

IMPLEMENTASI METODE MULTISCALE RETINEX UNTUK IMAGE ENHANCEMENT MENGGUNAKAN DELPHI

Deni Supriadi¹, Panji Novantara²

Teknik Informatika SI

Fakultas Ilmu Komputer Universitas Kuningan

Jalan Tjut Nyak Dhien Cijoho Kuningan Telepon (0232) 873696 Fax. (0232) 874 824

denisupriadi666@gmail.com¹, panji@uniku.ac.id²

ABSTRAK

Sekarang ini penggunaan citra digital semakin meningkat karena banyak kelebihan yang dimiliki oleh citra digital tersebut, antara lain kemudahan dalam mendapatkan gambar, memperbanyak gambar, pengolahan gambar dan lain-lain. Tetapi tidak semua citra digital memiliki tampilan visual yang memuaskan mata manusia. Ketidakpuasan itu dapat timbul karena adanya gangguan pada kualitas citra. Situasi dan kondisi saat pengambilan citra juga berpengaruh terhadap kualitas citra yang membuat citra menjadi terlalu gelap atau terang. Perbaikan kualitas citra dapat dilakukan dengan berbagai metode, salah satunya adalah metode *Multiscale Retinex*. Kelebihan dari metode *Multiscale Retinex* dapat memperbaiki kualitas citra dengan meningkatkan warna asli citra dan kecerahan citra terutama saat pengambilan citra dalam kondisi gelap. Jenis citra yang digunakan berformat **.bmp*, **.jpg* dan **.png*. Proses diawali dengan memasukan citra kedalam program kemudian diproses dengan metode *Multiscale Retinex*. Perbandingan citra asal dan citra hasil bisa dilihat pada histogram. Penelitian yang dilakukan menghasilkan aplikasi pengolahan citra untuk meningkatkan kecerahan citra.

Kata Kunci : *Multiscale Retinex*, Citra, Histogram.

ABSTRACT

*Nowadays the use of digital image is increasing because there are many advantages possessed by the digital image. Furthermore, the advantages of digital image are easy in getting the picture, reproducing the image, processing image and others. But, not all digital images have a visual appearance that satisfies the human eye. While, disadvantage of digital image is that it can arise because of interference in the image quality. Hence, the situations and conditions of image capture also affect the image quality that make the image too dark or light. Improving image quality can be done with various methods, one of them is the Multiscale Retinex method. Meanwhile, the advantages of Multiscale Retinex method can improve image quality by increasing the original color of image and the image brightness especially when an image is took in dark condition. Then, the type of image are used *.bmp *.jpg and *.png. The next process begins by inserting the image into the program and then it is processed by Multiscale Retinex method. In addition, the comparison of the original image and the result image can be seen in the histogram.*

Keywords: *Multiscale, Retinex, Image, Histogram.*

1. PENDAHULUAN

Sekarang ini penggunaan citra digital semakin meningkat karena banyak kelebihan yang dimiliki oleh citra digital tersebut, antara lain kemudahan dalam mendapatkan gambar, memperbanyak gambar, pengolahan gambar

dan lain-lain. Tetapi tidak semua citra digital memiliki tampilan visual yang memuaskan mata manusia. Ketidakpuasan itu dapat timbul karena adanya gangguan pada kualitas citra (noise) dan kualitas pencahayaan pada citra yang terlalu gelap atau terlalu terang. Untuk itu perlu dilakukan proses perbaikan citra untuk meningkatkan mutu citra. Bidang ilmu

pengetahuan yang membahas tentang proses peningkatan mutu citra tersebut yaitu pengolahan citra (*Digital Image Processing*). Salah satu cabang dalam ilmu pengolahan citra yaitu perbaikan citra (*Image Enhancement*). Perbaikan citra (*Image Enhancement*) merupakan salah satu proses awal dalam pengolahan citra (*Image Processing*). Dengan pengolahan citra, citra dapat diubah menjadi citra yang sesuai keinginan. Adapun proses-proses yang termasuk dalam perbaikan kualitas citra adalah perubahan kecerahan gambar (*Image Brightness*), peregangan kontras (*Contrast Stretching*), perubahan histogram (histogram equalization), pelembutan citra (*Image Smoothing*), penajaman tepi (*Sharpening Edge*), pewarnaan semu (*Pseudocolouring*), perubahan geometrik.

Salah satu metode yang digunakan adalah metode *Multiscale Retinex*. Kelebihan dari metode *Multiscale Retinex* dapat memperbaiki kualitas citra dengan meningkatkan warna asli citra dan kecerahan citra terutama saat pengambilan citra dalam kondisi gelap. Dengan menggunakan metode *Multiscale Retinex* kita dapat memperbaiki kualitas citra pada pencerahannya sehingga menghasilkan citra yang lebih baik.

Berdasarkan latar belakang diatas maka penulis melakukan penelitian dengan judul **“IMPLEMENTASI METODE MULTISCALE RETINEX UNTUK IMAGE ENHANCEMENT MENGGUNAKAN DELPHI”**.

