

**ANALISIS DAN IMPLEMENTASI PENGGUNAAN METODE  
INTERPOLASI NEAREST NEIGHBOR DAN INTERPOLASI  
BILINEAR (STUDI KASUS : APLIKASI IMAGE VIEWER  
BERBASIS ANDROID)**

**Naskah Publikasi**



diajukan oleh

**Dikky Faiz Maulana**

**09.11.3374**

kepada

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER  
AMIKOM YOGYAKARTA  
YOGYAKARTA  
2013**

**NASKAH PUBLIKASI**

**ANALISIS DAN IMPLEMENTASI PENGGUNAAN METODE INTERPOLASI NEAREST  
NEIGHBOR DAN INTERPOLASI BILINEAR (STUDI KASUS : APLIKASI IMAGE  
VIEWER BERBASIS ANDROID)**

disusun oleh

**Dikky Faiz Maulana**

**09.11.3374**

**Dosen Pembimbing**

**Andi Sunyoto, M.Kom**

**NIK : 190302052**

Tanggal, 10 Juni 2013

**Ketua Jurusan**

**Teknik Informatika**



**Sudarmawan, M.T**

**NIK : 190302035**

**ANALYSIS AND IMPLEMENTATION USING METHOD NEAREST NEIGHBOR  
INTERPOLATION AND BILINEAR INTERPOLATION (CASE STUDY :  
IMAGE VIEWER BASED ON ANDROID APPLICATION)**

**ANALISIS DAN IMPLEMENTASI PENGGUNAAN METODE INTERPOLASI  
NEAREST NEIGHBOR DAN INTERPOLASI BILINEAR  
(STUDI KASUS : APLIKASI IMAGE VIEWER  
BERBASIS ANDROID)**

Dikky Faiz Maulana  
Andi Sunyoto  
Jurusan Teknik Informatika  
STMIK AMIKOM YOGYAKARTA

**ABSTRACT**

*The title of the undergraduate thesis which I will write is "Analysis and Implementation Using Method Nearest Neighbor Interpolation and Bilinear Interpolation (Case Study: Image Viewer Based on Android Application)". This undergraduate thesis discusses the analysis and implement the use of interpolation method on Image Viewer application. Image Viewer is an application that is used to display the digital image. Sometimes when zoom in using Image Viewer it will decrease the quality of the digital image. To minimize the loss of quality in the digital image, it is necessary to interpolate.*

*The interpolation to be analyzed and implemented on Image Viewer is Nearest Neighbor Interpolation and Bilinear Interpolation. Both of this type interpolation will be performed comparisons of the result of the output image and the result of the execution speed.*

*The Image Viewer application is built using Eclipse 3.7.2 as an IDE for Java programming language. Eclipse is free, open source, multi-language and cross-platform IDE which is used in the design of Android-based applications.*

**Keywords** : *Image Processing, Interpolation, Nearest Neighbor Interpolation, Bilinear Interpolation, Image Viewer, Java, Android.*

## 1. PENDAHULUAN

Di era modern sekarang ini, diikuti dengan perkembangan akan teknologi dan informasi. Perkembangan teknologi kamera yang semula berupa kamera Daguerreotype berbentuk balok kayu kini telah berevolusi menjadi kamera digital. Kamera digital mampu menghasilkan foto dengan format citra digital. Sekarang ini, kita juga mampu membuat gambar, simbol, huruf atau angka dengan format digital disamping menggunakan kamera digital. Citra digital memiliki kelebihan pada kecepatan dan kemudahan dalam memprosesnya. Salah satu kemudahan dari citra digital adalah kita dapat membuat atau mengaksesnya kapan saja dan dimana saja selama kita memiliki peralatan dan aplikasi yang dibutuhkan. Aplikasi yang digunakan untuk menampilkan citra digital biasa disebut sebagai *Image Viewer*. Terkadang saat melakukan *zoom in* menggunakan *Image Viewer* terjadi penurunan kualitas pada citra digital. Penurunan kualitas gambar pada citra digital tersebut disebabkan karena ketidakmampuan citra digital dalam memproduksi *pixel* pada titik dimana *zoom in* dilakukan. Untuk meminimalisasikan penurunan kualitas gambar tersebut, maka diperlukan Interpolasi. Interpolasi adalah metode untuk membangun titik data baru pada rentang satu stuktur dari titik data yang telah diketahui. Aplikasi *Image Viewer* banyak dijumpai di internet, baik aplikasi tersebut bersifat gratis maupun komersial. Selain itu, aplikasi *Image Viewer* tersedia dalam perangkat elektronik seperti *Desktop PC* maupun *Mobile Phone*.

