sistem akan memproses gambar original untuk mendapatkan piksel asli. 11. Sistem akan memproses gambar original untuk mendapatkan rerata piksel dengan matrik 3×3.

	<p>12. Sistem memproses input dari aktor, piksel asli, dan rerata piksel, yang akan dimasukkan dalam rumus Lee's Algorithm sehingga didapat piksel hasil.</p> <p>13. Sistem melakukan pengecekan piksel hasil sesuai dengan syarat metode Lee's Algorithm.</p> <p>14. Sistem akan melakukan set pixel hasil yang didapat ke dalam gambar original yang akan dikontraskan.</p> <p>15. Aktor menekan Button Histogram kembali untuk melihat histogram citra setelah dikontraskan.</p> <p>16. Sistem memproses gambar hasil pengontrasan sehingga menghasilkan gambar histogram.</p>
Alternative Paths	-
Postconditions	Kontras gambar original yang telah terbuka sebelumnya berubah kontrasnya sesuai dengan metode yang dipilih yaitu Lee's Algorithm dan gambar histogram sebelum gambar dikontraskan serta setelah dikontraskan dapat terlihat.
Exception Paths	-
Extends	-
Includes	-

3.1.5 Spesifikasi Use Case : LogarithmicImageProcessing

Tabel 3.5 Spesifikasi Use Case : LogarithmicImageProcessing

Use Case ID	UC-COSTREXRAPIC-05
Use Case Name	LogarithmicImageProcessing
Use Case Type	Essential
Priority	High
Actors	User

Description	Use case ini digunakan oleh aktor untuk melakukan pengkontrasan gambar dengan menggunakan metode Logarithmic Image Processing.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan dan aktor sudah memilih menu Processing.
Basic Path	<ol style="list-style-type: none"> 1. Sistem akan menampilkan beberapa sub menu dalam menu Processing. 2. Aktor memilih sub menu Logarithmic Image Processing. 3. Sistem me-load gambar original ke PictureBox pada form Logarithmic Image Processing. 4. Sistem menyesuaikan ukuran gambar original dengan ukuran PictureBox pada form Logarithmic Image Processing. 5. Sistem melakukan center gambar original pada form Logarithmic Image Processing. 6. Aktor menekan Button Histogram untuk melihat histogram gambar original sebelum dikontraskan. 7. Sistem memproses gambar original sehingga menghasilkan gambar histogram. 8. Aktor menginputkan nilai α dan β pada TextBox. 9. Aktor menekan Button Process untuk melakukan pengontrasan gambar. 10. Sistem akan melakukan alih ragam aras keabuan piksel gambar original. 11. Sistem akan memproses gambar original untuk mendapatkan rerata piksel dengan matrik 3×3. 12. Sistem akan melakukan operasi logaritma untuk

	<p>mendapatkan nilai piksel hasil operasi logaritma.</p> <p>13. Sistem akan memproses nilai input dan rerata piksel untuk mendapatkan piksel hasil pemrosesan logaritma sesuai dengan rumus metode Logarithmic Image Processing.</p> <p>14. Sistem akan menghitung nilai eksponen dari piksel hasil logaritma yang sudah didapat.</p> <p>15. Sistem melakukan penghitungan untuk mendapatkan piksel hasil sesuai dengan rumus.</p> <p>16. Sistem melakukan pengecekan piksel hasil sesuai dengan syarat metode Logarithmic Image Processing.</p> <p>17. Sistem akan melakukan set pixel hasil yang didapat ke dalam gambar original yang akan dikontraskan.</p> <p>18. Aktor menekan kembali Button Histogram untuk melihat gambar histogram setelah gambar original dikontraskan.</p> <p>19. Sistem memproses citra yang sudah dikontraskan sehingga menghasilkan gambar histogram.</p>
Alternative Paths	-
Postconditions	Kontras gambar original yang telah terbuka sebelumnya berubah sesuai dengan metode yang dipilih yaitu Logarithmic Image Processing dan gambar histogram sebelum gambar dikontraskan serta sesudah dikontraskan dapat terlihat.
Exception Paths	-
Extends	-
Includes	-

3.1.6 Spesifikasi Use Case : View

Tabel 3.6 Spesifikasi Use Case : View

Use Case ID	UC-COSTREXRAPIC-06
Use Case Name	View
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk melihat perbandingan gambar, histogram, dan parameter input setelah proses pengontrasan.
Preconditions	<ol style="list-style-type: none">1. Use Case : LinearContrastStretching sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar dan,2. Use Case : HistogramEqualization sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar dan,3. Use Case : Lee'sAlgorithm sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar dan,4. Use Case : LogarithmicImageProcessing sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar.
Basic Path	<ol style="list-style-type: none">1. Sistem menampilkan gambar yang sudah dikontraskan pada PictureBox dalam form Linear Contrast Stretching, Histogram Equalization, Lee's Algorithm, dan Logarithmic Image Processing.2. Aktor memilih menu Window dan sub menu View untuk melihat perbandingan keempat metode yang sudah dikontraskan.

	3. Sistem akan memunculkan perbandingan gambar, histogram, dan parameter input pada frmView.
Alternative Paths	-
Postconditions	Gambar, Histogram, maupun parameter input hasil pengontrasan akan muncul dalam frmView.
Exception Paths	1. Ada metode pengontrasan yang belum dijalankan (setelah Tabel 3.6 Basic Path 2) <ul style="list-style-type: none"> a. Sistem tidak akan menampilkan frmView. b. Kembali ke Tabel 3.2 Basic Path 2, atau Tabel 3.3 Basic Path 2, atau Tabel 3.4 Basic Path 2, Tabel 3.5 Basic Path 2.
Extends	-
Includes	-

3.1.7 Spesifikasi Use Case : SavePicture

Tabel 3.7 Spesifikasi Use Case : SavePicture

Use Case ID	UC-COSTREXRAPIC-07
Use Case Name	SavePicture
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk meyimpan file gambar.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan sehingga gambar original sudah terbuka atau, 2. Use Case : LinearContrastStretching sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar atau, 3. Use Case : HistogramEqualization sudah

	<p>dilaksanakan dan aktor sudah berhasil mengkontraskan gambar atau,</p> <p>4. Use Case : Lee'sAlgorithm sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar atau,</p> <p>5. Use Case : LogarithmicImageProcessing sudah dilaksanakan dan aktor sudah berhasil mengkontraskan gambar.</p>
Basic Path	<ol style="list-style-type: none"> 1. Sistem menampilkan gambar yang belum dikontraskan maupun yang sudah dikontraskan pada PictureBox dalam form Original Picture, Linear Contrast Stretching, Histogram Equalization, Lee's Algorithm, atau Logarithmic Image Processing. 2. Aktor memilih menu File dan sub menu Save untuk menyimpan gambar yang sudah dikontraskan. 3. Sistem akan menampilkan Dialog Box Save File Original untuk menempatkan direktori dan nama file yang akan disimpan. 4. Aktor memilih direktori tempat file gambar akan disimpan dan menuliskan nama file yang akan disimpan. 5. Sistem akan melakukan penyimpanan file gambar dengan nama file dan direktori yang sudah ditentukan oleh aktor. 6. Sistem akan menampilkan Message Box untuk memperlihatkan status penyimpanan.
Alternative Paths	-
Postconditions	File gambar yang belum dikontraskan atau yang

	sudah dikontraskan tersimpan.
Exception Paths	2. Nama file yang akan disimpan sama (setelah Tabel 3.7 Basic Path 5) <ul style="list-style-type: none"> a. Sistem menampilkan peringatan bahwa nama file yang akan disimpan sama. b. Kembali ke Tabel 3.7 Basic Path 2.
Extends	-
Includes	-

3.1.8 Spesifikasi Use Case : ZoomIn

Tabel 3.8 Spesifikasi Use Case : ZoomIn

Use Case ID	UC-COSTREXPIC-08
Use Case Name	ZoomIn
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk memperbesar ukuran gambar.
Preconditions	<ol style="list-style-type: none"> 1. Use Case : OpenOriginalPicture sudah dilaksanakan dan gambar original sudah terbuka atau, 2. Use Case : LinearContrastStretching sudah dilaksanakan dan gambar original sudah dikontraskan atau, 3. Use Case : HistogramEqualization sudah dilaksanakan dan gambar original sudah dikontraskan atau, 4. Use Case : Lee'sAlgorithm sudah dilaksanakan dan gambar original sudah dikontraskan atau, 5. Use Case : LogarithmicImageProcessing sudah

	dilaksanakan dan gambar original sudah dikontraskan.
Basic Path	<ol style="list-style-type: none"> 1. Sistem menampilkan gambar yang belum atau yang sudah dikontraskan. 2. Aktor memilih menu Zoom dan sub menu Zoom In. 3. Aktor dapat melakukan Zoom In beberapa kali sampai mencapai batas ukuran terbesar. 4. Sistem akan memperbesar ukuran gambar dengan melakukan center image, dimana gambar zoom in diletakkan di bagian tengah PictureBox. 5. Sistem akan melakukan perbesaran sampai dengan 5 kali zoom in.
Alternative Paths	-
Postconditions	Gambar akan diperbesar ukurannya.
Exception Paths	<ol style="list-style-type: none"> 1. Gambar sudah di-Zoom In beberapa kali sehingga proses Zoom In akan berhenti (setelah Tabel 3.8 Basic Path 5) <ol style="list-style-type: none"> a. Sistem tidak akan memperbesar ukuran gambar lagi karena sudah mencapai ukuran maksimum yang di-set sistem. b. Kembali ke Tabel 3.9 Basic Path 2 atau kembali ke Tabel 3.10 Basic Path 1.
Extends	Use Case : LinearContrastStretching, HistogramEqualization, Lee'sAlgorithm, LogarithmicImageProcessing
Includes	-

3.1.9 Spesifikasi Use Case : ZoomOut

Tabel 3.9 Spesifikasi Use Case : ZoomOut

Use Case ID	UC-COSTREXRAPIC-09
Use Case Name	ZoomOut
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk memperkecil ukuran gambar.
Preconditions	<ol style="list-style-type: none">1. Use Case : OpenOriginalPicture sudah dilaksanakan dan gambar original sudah terbuka atau,2. Use Case : LinearContrastStretching sudah dilaksanakan dan gambar original sudah dikontraskan atau,3. Use Case : HistogramEqualization sudah dilaksanakan dan gambar original sudah dikontraskan atau,4. Use Case : Lee'sAlgorithm sudah dilaksanakan dan gambar original sudah dikontraskan atau,5. Use Case : LogarithmicImageProcessing sudah dilaksanakan dan gambar original sudah dikontraskan.
Basic Path	<ol style="list-style-type: none">1. Sistem menampilkan gambar yang belum atau yang sudah dikontraskan.2. Aktor memilih menu Zoom dan sub menu Zoom Out.3. Aktor dapat melakukan Zoom Out beberapa kali sampai mencapai batas ukuran terkecil.

	<p>4. Sistem akan memperkecil ukuran gambar dengan melakukan center image, dimana gambar zoom out diletakkan di bagian tengah PictureBox.</p> <p>5. Sistem akan melakukan pengecilan sampai dengan 10 kali zoom out.</p>
Alternative Paths	-
Postconditions	Gambar akan menjadi kecil ukurannya.
Exception Paths	<p>1. Gambar sudah di-Zoom Out beberapa kali sehingga proses Zoom Out akan berhenti (setelah Tabel 3.9 Basic Path 3)</p> <p>a. Sistem tidak akan memperkecil ukuran gambar lagi karena sudah mencapai ukuran minimum yang di-set sistem.</p> <p>b. Kembali ke Tabel 3.8 Basic Path 2 atau kembali ke Tabel 3.10 Basic Path 1.</p>
Extends	Use Case : LinearContrastStretching, HistogramEqualization, Lee'sAlgorithm, LogarithmicImageProcessing
Includes	-

3.1.10 Spesifikasi Use Case : AutoFit

Tabel 3.10 Spesifikasi Use Case : AutoFit

Use Case ID	UC-COSTREXPIC-10
Use Case Name	AutoFit
Use Case Type	Essential
Priority	High
Actors	User
Description	Use case ini digunakan oleh aktor untuk menyesuaikan ukuran gambar sesuai dengan picture box yang disediakan agar seluruh gambar dapat

	terlihat dengan jelas.
Preconditions	1. Use Case : OpenOriginalPicture sudah dilaksanakan dan gambar original sudah terbuka.
Basic Path	<ol style="list-style-type: none"> 1. Aktor memilih menu Zoom dan sub menu Auto Fit. 2. Sistem akan menyesuaikan ukuran gambar original, sesuai dengan ukuran picture box. 3. Sistem akan menyesuaikan ukuran gambar dengan ukuran PictureBox apabila lebar, dan tinggi gambar lebih besar dari ukuran PictureBox. 4. Sistem akan mengeset gambar di tengah PictureBox. 5. Sistem akan mengeset gambar di tengah PictureBox apabila lebar dan tinggi gambar lebih kecil dari ukuran PictureBox.
Alternative Paths	-
Postconditions	Gambar akan menyesuaikan dengan ukuran picture box pada layar monitor.
Exception Paths	<ol style="list-style-type: none"> 1. Gambar sudah di-Auto Fit sehingga proses Auto Fit akan berhenti (setelah Tabel 3.10 Basic Path 2). <ol style="list-style-type: none"> a. Apabila ukuran gambar lebih besar dari ukuran picture box, sistem akan menyesuaikan ukuran gambar sesuai dengan picture box. b. Apabila ukuran gambar lebih kecil dari ukuran picture box, sistem akan meletakkan gambar di tengah picture box tanpa mengubah ukuran aslinya. c. Kembali ke Tabel 3.10 Basic Path 2.

Extends	Use Case : LinearContrastStretching, HistogramEqualization, Lee'sAlgorithm, LogarithmicImageProcessing
Includes	-

3.2 Spesifikasi Kebutuhan Non-Fungsionalitas

3.2.1 Kebutuhan Antarmuka Eksternal

Kebutuhan antar muka eksternal pada sistem *COSTREXRAPIC* mencakup kebutuhan antarmuka pemakai, antarmuka perangkat keras dan antarmuka perangkat lunak.

3.2.2 Antarmuka Pemakai

Pemakai berinteraksi langsung dengan sistem *COSTREXRAPIC* dengan antarmuka berbasis GUI. Piranti masukan yang digunakan untuk memasukkan data masukan adalah *keyboard* dan *mouse*. Sedangkan keluaran dari sistem berupa file gambar yang akan disimpan serta akan ditampilkan langsung ke layar monitor.