1.1 Identifikasi Masalah

Dari latar belakang yang sudah dijabarkan di atas, maka penulis mengidentifikasi masalah. Berikut identifikasi masalah yang perlu diketahui :

1. Bagaimana caranya memperbaiki kecerahan dari sebuah citra agar bagian yang terlalu gelap menjadi terang dengan menggunakan metode *Multiscale Retinex*?
2. Bagaimana kualitas citra setelah melalui proses metode *Multiscale Retinex* ?

1.2 Batasan Masalah

Berikut batasan masalah yang penulis sesuaikan dengan penelitian di atas :

1. Membangun aplikasi *Image Enhancement* menggunakan metode *Multiscale Retinex*.

2. Penggunaan metode *Multiscale Retinex* hanya untuk kecerahan citra digital.
3. Jumlah *scale* yang digunakan *Multiscale Retinex* pada aplikasi berjumlah tiga.
4. Citra yang digunakan adalah jenis citra warna dengan format file **.bmp *.png dan *.jpg..*
5. Ukuran maksimal *input* citra adalah 1200x1200 piksel.
6. *File* citra yang sudah diproses disimpan dengan nama baru.
7. Perbandingan kecerahan citra awal dan citra hasil bisa dilihat pada histogramnya.
8. Perancangan dan pembuatan perangkat lunak ini menggunakan bahasa pemrograman Borland Delphi 2010 Embarcadero.

1.4 Tujuan Penelitian

Adapun tujuan yang akan dicapai dari penulis skripsi ini adalah sebagai berikut :

1. Mengimplementasikan metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan bahasa pemrograman Borland Delphi 2010 Embarcadero.
2. Sistem dapat memperbaiki pencerahan dari sebuah citra agar bagian yang terlalu gelap menjadi terang dengan menggunakan Metode *Multiscale Retinex*.
3. Untuk membuat aplikasi *Image Enhancement* dengan metode *Multiscale Retinex* agar menarik dan mudah digunakan oleh *user*.

2. Metodologi Penelitian

2.1 Metode Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam pelaksanaan penulisan ini adalah dengan menggunakan metode studi kepustakaan, dengan metode ini penulis memperoleh data dan informasi mengenai tema yang diangkat dari berbagai jurnal ilmiah, karya ilmiah, buku-buku dan sumber lainnya yang berhubungan dengan metode *Multiscale Retinex* untuk perbaikan citra.

2.2 Metode Pengembangan Sistem

Metode penyelesaian masalah yang digunakan dalam penelitian ini adalah

menggunakan metode RUP (*Rational Unified Process*).

1. *Permulaan (Inception)*

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirements*). Pada fase ini akan dilakukan tugas-tugas sebagai berikut :

- a) Pengenalan masalah, yaitu memahami permasalahan dalam membangun aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan delphi.
- b) Pembuatan proposal untuk membangun aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan delphi mencakup latar belakang permasalahan, pokok permasalahan, tujuan, batasan masalah dan jadwal pelaksanaan kegiatan penelitian untuk membangun aplikasi tersebut.
- c) Studi literatur yang berhubungan dengan pembangunan aplikasi, mencakup pengumpulann data - data, penelusuran teori - teori yang berhubungan dengan permasalahan.

2. *Perluasan / Perencanaan (Elaboration)*

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur yang diinginkan dapat dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih analisis dan desain sistem serta implementasi sistem yang fokus pada *prototype*. Pada fase *elaboration* akan dilakukan tugas-tugas sebagai berikut:

- a) Pembuatan deskripsi arsitektur aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan delphi.
- b) Deskripsi kebutuhan aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan delphi.
- c) Deskripsi pengguna.
- d) Penyempurnaan model *use case* (>80%).
- e) Penyempurnaan perancangan sistem dengan *Unified Modeling Language* (UML).

- f) Perancangan tampilan input dan output sistem yang akan dibangun.

3. *Konstruksi (Construction)*

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak secara lengkap yang siap diserahkan kepada pemakai. Pada fase ini akan dilakukan tugas – tugas berikut :

- a) Membangun aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* yang berpedoman pada model *use case*, menggunakan bahasa pemrograman Borland Delphi 2010.
- b) Melakukan pengujian *white box*, apakah sudah memenuhi fungsi yang diharapkan atau belum.
- c) Implementasi aplikasi pada perangkat komputer atau laptop.

4. *Transisi (transition)*

Tahap ini lebih pada instalasi sistem agar dapat dimengerti oleh *user*. Aktifitas pada tahap ini termasuk pelatihan *user*, pemeliharaan dan pengujian sistem apakah sistem sudah memenuhi harapan *user*. Pada fase ini akan dilakukan tugas – tugas sebagai berikut :

- a) Melakukan pengujian menggunakan *black box testing*.
- b) Mengamati kekurangan yang ada terhadap aplikasi yang telah dilakukan pengujian.
- c) Menyimpulkan fitur – fitur tambahan guna pengembangan aplikasi implementasi metode *Multiscale Retinex* untuk *Image Enhancement* menggunakan delphi.

