

Pengembangan Aplikasi Konversi Video Digital Greyscale Menjadi Video Digital Berwarna Dengan Menggunakan Metode Global Image Matching

Ketut Widianara¹, Made Windu Antara Kesiman², I Made Gede Sunarya³
Jurusan Pendidikan Teknik Informatika

Universitas Pendidikan Ganesha
Singaraja, Bali

E-mail: widian45@yahoo.com¹, dekndu@yahoo.com², imagededesunarya@gmail.com³

Abstrak— Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah aplikasi yang dapat mengkonversi video digital *greyscale* menjadi video digital berwarna secara cepat, mudah dan hasil pewarnaan yang memuaskan. *Input*-an dari aplikasi ini berupa video *greyscale* dan citra warna acuan, dan *output* dari aplikasi ini adalah sebuah video berwarna dengan nuansa warna yang mengacu pada citra warna acuan yang di-*input*-kan.

Aplikasi ini dikembangkan dengan menggunakan metode *Global Image Matching* yang diimplementasikan pada lingkungan bahasa pemrograman *Java* dengan menggunakan editor *NetBeans 7.2.1*. Pada aplikasi yang dibuat terdapat tiga proses utama yaitu proses ekstrak video, proses pewarnaan, dan proses *render*. Proses ekstraksi merupakan proses memecah video menjadi *frame-frame* dan audio. Proses pewarnaan merupakan proses yang mengimplementasikan metode *Global Image Matching*, dimana *frame-frame* hasil ekstraksi akan dilakukan pewarnaan pada proses ini. Sedangkan proses *rendering* merupakan proses penggabungan *frame-frame* hasil pewarnaan dengan *file* audio hasil ekstraksi menjadi sebuah video berwarna. Berdasarkan hasil pengujian, aplikasi konversi video digital *greyscale* menjadi video digital berwarna ini dapat melakukan proses konversi dengan sangat mudah dan cepat.

Kata kunci— *Citra Warna Acuan, Ekstraksi, Frame, Global Image Matching, Java, NetBeans, Rendering, Video Digital Greyscale.*

Abstract— *This research aims to design and implement an application that could convert a greyscale video into a colored video digital, easily and*

staining results were satisfactory. Input from the applications is greyscale video and color reference image, and the output of this application was a video-colored with shades of color that refers to the reference image input.

This application was developed by using the method of Global Image Matching was implemented in the Java programming language environment with editor used NetBeans 7.2.1. On an application made there were three main processes that extract the video, the process of coloring, and render process. The extraction process was the process of breaking into the video and audio frames. Coloring process was a process that implements the Global Image Matching method, in which frame-frame extraction results would be coloring in this process. While the rendering process was the process of combining frames staining results with the extracted audio files into a video-colored. Based on test results, greyscale digital video conversion application to digital video color conversion process can be done very easily and quickly.

Keywords— *Color Reference Image, Extraction, Frame, Global Image Matching, Java, NetBeans, Rendering, Greyscale Digital Video.*

I. PENDAHULUAN

Video *greyscale* merupakan video yang *frame* hasil ekstraksinya hanya memiliki satu nilai kanal pada setiap *pixel*, dengan kata lain *RED = GREEN = BLUE*. Di Indonesia video *greyscale* tentunya memiliki nilai sejarah tersendiri, ini dapat dilihat dari dokumentasi/ arsip video *greyscale* yang ada kebanyakan menceritakan kehidupan Indonesia pada zaman penjajahan dan menjelang

kemerdekaan. Video seperti ini tentunya penting untuk disimak bagi generasi penerus bangsa, namun gambar video yang *greyscale* menjadikan video ini kurang diminati dan agak sulit untuk diinterpretasikan makna dari video tersebut. Untuk itu diperlukan suatu upaya restorasi video untuk menjadikan video *greyscale* menjadi lebih menarik dan lebih mudah untuk diinterpretasikan makna dari video tersebut. Salah satu yang mungkin dilakukan adalah mengubah/ mewarnai video *greyscale* menjadi video berwarna dengan menggunakan teknik pentransferan warna.

Pentransferan warna dapat dilakukan dengan cara manual, yaitu dengan menggunakan *software* desain grafis untuk mewarnainya. Hal ini tidak akan masalah jika durasi dari video hanya 5 detik atau 10 detik, seperti yang kita ketahui fps (*frame rate/seconds*) dari suatu video berkisar 25-30 fps [1]. Jadi bisa dibayangkan waktu yang dibutuhkan untuk memproses video yang berdurasi 60 detik saja akan membutuhkan waktu yang lama, mungkin akan berjam-jam. Agar hasil pewarnaan *frame* lebih bagus maka dibutuhkan tenaga ahli yang paham mengenai proses pewarnaan *frame* pada perangkat lunak desain grafis, dan untuk mendatangkan tenaga ahli tentunya membutuhkan biaya yang tidak sedikit.

Berdasarkan permasalahan tersebut perlu dikembangkannya suatu aplikasi yang dapat mengkonversi video *greyscale* menjadi video berwarna, sehingga aplikasi ini nantinya dapat menghemat biaya untuk mewarnai sebuah video *greyscale*, dapat mengefisienkan waktu, dan hasil dari konversi video ini mampu memberikan persepsi yang jelas dan lebih menarik untuk disimak.