3.2.3 Antarmuka Perangkat Keras

Antarmuka perangkat keras yang dibutuhkan dalam penggunaan perangkat lunak *COSTREXRAPIC* adalah:

- *Personal Computer*
- *Keyboard*
- *Mouse*

3.2.4 Antarmuka Perangkat Lunak

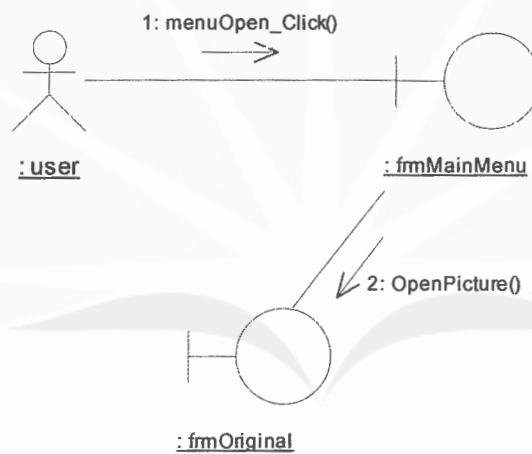
Perangkat lunak yang dibutuhkan untuk mendukung berjalannya perangkat lunak *COSTREXRAPIC* adalah :

- Nama : *Windows Me/NT/2000/XP*
- Sumber : *Microsoft*
- Fungsi : *Sistem Operasi Komputer*
- Nama : *C# .NET*
- Sumber : *Microsoft*
- Fungsi : *Tool perancang antarmuka aplikasi*

4 Realisasi Use Case

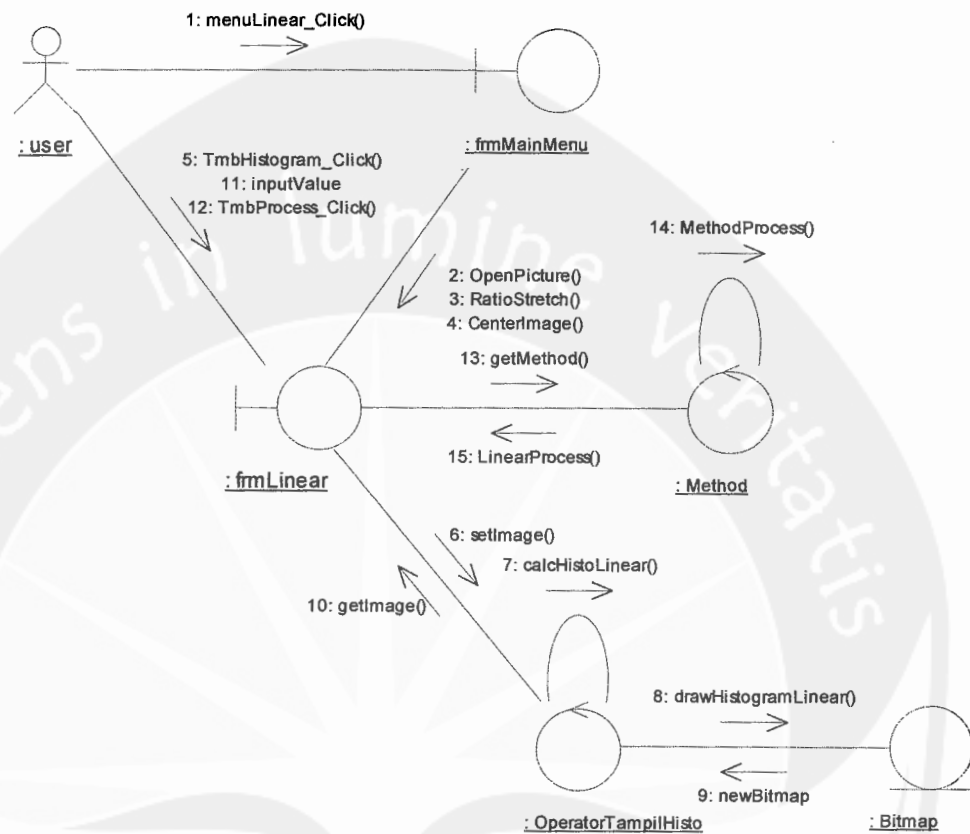
4.1 Interaction Diagram

4.1.1 Analysis Collaboration Diagram : Use Case OpenOriginalPicture



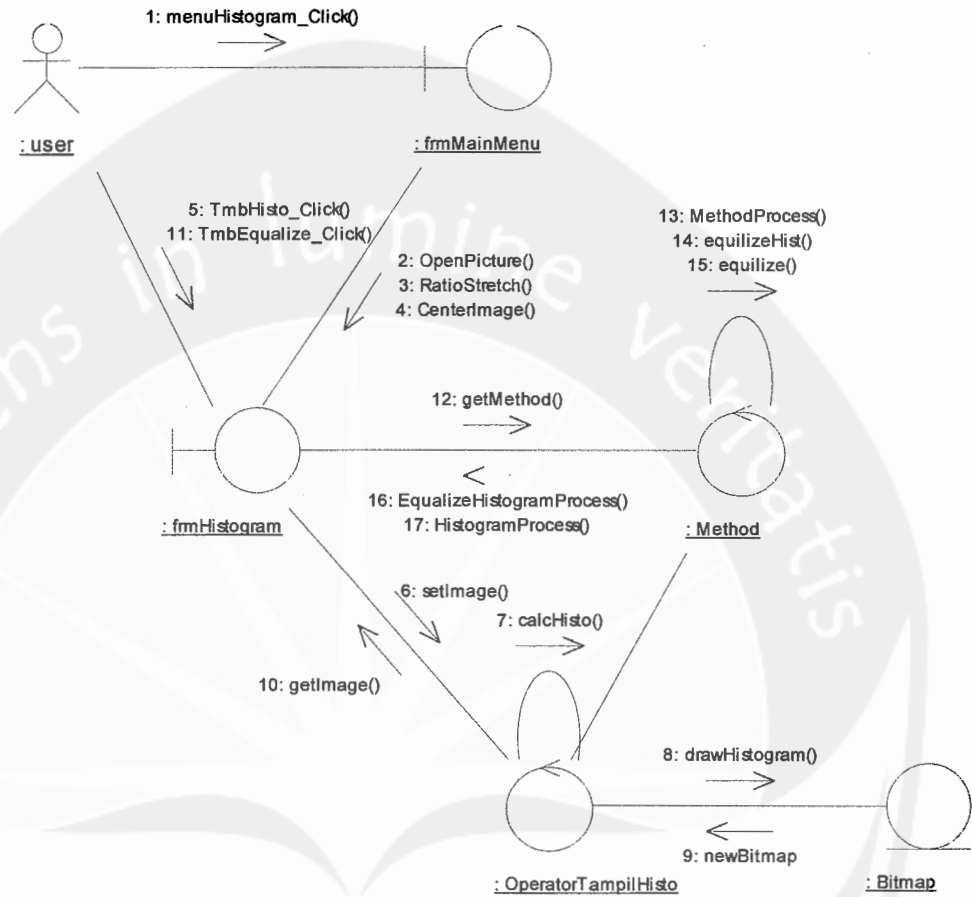
Gambar 4.1 Analysis Collaboration Diagram : Use Case OpenOriginalPicture

4.1.2 Analysis Collaboration Diagram : Use Case LinearContrastStretching



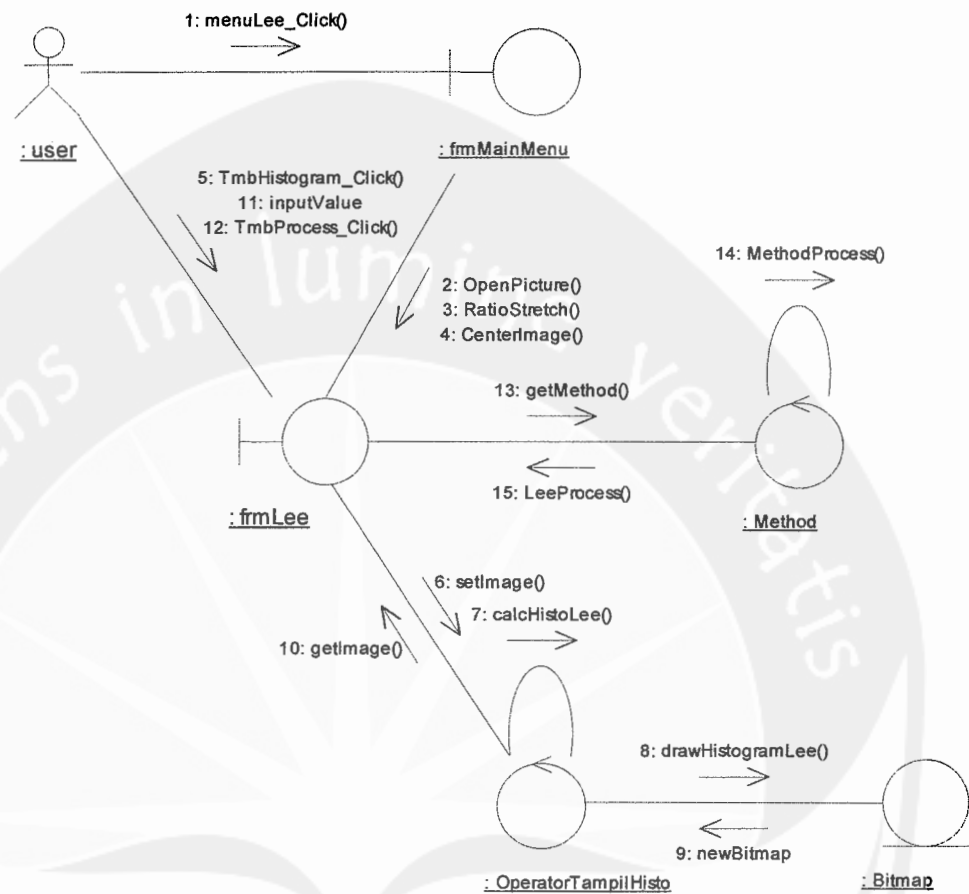
Gambar 4.2 Analysis Collaboration Diagram : Use Case LinearContrastStretching

4.1.3 Analysis Collaboration Diagram : Use Case HistogramEqualization



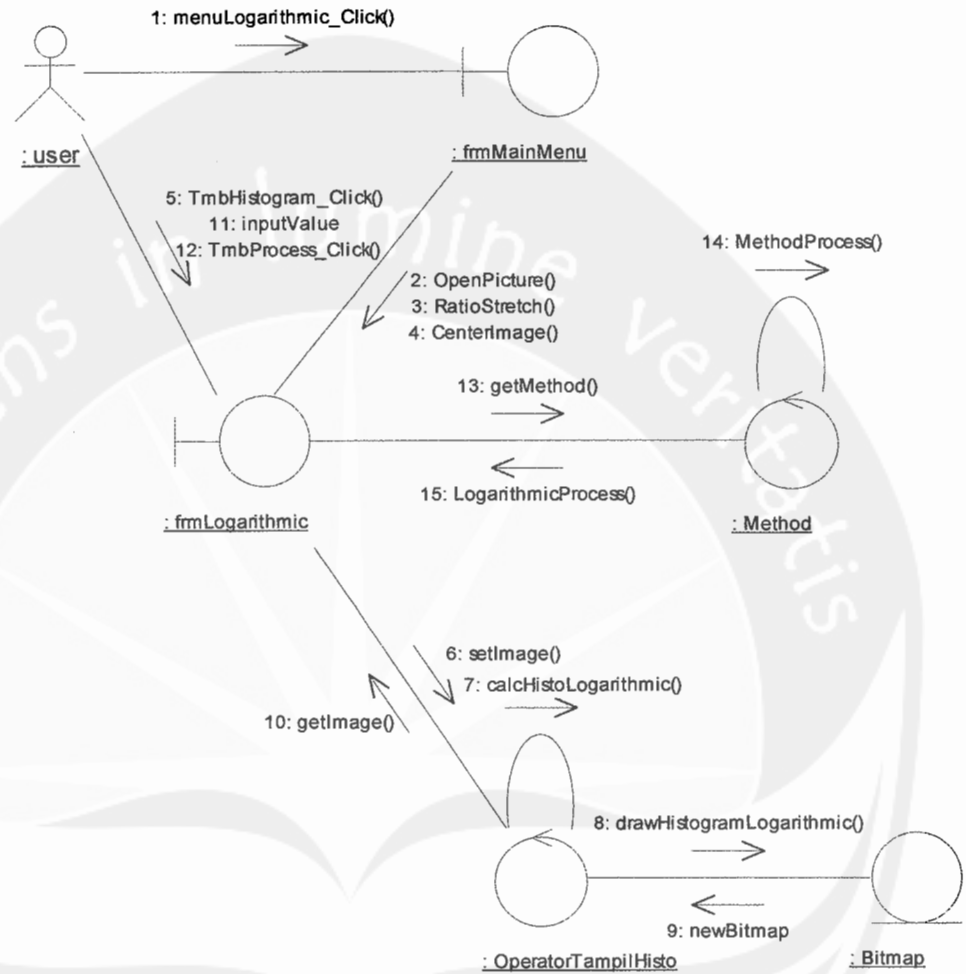
Gambar 4.3 Analysis Collaboration Diagram : Use Case HistogramEqualization

4.1.4 Analysis Collaboration Diagram : Use Case Lee'sAlgorithm



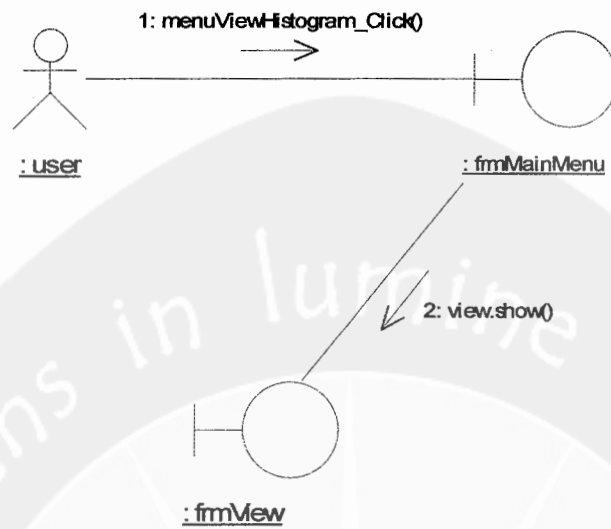
Gambar 4.4 Analysis Collaboration Diagram : Use Case Lee'sAlgorithm

4.1.5 Analysis Collaboration Diagram : Use Case LogarithmicImageProcessing



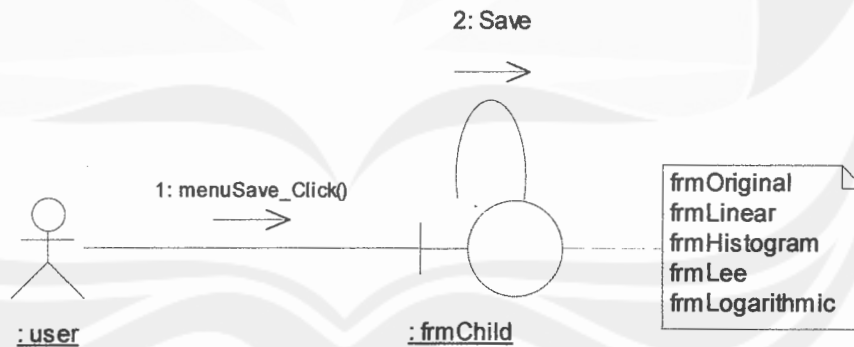
Gambar 4.5 Analysis Collaboration Diagram : Use Case LogarithmicImageProcessing

4.1.6 Analysis Collaboration Diagram : Use Case View



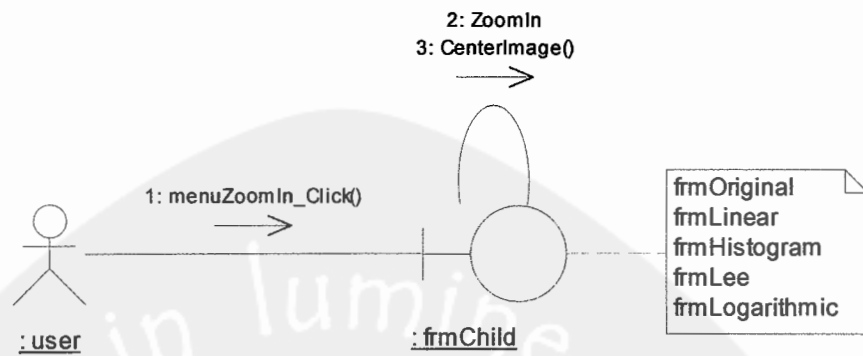
Gambar 4.6 Analysis Collaboration Diagram : Use Case View

4.1.7 Analysis Collaboration Diagram : Use Case SavePicture



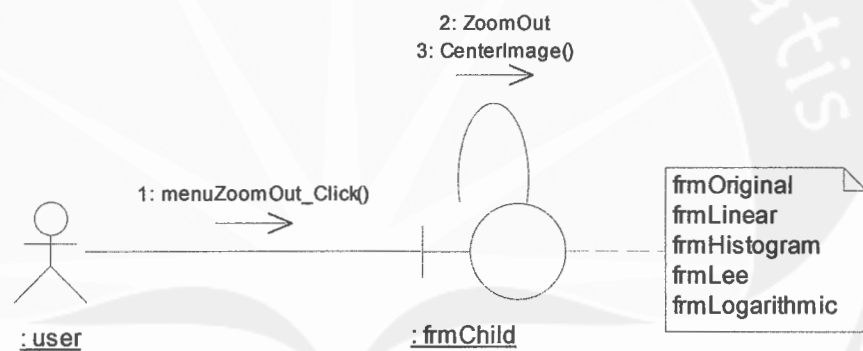
Gambar 4.7 Analysis Collaboration Diagram : Use Case SavePicture

4.1.8 Analysis Collaboration Diagram : Use Case ZoomIn



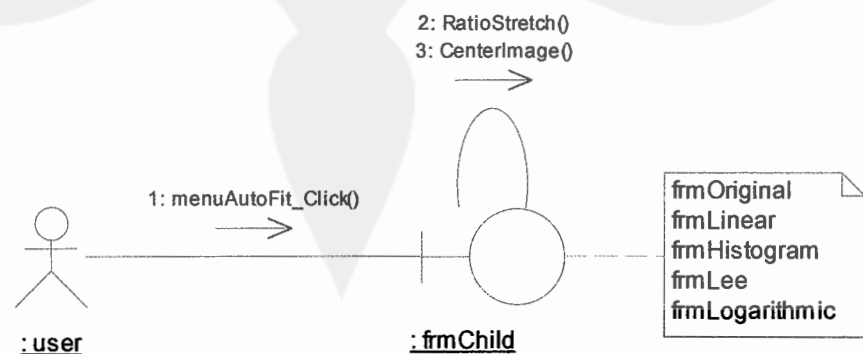
Gambar 4.8 Analysis Collaboration Diagram : Use Case ZoomIn

4.1.9 Analysis Collaboration Diagram : Use Case ZoomOut



Gambar 4.9 Analysis Collaboration Diagram : Use Case ZoomOut

4.1.10 Analysis Collaboration Diagram : Use Case AutoFit



Gambar 4.10 Analysis Collaboration Diagram : Use Case AutoFit

Apendiks A : Daftar Istilah dan Singkatan

Actor adalah representasi pemakai sistem (manusia atau sistem yang lain).