3 LANDASAN TEORI

3.1 *Pengertian Implementasi*

Implementasi merupakan proses untuk melaksanakan ide, proses atau seperangkat aktivitas baru dengan harapan orang lain dapat menerima dan melakukan penyesuaian dalam tubuh birokrasi demi terciptanya satu tujuan yang bisa tercapai dengan jaringan pelaksana yang bisa dipercaya (*Septian, 2010*).

Implementasi adalah suatu tindakan atau pelaksanaan dari sebuah rencana yang sudah disusun secara matang dan terperinci. Implementasi biasanya dilaksanakan setelah perencanaan sudah dianggap fix. Secara umum istilah implementasi berarti pelaksanaan atau penerapan. Istilah implementasi biasanya dikaitkan dengan suatu kegiatan yang dilaksanakan untuk mencapai tujuan tertentu. (Al Fatta, 2007).

Kesimpulan dari implementasi adalah suatu proses untuk melaksanakan ide dari sebuah rencana yang sudah disusun secara matang dan terperinci kemudian dilaksanakannya setelah perencanaan sudah dianggap fix.

3.2 Pengertian Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau ilustrasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpan (Sutoyo, T et al, 2009).

Penampilan citra dapat dibagi jadi dua bagian yaitu citra diam (*still image*) dan citra bergerak (*moving image*). Citra diam adalah citra tunggal yang tidak bergerak. Citra bergerak adalah rangkaian citra diam yang ditampilkan secara beruntun (*sequential*) hingga memberi kesan pada mata seolah-olah gambar tersebut bergerak (Munir, R, 2004).

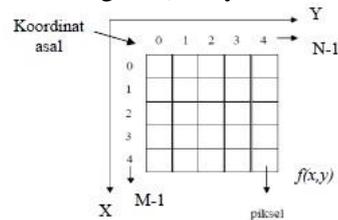
Kesimpulannya citra merupakan suatu representasi atau ilustrasi dari suatu objek, penampilan citra dibagi dua yaitu diam (*still image*) dan citra bergerak (*moving image*).

3.3 Pengertian Citra Digital

Citra digital adalah citra yang dapat diolah oleh komputer. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Citra digital dihasilkan dengan proses digitalisasi terhadap citra kontinu. Yang mana proses ini dihasilkan dari peralatan digital yang langsung bisa diproses oleh komputer.

Citra digital yang tersusun dalam bentuk *grid*. Setiap kotak yang disebut piksel dan memiliki koordinat (x,y) adalah f(x,y). Sumbu x yaitu baris, sedangkan sumbu y yaitu

kolom. Setiap piksel memiliki nilai yang menunjukkan intensitas warna pada piksel tersebut, Gambar 2.1 menunjukkan posisi koordinat citra digital (Sutoyo, T et al, 2009).



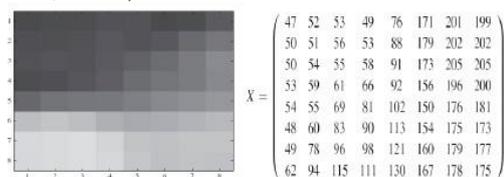
Gambar 3.1 Koordinat Citra Digital (Sutoyo, T et al : 2009)

Citra digital dinyatakan dengan matriks berukuran N x M (kolom / tinggi = N, baris / lebar = M). Piksel mempunyai dua parameter, yaitu koordinat dan intensitas warna dari piksel di titik itu. Sehingga citra digital dapat ditulis dalam bentuk matriks sebagai berikut :

$$f(x,y) = \begin{bmatrix} f(x-1,0) & f(x-1,1) & \dots & f(x-1,N-1) \\ \dots & \dots & \dots & \dots \\ f(x,0) & \dots & \dots & f(x,N-1) \\ f(x,0) & f(x,1) & \dots & f(x,N-1) \end{bmatrix}$$

Gambar 3.2 Matriks Citra Digital N x M (Sutoyo, T et al, 2009)

Berdasarkan gambaran tersebut, citra digital dapat dituliskan sebagai fungsi intensitas f(x,y), dimana harga x (baris) dan y (kolom) merupakan koordinat posisi dan f(x,y) adalah nilai fungsi pada setiap titik (x,y) yang dititik tersebut. Merepresentasikan sebuah citra ukuran 8 x 8 piksel dengan mengambil derajat keabuan pada tiap piksel serta matriks yang terdiri dari 8 baris dan 8 kolom (Sutoyo, T et al, 2009).



Gambar 3.3 Ilustrasi Sistem Matrik Citra Digital 8 x 8 Piksel (Sutoyo, T et al : 2009)

Dapat dilihat dari gambar 3.3 bahwa pada piksel pertama pada koordinat (1,1) mempunyai derajat 47 yang mewakili dari beberapa gradasi warna, selanjutnya piksel kedua pada koordinat (2,1) memiliki derajat keabuan 52 yang mewakili beberapa gradasi warna dan seterusnya.