II. KAJIAN TEORI

A. Video Digital

Video merupakan gabungan gambar-gambar mati yang dibaca berurutan dalam suatu waktu dengan kecepatan tertentu. Gambar-gambar yang digabung tersebut dinamakan *frame* dan kecepatan pembacaan gambar disebut dengan *frame rate*, dengan satuan fps (*frame per second*). Karena dimainkan dalam kecepatan yang tinggi maka tercipta ilusi gerak yang halus, semakin besar nilai *frame rate* maka akan semakin halus pergerakan yang ditampilkan [2].

Karakteristik dari Video Digital mempengaruhi kualitas terhadap suatu video. Adapun karakteristik yang dimiliki oleh suatu video ada

tiga yaitu resolusi, kedalaman bit, dan laju *frame* [3].

B. Citra Digital

Citra menurut kamus Webster, adalah suatu representasi, kemiripan, atau imitasi dari suatu objek atau benda. Citra merupakan istilah lain untuk gambar sebagai salah satu komponen multimedia yang memegang peranan penting dalam informasi visual. Citra mempunyai karakteristik yang tidak dimiliki oleh data teks yaitu citra kaya dalam informasi.

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut, jika nilai x, y, dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital [4].

Citra terbagi menjadi 2 yaitu :

- Citra diam, adalah citra tunggal yang tidak bergerak
- Citra bergerak (video) adalah rangkaian citra diam yang ditampilkan secara beruntun sehingga memberi kesan pada mata kita sebagai gambar yang bergerak.

C. Jenis Citra

Berdasarkan nilai *pixel* dalam , jenis citra dapat dibedakan sebagai berikut [4].

- Citra biner, merupakan citra dengan nilai *pixel* antara intensitas maksimum dan intensitas minimum. Citra biner juga disebut sebagai citra B&W (black and white) atau citra monokrom.
- Citra Greyscale, merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap *pixel*, dengan kata lain nilai bagian RED = GREEN = BLUE. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. warna yang dimiliki adalah warna dari hitam, keabuan, keabuan dan putih.
- Citra warna, adalah citra dengan lebih dari 1 nilai pada setiap *pixel*-nya. Citra warna mengkombinasikan beberapa derajat intensitas warna untuk menghasilkan berbagai warna. Warna merah, hijau, dan biru (Red, Green, Blue atau RGB) adalah warna primer untuk mencampur cahaya yang disebut dengan

additive primary color yang berbeda dengan subtractive primary color untuk mencampur cat (Cyan, Magenta, Yellow).

III. GLOBAL IMAGE MATCHING

Menurut Simon Premoze (dalam [4] : 5), pentransferan warna secara keseluruhan dari gambar warna ke gambar *greyscale* dapat dilakukan dengan mencocokkan tingkat kecerahan dan informasi tekstur diantara kedua gambar Teknik yang digunakan dalam pentransferan warna dari citra warna ke dalam citra *greyscale* adalah teknik "Global Image Matching", dimana dalam metode ini campur tangan manusia sangat sedikit, selain itu dalam proses pentransferan warna dengan menggunakan teknik *Global Image Matching* dibutuhkan beberapa tahapan yang harus dilakukan. Berikut adalah penjelasan dari tahapan-tahapan tersebut [5].

a) Merubah Ruang Warna RGB ke dalam Ruang Warna $l\alpha\beta$

Ruang warna $l\alpha\beta$ merupakan ruang warna yang dikembangkan oleh Ruderman. Ruang warna ini memiliki 3 buah *channel* dimana l merupakan *luminance*, komponen α merupakan komponen penyusun warna yaitu dari kuning sampai biru dan komponen β merupakan komponen penyusun warna yaitu dari merah sampai dengan hijau. Cara untuk mengubah citra dengan ruang warna RGB menjadi citra dengan ruang warna $l\alpha\beta$, dapat menggunakan matriks konversi sebagai berikut.

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \dots(1)$$

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \dots(2)$$

b) Penyesuaian Tingkat Kecerahan Warna

Citra warna dan citra *greyscale* memiliki perbedaan tingkat kecerahan yang cukup tinggi, hal ini dapat berpengaruh di dalam proses pewarnaan sehingga diperlukan adanya suatu proses *luminance remapping*, yaitu suatu proses menggeser dan menskalakan tingkat kecerahan citra warna agar sesuai dengan tingkat kecerahan citra *greyscale* [3]. Terdapat beberapa tahapan dalam melaksanakan proses penyesuaian tingkat

kecerahan citra, adapun tahapan tersebut adalah sebagai berikut.

1. Membuat histogram tingkat kecerahan gambar warna
2. Membuat histogram tingkat kecerahan gambar *greyscale*
3. Melakukan proses *luminance remapping*

c) Menghitung Statistik Pixel

Proses ini bertujuan untuk mencari kecocokan antara citra warna dengan citra *greyscale*. Terdapat beberapa tahapan yang harus dilakukan untuk mencari kecocokan antara citra warna dengan citra *greyscale* yaitu:

1. Pemilihan sampel *pixel* warna pada citra warna

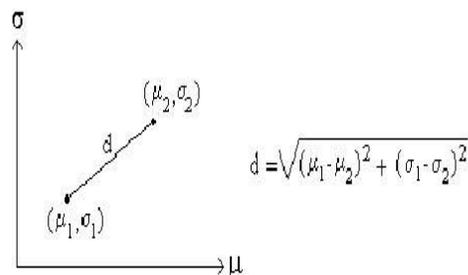
Sampel warna merupakan contoh *pixel* warna yang nantinya akan diambil nilai kromatik warnanya yang akan digunakan sebagai pigmen warna *pixel* pada citra *greyscale*.