Auto Fit adalah cara untuk menyesuaikan ukuran gambar dalam picture box, agar gambar yang ditampilkan dapat terlihat semua. Apabila ukuran gambar lebih besar dari ukuran picture box maka gambar akan disesuaikan ukurannya dengan ukuran picture box, tetapi apabila ukuran gambar lebih kecil dari ukuran picture box maka ukuran gambar tidak akan diubah.

C# adalah bahasa pemrograman yang sederhana, moderen, berorientasi objek dan type-safe, yang diturunkan dari C dan C++. C# menggabungkan produktifitas dari Visual Basic dan kemampuan dari C++.

Collaboration Diagram adalah diagram yang menggambarkan interaksi antar kelas dan asosiasi. Dimodelkan sebagai pertukaran pesan antar kelas melalui asosiasi kelas dan merupakan diagram interaksi yang menekankan pada organisasi objek yang mengirim dan menerima pesan.

COSTREXPIC adalah singkatan dari *Contrast Stretching X-Ray Picture*, yaitu perangkat lunak yang ditujukan untuk dunia kedokteran, berfungsi untuk meningkatkan kualitas citra *x-ray* dengan mengkontraskan citra dengan 3 metode yaitu *Linear Contrast Stretching*, *Histogram Equalization*, dan *Logarithmic Image Processing*.

Histogram Equalization adalah salah satu metode peningkatan kualitas citra dengan melibatkan fungsi alih non linear dan tak monoton untuk memetakan intensitas piksel.

Lee's Algorithm adalah algoritma peningkatan kualitas citra oleh Lee yang nantinya akan diusulkan oleh Deng menjadi *Logarithmic Image Processing*.

Linear Contrast Stretching adalah salah satu metode peningkatan kualitas citra dengan merentangkan kisaran nilai intensitas yang sempit menjadi kisaran intensitas yang diinginkan [0,255].

Logarithmic Image Processing adalah salah satu metode peningkatan kualitas citra yang didasarkan pada suatu struktur matematika untuk pengolahan gambar yang logaritmis.

Microsoft Visual Studio .NET adalah sebuah *platform* untuk membangun, menjalankan dan meningkatkan generasi lanjut dari aplikasi terdistribusi. Referensi lengkap mengenai Microsoft Visual Studio .NET dapat dilihat pada <http://msdn.microsoft.com/vstudio/>

Use Case adalah representasi fungsionalitas atau layanan yang diberikan sistem kepada pemakai.

Use Case Diagram adalah penggambaran fungsionalitas sistem serta pemakai sistem.

User adalah pengguna perangkat lunak *COSTREXRAPIC*, meliputi **Use Case : Open Original Picture, Contrast Stretching, Linear Contrast Stretching, Histogram Equalization, Logarithmic Image Processing, dan Save Picture.**

Zoom adalah cara untuk merubah ukuran gambar. Zoom terbagi menjadi 2 yaitu zoom in dan zoom out. Zoom in digunakan untuk memperbesar ukuran gambar, sedangkan zoom out digunakan untuk memperkecil ukuran gambar.

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK


**CONTRAST STRETCHING X-RAY PICTURE
(COSTREXRAPIC)**

Disusun oleh:

Ningrum Pratiwi

02 07 03474

**Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		DPPL- COSTREXRAPIC		1/55
	Universitas Atma Jaya Yogyakarta	Revisi	A	08/12/2006

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	<ul style="list-style-type: none"> ▪ Mengubah Design Class ▪ Menambah Design Sequence Diagram View ▪ Menambah Rancangan Antarmuka frmView
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F
Ditulis oleh	NP						
Diperiksa oleh	BYD KSN						
Disetujui oleh	BYD KSN						

NOTASI DOKUMEN

Notasi yang digunakan dalam dokumen ini adalah sebagai berikut :

- Teks normal ditulis dalam font Times New Roman 12 pt, plain.
- Teks yang ditulis dalam font **bold** merupakan teks yang mengacu pada bab, sub-bab, gambar, atau tabel dalam dokumen ini.
- Teks yang ditulis dalam font Courier New merupakan teks yang mengacu pada model, diagram, atau file yang disebutkan dalam dokumen ini.



DAFTAR ISI

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim, dan Singkatan	6
1.4	Referensi	6
1.5	Deskripsi Umum (Overview)	7
2	Deskripsi Perancangan Arsitektural	7
2.1	Design Class	7
2.1.1	Pengantar	7
2.1.2	Package Dependencies	8
2.1.3	Package OpenOriginalPicture	8
2.1.3.1	Class Diagram Package OpenOriginalPicture	8
2.1.3.2	Class frmMainMenu	9
2.1.3.3	Class frmOriginal	11
2.1.4	Package MethodContrastStretching	12
2.1.4.1	Class Diagram Package MethodContrastStretching	12
2.1.4.2	Class frmLinear	13
2.1.4.3	Class frmHistogram	15
2.1.4.4	Class frmLee	16
2.1.4.5	Class frmLogarithmic	18
2.1.4.6	Class OperatorTampilHisto	19
2.1.4.7	Class Method	21
2.2	Realisasi Use Case	24
2.2.1	Use Case : OpenOriginalPicture	24
2.2.2	Use Case : LinearContrastStretching	25
2.2.3	Use Case : HistogramEqualization	27
2.2.4	Use Case : Lee'sAlgorithm	29
2.2.5	Use Case : LogarithmicImageProcessing	31
2.2.6	Use Case : View	32
2.2.7	Use Case : SavePicture	33
2.2.8	Use Case : Zoom In	34
2.2.9	Use Case : ZoomOut	35
2.2.10	Use Case : AutoFit	36
3	Deskripsi Perancangan Antarmuka	37
3.1	Antarmuka Main Menu	37
3.2	Use Case : OpenOriginalPicture	38
3.3	Use Case : LinearContrastStretching	40
3.4	Use Case : HistogramEqualization	42
3.5	Use Case : Lee'sAlgorithm	44
3.6	Use Case : LogarithmicImageProcessing	46
3.7	Use Case : View	48
3.8	Use Case : SavePicture	49
3.9	Use Case : ZoomIn	50
3.10	Use Case : ZoomOut	50
3.11	Use Case : AutoFit	51
	Apendiks A : Daftar Istilah dan Singkatan	54

DAFTAR GAMBAR

<i>Gambar 2.1 Package Dependencies COSTREXPIC</i>	8
<i>Gambar 2.2 Class Diagram Package OpenOriginalPicture</i>	8
<i>Gambar 2.3 Class frmMainMenu</i>	9
<i>Gambar 2.4 Class frmOriginal</i>	11
<i>Gambar 2.5 Class Diagram Package MethodContrastStretching</i>	12
<i>Gambar 2.6 Class frmLinear</i>	13
<i>Gambar 2.7 Class frmHistogram</i>	15
<i>Gambar 2.8 Class frmLee</i>	16
<i>Gambar 2.9 Class frmLogarithmic</i>	18
<i>Gambar 2.10 Class OperatorTampilHisto</i>	19
<i>Gambar 2.11 Class Method</i>	21
<i>Gambar 2.12 Design Sequence Diagram : Use Case OpenOriginalPicture</i>	24
<i>Gambar 2.13 Design Sequence Diagram : Use Case LinearContrastStretching</i>	25
<i>Gambar 2.14 Design Sequence Diagram : Use Case HistogramEqualization</i>	27
<i>Gambar 2.15 Design Sequence Diagram : Use Case Lee'sAlgorithm</i>	29
<i>Gambar 2.16 Design Sequence Diagram : Use Case LogarithmicImageProcessing</i>	31
<i>Gambar 2.17 Design Sequence Diagram : Use Case View</i>	32
<i>Gambar 2.18 Design Sequence Diagram : Use Case SavePicture</i>	33
<i>Gambar 2.19 Design Sequence Diagram : Use Case ZoomIn</i>	34
<i>Gambar 2.20 Design Sequence Diagram : Use Case ZoomOut</i>	35
<i>Gambar 2.21 Design Sequence Diagram : Use Case AutoFit</i>	36
<i>Gambar 3.1 Rancangan Antarmuka frmMainMenu</i>	37
<i>Gambar 3.2 Rancangan Antarmuka frmOriginal</i>	38
<i>Gambar 3.3 Rancangan Antarmuka frmLinear</i>	40
<i>Gambar 3.4 Rancangan Antarmuka frmHistogram</i>	42
<i>Gambar 3.5 Rancangan Antarmuka frmLee</i>	44
<i>Gambar 3.6 Rancangan Antarmuka frmLogarithmic</i>	46
<i>Gambar 3.7 Rancangan Antarmuka frmView</i>	48

1 Pendahuluan

1.1 Tujuan

Dokumen ini berisi penjelasan tentang Deskripsi Perancangan Perangkat Lunak (DPPL). Dokumen DPPL ini merupakan dokumen deskripsi perancangan perangkat lunak untuk *CONtrast STREtching X-RAY PICTure (COSTREXRAPIC)* versi 1.0 yang akan dibangun. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis untuk pengembangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat lunak *COSTREXRAPIC* dikembangkan dengan tujuan untuk :

1. Menangani proses perentangan kontras citra dengan menggunakan *Linear Contrast Stretching*.
2. Menangani proses perentangan kontras citra dengan menggunakan *Histogram Equalization*.
3. Menangani proses perentangan kontras citra dengan menggunakan *Lee's Algorithm*.
4. Menangani proses perentangan kontras citra dengan menggunakan *Logarithmic Image Processing*.

1.3 Definisi, Akronim, dan Singkatan

Untuk definisi istilah dan singkatan yang digunakan dalam dokumen ini dapat mengacu pada **Apendiks A : Daftar Istilah dan Singkatan**.

1.4 Referensi

Referensi yang digunakan dalam pembuatan dokumen ini adalah :

1. Nugroho, Adi. *Rational Rose untuk Pemodelan Berorientasi Objek*. Informatika. 2005.

2. Ivar Jacobson, Grady Booch, James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley. 1998.
3. Martin Fowler, Kendall Scott. *UML Distilled – Second Edition*. Addison Wesley. 1999.
4. Stephen R Schach. *An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process*. Mc Graw-Hill. 2004.
5. Wendy Boggs, Michael Boggs. *Mastering UML With Rational Rose*. 2002.

1.5 Deskripsi Umum (Overview)

Dokumen ini terdiri dari tiga bab. Bab pertama adalah **Pendahuluan**, yang berisi deskripsi dokumen. Bab kedua adalah **Deskripsi Perancangan Arsitektural**, yang berisi deskripsi arsitektur sistem. Bab ketiga adalah **Deskripsi Perancangan Antarmuka**, yang berisi deskripsi rancangan GUI yang digunakan sistem untuk berinteraksi dengan user.

2 Deskripsi Perancangan Arsitektural

2.1 Design Class

2.1.1 Pengantar

Stereotype yang digunakan dalam design class adalah :

- << boundary >>

Boundary class merupakan class yang berfungsi untuk menghubungkan sistem dengan user di luar sistem.

- << control >>

Control class adalah suatu class yang objek-nya melakukan interaksi antar sekelompok objek lain. Control class biasanya memiliki karakteristik yang spesifik untuk satu use case, dan objek class ini biasanya hanya aktif pada realisasi use case.

- << entity >>

Entity class adalah class yang bersifat pasif, dalam arti class tersebut tidak memulai interaksi dengan class lain. Entity class ini biasanya merepresentasikan suatu objek yang disimpan dalam persistent storage.

2.1.2 Package Dependencies



Gambar 2.1 Package Dependencies COSTREXRAPIC

2.1.3 Package OpenOriginalPicture

Package ini menyediakan class-class yang digunakan oleh program untuk mengelola hal-hal yang berhubungan dengan open picture.

2.1.3.1 Class Diagram Package OpenOriginalPicture



Gambar 2.2 Class Diagram Package OpenOriginalPicture

2.1.3.2 Class frmMainMenu



Gambar 2.3 Class frmMainMenu

Deskripsi

Class yang berperan sebagai boundary class. Class ini merupakan GUI yang digunakan sebagai main form yang digunakan untuk mendeklarasikan child form dan berhubungan dengan open picture, dan pemrosesan contrast picture.

Atribut

- - tampung : integer = -1
Merepresentasikan variabel tampung bertipe integer yang di set sama dengan -1.
- - tampung2 : integer = -1
Merepresentasikan variabel tampung2 bertipe integer yang di set sama dengan -1.
- - tampung3 : integer = -1
Merepresentasikan variabel tampung3 bertipe integer yang di set sama dengan -1.
- - tampung4 : integer = -1
Merepresentasikan variabel tampung4 bertipe integer yang di set sama dengan -1.
- - aktifHisto : integer = 0

Merepresentasikan variabel aktifHisto bertipe integer yang di set sama dengan 0.

- - aktifLin : integer = 0

Merepresentasikan variabel aktifLin bertipe integer yang di set sama dengan 0.

- - aktifLee : integer = 0

Merepresentasikan variabel aktifLee bertipe integer yang di set sama dengan 0.

- - aktifLog : integer = 0

Merepresentasikan variabel aktifLog bertipe integer yang di set sama dengan 0.

- + k : integer = 0

Merepresentasikan variabel k yang di set sama dengan 0.

- + l : integer = 0

Merepresentasikan variabel l yang di set sama dengan 0.

- + fileLoc : string

Merepresentasikan lokasi file gambar yang bertipe string.

- + fileLoc2 : string

Merepresentasikan lokasi file gambar (2) yang bertipe string.

- + imageLinear : Bitmap

Merepresentasikan imageLinear yang bertipe Bitmap.

- + imageHistoLinear : Bitmap

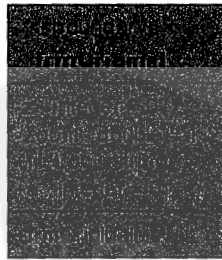
Merepresentasikan imageHistoLinear yang bertipe Bitmap.

Method

- + frmMainMenu()

Operasi ini digunakan untuk melakukan open picture yang merujuk ke frmOriginal dan merujuk ke frmLinear, frmHistogram, frmLee, dan frmLogarithmic untuk memanggil menu untuk mengontraskan gambar serta digunakan untuk melakukan view gambar maupun histogram dari keempat metode yang diproses.

2.1.3.3 Class frmOriginal



Gambar 2.4 Class frmOriginal

Deskripsi

Class yang berperan sebagai boundary class. Class ini merupakan GUI yang digunakan sebagai form yang berfungsi untuk melakukan pembukaan gambar original, melakukan zoom in, zoom out, dan auto fit.

Atribut

- - z : integer = 0
Merepresentasikan variabel z yang bertipe integer dan diset sama dengan 0.
- - Zoom : double = 1.2
Merepresentasikan variabel Zoom yang bertipe double dan diset dengan nilai 1.2 yang merupakan skala perbesaran.
- + fileLoc : string
Merepresentasikan lokasi file gambar dengan tipe string.
- + a : integer = 0
Merepresentasikan variabel a yang bertipe integer dan diset sama dengan 0.

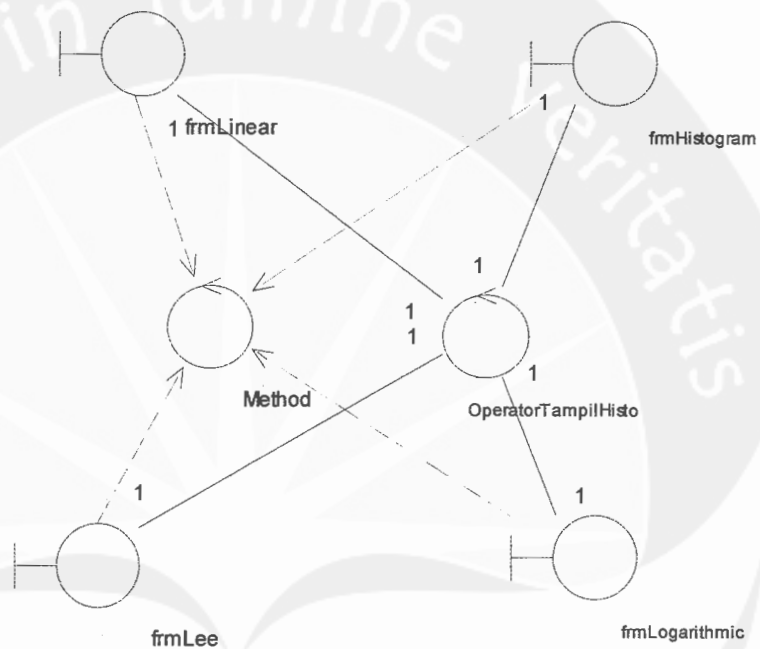
Method

- + frmOriginal()
Operasi ini digunakan untuk membuka gambar original yang akan diproses atau dikontraskan.