3.4 Perbaikan Kualitas Citra

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

Operasi-operasi yang digolongkan sebagai perbaikan kualitas citra cukup beragam antara lain, perubahan kecerahan gambar (*image brightness*), peregangan kontras (*contrast stretching*), perataan histogram (*histogram equalization*), pelembutan citra (*image smoothing*), penajaman (*sharpening*) tepi (*edge*), pewarnaan semu (*pseudocolouring*), perubahan geometrik, dan sebagainya. (Munir, 2004)

3.4 Metode SSR (Single-Scale Retinex)

Single-Scale Retinex (SSR) merupakan *Retinex* versi dinamis untuk meniru sistem kerja neuron pada sistem persepsi warna manusia. Berikut ini adalah formulasi *Single-Scale Retinex*

$$R_i(x,y) = \log I_i(x,y) - \log [F(x,y) * I_i(x,y)]$$

Dimana $R_i(x,y)$ merupakan *output Retinex* $I_i(x,y)$ adalah distribusi citra pada piksel ke (x,y) . Simbol "*" menyatakan operator konvolusi. Simbol i menyatakan *channel* warna, misalnya *Red*(R), *Green*(G), dan *Blue*(B). $F(x,y)$ merupakan fungsi *Gaussian* yang didefinisikan sebagai berikut :

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Dimana $G(x,y)$ merupakan *Gaussian kernel* pada piksel (x,y) . Simbol π atau pi merupakan konstanta yaitu $22/7$. Simbol σ merupakan nilai sigma. Simbol (x,y) adalah kordinat piksel. Sedangkan simbol e merupakan ketetapan yaitu 2.7182818246 *Gaussian filter* digunakan pada citra supaya citra lebih halus dalam memproses gambar. *Gaussian filter* juga bertujuan untuk

menghilangkan *noise* pada citra dan meningkatkan kualitas detail citra.

Konvolusi merupakan perkalian total dari dua buah fungsi matriks f dan kernel g . Untuk mengaplikasikan penapis/*filter* pada citra, digunakan metode konvolusi. Konvolusi bisa dinyatakan dalam matriks, dimana setiap elemen matriks penapis merupakan koefisien konvolusi. Operasi konvolusi merupakan penggeseran kernel piksel per piksel dan hasil disimpan dalam matriks baru.

Berikut hasil ilustrasi konvolusi, yaitu :

		A1	A2	A3		
		A4	A5	A6		
		A7	A8	A9		

$f(x,y)$

A	B	C
D	E	F
G	H	I

$g(x,y)$

Maka proses perhitungan konvolusi dapat dilakukan dengan : $f(x,y) * g(x,y)$

$$f(x,y) = AA1 + BA + CA + DA + EA + FA + GA + HA + IA$$

3.5 Metode MSR (Multiscale Retinex)

Multi-Scale Retinex (MSR) perkembangan dari *Singlescale Retinex* karena keterbatasan yang dimiliki oleh *Single-Scale Retinex* (SSR), yaitu sebagai berikut.

- 1) SSR dapat melakukan kompresi jarak dinamis pada citra dengan skala rendah, sehingga memungkinkan citra dengan jarak dinamis sangat lebar dikompresi dengan melakukan penguatan bagian gelap dan melemahkan bagian yang terang. Sedangkan skala besar, SSR dapat menghasilkan citra lebih alami. Tetapi SSR tidak dapat melakukan kedua hal tersebut sekaligus.

- 2) SSR cenderung menghasilkan area berwarna sama menjadi berwarna abu-abu, terutama jika digunakan skala rendah.

Ide dari dasar *Multi-Scale Retinex* untuk masalah SSR terdapat pada skala yang berbeda yang digunakan dan diberi bobot yang berbeda untuk menggabungkan kelebihan dan menghilangkan kelemahan yang dimiliki dari skala rendah dan skala besar. Berikut adalah formulasi original *Multiscale Retinex*.

$$R_{msr_i} = \sum_{n=1}^N W_n R_{n_i}$$

R_{msr_i} merupakan *output* dari *Multi-Scale Retinex* (MSR) yang merupakan jumlah dari *output* SSR yang masing-masing diberi bobot. N adalah jumlah skala yang digunakan. w_n adalah bobot yang diasosiasikan dengan skala ke- n . R_{n_i} merupakan *output* dari SSR yang diasosiasikan dengan skala ke- n . Simbol i menyatakan *channel* warna, misalnya *Red*(R), *Green*(G), dan *Blue*(B). Berdasarkan formula MSR tersebut, konstruksi MSR ditentukan oleh parameter-parameter berikut ini.

- 1) Jumlah Skala (N)
Jumlah skala yang digunakan adalah 3 sebagai jumlah skala minimum yang menyediakan *output* yang baik berdasarkan persepsi visual dan waktu komputasi yang cepat.
- 2) Sigma yang digunakan (σ_n)
Sigma yang digunakan sebaiknya kombinasi dari tiga tingkatan yaitu rendah, sedang, dan tinggi tergantung dengan dimensi pada citra.
- 3) Bobot untuk setiap input SSR (w_n)
Bobot yang digunakan dalam proses MSR harus sama dengan satu. Jika $N=3$, maka $W_n = W_1+W_2+W_3$ dimana $W_n = 1$. Nilai W_1 , W_2 , dan W_3 bisa disesuaikan dengan kebutuhan.