2. Perhitungan statistik

Perhitungan statistik yang dimaksud disini adalah perhitungan *mean* (rata-rata) dan perhitungan standar deviasi (simpangan baku). Perhitungan statistik dilakukan pada kedua citra inputan. Pada citra warna, perhitungan dilakukan pada masing-masing sampel sedangkan pada citra *greyscale* dilakukan pada keseluruhan masing-masing *pixel*. Dalam perhitungan nilai rata-rata dan simpangan baku setiap *pixel* dilakukan dengan menggunakan matriks 5 X 5.

d) Mencocokkan Pixel

Setiap *pixel* pada citra *greyscale* dicocokkan dengan *pixel* sampel yang didapat dari citra warna. Proses pencocokan dilakukan dengan menghitung bobot rata-rata dan simpangan baku di sekelilingnya. Ilustrasi pencocokan *pixel* dapat dilihat seperti Gambar 1 berikut.



Gambar 1. Ilustrasi Pencocokan Pixel [5]

e) Pewarnaan Citra Greyscale

Pewarnaan citra *greyscale* dilakukan dengan mengubah ruang warna *lab* ke dalam ruang warna RGB dibutuhkan matriks konversi berikut

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{6}}{6} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \quad (4)$$

III METODOLOGI

A. Analisis Masalah dan Solusi

Dalam mentransfer warna secara manual, waktu yang dibutuhkan sangat lama serta membutuhkan biaya yang tidak sedikit agar video *greyscale* menjadi video berwarna. Selain itu implementasi metode *Global Image Matching* pada aplikasi yang sudah ada waktu prosesnya agak lama yaitu berkisar 20 ó 60 detik dan citra warna acuan yang digunakan hanya berupa citra *bitmap* [6].

Berdasarkan analisis masalah diatas maka solusi yang dapat diusulkan adalah sebuah perangkat lunak berupa aplikasi pewarnaan video *greyscale* menjadi video berwarna. Adapun solusi yang ditawarkan pada aplikasi pewarnaan video adalah :

1. Proses pewarnaan video digital *greyscale* menjadi video digital berwarna dapat dilakukan oleh sistem secara otomatis dengan meminimalisir keterlibatan pengguna sehingga waktu yang dibutuhkan akan lebih cepat jika dibandingkan dengan cara manual.
2. Proses pewarnaan yang dilakukan oleh sistem akan mempermudah orang tidak paham mengenai desain grafis dalam mewarnai video digital *greyscale* menjadi video digital berwarna, sehingga dapat meminimalisir biaya yang dikeluarkan.
3. Format citra acuan pada aplikasi pewarnaan video digital *greyscale* yang menggunakan metode *Global Image Matching* dapat berupa citra berformat *bitmap* (*.bmp) , PNG (*.png) dan JPEG/JPG (*.jpg/ *.jpeg).

4. Waktu yang dibutuhkan untuk mewarnai satu *frame* dari hasil ekstraksi suatu video bisa lebih cepat dari implementasi metode *global image matching* pada aplikasi pewarnaan citra *greyscale* yang sudah ada.

B. Analisis Perangkat Lunak

Pada sub bab analisis perangkat lunak ini, akan dipaparkan tahapan dari *Software/ System Development Life Cycle* (SDLC) yang meliputi hal-hal berikut.

1. Kebutuhan Perangkat Lunak

Berdasarkan analisis terhadap pengembangan aplikasi konversi video digital *greyscale* menjadi video digital berwarna, maka terdapat beberapa proses yang dapat implementasikan, yaitu :

- a. Melakukan ekstraksi dari video *greyscale* menjadi *frame-frame* dengan format *bitmap* dan sebuah *file* audio dengan format mp3.
- b. Melakukan operasi perubahan ruang warna dari ruang warna RGB ke ruang warna *lab* dan sebaliknya dari ruang warna *lab* ke ruang warna RGB.
- c. Melakukan penyesuaian tingkat kecerahan *pixel* dengan cara melakukan perataan histogram pada *frame greyscale* dan citra warna acuan dan melakukan proses spesifikasi histogram pada *frame* dan citra acuan hasil perataan histogram.
- d. Melakukan pengambilan *pixel* sampel dari citra warna acuan dengan jumlah yang sesuai dengan yang di-*input*-kan *user*.
- e. Melakukan penghitungan statistika *pixel* yang meliputi penghitungan rata-rata dan penghitungan standar deviasi *pixel*.
- f. Melakukan pencocokan nilai *pixel*, pencocokan nilai *pixel* didasarkan atas nilai perhitungan pada operasi statistik.
- g. Melakukan proses pewarnaan *frame greyscale* menjadi *frame* berwarna.
- h. Melakukan penggabungan antara *frame-frame* yang sudah diwarnai dengan *file* audio menjadi sebuah video berwarna.