2.1.4 Package MethodContrastStretching

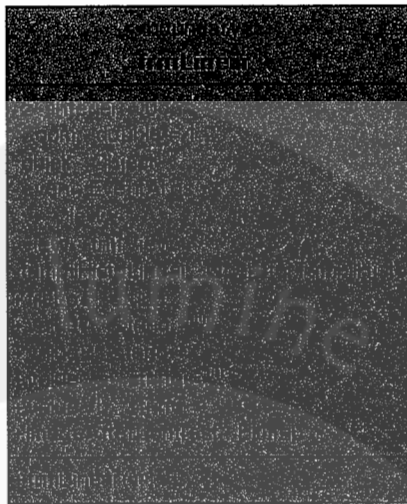
Package ini menyediakan class-class yang digunakan oleh program untuk mengelola hal-hal yang berhubungan dengan metode pengontrasan gambar.

2.1.4.1 Class Diagram Package MethodContrastStretching



Gambar 2.5 Class Diagram Package MethodContrastStretching

2.1.4.2 Class frmLinear



Gambar 2.6 Class frmLinear

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan GUI yang digunakan untuk melakukan pengontrasan original picture menjadi contrast picture dengan metode Linear Contrast Stretching.

Atribut

- - z : integer = 0
Merepresentasikan variabel z yang bertipe integer, dan z diset sama dengan 0.
- - Zoom : double = 1.2
Merepresentasikan variabel Zoom yang bertipe double dan Zoom diinisialisasi sama dengan 1.2 sebagai skala perbesaran.
- - objek : Object
Merepresentasikan variabel objek yang bertipe Object.
- - even : EventArgs
Merepresentasikan variabel even yang bertipe EventArgs.
- - v : View
Merepresentasikan variabel v yang bertipe View.
- - view : frmView
Merepresentasikan variabel view yang merujuk ke frmView.

- - `tampilHistoLinear` : `OperatorTampilHisto`
Merepresentasikan variabel `tampilHistoLinear` yang mengacu ke class `OperatorTampilHisto`.
- + `fileLoc` : `string`
Merepresentasikan lokasi file gambar yang bertipe `string`.
- + `valueA` : `string = ""`
Merepresentasikan variabel `valueA` yang bertipe `string`, dan `valueA` diset sama dengan `""`.
- + `valueB` : `string = ""`
Merepresentasikan variabel `valueB` yang bertipe `string`, dan `valueB` diset sama dengan `""`.
- + `valueVa` : `string = ""`
Merepresentasikan variabel `valueVa` yang bertipe `string`, dan `valueVa` diset sama dengan `""`.
- + `valueVb` : `string = ""`
Merepresentasikan variabel `valueVb` yang bertipe `string`, dan `valueVb` diset sama dengan `""`.
- + `ImageAfterContrast` : `Bitmap`
Merepresentasikan variabel `ImageAfterContrast` yang bertipe `Bitmap`.

Method

- + `frmLinear()`
Operasi ini digunakan untuk melakukan pengontrasan gambar dengan menggunakan metode `Linear Contrast Stretching`. Operasi ini akan mengacu ke class `OperatorTampilHisto` untuk menampilkan histogram gambar, mengacu ke class `Method` untuk memanggil `LinearProcess`, dan mengacu ke `frmView` untuk menampilkan hasil gambar, histogram dan parameter input setelah pengontrasan.

2.1.4.3 Class frmHistogram



Gambar 2.7 Class frmHistogram

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan pengontrasan citra dengan metode Histogram Equalization.

Atribut

- - z : integer = 0
Merepresentasikan variabel z yang bertipe integer, dan z diset sama dengan 0.
- - Zoom : double = 1.2
Merepresentasikan variabel Zoom yang bertipe double dan Zoom diinisialisasi sama dengan 1.2 sebagai skala perbesaran.
- - objek : Object
Merepresentasikan variabel objek yang bertipe Object.
- - even : EventArgs
Merepresentasikan variabel even yang bertipe EventArgs.
- - tampilHisto : OperatorTampilHisto
Merepresentasikan variabel tampilHisto yang mengacu ke class OperatorTampilHisto.
- + ImageAfterContrast : Bitmap
Merepresentasikan variabel ImageAfterContrast yang bertipe Bitmap.
- + fileLoc : string
Merepresentasikan lokasi file gambar yang bertipe string.
- + k : integer = 0

Merepresentasikan variabel k yang bertipe integer dan diset sama dengan 0.

- $+ h : \text{integer} = 0$

Merepresentasikan variabel h yang bertipe integer dan diset sama dengan 0.

Method

- $+ \text{frmHistogram}()$

Operasi ini digunakan untuk melakukan pengontrasan gambar dengan menggunakan metode Histogram Equalization. Operasi ini akan mengacu ke class `OperatorTampilHisto` untuk menampilkan histogram gambar dan mengacu ke class `Method` untuk memanggil prosedur pengontrasan sesuai dengan metode Histogram Equalization. Operasi ini juga akan mengacu ke `frmView` untuk menampilkan gambar dan histogram setelah proses pengontrasan.

2.1.4.4 Class frmLee



Gambar 2.8 Class frmLee

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan pengontrasan citra dengan metode Lee's Algorithm.

Atribut

- $- z : \text{integer} = 0$

Merepresentasikan variabel *z* yang bertipe integer, dan *z* diset sama dengan 0.

- - *Zoom* : double = 1.2

Merepresentasikan variabel *Zoom* yang bertipe double dan *Zoom* diinisialisasi sama dengan 1.2 sebagai skala perbesaran.

- - objek : Object

Merepresentasikan variabel objek yang bertipe Object.

- - *even* : EventArgs

Merepresentasikan variabel *even* yang bertipe EventArgs.

- - *tampilHistoLee* : OperatorTampilHisto

Merepresentasikan variabel *tampilHistoLee* yang mengacu ke class *OperatorTampilHisto*.

- + *fileLoc* : string

Merepresentasikan lokasi file gambar yang bertipe string.

- + *valueAlphaLee* : string = ""

Merepresentasikan variabel *valueAlphaLee* yang bertipe string dan diset sama dengan "".

- + *valueBetaLee* : string = ""

Merepresentasikan variabel *valueBetaLee* yang bertipe string sama dengan "".

- + *ImageAfterContrast* : Bitmap

Merepresentasikan variabel *ImageAfterContrast* yang bertipe Bitmap.

Method

- + *frmLee*()

Operasi ini digunakan untuk melakukan pengontrasan gambar dengan menggunakan metode Lee's Algorithm. Operasi ini akan mengacu ke class *OperatorTampilHisto* untuk menampilkan histogram gambar dan mengacu ke class *Method* untuk memanggil prosedur *LeeProcess*. Operasi ini juga mengacu ke *frmView* yang digunakan untuk menampilkan

gambar, histogram, dan parameter input setelah proses pengontrasan.

2.1.4.5 Class frmLogarithmic



Gambar 2.9 Class frmLogarithmic

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan pengontrasan citra dengan metode Logarithmic Image Processing.

Atribut

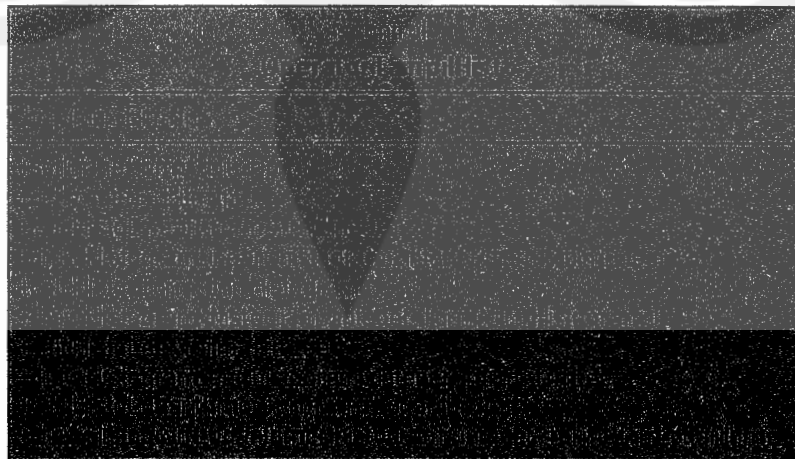
- - z : integer = 0
Merepresentasikan variabel z yang bertipe integer, dan z diset sama dengan 0.
- - Zoom : double = 1.2
Merepresentasikan variabel Zoom yang bertipe double dan Zoom diinisialisasi sama dengan 1.2 sebagai skala perbesaran.
- - objek : Object
Merepresentasikan variabel objek yang bertipe Object.
- - even : EventArgs
Merepresentasikan variabel even yang bertipe EventArgs.
- - tampilHistoLogarithmic : OperatorTampilHisto
Merepresentasikan variabel tampilHistoLogarithmic yang mengacu ke class OperatorTampilHisto.
- + fileLoc : string
Merepresentasikan lokasi file gambar yang bertipe string.

- + valueAlphaLog : string = ""
Merepresentasikan variabel valueAlphaLog yang bertipe string dan diset sama dengan "".
- + valueBetaLee : string = ""
Merepresentasikan variabel valueBetaLog yang bertipe string sama dengan "".
- + ImageAfterContrast : Bitmap
Merepresentasikan variabel ImageAfterContrast yang bertipe Bitmap.

Method

- + frmLogarithmic()
Operasi ini digunakan untuk melakukan pengontrasan gambar dengan menggunakan metode Logarithmic Image Processing. Operasi ini akan mengacu ke class OperatorTampilHisto untuk menampilkan histogram gambar dan mengacu ke class Method untuk memanggil prosedur LogarithmicProcess. Operasi ini juga mengacu ke frmView yang digunakan untuk menampilkan gambar, histogram dan parameter input setelah proses pengontrasan.

2.1.4.6 Class OperatorTampilHisto



Gambar 2.10 Class OperatorTampilHisto

Deskripsi

Class yang berperan sebagai control class. Class ini digunakan untuk melakukan pemrosesan gambar histogram dari citra baik sebelum dikontraskan maupun yang sudah dikontraskan.

Atribut

- - bmpimg : Bitmap
Merepresentasikan variabel bmpimg yang bertipe Bitmap.

Method

- + setImage(bmp : Bitmap)
Operasi ini digunakan untuk proses set image .
- + getImage() : Bitmap
Operasi ini digunakan untuk proses get image dengan fungsi balikan Bitmap.
- + calcHistoLinear(form : frmLinear)
Operasi ini digunakan untuk memanggil histogram citra pada frmLinear baik sebelum maupun sesudah pengontrasan.
- + drawHistogramLinear(histogram : integer, form : frmLinear)
Operasi ini digunakan untuk menggambarkan histogram citra pada frmLinear baik sebelum maupun sesudah pengontrasan.
- + calcHisto(form : frmHistogram)
Operasi ini digunakan untuk memanggil histogram citra pada frmHistogram baik sebelum maupun sesudah pengontrasan.
- + drawHistogram(histogram : integer, form : frmHistogram)
Operasi ini digunakan untuk menggambarkan histogram citra pada frmHistogram baik sebelum maupun sesudah pengontrasan.
- + calcHistoLee(form : frmLee)
Operasi ini digunakan untuk memanggil histogram citra pada frmLee baik sebelum maupun sesudah pengontrasan.

- + drawHistogramLee(histogram : integer, form : frmLee)

Operasi ini digunakan untuk menggambar histogram citra pada frmLee baik sebelum maupun sesudah pengontrasan.

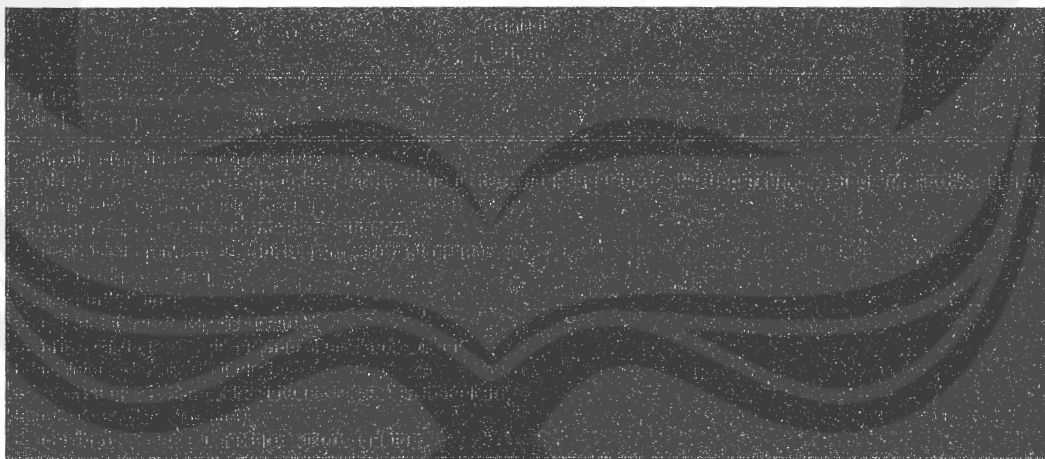
- + calcHistoLogarithmic(form : frmLogarithmic)

Operasi ini digunakan untuk memanggil histogram citra pada frmHistogram baik sebelum maupun sesudah pengontrasan.

- + drawHistogramLogarithmic(histogram : integer, form : frmLogarithmic)

Operasi ini digunakan untuk menggambar histogram citra pada frmLogarithmic baik sebelum maupun sesudah pengontrasan.

2.1.4.7 Class Method



Gambar 2.11 Class Method

Deskripsi

Class yang berperan sebagai control class, merepresentasikan metode-metode yang digunakan dalam pemrosesan pengontrasan citra.

Atribut

- + bmpimg : Bitmap

Merepresentasikan variabel bmpimg yang bertipe Bitmap.

- + bitmap : Bitmap

Merepresentasikan variabel bitmap yang bertipe Bitmap.

Method

- + `getMethod(method : string) : string`
Operasi ini digunakan untuk memanggil method yang dideklarasikan dengan tipe string.
- + `MethodProcess(method : string, lee : frmLee, linear : frmLinear, logarithmic : Logarithmic, histogram : frmHistogram)`
Operasi ini digunakan untuk mendeklarasikan method-method yang digunakan untuk melakukan pengontrasan citra yaitu Linear Contrast Stretching, Histogram Equalization, Lee's Algorithm, dan Logarithmic Image Processing.
- + `LinearProcess(linear : frmLinear)`
Operasi ini digunakan untuk mendeklarasikan prosedur yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Linear Contrast Stretching.
- + `HistogramProcess(histogram : frmHistogram)`
Operasi ini digunakan untuk mendeklarasikan prosedur yang digunakan untuk melakukan pemrosesan histogram.
- + `EqualizeHistogramProcess(histogram:frmHistogram)`
Operasi ini digunakan untuk mendeklarasikan prosedur yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Histogram Equalization.
- + `setImage bmp : Bitmap)`
Operasi ini digunakan untuk melakukan set Image.
- + `getImage() : Bitmap`
Operasi ini digunakan untuk melakukan get Image dengan tipe balikan Bitmap.
- + `equilizeHist(form : frmHistogram)`
Operasi ini digunakan untuk melakukan penyebaran histogram citra sesuai dengan metode Histogram Equalization.
- + `equilize(histogram : integer, numPixel : long) : float`

Operasi ini digunakan untuk melakukan penyebaran histogram dengan tipe variabel balikan float.

- + `calcHisto(form : frmHistogram)`

Operasi ini digunakan untuk memanggil histogram citra pada `frmHistogram`.

- + `drawHistogram(histogram : integer, form : frmHistogram)`

Operasi ini digunakan untuk menggambar histogram citra pada `frmHistogram`.

- + `LeeProcess(lee : frmLee)`

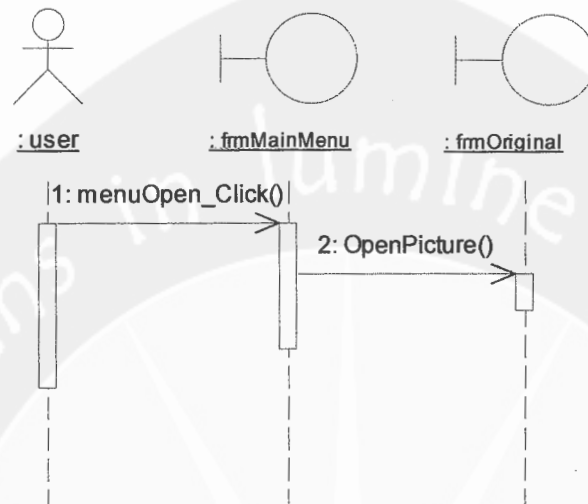
Operasi ini digunakan untuk mendeklarasikan prosedur yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Lee's Algorithm.