4 ANALISIS DAN PERANCANGAN

4.1 Analisis Sistem

Rancangan aplikasi perbaikan citra diperlukan sesuai dengan perkembangan teknologi saat ini. Dengan disediakannya metode *Multiscale Retinex* sehingga dapat dibandingkan melalui hasil dan kualitas oleh aplikasi perbaikan citra ini. Seluruh kalangan masyarakat pasti dapat merasakan perkembangan teknologi yang

semakin pesat, khususnya untuk sebuah kualitas citra yang lebih baik sehingga aplikasi ini dapat dimanfaatkan untuk memperbaiki kualitas citra yang kurang baik menjadi lebih baik pada kecerahannya.

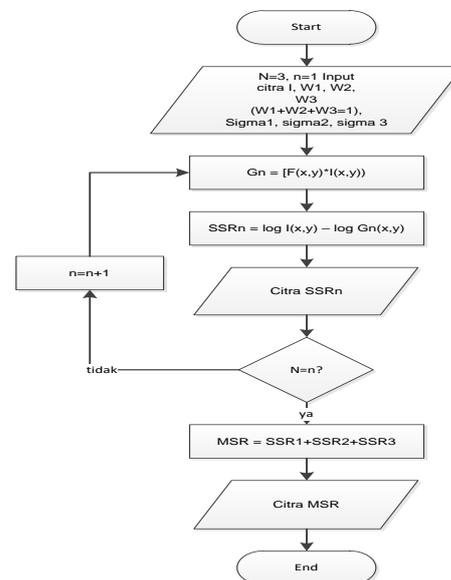
4.2 Analisis Kebutuhan Fungsional

Dalam perancangan sebuah perangkat lunak diperlukan rekayasa persyaratan yang merupakan sebuah proses untuk menemukan, menganalisis, mendokumentasikan dan memeriksa layanan serta batasan sistem. Terdapat beberapa hal yang harus dipenuhi pada persyaratan fungsional, antara lain:

- 1) Sistem akan melakukan perbaikan citra digital berformat *bmp*, *jpg*, *png* yang memiliki kualitas rendah.
- 2) Sistem melakukan perbaikan citra menggunakan metode *Multiscale Retinex*.
- 3) Sistem akan melakukan hasil histogram dari citra awal dan citra hasil.

4.3 Analisis Penyelesaian Masalah

4.3.1 Flowchart Metode Multiscale Retinex



Gambar 4.1 Proses Metode Multiscale Retinex
Keterangan :

- 1) Mulai
- 2) Diketahui $N=3$ adalah jumlah skala yang digunakan, $n=1$ untuk memulai proses SSR pertama, input citra yang akan diproses, W adalah nilai weight dimana $W_1+W_2+W_3$ jumlah nilainya harus sama dengan 1, Sigma_1 , Sigma_2 dan Sigma_3

adalah sigma yang akan digunakan terdiri dari nilai rendah, sedang dan tinggi.

- 3) Mencari nilai *gaussian* dengan rumus $G_n = [F(x,y) * [x,y]]$
- 4) Melakukan proses SSR dengan rumus $SSR_n = \log \ln(x,y) - \log G_n(x,y)$
- 5) Setelah melakukan proses SSR_n akan mendapat citra hasil SSR_n
- 6) Jika $N = (n=1)$ maka akan mengulangi ke proses mencari *gaussian* dengan n ditambah 1, sampai mengulangi tiga kali proses sehingga didapat tiga *Singlescale Retinex* maka akan masuk ke proses *Multiscale Retinex*.
- 7) Melakukan proses *Multiscale Retinex* dengan cara menambahkan SSR_1, SSR_2 dan SSR_3 .
- 8) Setelah melakukan proses *Multiscale Retinex* akan mendapatkan citra hasil *Multiscale Retinex*.
- 9) Selesai.

4.3.2 Implementasi Singlescale Retinex

Tahap awal yang akan dilakukan untuk perbaikan citra menggunakan metode *Multiscale Retinex* (MSR), harus terlebih dahulu melewati tahap *Singlescale Retinex* (SSR), dengan melewati tiga tahap SSR. SSR adalah *Retinex* versi dinamis untuk meniru sistem kerja neuron pada sistem persepsi warna manusia. Berbeda dengan *Retinex* statis, *Retinex* versi dinamis tidak menggunakan jalur, tetapi menggunakan fungsi sekitar (*surround function*) untuk meradiasikan intensitas ke piksel sekitarnya, dengan rumus:

$$SSR_n(x,y) = \log \ln(x,y) - \log G_n(x,y)$$

4.3.4 Implementasi Multiscale Retinex

Tahap selanjutnya yang dilakukan untuk perbaikan citra (*Image Enhancement*) adalah metode *Multiscale Retinex* (MSR), dimana metode MSR merupakan pengembangan dari SSR, dengan rumus:

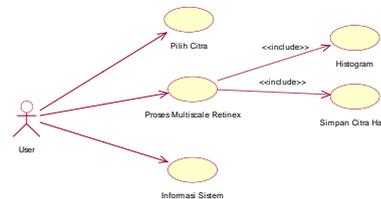
$$R_{MSR_i} = \sum_{n=1}^x W_n R_{ni}$$

R_{MSR_i} merupakan output dari *Multi-Scale Retinex* (MSR) yang merupakan jumlah dari output SSR yang masing-masing diberi bobot. N adalah jumlah skala yang digunakan. w_n adalah bobot yang diasosiasikan dengan

skala ke- n . R_{ni} merupakan output dari SSR yang diasosiasikan dengan skala ke- n . Simbol i menyatakan *channel* warna, misalnya *Red*(R), *Green*(G), dan *Blue*(B).

4.4 Use Case Diagram

Sebuah diagram use case adalah sebuah diagram yang menunjukkan hubungan antara aktor dan kasus penggunaan dalam sistem. Berdasarkan analisis kebutuhan sistem, secara garis besar sistem melakukan proses perbaikan kualitas citra dengan metode *Multiscale Retinex*. Gambar 3.3 adalah diagram use case yang dirancang sebagai pemodelan persyaratan sistem berdasarkan informasi kebutuhan sistem dan user yang berperan didalamnya.

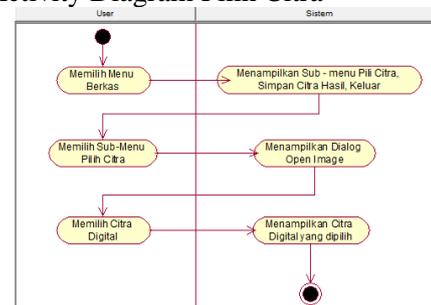


Gambar 4.2 Use Case Diagram

4.5 Activity Diagram

Activity Diagram menggambarkan berbagai alur proses bisnis secara keseluruhan atau aktivitas secara umum dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, decision/keputusan yang mungkin terjadi dan bagaimana alur proses tersebut berakhir.

1. Activity Diagram Pilih Citra

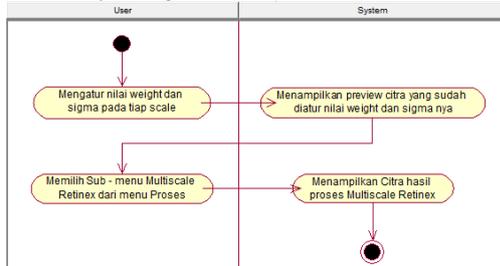


Gambar 4.3 Activity Diagram Pilih Citra
Keterangan :

- 1) User memilih menu Berkas.
- 2) Sistem menampilkan Sub-menu Pilih Citra, Simpan Citra Hasil, dan Keluar.
- 3) User memilih Sub-menu Pilih Citra.

- 4) Sistem menampilkan Dialog Open Image.
- 5) User memilih Citra Digital.
- 6) Sistem menampilkan Citra Digital yang dipilih.

2. Activity Diagram Proses *Multiscale Retinex*



Gambar 4.4 Activity Diagram Proses *Multiscale Retinex*

Keterangan :

- 1) User mengatur nilai *weight* dan *sigma* pada tiap scale.
- 2) Sistem menampilkan *preview* citra yang sudah diatur nilai *weight* dan *sigma* nya.
- 3) User memilih *Sub – Menu Multiscale Retinex* dari menu Proses.
- 4) Sistem menampilkan Citra hasil proses *Multiscale Retinex*.

3. Activity Diagram Informasi Sistem



Gambar 4.5 Activity Diagram Informasi Sistem

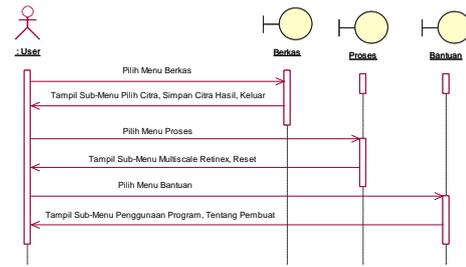
Keterangan :

- 1) User memilih Menu Bantuan.
- 2) Sistem menampilkan *Sub-menu* Penggunaan Program dan Tentang Pembuat
- 3) User memilih Penggunaan Program atau Tentang Pembuat
- 4) Sistem menampilkan *form menu* aplikasi sistem yang dipilih *user*

4.6 Sequence Diagram

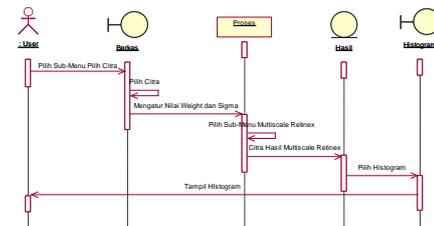
Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah – langkah yang dilakukan sebagai respons dari sebuah *event* (proses) untuk menghasilkan *output* tertentu.