2. Tujuan Pengembangan Perangkat Lunak

Adapun tujuan dari pengembangan perangkat lunak aplikasi konversi video digital *greyscale* menjadi video digital berwarna adalah sebagai berikut.

- a. Dapat Melakukan ekstraksi dari video *greyscale* menjadi *frame-frame* dengan

- format *bitmap* dan sebuah *file* audio dengan format mp3.
- b. Dapat Melakukan operasi perubahan ruang warna dari ruang warna RGB ke ruang warna *lab* dan sebaliknya dari ruang warna *lab* ke ruang warna RGB.
- c. Dapat Melakukan penyesuaian tingkat kecerahan *pixel* dengan cara melakukan perataan histogram pada *frame greyscale* dan citra acuan dan melakukan proses spesifikasi histogram pada *frame* dan citra acuan hasil perataan histogram.
- d. Dapat melakukan pengambilan *pixel* sampel dari citra warna acuan dengan jumlah yang sesuai dengan yang di-*input*-kan *user*.
- e. Dapat melakukan penghitungan statistika *pixel* yang meliputi penghitungan rata-rata dan penghitungan standar deviasi *pixel*.
- f. Dapat melakukan pencocokan nilai *pixel*, pencocokan nilai *pixel* didasarkan atas nilai perhitungan pada operasi statistic.
- g. Dapat Melakukan proses pewarnaan *frame-frame greyscale* menjadi *frame-frame* berwarna.
- h. Dapat Melakukan penggabungan antara *frame-frame* yang sudah diwarnai dengan *file* audio menjadi sebuah video berwarna.

3. Masukan dan Keluaran Perangkat Lunak

Adapun masukan dan keluaran dari perangkat lunak aplikasi konversi video digital *greyscale* menjadi video digital berwarna, adalah sebagai berikut.

a. Masukan Perangkat Lunak

- Video *greyscale* dengan format AVI , FLV, WEBM, MKV, MPEG, MP4, dan WMV.
- Parameter jumlah *frame* per-detik (fps) untuk proses ekstrasi video.
- Citra acuan pewarnaan dengan format *bitmap* (*.*bmp*) 24 bit.
- Parameter jumlah *pixel* sampel yang akan digunakan sebagai acuan pewarnaan.
- Format *file* video *output* hasil pewarnaan video *greyscale*.

b. Keluaran Perangkat Lunak

- *Frame* hasil ekstrasi video *greyscale* dengan format *bitmap* (*.*bmp*).

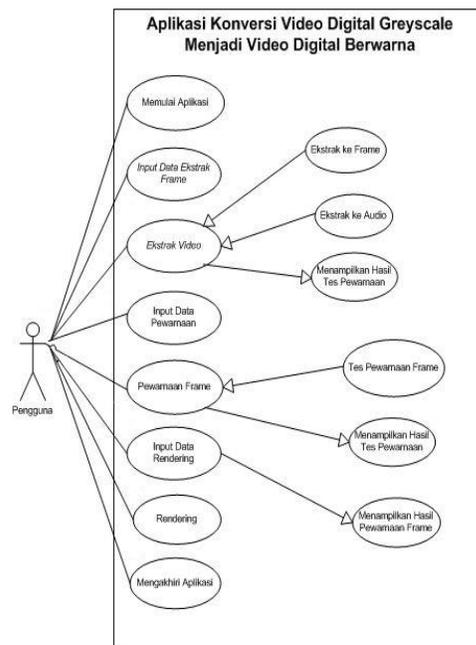
- Audio hasil ekstrasi video dengan format mp3.
- *Frame* hasil pewarnaan dengan format *bitmap* (*.*bmp*).
- Video digital berwarna dengan format yang sesuai dengan apa yang telah ditentukan oleh *user*.

4. Model Fungsional Perangkat Lunak

Pada model fungsional perangkat lunak menjelaskan gambaran umum terhadap proses yang terjadi dalam perangkat lunak. Model fungsional dapat memberikan gambaran terhadap proses yang terjadi antara perangkat lunak dengan pengguna luar (*user*). Dalam aplikasi yang akan dikembangkan model fungsionalnya digambarkan dalam bentuk *use case diagram* dan *activity diagram*.

a. Use Case Diagram

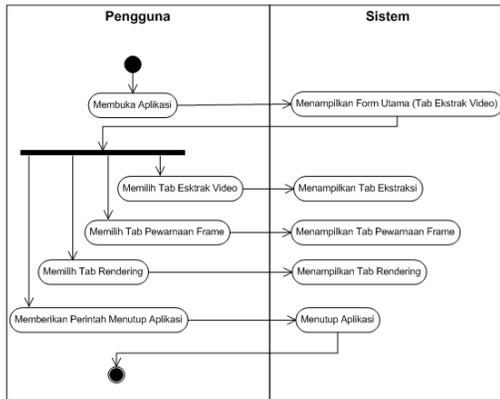
Use case diagram menggambarkan fungsionalitas yang diharapkan dari aplikasi konversi video *greyscale* ke berwarna , kebutuhan sistem dari sudut pandang pengguna serta hubungan antara *actor* dan *use case*. *Use case diagram* aplikasi konversi video *greyscale* ke berwarna ditunjukkan oleh Gambar 2.