- + `LogarithmicProcess(logarithmic : frmLogarithmic)`

Operasi ini digunakan untuk mendeklarasikan prosedur yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Logarithmic Image Processing.

2.2 Realisasi Use Case

2.2.1 Use Case : OpenOriginalPicture

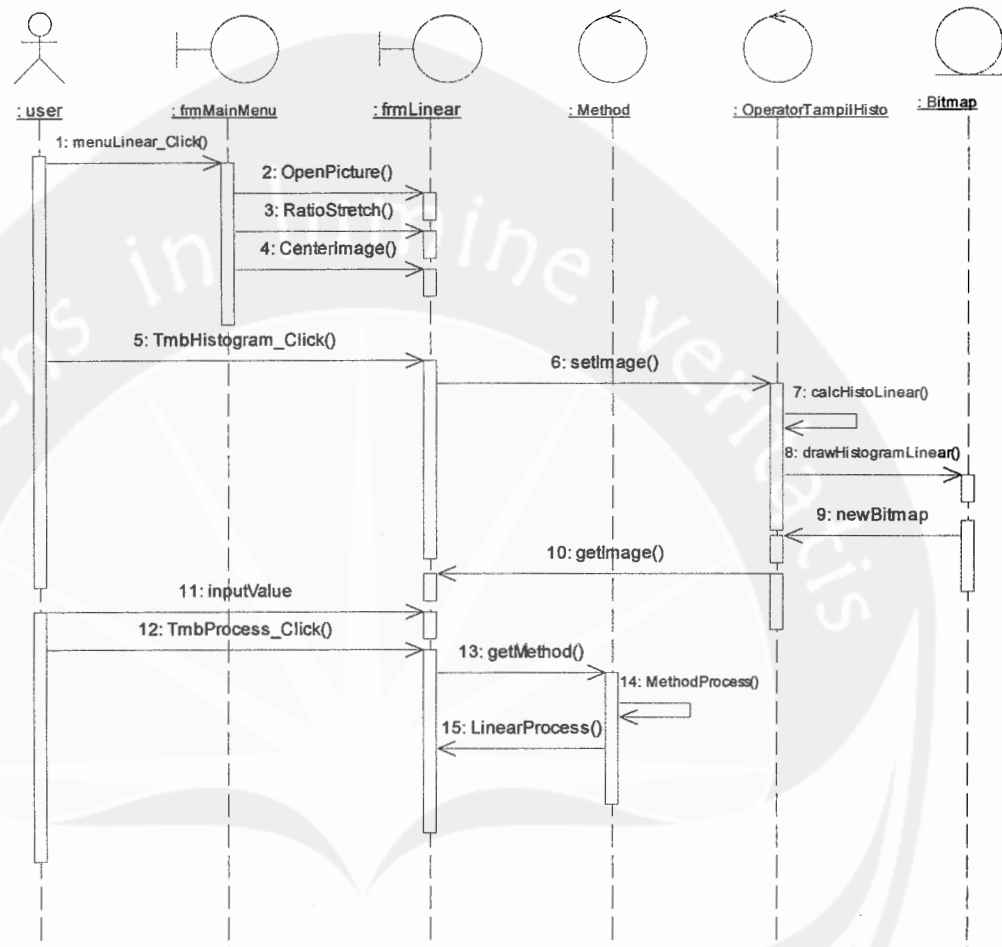


Gambar 2.12 Design Sequence Diagram : Use Case OpenOriginalPicture

Flow of events :

1. User meminta sistem untuk membuka `frmOriginal` dengan menekan menu Open pada `frmMainMenu` yang merupakan boundary class.
2. Sistem akan melakukan `OpenPicture` ke boundary class `frmOriginal`.

2.2.2 Use Case : LinearContrastStretching



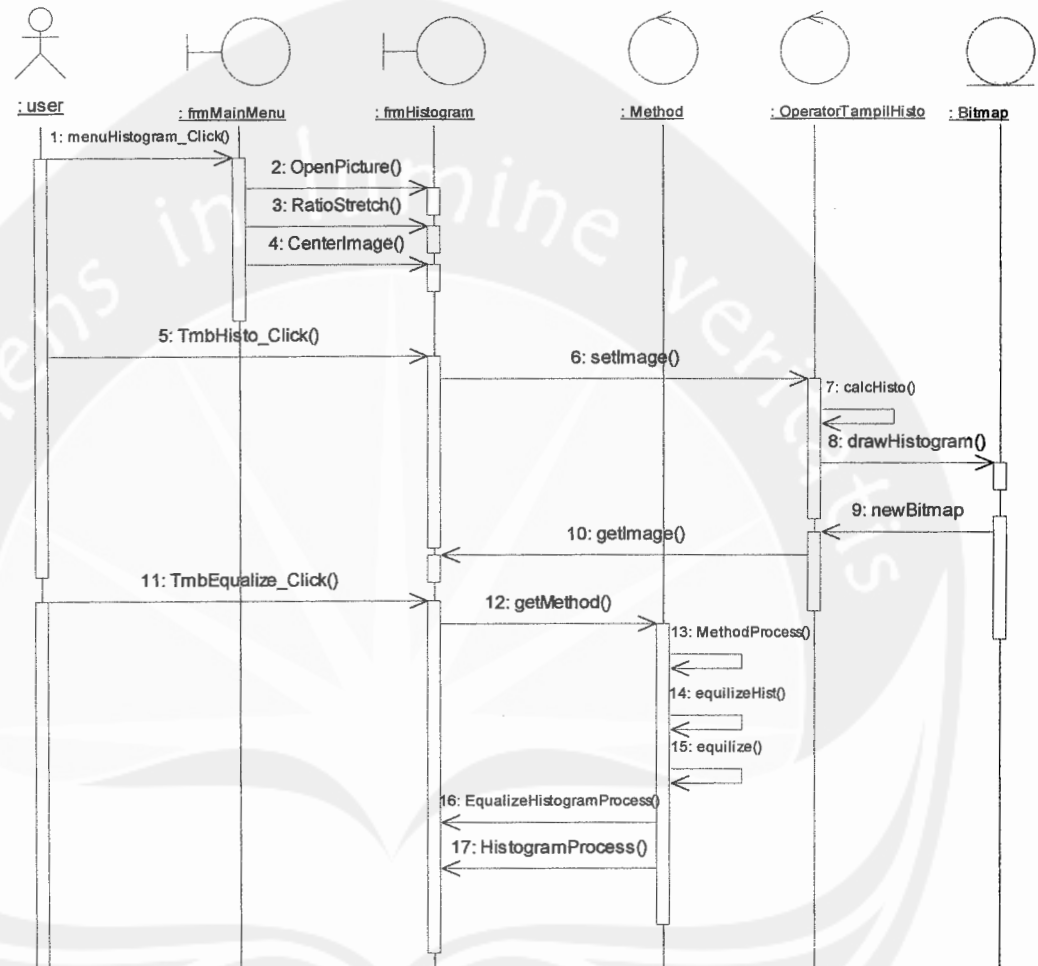
Gambar 2.13 Design Sequence Diagram : Use Case LinearContrastStretching

Flow of events :

1. User menampilkan antarmuka untuk melakukan pengontrasan citra dengan metode Linear Contrast Stretching, dari boundary class `frmMainMenu` ke boundary class `frmLinear`.
2. Sistem memanggil prosedur `OpenPicture()` untuk membuka gambar asli, `RatioStretch()` untuk melakukan perentangan gambar, `CenterImage()` untuk melakukan penengahan gambar ke boundary class `frmLinear`.

3. User menekan button Histogram untuk melihat histogram gambar ke boundary class frmLinear.
4. Sistem kemudian akan memanggil prosedur setImage() dari control class OperatorTampilHisto kemudian sistem akan memproses prosedur calcHistoLinear() pada control class OperatorTampilHisto.
5. Sistem kemudian memanggil prosedur drawHistogramLinear() ke entity class Bitmap untuk membuat histogram gambar, dan melakukan getImage().
6. User kemudian melakukan input value dan menekan button Process.
7. Sistem kemudian akan memanggil prosedur getMethod() pada control class Method, kemudian sistem akan memproses method dengan prosedur MethodProcess(), dan memproses method dengan memanggil prosedur LinearProcess().

2.2.3 Use Case : HistogramEqualization



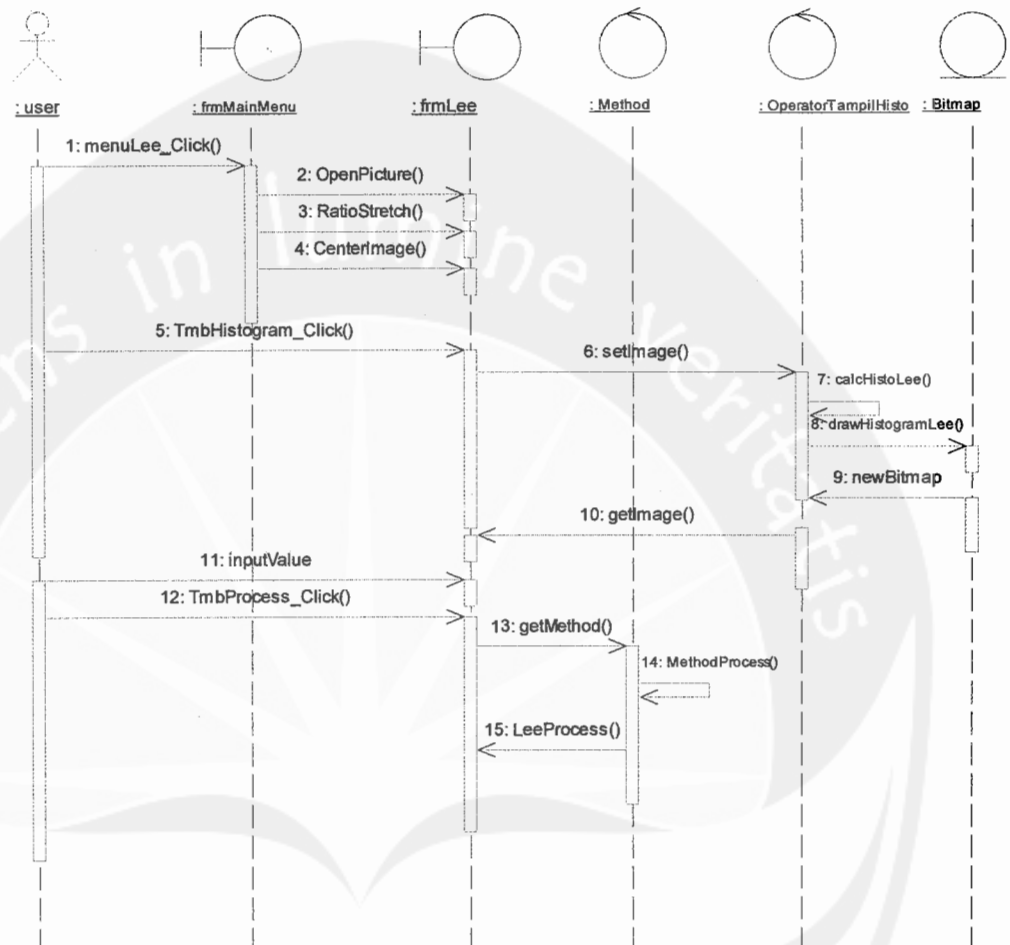
Gambar 2.14 Design Sequence Diagram : Use Case HistogramEqualization

Flow of events :

1. User menampilkan antarmuka untuk melakukan pengontrasan citra dengan metode Histogram Equalization, dari boundary class frmMainMenu ke boundary class frmHistogram.
2. Sistem memanggil prosedur OpenPicture() untuk membuka gambar asli, RatioStretch() untuk melakukan perentangan gambar, CenterImage() untuk melakukan penengahan gambar ke boundary class frmHistogram.

3. User menekan button Histogram untuk melihat histogram gambar ke boundary class frmHistogram.
4. Sistem kemudian akan memanggil prosedur setImage() dari control class OperatorTampilHisto kemudian sistem akan memproses prosedur calcHistoLinear() pada control class OperatorTampilHisto.
5. Sistem kemudian memanggil prosedur drawHistogramLinear() ke entity class Bitmap untuk membuat histogram gambar, dan melakukan getImage().
6. User kemudian menekan button Equalize ke boundary class frmHistogram.
7. Sistem kemudian akan memanggil prosedur getMethod() pada control class Method, kemudian sistem akan memproses method dengan prosedur MethodProcess(), dan memproses method dengan memanggil prosedur equilizeHist() dan equilize() untuk melakukan penyebaran histogram citra.
8. Sistem kemudian akan memanggil prosedur EqualizeHistogramProcess() dan HistogramProcess().

2.2.4 Use Case : Lee'sAlgorithm



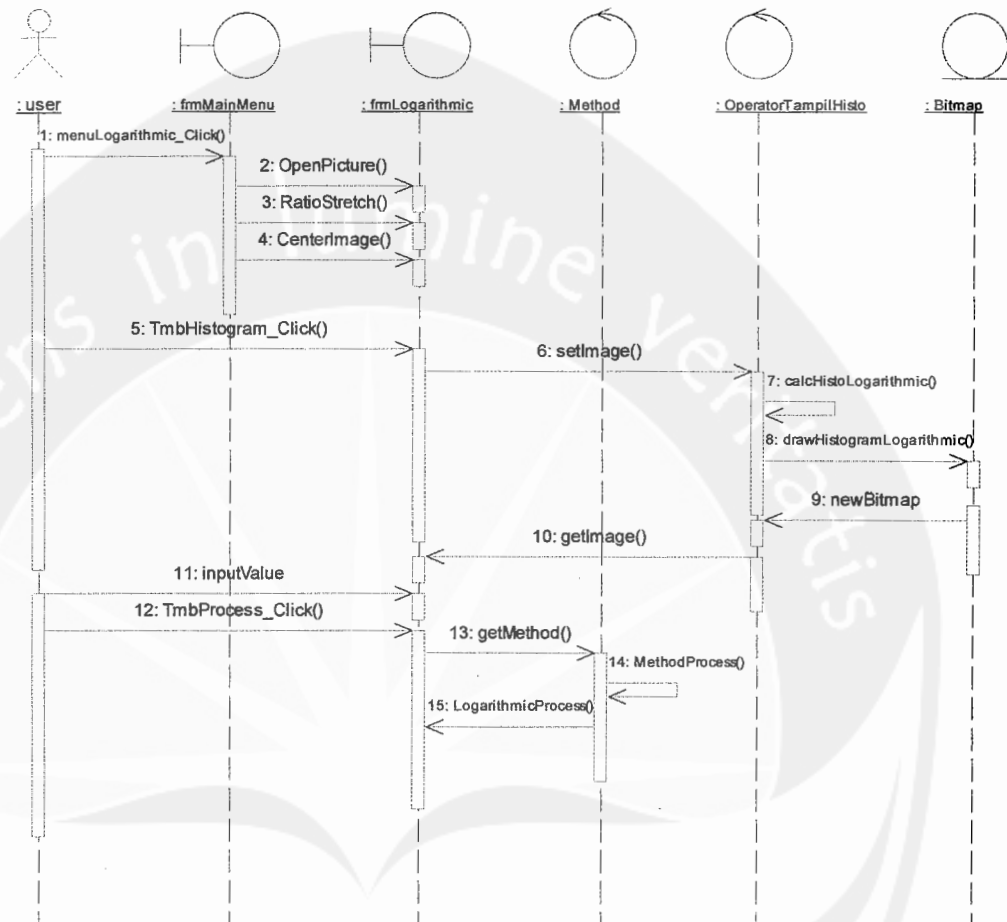
Gambar 2.15 Design Sequence Diagram : Use Case Lee'sAlgorithm

Flow of events :

1. User menampilkan antarmuka untuk melakukan pengontrasan citra dengan metode Lee's Algorithm, dari boundary class frmMainMenu ke boundary class frmLinear.
2. Sistem memanggil prosedur OpenPicture() untuk membuka gambar asli, RatioStretch() untuk melakukan perentangan gambar, CenterImage() untuk melakukan penengahan gambar ke boundary class frmLee.