1. Sequence Diagram Antar Menu



Gambar 4.5 Sequence Diagram Antar Menu

2. Sequence Diagram Proses *Multiscale Retinex*



Gambar 4.6 Sequence Diagram Proses *Multiscale Retinex*

5. IMPLEMENTASIDAN PENGUJIAN

5.1 Implementasi Antarmuka

Di dalam implementasi antarmuka, hanya menerangkan mengenai tampilan yang ada di dalam “Implementasi Metode *Multiscale Retinex* Untuk *Image Enhancement*”.

1. Halaman Utama



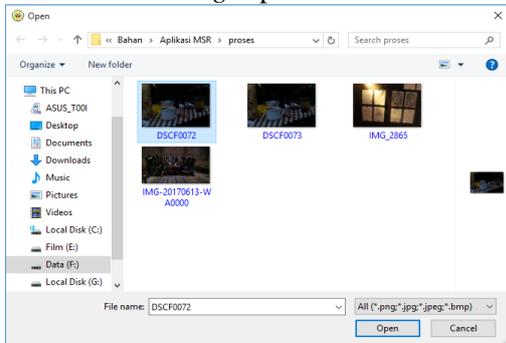
Gambar 5.1 Halaman Utama

Gambar 5.1 Form Halaman Awal adalah halaman yang akan dilihat pertama kali oleh *user* ketika menjalankan *program*. Dimana didalam *program* ini memiliki beberapa *menu* yang diantaranya :

- *Menu* Berkas, *menu* ini memiliki beberapa *sub-menu* diantaranya Pilih Citra, Simpan Citra Hasil, dan keluar.
- *Menu* Proses, *menu* ini memiliki *sub-menu* proses *Multiscale Retinex* dan *Reset*.

- *Menu Bantuan*, *menu* ini memiliki *sub-menu* tentang untuk menampilkan informasi pembuat aplikasi dan *sub-menu* bantuan untuk menampilkan cara penggunaan dari aplikasi.

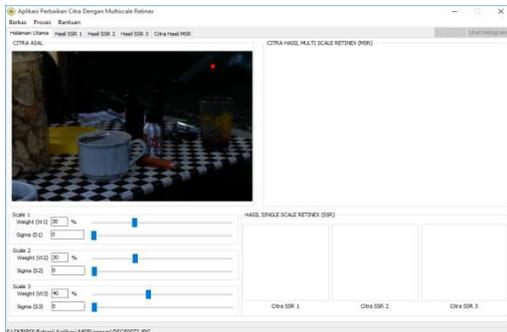
2. Halaman *Dialog Input Citra*



Gambar 5.2 Halaman *Dialog Input Citra*

Gambar 5.2 *Dialog Input Citra* adalah dialog yang akan muncul setelah *sub-menu* Buka Gambar di klik dan *user* diminta untuk memilih *file* citra yang akan di proses oleh aplikasi.

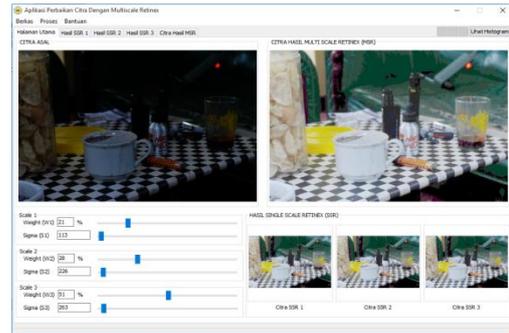
3. Halaman Citra Asal



Gambar 5.3 Halaman Citra Asal

Gambar 5.3 Halaman Citra Asal adalah halaman yang menampilkan citra yang di pilih oleh *user* untuk di proses menggunakan Metode *Multiscale Retinex* dan bisa dilihat *histogram*-nya untuk membandingkan citra asal dan nilai citra hasil Proses *Multiscale Retinex* pada Histogram.

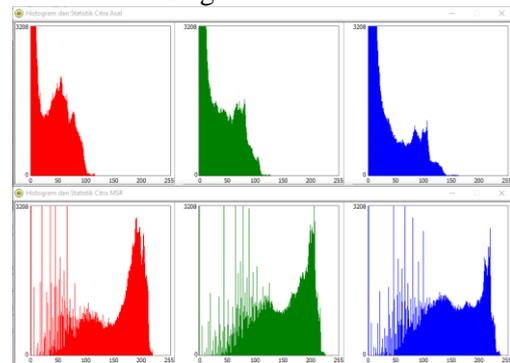
4. Halaman *Multiscale Retinex*



Gambar 5.4 Halaman *Multiscale Retinex*

Gambar 5.4 Halaman *Multiscale Retinex* terdapat pada *form* utama dan *user* harus mengatur nilai *weight* dan *sigma*. Dimana nilai $W1+W2+W3$ jumlahnya harus 100% dan nilai *sigma* yang digunakan sebaiknya kombinasi dari tiga tingkatan yaitu rendah, sedang, dan tinggi tergantung dengan dimensi pada citra. Dimana setelah mengatur nilai *weight* dan *sigma* *user* memilih *sub-menu* proses *Multiscale Retinex* maka akan muncul citra hasil *Singlescale Retinex* dan *Multiscale Retinex*.