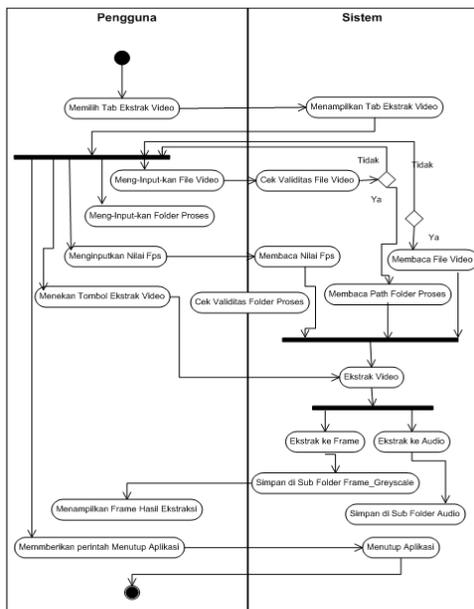
Gambar 2. Use Case Diagram Aplikasi Konversi Video Digital *Greyscale* Menjadi Video Digital Berwarna

b. Activity Diagram

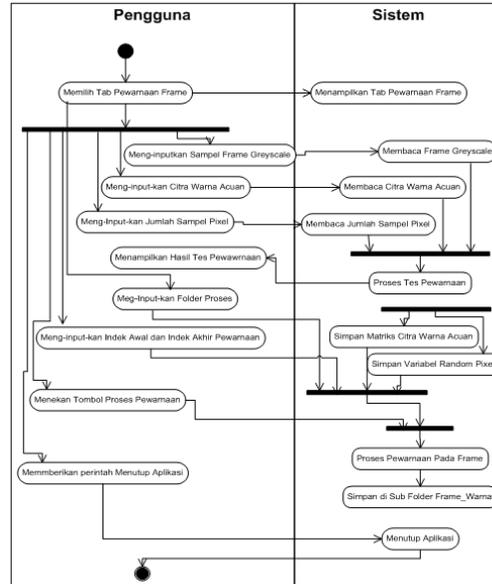
Activity diagram menggambarkan proses, urutan aktivitas dalam proses-proses sistem tersebut, bagaimana setiap proses dimulai dan bagaimana suatu proses akan berakhir. *Activity diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram*. *Activity diagram* aplikasi konversi video digital *greyscale* menjadi video digital berwarna ditunjukkan oleh Gambar 3, dan Gambar 4, Gambar 5 dan Gambar 6 merupakan sub aktivitas dari *activity diagram* aplikasi konversi video digital *greyscale* menjadi video digital berwarna.



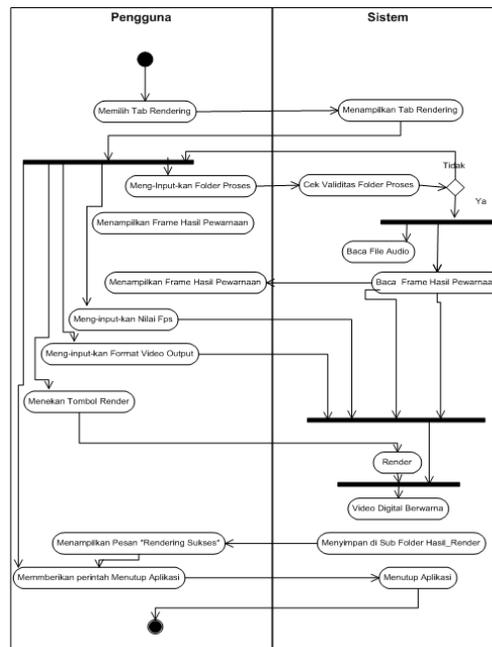
Gambar 3. *Activity Diagram* Aplikasi Konversi Video Digital *Greyscale* Menjadi Video Digital Berwarna



Gambar 4. *Activity Diagram* Sub Aktivitas Memilih Tab Ekstrak Video



Gambar 5. *Activity Diagram* Sub Aktivitas Memilih Tab Pewarnaan Frame



Gambar 6. *Activity Diagram* Sub Aktivitas Memilih Tab Rendering

5. Perancangan Perangkat Lunak

Terdapat beberapa tahapan dalam perancangan perangkat lunak aplikasi konversi video digital

greyscale menjadi video digital berwarna. Adapun tahapan-tahapan tersebut yaitu:

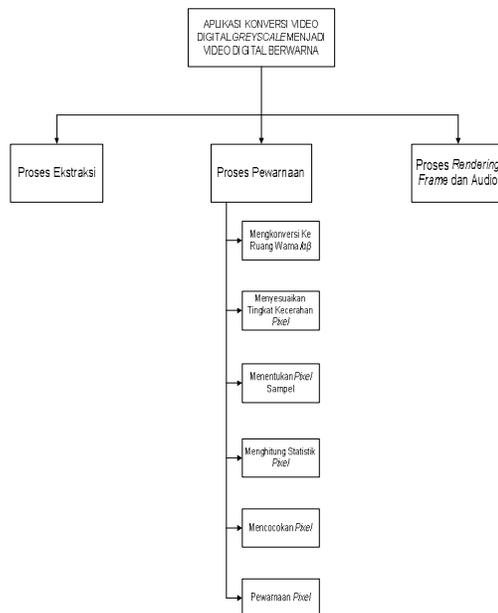
a. Batasan Perancangan Perangkat Lunak

Masalah yang akan dibahas pada skripsi ini dibatasi pada hal-hal sebagai berikut.