3. User menekan button Histogram untuk melihat histogram gambar ke boundary class frmLee.
4. Sistem kemudian akan memanggil prosedur setImage() dari control class OperatorTampilHisto kemudian sistem akan memproses prosedur calcHistoLee() pada control class OperatorTampilHisto.
5. Sistem kemudian memanggil prosedur drawHistogramLee() ke entity class Bitmap untuk membuat histogram gambar, dan melakukan getImage().
6. User kemudian melakukan input value dan menekan button Process.
7. Sistem kemudian akan memanggil prosedur getMethod() pada control class Method, kemudian sistem akan memproses method dengan prosedur MethodProcess(), dan memproses method dengan memanggil prosedur LeeProcess().

2.2.5 Use Case : LogarithmicImageProcessing



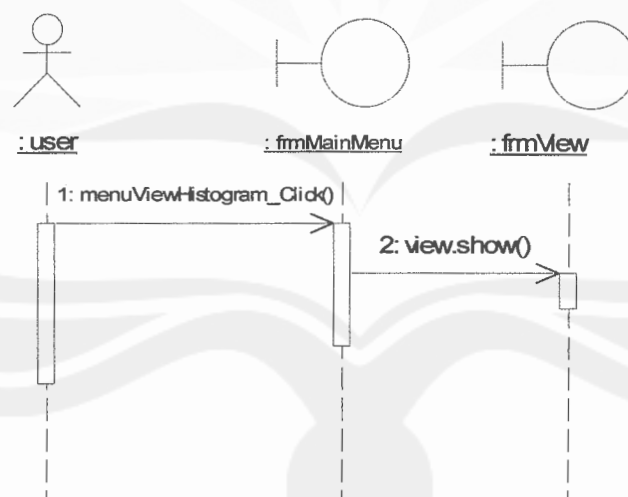
Gambar 2.16 Design Sequence Diagram : Use Case LogarithmicImageProcessing

Flow of events :

1. User menampilkan antarmuka untuk melakukan pengontrasan citra dengan metode Logarithmic Image Processing, dari boundary class frmMainMenu ke boundary class frmLee.
2. Sistem memanggil prosedur OpenPicture() untuk membuka gambar asli, RatioStretch() untuk melakukan perentangan gambar, CenterImage() untuk melakukan penengahan gambar ke boundary class frmLogarithmic.
3. User menekan button Histogram untuk melihat histogram gambar ke boundary class frmLogarithmic.

4. Sistem kemudian akan memanggil prosedur `setImage()` dari control class `OperatorTampilHisto` kemudian sistem akan memproses prosedur `calcHistoLogarithmic()` pada control class `OperatorTampilHisto`.
5. Sistem kemudian memanggil prosedur `drawHistogramLogarithmic()` ke entity class `Bitmap` untuk membuat histogram gambar, dan melakukan `getImage()`.
6. User kemudian melakukan input value dan menekan button `Process`.
7. Sistem kemudian akan memanggil prosedur `getMethod()` pada control class `Method`, kemudian sistem akan memproses method dengan prosedur `MethodProcess()`, dan memproses method dengan memanggil prosedur `LogarithmicProcess()`.

2.2.6 Use Case : View



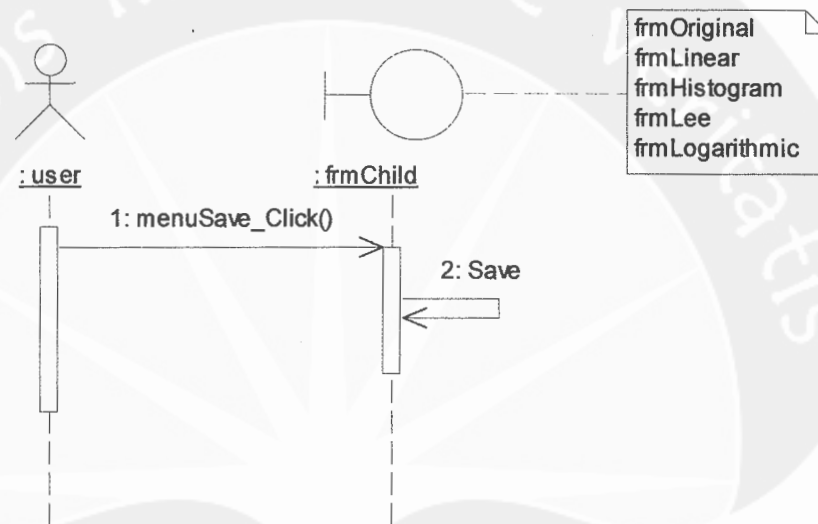
Gambar 2.17 Design Sequence Diagram : Use Case View

Flow of events :

1. User menampilkan antarmuka untuk membandingkan hasil pengontrasan keempat metode, dari boundary class `frmMainMenu` ke boundary class `frmView`.

2. Sistem mengecek form yang telah terbuka dengan mengecek label form anak yaitu “Linear Contrast Stretching”, “Histogram Equalization”, “Lee’s Algorithm” dan “Logarithmic Image Processing”. Apabila terpenuhi maka sistem akan memunculkan `frmView` dengan kode program `view.show()`.

2.2.7 Use Case : SavePicture

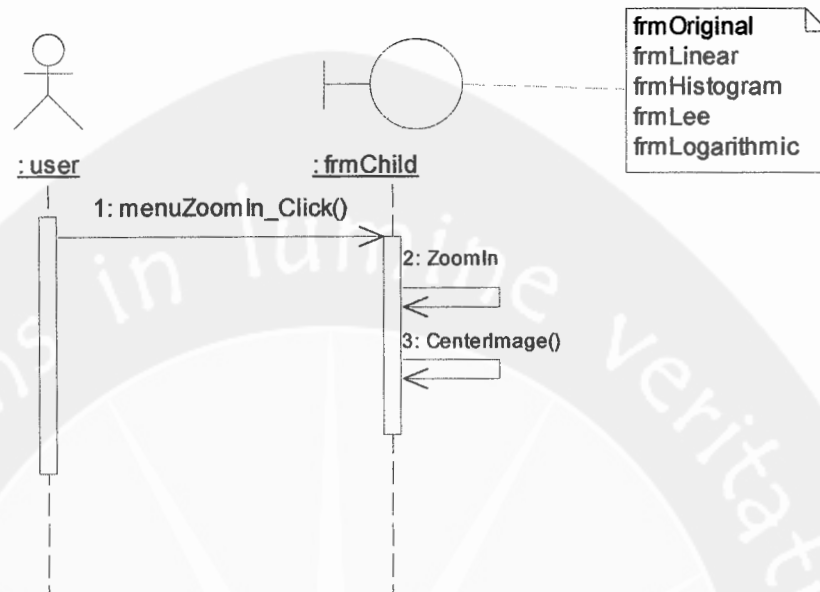


Gambar 2.18 Design Sequence Diagram : Use Case SavePicture

Flow of events :

1. User menampilkan antarmuka untuk melakukan penyimpanan citra hasil proses pengontrasan dengan memilih menu Save pada boundary class `frmChild` atau form anak yang terdiri dari `frmLinear`, `frmHistogram`, `frmLee`, dan `frmLogarithmic`.
2. Sistem akan melakukan penyimpanan gambar yang telah di kontraskan.

2.2.8 Use Case : Zoom In

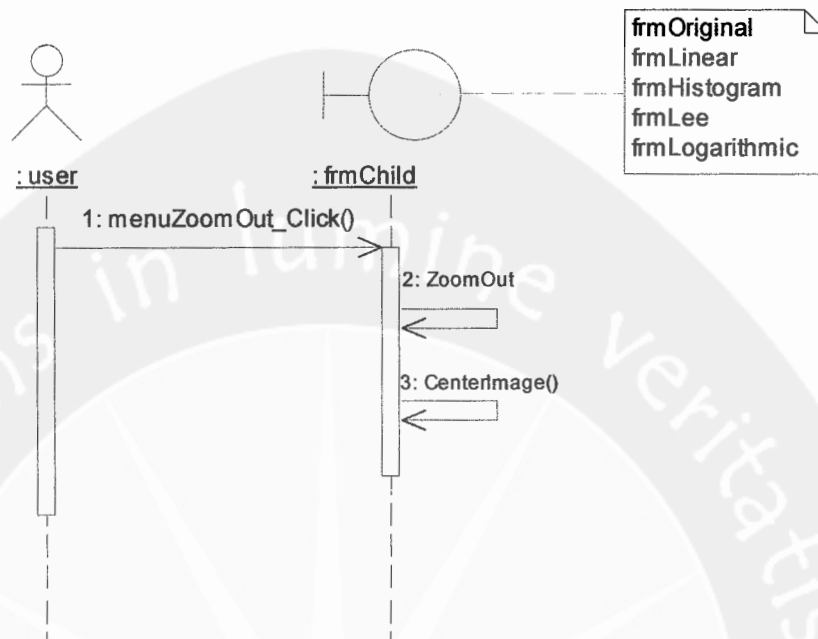


Gambar 2.19 Design Sequence Diagram : Use Case ZoomIn

Flow of events :

1. User menampilkan antarmuka untuk memperbesar ukuran citra dengan memilih menu Zoom In pada setiap boundary class frmChild atau form anak yang meliputi frmLinear, frmHistogram, frmLee, dan frmLogarithmic.
2. Sistem kemudian melakukan zoom in dan memanggil prosedur CenterImage() yang digunakan untuk menengahkan gambar.

2.2.9 Use Case : ZoomOut

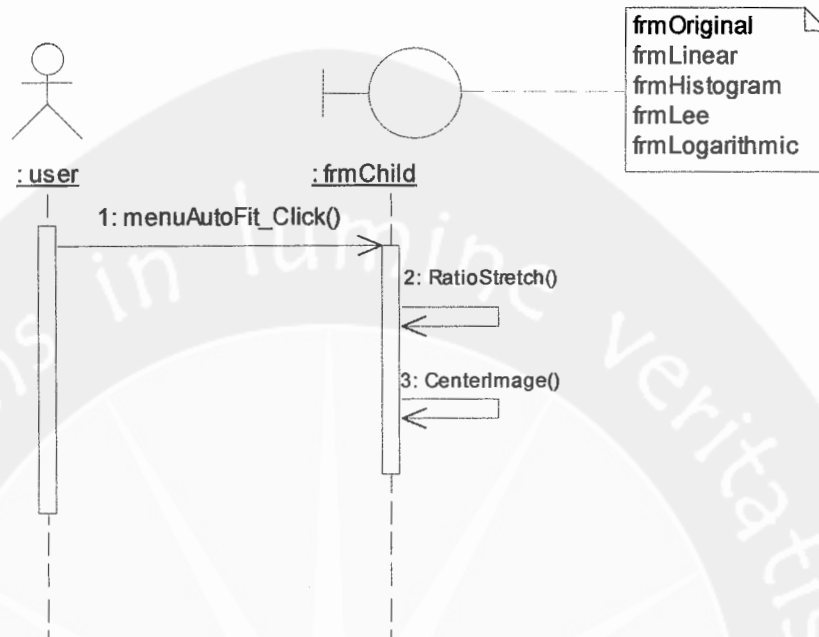


Gambar 2.20 Design Sequence Diagram : Use Case ZoomOut

Flow of events :

1. User menampilkan antarmuka untuk memperkecil ukuran citra dengan memilih menu Zoom Out pada setiap boundary class frmChild atau form anak yang meliputi frmLinear, frmHistogram, frmLee, dan frmLogarithmic.
2. Sistem kemudian melakukan zoom out dan memanggil prosedur CenterImage() yang digunakan untuk menengahkan gambar.

2.2.10 Use Case : AutoFit



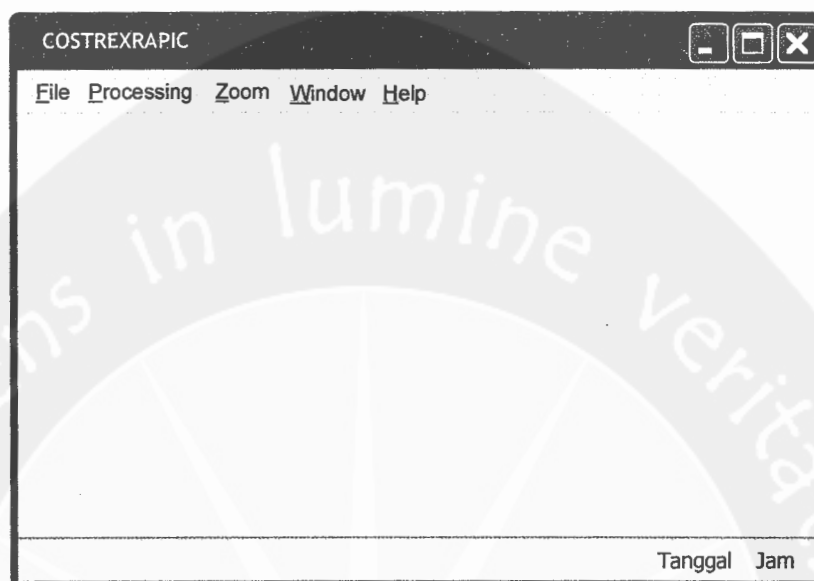
Gambar 2.21 Design Sequence Diagram : Use Case AutoFit

Flow of events :

1. User menampilkan antarmuka untuk menyesuaikan ukuran citra dengan pictureBox dengan memilih menu Auto Fit pada setiap boundary class frmChild atau form anak yang meliputi frmLinear, frmHistogram, frmLee, dan frmLogarithmic.
2. Sistem kemudian melakukan memanggil prosedur RatioStretch(), dan CenterImage() yang digunakan untuk menengahkan gambar.

3 Deskripsi Perancangan Antarmuka

3.1 Antarmuka Main Menu



Gambar 3.1 Rancangan Antarmuka frmMainMenu

Deskripsi

- Antarmuka ini merupakan menu utama yang digunakan untuk memilih menu maupun sub menu dalam sistem.
- Menu yang tersedia adalah menu File, Processing, Zoom, Window, dan Help.
- Dalam menu File terdapat sub menu Open (UC-COSTREXRAPIC-01), Save (UC-COSTREXRAPIC-07), dan Exit.
- Dalam menu Processing terdapat sub menu Linear Contrast Stretching (UC-COSTREXRAPIC-02), Histogram Equalization (UC-COSTREXRAPIC-03), Lee's Algorithm (UC-COSTREXRAPIC-04), dan Logarithmic Image Processing (UC-COSTREXRAPIC-05).
- Dalam menu Zoom terdapat sub menu Zoom In (UC-COSTREXRAPIC-08), Zoom Out (UC-COSTREXRAPIC-09), dan Auto Fit (UC-COSTREXRAPIC-10).

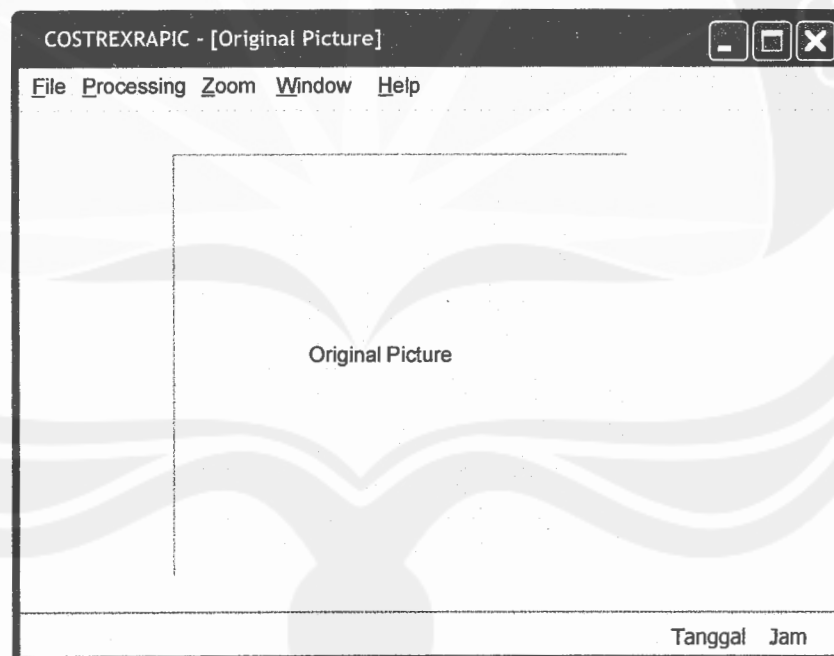
- Dalam menu Window terdapat sub menu Cascade, Tile Horizontal, Tile Vertical, View (UC-COSTREXRAPIC-06), Close, dan Close All Document.
- Dalam menu Help terdapat sub menu About dan Help.

Event

Urutan aksi yang terjadi :

1. User masuk ke sistem.
2. Sistem akan menampilkan antarmuka main menu.
3. User dapat memilih menu maupun sub menu yang diinginkan.