5. Halaman Histogram



Gambar 5.5 Halaman Histogram

Gambar 5.3 Halaman *Histogram Citra Asli* dan Citra Hasil proses *Multiscale Retinex* adalah halaman yang menampilkan *histogram RGB* dari citra hasil atau citra yang sudah di proses. Dapat dilihat bahwa proses pengolahan citra dengan metode *Multiscale Retinex* akan menghasilkan citra yang lebih terang, dan citra akan semakin terang jika nilai *sigma* semakin besar seperti. Gambar 5.5 menjelaskan histogram pada citra asli dan *Multiscale Retinex*. Halaman ini bisa dapat tampil, pertama kita klik *TabSheet* Citra Hasil MSR. Setelah klik *TabSheet* Citra Hasil MSR berikutnya kita langsung saja klik *Button* Lihat Statistik. Maka Secara otomatis sistem akan menampilkan *Form* atau halaman seperti gambar 5.5 di atas.

6. Halaman *Sub-menu* Tentang Pembuat



Gambar 5.6 Halaman *Sub-menu* Tentang Pembuat

Gambar 5.6 Halaman *Sub-menu* Tentang pembuat adalah halaman yang menampilkan pembuat dari program.

6. KESIMPULAN

Kesimpulan yang diambil penulis setelah mengimplementasikan metode *Multiscale Retinex* ini adalah sebagai berikut :

1. Sistem ini menunjukkan proses kerja perbaikan citra (*Image enhancement*) dengan metode *Multiscale Retinex*.
2. Kelebihan pada metode *Multiscale Retinex* ini terdapat pada citra yang gelap, karena dapat menambah kecerahan gambar tersebut tanpa merubah warna asli citra dan kekurangan metode ini terdapat pada citra cerah, karena citra yang dihasilkan akan semakin cerah dan semakin pudar.

7. SARAN

Setelah mengimplementasikan metode *Multiscale Retinex* pada skripsi ini penulis dapat memberikan beberapa saran antara lain :

1. Pada sistem yang telah dibuat format *file* citra yang bisa digunakan adalah citra yang berformat *.png*, *.bmp* dan *.jpg*. Untuk pengembangan selanjutnya dapat menggunakan format citra yang lain.
2. Objek penelitian ini hanya sampai pada *Multiscale Retinex*, akan lebih baik jika *Multiscale Retinex* dapat dikembangkan lagi dengan *Multiscale Retinex with Color Restoration (MSRCR)*.

DAFTAR PUSTAKA

Al Fatta, H. (2007). *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern*. Yogyakarta : CV. Andi Offset.

A.S, Rosa dan M. Salahudin. 2011. Modul Pembelajaran Rekayasa Perangkat Lunak. Bandung : Modula

A.S, Rosa dan M. Salahuddin. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung : Informatika Bandung

Gonzalez, Rafael C, Woods, Richard E. (1993). *Digital Image Processing*. Addison-Wesley Publishing Company Inc., USA.

Jobson, D.J., Rahman, Zia-ur., Woodell & Glenn A. 1997. A Multiscale Retinex for Bridging the Gap Between Color Images and the Human Observation of Scenes. Volume 6, No. 7. IEEE Transactions on Image Processing.

Komputer, W. (2010). *Delphi 2010 Programming Konsep dan Implementasi*, Yogyakarta : C.V ANDI OFFSET.

Ladjamudin B, Al-Bahra. 2004. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu

Munir, R. (2004). *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*. Bandung : Informatika.

Murinto., Aribowo, E., & Yustina, E. 2009. Implementasi Metode Retinex untuk Pencerahan Citra. Volume 3. Jurnal Informatika. Universitas Ahmad Dahlan Yogyakarta.

Nugroho, Adi. 2005. "Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek". Bandung: Informatika.

Septian, D. (2010). *Implementasi Kebijakan Peraturan Pemerintah Nomor 98 Tahun 2000 Tentang Pengadaan Pegawai Negeri Sipil Di Badan Cianjur*. Bandung. Universitas Komputer Indonesia.

Siahaan, D. (2012). *Analisis Kebutuhan Dalam Rekayasa Perangkat Lunak* (edisi pertama). Yogyakarta : ANDI.

Supardi, Y. (2013). *Koleksi Program Tugas Akhir dan Skripsi dengan FoxPro 9*. Jakarta : PT Elex Media Komputindo.

Sutoyo, T.,Et. Al. (2009). *Teori Pengolahan Citra Digital*. Semarang : ANDI Yogyakarta dengan UDINUS Semarang.

Wahono, R. S. Dan Dharwiyanti, S. (2006). *Pengantar Unified Modelling Language*. Jakarta : Ilmu Komputer.