1. Video digital greyscale yang dimaksud adalah video yang apabila diekstrak akan terdiri dari *frame-frame* yang memiliki warna skala keabuan (*greyscale*) berekstensi *bitmap (*.bmp)* 24 bit.
2. Format video masukan berupa AVI, FLV, WEBM, MKV, MPEG, MP4, dan WMV.
3. Citra warna/ acuan menggunakan format RGB - 24 bit dan berekstensi BMP, PNG dan JPEG/ JPG

b. Perancangan Arsitektur Perangkat Lunak

Pada tahapan ini menggambarkan bagian-bagian modul, struktur ketergantungan antar modul, dan hubungan antar modul dari perangkat lunak yang dibangun. Pada bagian ini terdapat *structure chart* sebagai kendali fungsional yang digambarkan seperti Gambar 7 untuk perangkat lunak aplikasi konversi video digital greyscale menjadi video digital berwarna.



Gambar 7. Structure Chart Aplikasi Konversi Video Digital Greyscale Menjadi Video Digital Berwarna

c. Perancangan struktur data perangkat Lunak

Struktur data utama yang digunakan dalam perangkat lunak aplikasi konversi video digital

greyscale menjadi video digital berwarna ini adalah struktur data matriks *array of array of short*, matriks *array of array of integer*, matriks *array of array of double*, larik *array of integer*, larik *array of double*, *string*, *bufferedImage* dan *integer*. Sedangkan struktur data bentukan meliputi tipe dan objek. Tipe data objek pada perangkat lunak aplikasi konversi video digital greyscale menjadi video digital berwarna dapat berupa tampilan GUI (*Graphic Unit Interface*). Objek untuk tampilan GUI yang digunakan yaitu *Jframe*, *Jinternalframe*, *JtabbedPane*, *Jpopupmenu*, *Jpanel*, *Jlabel*, *Jbutton*, *Jtextfield*, *Jcheckbox*, *Jprogressbar*, *Jradiobutton*, *Jfilechooser* dan *Jspinner*, *Jscrollpane*

d. Perancangan Antar Muka Perangkat Lunak

Perancangan antar muka perangkat lunak bertujuan untuk menggambarkan bentuk rancangan atarmuka perangkat lunak yang meliputi rancangan layar utama, rancangan layar ekstrak video, rancangan layar pewarnaan frame dan rancangan layar rendering dan rancangan layar hasil pewarnaan.

IV. PEMBAHASAN

A. Implementasi Arsitektur Perangkat Lunak konversi video digital greyscale menjadi video digital berwarna.

Data Flow Diagram (DFD) dan Structure Chart perangkat lunak aplikasi pewarnaan citra greyscale diimplementasikan menggunakan bahasa pemrograman Java dengan editor *Netbeans 7.2.1*. Tabel 1 berikut ini merupakan tabel pemetaan *class* implementasi.

Tabel 1. Tabel Pemetaan Arsitektur Terhadap Class Implementasi

Bagian Perancangan Arsitektur	Unit Implementasi	Penjelasan Unit Implementasi
Proses Ekstraksi	<i>Class Utama.java</i>	Pada <i>class</i> ini terdapat <i>method</i> untuk mengekstraksi video <i>greyscale</i> yang di- <i>input</i> -kan oleh <i>user</i> menjadi <i>frame-frame</i> dengan format <i>bitmap</i> dan menjadi sebuah <i>file</i> audio dengan format <i>mp3</i> .

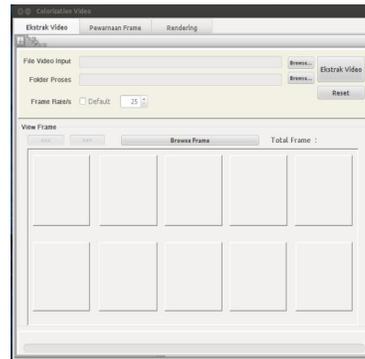
Bagian Perancangan Arsitektur	Unit Implementasi	Penjelasan Unit Implementasi
Proses Pewarnaan	<i>ClassUtama.java</i>	Pada <i>class</i> ini terdapat beberapa <i>method</i> untuk meminta <i>input</i> -an dari <i>user</i> yang terkait dengan proses pewarnaan
	<i>ImageProcessing.java</i>	Pada <i>class</i> ini diimplementasikan <i>method-method</i> untuk melakukan proses pewarnaan yang sesuai dengan metode <i>Global Image Matching</i> .
	<i>ImageMatrix.java</i>	Pada <i>class</i> ini diimplementasikan <i>method-method</i> untuk melakukan pembacaan file citra, untuk mendapatkan nilai R G B dari suatu citra, untuk mengubah nilai-nilai R G B menjadi suatu file citra.
	<i>ImagePreview.java</i>	<i>Class</i> ini digunakan untuk menampilkan hasil dari testing pewarnaan <i>frame greyscale</i> .
Proses Rendering	<i>ClassUtama.java</i>	Pada <i>class ClassUtama.java</i> terdapat <i>method</i> untuk menggabungkan <i>frame-frame</i> hasil pewarnaan dan <i>file</i> audio menjadi sebuah video.