3.2 Use Case : OpenOriginalPicture



Gambar 3.2 Rancangan Antarmuka frmOriginal

Deskripsi

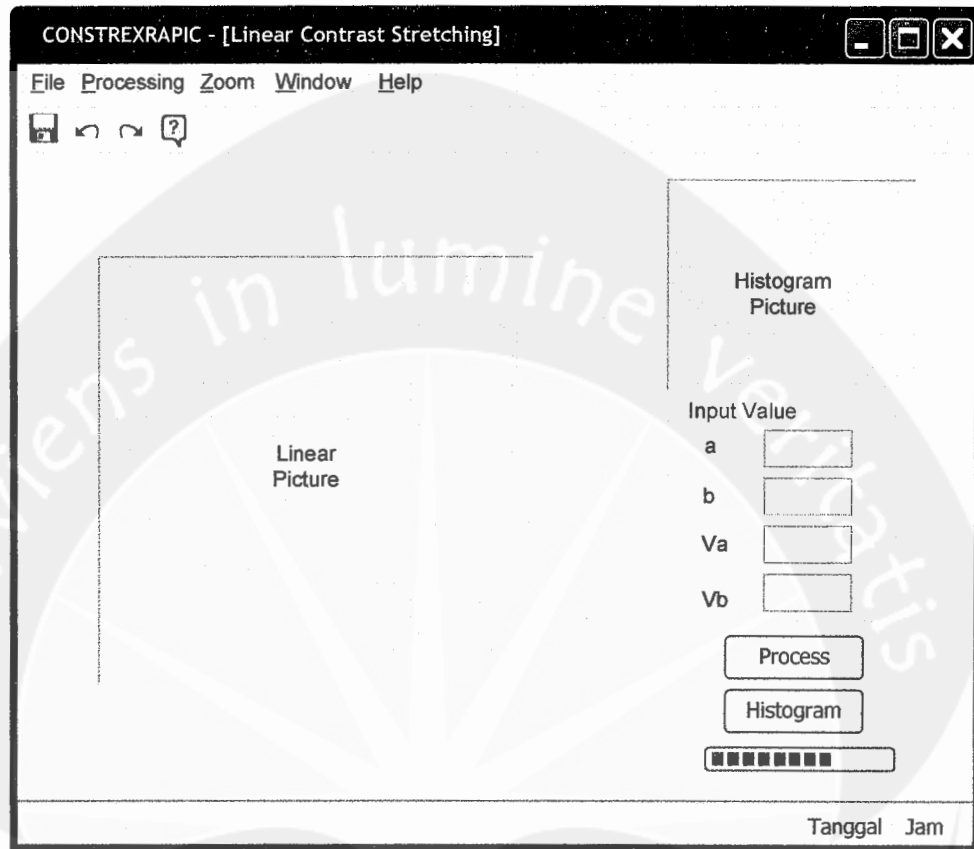
- Antarmuka ini digunakan pada use case OpenOriginalPicture (UC-COSTREXRAPIC-01).
- Sub menu Open ini digunakan untuk membuka citra asli yang akan diproses.

Event

Urutan aksi yang terjadi :

1. User memilih sub menu **Open** pada **frmMainMenu**.
2. Sistem akan membuka **OpenFileDialog** dengan title **Open File Original** yang memunculkan direktori penyimpanan citra asli.
3. User dapat memilih citra asli yang diinginkan.
4. Jika **openFileOriginal.ShowDialog = DialogResult.OK** maka **fileLoc** di set dengan **openFileOriginal.FileName** dan **pictureBox picOriginal** di set dengan **FromFile(fileLoc)**.
5. Kemudian untuk menampilkan gambar originalnya dengan perintah **Show()**.
6. Apabila proses pembukaan gambar error maka akan muncul **MessageBox "Error opening image"**.

3.3 Use Case : LinearContrastStretching



Gambar 3.3 Rancangan Antarmuka frmLinear

Deskripsi

- Antarmuka ini adalah frmLinear yang digunakan pada use case LinearContrastStretching (UC-COSTREXPIC-02), untuk pengontrasan citra dengan metode Linear Contrast Stretching.
- Metode Linear Contrast Stretching ini merupakan salah satu metode yang akan digunakan untuk mengontraskan citra.

Event

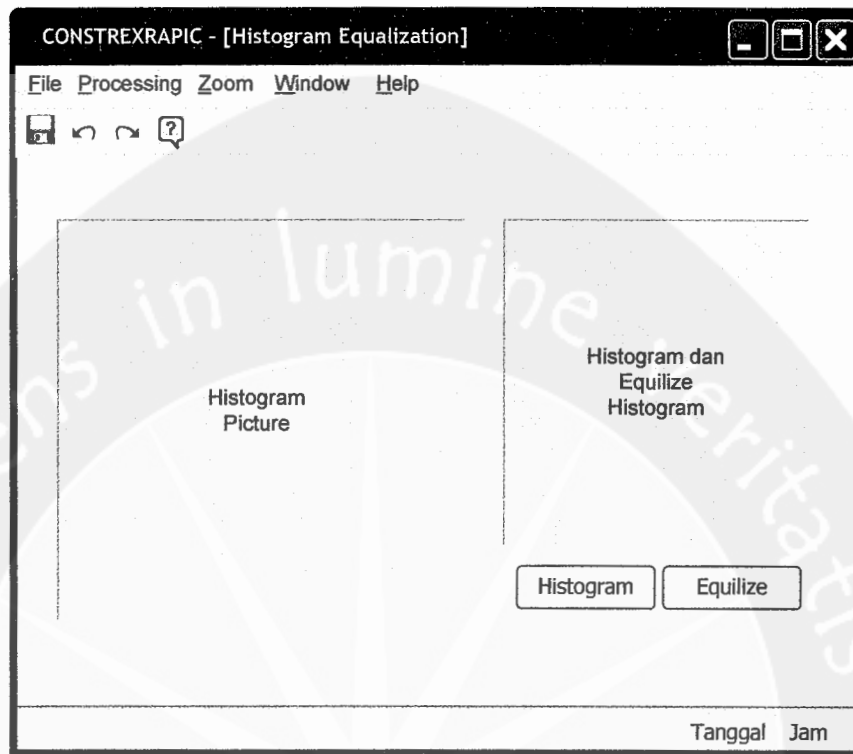
Urutan aksi yang terjadi :

1. User memilih sub menu Linear Contrast Stretching pada form main menu.
2. Sistem akan melakukan **OpenPicture()** dengan mengeset **picLinear** yang merupakan **pictureBox** dalam **frmLinear** dengan

FromFile(fileLoc). Show() digunakan untuk menampilkan citra ke dalam `pictureBox`.

3. User menekan button Histogram untuk mengetahui histogram citra asli.
4. Sistem akan memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLinear()** dan **drawHistogramLinear()**.
5. Kemudian user menginputkan nilai a , b , V_a , dan V_b dalam textbox dan menekan button Process.
6. Sistem akan memanggil class **Method** yang di dalamnya terdapat **LinearProcess()** yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Linear Contrast Stretching.
7. Sistem akan mencari nilai **alpha**, **beta**, dan **teta** sesuai dengan nilai input a , b , V_a , dan V_b . Kemudian dilakukan **getPixel** untuk mendapatkan **pixelAsli**. Sistem akan melakukan perulangan untuk mendapatkan **pixelHasil**. **pixelHasil** yang sudah didapat dicek, baru kemudian dilakukan pengesetan **Color** dengan **pixelHasil** untuk komponen RGB nya.
8. User kemudian menekan button Histogram kembali untuk melihat perubahan histogram citra yang telah dikontraskan dengan metode Linear Contrast Stretching.
9. Sistem akan kembali memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLinear()** dan **drawHistogramLinear()**.

3.4 Use Case : HistogramEqualization



Gambar 3.4 Rancangan Antarmuka frmHistogram

Deskripsi

- Antarmuka ini digunakan pada use case HistogramEqualization (UC-COSTREXRAPIC-03).
- Metode Histogram Equalization ini merupakan salah satu metode yang akan digunakan untuk mengontraskan citra.

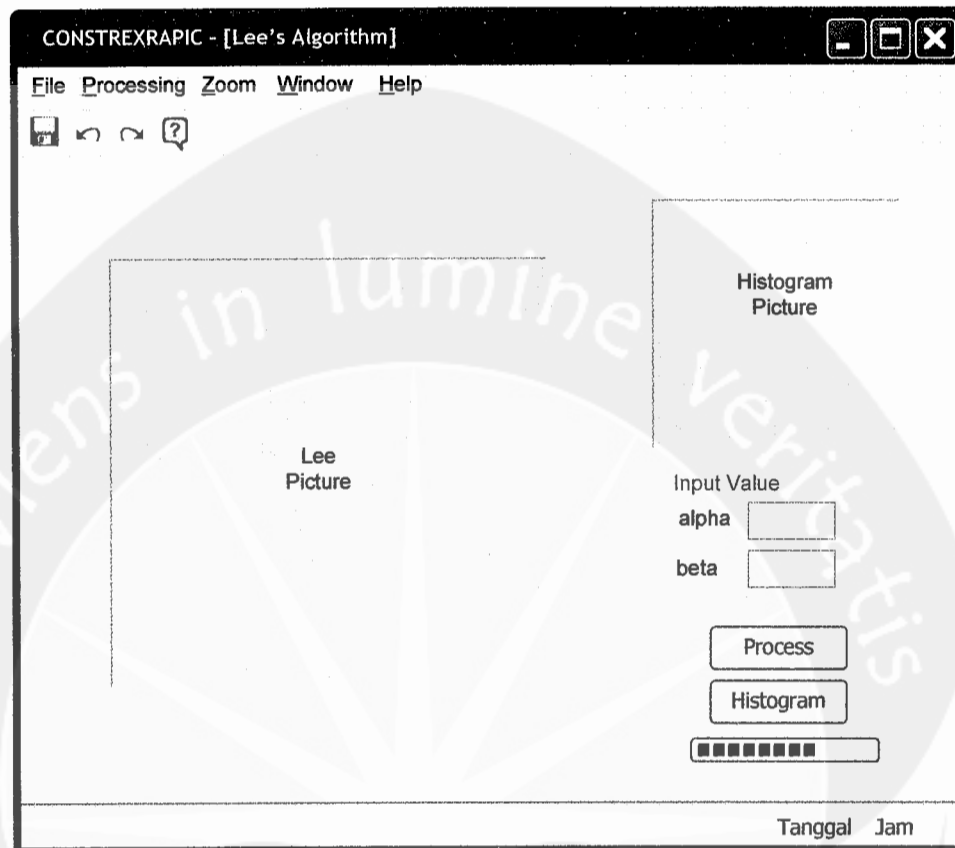
Event

Urutan aksi yang terjadi :

1. User memilih sub menu Histogram Equalization pada form main menu.
2. Sistem akan melakukan **OpenPicture()** dengan mengeset **picHistogram** yang merupakan pictureBox dalam **frmHistogram** dengan **FromFile(fileLoc)**. **Show()** digunakan untuk menampilkan citra ke dalam pictureBox.

3. User menekan button Histogram untuk mengetahui histogram citra asli.
4. Sistem akan memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHisto()** dan **drawHistogram()**.
5. User menekan button Equalize untuk melakukan penyebaran histogram dan pengontrasan citra.
6. Sistem akan memanggil class **Method** yang di dalamnya terdapat prosedur **HistogramProcess()** yang digunakan untuk memanggil prosedur **calcHisto()** serta terdapat **EquilizeHistogramProcess()** yang digunakan untuk memanggil **equilizeHist()** yang di dalamnya terdapat perintah untuk memanggil fungsi **equilize()** dengan nilai balikan **hist. equilizeHist()** digunakan untuk melakukan penyebaran histogram citra.

3.5 Use Case : Lee'sAlgorithm



Gambar 3.5 Rancangan Antarmuka frmLee

Deskripsi

- Antarmuka ini digunakan pada use case Lee's Algorithm (UC-COSTREXPIC-04).
- Metode Lee's Algorithm ini merupakan salah satu metode yang akan digunakan untuk mengontraskan citra.

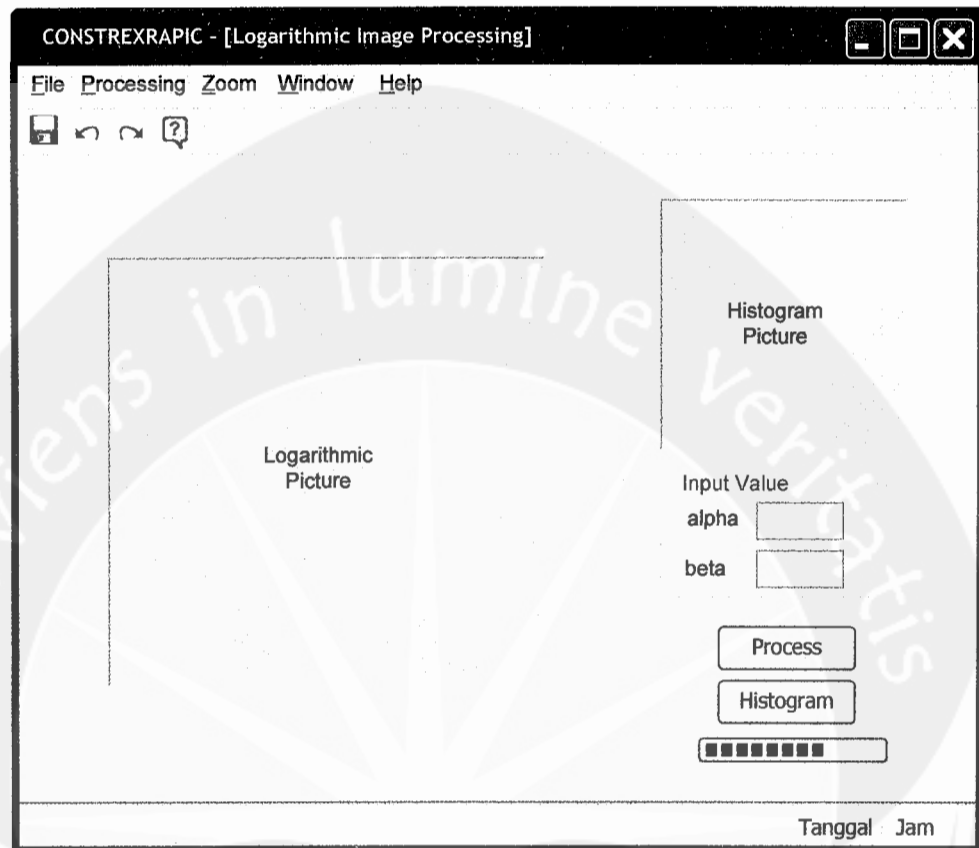
Event

Urutan aksi yang terjadi :

1. User memilih sub menu Lee's Algorithm pada form main menu.
2. Sistem akan melakukan **OpenPicture()** dengan mengeset **picLee** yang merupakan **pictureBox** dalam **frmLee** dengan **FromFile(fileLoc). Show()** digunakan untuk menampilkan citra ke dalam **pictureBox**.

3. User menekan button Histogram untuk mengetahui histogram citra asli.
4. Sistem akan memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLee()** dan **drawHistogramLee()**.
5. Kemudian user menginputkan nilai alpha, dan beta dalam textbox dan menekan button Process.
6. Sistem akan memanggil class **Method** yang di dalamnya terdapat **LeeProcess()** yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Lee's Algorithm.
7. Dalam prosedur **LeeProcess()** dilakukan pencarian rerata pixel untuk pixel **3x3**, kemudian dilakukan **getPixel pixelAsli** untuk nilai **R**. Kemudian dicari **pixel hasil** sesuai dengan rumus Lee's Algorithm. Pixel hasil yang telah diperoleh dicek nilainya agar tidak terjadi error. Setelah itu baru dilakukan **SetPixel** dengan nilai pixel hasil yang sudah didapatkan untuk komponen RGB.
8. User kemudian menekan button Histogram kembali untuk melihat perubahan histogram citra yang telah dikontraskan dengan metode Lee's Algorithm.
9. Sistem akan kembali memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLee()** dan **drawHistogramLee()**.

3.6 Use Case : LogarithmicImageProcessing



Gambar 3.6 Rancangan Antarmuka frmLogarithmic

Deskripsi

- Antarmuka ini digunakan pada use case LogarithmicImageProcessing (UC-COSTREXRAPIC-05).
- Metode Logarithmic Image Processing ini merupakan salah satu metode yang akan digunakan untuk mengontraskan citra.