B. Implementasi Layar Antarmuka Perangkat Lunak

Rancangan layar antarmuka perangkat lunak aplikasi konversi video digital *greyscale* menjadi video digital berwarna diimplementasikan menggunakan *class-class* yang terdapat pada *frame* di editor *Netbeans 7.2.1 IDE*. Gambar 8

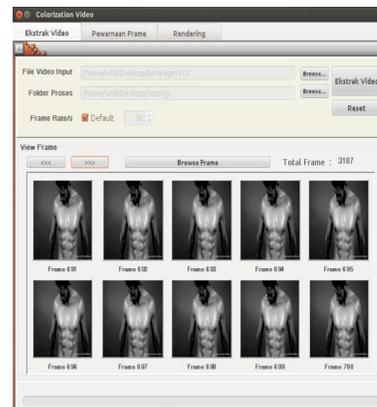
sampai Gambar 12 merupakan implementasi dari rancangan antarmuka yang direncanakan.

1. Layar Utama Perangkat Lunak



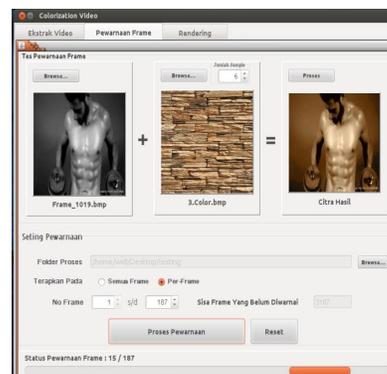
Gambar 8. Layar Utama Perangkat Utama

2. Layar Tab Ekstrak Video



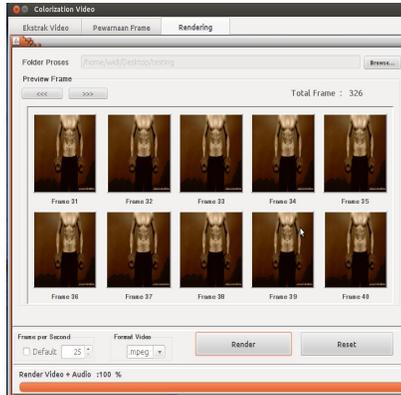
Gambar 9. Layar Tab Ekstrak Video

3. Layar Tab Pewarnaan Frame



Gambar 10. Layar Tab Pewarnaan Frame

4. Layar Tab Rendering



Gambar 11. Layar Tab Rendering

5. Layar Hasil Pewarnaan



Gambar 12. Layar Hasil Pewarnaan

C. Pengujian Perangkat Lunak

Secara umum hasil pengujian fungsional menunjukkan bahwa sistem sudah dapat menangani data masukan yang tidak valid dan valid dan dapat melakukan proses pewarnaan video greyscale menjadi video berwarna dengan baik. Berikut akan ditampilkan hasil perbandingan *frame-frame* hasil pengujian aplikasi konversi video digital greyscale menjadi video digital berwarna.

1. Video "Male Fitness Model Slow Motion Shower 3"

File video "Male Fitness Model Slow Motion Shower 3" dapat anda lihat di situs Youtube dengan alamat <https://www.youtube.com/watch?v=7oMBYN8Q7bg>. Dengan nilai fps sama dengan 30, *frame* yang didapat setelah dilakukan ekstraksi yaitu 2679 *frame* dengan resolusi *frame* 426 x 240 *pixel*.

Tabel hasil perbandingan *frame* sebelum dan sesudah diwarnai ditunjukkan pada Tabel 2 berikut.

Tabel 2. Hasil Perbandingan *Frame* Sebelum Diwarnai dan Setelah Diwarnai pada Video "Male Fitness Model Slow Motion Shower 3"

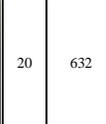
Hasil Pengujian			
Citra Warna Acuan			
Pixel Sampel Jumlah = 30			
No Frame	Frame Greyscale	Frame Output	Waktu
1			631
100			631
500			631
900			631
1200			631

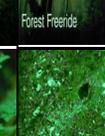
1500		631
1700		631
1900		631
2300		631

2. Video "Forest Freeride"

File video "Forest Freeride" merupakan sebuah video yang menceritakan pengendara sepeda yang bersepeda di hutan, video ini berdurasi 2 menit 7 detik. Video "Forest Freeride" dapat anda saksikan di situs Youtube dengan alamat http://www.youtube.com/watch?v=3u_Vi4Eb_nQ. Setelah dilakukan ekstraksi dengan fps sama dengan 30, didapatkan frame sebanyak 3812 frame dengan resolusi 426 x 240 pixel. Pada proses pewarnaan video "Forest Freeride" ini menggunakan beberapa citra warna acuan. Pemakaian citra warna acuan yang lebih dari satu disebabkan karena pada video ini terdiri dari latar tempat yang berbeda-beda. Tabel hasil perbandingan frame sebelum dan sesudah diwarnai ditunjukkan pada Tabel 3 berikut.

Tabel 3. Hasil Perbandingan Frame Sebelum Diwarnai dan Setelah Diwarnai pada Video "Forest Freeride"

No	Input		Frame Output	S	T
	Frame Greyscale	Citra Warna Acuan			
104				20	632

No	Input		Frame Output	S	T
	Frame Greyscale	Citra Warna Acuan			
140					
200					
220				50	937
300					
480				30	676
590					
1110				50	980
900				30	679
415				30	780

D. Evaluasi Hasil Pengujian

Berdasarkan hasil pengujian di atas, diketahui bahwa aplikasi konversi video digital greyscale menjadi video digital berwarna sudah mampu

melakukan proses pewarnaan pada video *greyscale*. Hal ini dibuktikan dengan terjadinya perubahan warna pada frame *greyscale* yang awal mulanya hanya memiliki nuansa warna keabuan saja tapi setelah dilakukan proses pewarnaan maka frame yang dihasilkan memiliki warna yang beragam (*true color*). Keberhasilan sistem sangat ditentukan oleh citra warna acuan yang di-input-kan. Citra warna acuan yang memiliki tingkat kecerahan yang mendekati tingkat kecerahan frame *greyscale* akan menyebabkan hasil pewarnaan lebih baik jika dibandingkan dengan citra warna acuan yang memiliki tingkat kecerahan yang sangat berbeda dengan frame *greyscale*.

V. PENUTUP

A. Simpulan

Berdasarkan penelitian yang telah dilakukan diperoleh simpulan sebagai berikut.

1. Aplikasi konversi video digital *greyscale* menjadi video digital berwarna diimplementasikan dengan menerapkan metode *Global Image Matching*.
2. Pengembangan aplikasi dilakukan dengan menggunakan bahasa pemrograman *Java* dengan *java version 1.7.0_10* dengan editor *Netbeans 7.2.1 IDE*.
3. Implementasi metode *Global Image Matching* pada aplikasi konversi video digital *greyscale* menjadi video digital berwarna menghasilkan suatu aplikasi yang mampu melakukan pewarnaan video *greyscale* dengan cepat dan mudah sehingga dapat dilakukan oleh semua orang.
4. Kecepatan dalam pewarnaan video *greyscale* sangat ditentukan oleh jumlah pixel sampel yang di-input-kan.
5. Hasil pewarnaan video yang diperoleh sangat ditentukan oleh citra warna acuan yang di-input-kan.

Untuk pengembangan lebih lanjut dari aplikasi konversi video digital *greyscale* menjadi video digital berwarna, adapun saran yang dapat diberikan adalah sebagai berikut.

- a. Adanya pengembangan algoritma penyesuaian tingkat kecerahan yang lebih spesifik sehingga menghasilkan tingkat kecerahan frame *greyscale* dan citra warna acuan akan memiliki lebih baik dan nantinya akan

berimbang pada hasil pewarnaan yang lebih bagus.

- b. Adanya pengembangan algoritma yang dapat melakukan restorasi pada frame hasil pewarnaan, sehingga dapat meminimalisir bercak-bercak yang dihasilkan dari proses pewarnaan frame.
- c. Adanya pengembangan algoritma yang dapat mendeteksi bentuk, sehingga kita dapat menentukan dengan lebih tepat objek yang akan diwarnai dari suatu frame hasil ekstraksi.

REFERENSI

- [1] Pusat Pengembangan Bahan Ajar Universitas Mercu Buana . 2011. *Video & Televisi*. Tersedia pada <http://kk.mercubuana.ac.id/files/92052-3-574637347358.doc>.
- [2] Budiman, Asep. 2009. Aplikasi *Steganography* pada Video dengan Metode *Least Significant Bit (LSB)*. *Skripsi* (tidak diterbitkan). Teknik Informatika S1, Universitas Indonesia. Tersedia pada <http://elib.unikom.ac.id/gdl.php?mod=browse&op=read&id=ibptunikompp-gdl-asepbudima-18351>.
- [3] Andikaputra, Gusti Bagus Agung Pradnya. 2011. Karakteristik Video Digital. Tersedia pada http://digilib.itelkom.ac.id/index.php?option=com_content&view=article&id=866:karakteristik-video-digital&catid=21:itp-informatika-teori-dan-pemrograman&Itemid=14 (diakses pada tanggal 20 Desember 2012).
- [4] Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta: Andi.
- [5] Dewi, Lilyana. 2003. Perencanaan Dan Pembuatan Aplikasi Untuk Transfer Warna Ke Gambar *Greyscale* Dengan Metode *Global Image Matching*. *Skripsi* (tidak diterbitkan). Teknik Informatika S1, Universitas Kristen Petra. Tersedia pada http://dewey.petra.ac.id/jiunkpe_dg_1519.html.
- [6] Putra, I Made Agus Wirahadi. 2012. Pengembangan Aplikasi Citra Digital Untuk Mengubah Citra *Greyscale* Menjadi Citra Berwarna. *Skripsi* (tidak diterbitkan). Jurusan Pendidikan Teknik Informatika, Undiksha Singaraja.