Event

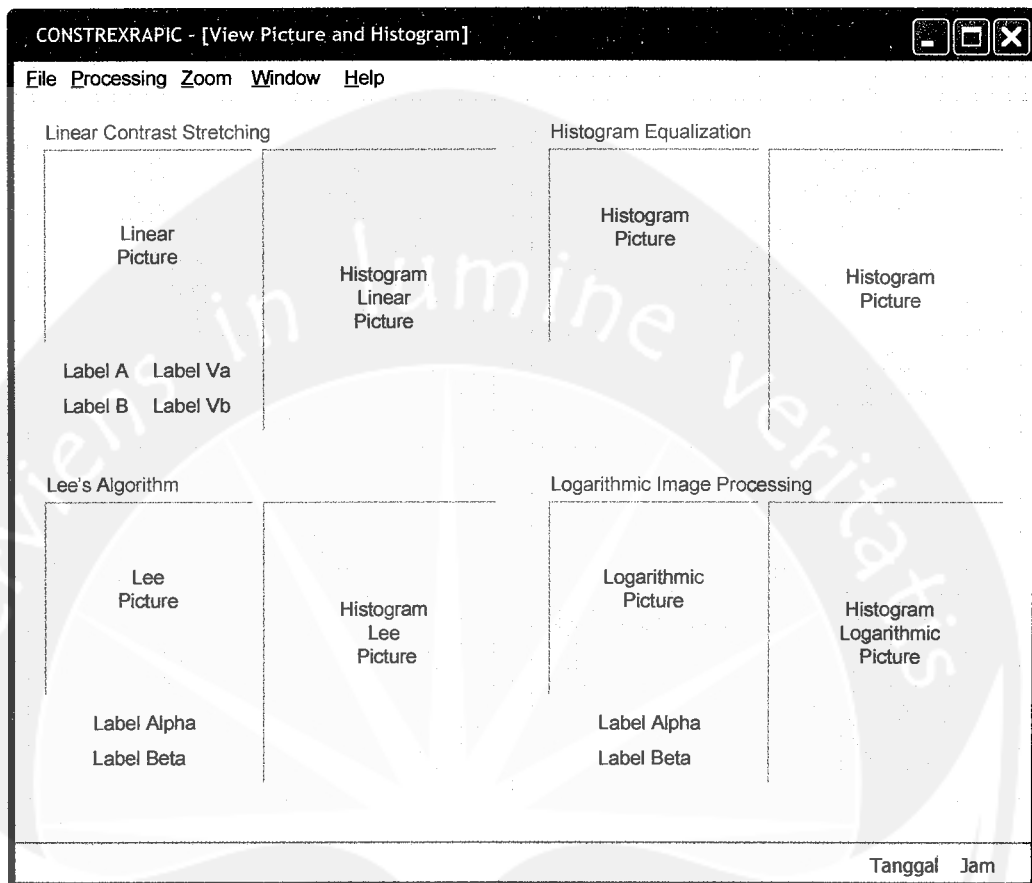
Urutan aksi yang terjadi :

1. User memilih sub menu Logarithmic Image Processing pada form main menu.
2. Sistem akan melakukan **OpenPicture()** dengan mengeset **picLogarithmic** yang merupakan pictureBox dalam

frmLogarithmic dengan **FromFile(fileLoc)**. **Show()** digunakan untuk menampilkan citra ke dalam **pictureBox**.

3. User menekan button Histogram untuk mengetahui histogram citra asli.
4. Sistem akan memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLogarithmic()** dan **drawHistogramLogarithmic()**.
5. Kemudian user menginputkan nilai alpha, dan beta dalam textbox dan menekan button Process.
6. Sistem akan memanggil class **Method** yang di dalamnya terdapat **LogarithmicProcess()** yang digunakan untuk melakukan pengontrasan citra dengan menggunakan metode Logarithmic Image Processing.
7. Dalam prosedur **LogarithmicProcess()** dilakukan alih ragam pixel, kemudian dilakukan pencarian nilai rerata (seperti Lee's Algorithm tetapi dicari nilai log nya). Setelah itu dilakukan pencarian **pixelAsli** yang juga dialihragamkan. Dengan input **alpha** dan **beta** maka didapat **pixel hasil** yang nantinya dilakukan operasi **exponen**. Yang terakhir dilakukan pengurangan nilai **L = 255** dengan **pixel hasil exponen**. Kemudian dilakukan pengecekan untuk menghindarkan error, dan dilakukan **setPixel** dengan **pixelHasil** dalam komponen RGB.
8. User kemudian menekan button Histogram kembali untuk melihat perubahan histogram citra yang telah dikontraskan dengan metode Logarithmic Image Processing.
9. Sistem akan kembali memanggil class **OperatorTampilHisto**, kemudian melakukan **calcHistoLogarithmic()** dan **drawHistogramLogarithmic()**.

3.7 Use Case : View



Gambar 3.7 Rancangan Antarmuka frmView

Deskripsi

- Antarmuka ini digunakan pada use case View (UC-COSTREXRAPIC-06).
- Antarmuka ini digunakan untuk menampilkan perbandingan hasil pengontrasan keempat metode, yaitu Linear Contrast Stretching, Histogram Equalization, Lee's Algorithm, dan Logarithmic Image Processing.

Event

Urutan aksi yang terjadi :

1. User memilih sub menu View pada form main menu.

2. Sistem akan mengecek label keempat form anak yang telah terbuka yaitu “**Linear Contrast Stretching**”, “**Histogram Equalization**”, “**Lee’s Algorithm**”, dan “**Logarithmic Image Processing**”.
3. Sistem akan menampung hasil pengontrasan dengan mendeklarasikan variabel **tampung**, **tampung2**, **tampung3**, dan **tampung4** yang bertipe **integer** dan diset sama dengan -1.
4. Jika form semua metode aktif (**aktifLin=1**, **aktifHis=1**, **aktifLee=1**, dan **aktifLog=1**) maka gambar (**picLinear**, **picHistogram**, **picLee**, **picLogarithmic**), dan histogram (**picHistoLinear**, **picHistoHistogram**, **picHistoLee**, **picHistoLogarithmic**), serta parameter input (**frmLinear** yaitu **valueA**, **valueB**, **valueVa**, **valueVb**, **frmLee** yaitu **valueAlphaLee**, **valueBetaLee**, dan **frmLogarithmic** yaitu **valueAlphaLog**, **valueBetaLog**) akan diset ke **frmView**.

3.8 Use Case : SavePicture

Deskripsi

- Antarmuka ini digunakan pada use case Save Picture (UC-COSTREXRAPIC-07).
- Penyimpanan dilakukan setelah proses pengontrasan citra dengan metode Linear Contrast Stretching, Histogram Equalization, Lee’s Algorithm, maupun Logarithmic Image Processing.

Event

Urutan aksi yang terjadi :

1. Setelah proses pengontrasan selesai selanjutnya dilakukan penyimpanan, apabila citra ingin disimpan.
2. User memilih sub menu Save untuk melakukan penyimpanan citra hasil pengontrasan.
3. Sistem akan menampilkan **SaveFileDialog**. Pertama **CheckPathExists**, **ValidateNames** didet **true**. File gambar yang akan disimpan diberi **filter** hanya disimpan dengan tipe **.jpg** dan

.bmp. SaveFileDialog diberi **Title**. Kemudian dilakukan proses penyimpanan, apabila penyimpanan sukses maka akan muncul **MessageBox** “**The image has been saved**”, dan apabila penyimpanan error akan muncul **MessageBox** “**Error Saving**”.

3.9 Use Case : ZoomIn

Deskripsi

- Antarmuka ini digunakan pada use case ZoomIn (UC-COSTREXRAPIC-08).
- Zoom In digunakan untuk memperbesar ukuran citra sehingga dapat dilihat dengan jelas hasil pengontrasan.

Event

Urutan aksi yang terjadi :

1. Untuk memperbesar ukuran citra dilakukan zoom in.
2. User memilih sub menu Zoom In pada form main menu.
3. Sistem akan menginisialisasi z yang bertipe **integer** menjadi $z = 0$, kemudian meng-increment $z++$, jika $z \leq 5$ maka **pictureBox** dalam form anak yang dikenai operasi zoom in akan melakukan **StretchImage**. Kemudian **Width** dan **Height** **pictureBox** akan dilakukan operasi perkalian dengan suatu konstansta **Zoom** yang sebelumnya telah diinisialisasi **Zoom = 1.2** dan bertipe **double**. Apabila $z > 5$ maka z diinisialisasi sama dengan 5.
4. User dapat melakukan zoom in sampai 5 kali perbesaran dari posisi normal sesuai dengan skala yang ditentukan sistem yaitu 1.2.

3.10 Use Case : ZoomOut

Deskripsi

- Antarmuka ini digunakan pada use case ZoomOut (UC-COSTREXRAPIC-09).

- Zoom Out digunakan untuk memperkecil ukuran citra sesuai yang diinginkan user.

Event

Urutan aksi yang terjadi :

1. Untuk memperkecil ukuran citra dilakukan zoom out.
2. User memilih sub menu Zoom Out pada form main menu.
3. Sistem akan menginisialisasi z yang bertipe **integer** menjadi $z = 0$, kemudian melakukan decrement $z--$, jika $z \leq 5$ dan $z \geq -10$ maka pictureBox dalam form anak yang dikenai operasi zoom out akan melakukan **StretchImage**. Kemudian **Width** dan **Height** pictureBox akan dilakukan operasi pembagian dengan suatu konstanta **Zoom** yang sebelumnya telah diinisialisasi **Zoom = 1.2** dan bertipe **double**. Kemudian sistem akan memanggil prosedur **CenterImage()** untuk menengahkan gambar ketika dilakukan zoom out. Apabila kondisi di atas tidak terpenuhi maka z diinisialisasi sama dengan -10 .
4. User dapat melakukan zoom out sampai 10 kali pengecilan dari posisi normal sesuai dengan skala yang ditentukan sistem yaitu **1.2**.

3.11 Use Case : AutoFit

Deskripsi

- Antarmuka ini digunakan pada use case AutoFit (UC-COSTREXRAPIC-10).
- Auto Fit digunakan untuk menyesuaikan ukuran citra sesuai dengan ukuran pictureBox.

Event

Urutan aksi yang terjadi :

1. Untuk menyesuaikan ukuran citra dengan ukuran picture box dilakukan auto fit.
2. User memilih sub menu Auto Fit pada form main menu.
3. Sistem akan memanggil prosedur **RatioStretch()**. Didalam prosedur ini dilakukan pengesetan **pRatio** dan **imRatio** yang bertipe **float**.

pRatio diset dengan **Width** dibagi **Height** dan **imRatio** diset dengan **Width pictureBox** dibagi dengan **Height pictureBox**.

4. Jika **Width \geq Width pictureBox** dan **Height \geq Height pictureBox**, dan jika **Width panel \leq Width pictureBox** dan **Height panel \leq Height pictureBox** maka dilakukan **StretchImage** untuk **SizeMode**nya kemudian set **Width** dan **Height pictureBox** dengan **Width** dan **Height panel**, setelah itu baru dilakukan **CenterImage()** untuk menengahkan gambar. Jika kondisi diatas tidak terpenuhi maka tidak perlu dilakukan **StretchImage**, hanya dilakukan pengesetan **Width** dan **Height pictureBox** dengan **Width** dan **Height pictureBox** itu sendiri, kemudian memanggil prosedur **CenterImage()**.
5. Jika **Width $>$ Width pictureBox** dan **Height $<$ Height pictureBox**, maka dilakukan pengecekan kembali jika **Width panel $<$ Width pictureBox** dan **Height panel $<$ Height pictureBox**, dilakukan pengecekan lagi jika **Width pictureBox $<$ Height pictureBox** maka sistem akan melakukan **StretchImage** pada **SizeMode**nya, kemudian mengeset **Width pictureBox** dengan membagi **Height panel** dengan **pRatio**. Setelah itu mengeset **Height pictureBox** dengan perkalian antara **Width pictureBox** dengan **pRatio**. Baru kemudian dilakukan **CenterImage()** yang digunakan untuk menengahkan gambar.
6. Apabila **Width pictureBox $>$ Height pictureBox** maka dilakukan **StretchImage** pada **SizeMode**nya, **Height pictureBox** diset sama dengan **Height panel** dan **Width pictureBox** diset sama dengan **Width panel**. Setelah itu sistem akan memanggil **CenterImage()**.
7. Apabila **Width panel $>$ Width pictureBox** dan **Height panel $>$ Height pictureBox** maka dilakukan **StretchImage**, mengeset **Height pictureBox** dengan **Height panel**, mengeset **Width pictureBox** dengan perkalian **Height panel** dengan **imRatio**. Baru kemudian melakukan **CenterImage()**.

8. Jika $\text{Width} < \text{Width pictureBox}$ dan $\text{Height} > \text{Height pictureBox}$, maka Width pictureBox diset dengan Width , dan Height pictureBox diset dengan Width dibagi imRatio .
9. Jika $\text{Width} < \text{Width pictureBox}$ dan $\text{Height} < \text{Height pictureBox}$, dan jika $\text{Width} \geq \text{Height}$, dan jika $\text{Height pictureBox} > \text{Width pictureBox}$ maka set SizeMode nya dengan StretchImage , set Width pictureBox dengan $\text{Width panel} - 20$, set Height pictureBox dengan pembagian Width panel dengan imRatio .
10. Tetapi jika $\text{Height pictureBox} < \text{Width pictureBox}$ maka SizeMode diset StretchImage , Width pictureBox diset dengan Width panel , Height pictureBox diset dengan Height panel , baru kemudian dilakukan $\text{CenterImage}()$.
11. Tetapi jika $\text{Width pictureBox} < \text{Height pictureBox}$ dan $\text{imRatio} < \text{pRatio}$ dan jika $\text{Height pictureBox} > \text{Width pictureBox}$ maka StretchImage , set $\text{Height pictureBox} = \text{Height panel}$, $\text{Width pictureBox} = \text{perkalian Height panel dengan imRatio}$, baru kemudian dilakukan $\text{CenterImage}()$.
12. Tetapi jika $\text{Height pictureBox} < \text{Width pictureBox}$ maka StretchImage , Height pictureBox diset dengan Height panel , Width pictureBox diset dengan Width panel kemudian dilakukan ZoomOut dan $\text{CenterImage}()$.
13. Jika $\text{Width} < \text{Height}$ dan jika $\text{Width pictureBox} < \text{Height pictureBox}$ dan $\text{imRatio} < \text{pRatio}$ maka StretchImage , Height pictureBox diset sama dengan Height panel , dan Width pictureBox diset sama dengan Height panel dikali imRatio .
14. Tetapi jika $\text{Width pictureBox} > \text{Height pictureBox}$ dan $\text{imRatio} > \text{pRatio}$ maka dilakukan StretchImage , Width pictureBox diset sama dengan Width panel dan Height pictureBox diset sama dengan Width panel dibagi imRatio .

Apendiks A : Daftar Istilah dan Singkatan

Class Diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang dikembangkan.

COSTREXPAPIC adalah singkatan dari *Contrast Stretching X-Ray Picture*, yaitu perangkat lunak yang ditujukan untuk dunia kedokteran, berfungsi untuk meningkatkan kualitas citra *x-ray* dengan mengkontraskan citra dengan 3 metode yaitu *Linear Contrast Stretching*, *Histogram Equalization*, dan *Logarithmic Image Processing*.

C# adalah bahasa pemrograman yang sederhana, moderen, berorientasi objek dan type-safe, yang diturunkan dari C dan C++. C# menggabungkan produktifitas dari Visual Basic dan keampuhan dari C++.

Flow of events adalah aliran event-event yang terjadi dalam suatu proses, yang digunakan untuk mendeskripsikan urutan jalannya proses.

Histogram Equalization adalah salah satu metode peningkatan kualitas citra dengan melibatkan fungsi alih non linear dan tak monoton untuk memetakan intensitas piksel.

Lee's Algorithm adalah algoritma peningkatan kualitas citra oleh Lee yang nantinya akan diusulkan oleh Deng menjadi *Logarithmic Image Processing*.

Linear Contrast Stretching adalah salah satu metode peningkatan kualitas citra dengan merentangkan kisaran nilai intensitas yang sempit menjadi kisaran intensitas yang diinginkan [0,255].

Logarithmic Image Processing adalah salah satu metode peningkatan kualitas citra yang didasarkan pada suatu struktur matematika untuk pengolahan gambar yang logaritmis.

Microsoft Visual Studio .NET adalah sebuah *platform* untuk membangun, menjalankan dan meningkatkan generasi lanjut dari aplikasi terdistribusi. Referensi lengkap mengenai Microsoft Visual Studio .NET dapat dilihat pada <http://msdn.microsoft.com/vstudio/>

Sequence Diagram adalah diagram interaksi antar kelas yang menekankan pada waktu urutan waktu pesan dan berfokus pada kelas dan pesan yang dipertukarkan untuk mendapatkan perilaku yang diinginkan.