Perkembangan *Mobile Phone* yang sebelumnya hanyalah alat untuk mempermudah dalam berkomunikasi kini telah berevolusi menjadi *Smartphone* dimana *Mobile Phone* mampu melakukan operasi yang biasa dilakukan dalam *Desktop PC*. Pesatnya perkembangan dari *Smartphone* ini berdampak mulai banyak digunakannya sebagai perangkat elektronik untuk menampilkan bahkan memproses atau memproduksi citra digital.

*Smartphone* merupakan *Mobile Phone* yang memiliki spesifikasi hampir sama dengan sebuah *Personal Computer* dimana perangkat ini memerlukan sebuah sistem operasi untuk mengoperasikannya. Sistem operasi mobile yang populer sekarang ini adalah **Google** Android dan **Apple** iOS. Kedua produk tersebut terus bersaing melakukan inovasi dan berbagai strategi untuk menguasai pasar *smartphone platform*. Berdasarkan laporan survei penggunaan *smartphone* di Negara **U.S.** (*United State of America*) yang bersumber dari **comScore**, memperlihatkan bahwa **Google** Android menguasai pasar *smartphone platform* pada Februari 2012 sebesar 50.1% mengalahkan pesaingannya **Apple** iOS yang saat itu sebesar 30.2% dan tetap menguasai pasar pada akhir Mei 2012 sebesar 50.9%.

## 2. LANDASAN TEORI

### 2.1 Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data yang dapat bersifat optik berupa foto, bersifat analog berupa sinyal – sinyal video seperti gambar pada layar televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan.<sup>[1]</sup>

Citra secara harfiah adalah gambar pada bidang dua dimensi. Dipandang dari sudut matematis, citra merupakan fungsi menerus (*Continue*) dari intensitas cahaya pada bidang dua dimensi. Sumber cahaya menerangi objek, kemudian memantulkan sebagian dari berkas cahaya tersebut. Pantulan cahaya tersebut diterima oleh alat – alat optik misalnya mata pada manusia, sehingga bayangan objek tersebut yang bernama citra terekam.<sup>[2]</sup>

Citra terdiri dari 2 jenis, yaitu *Continue Image* dan *Discreet Image*.<sup>[3]</sup>

1. *Continue Image*, diperoleh dari sistem optik yang menerima sinyal analog, seperti mata manusia dan kamera analog.
2. *Discreet Image*, dihasilkan melalui proses digitalisasi terhadap *Continue Image*.

#### 2.1.1 Definisi Citra Digital

Citra digital adalah suatu fungsi intensitas cahaya (*Brightness*) dua dimensi  $f(x,y)$  dimana  $(x)$  dan  $(y)$  merupakan koordinat spasial, dan  $f$  pada suatu titik  $(x, y)$  merupakan intensitas atau level keabuan dari citra.<sup>[4]</sup>

Setiap pasangan indek baris  $(y)$  dan kolom  $(x)$  menyatakan suatu titik pada citra. Nilai pada setiap titik tersebut menyatakan intensitas atau nilai kecerahan. Untuk selanjutnya titik – titik tersebut dinamakan sebagai *Pixel (Picture element)*.

Agar dapat diolah dengan komputer digital, maka suatu citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi malar (kontinu) menjadi nilai – nilai diskrit disebut digitalisasi. Citra yang dihasilkan inilah yang disebut citra digital (*Digital Image*). Pada umumnya citra digital berbentuk persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi x lebar (atau lebar x panjang).<sup>[5]</sup>

---

<sup>1</sup> T. Sutoyo, dkk, Teori Pengolahan Citra Digital (Semarang : ANDI, 2009) Hal. 9.

<sup>2</sup> Munir, Rinaldi, Pengolahan Citra Digital Dengan Pendekatan Algoritmik (Bandung : Informatika, 2004) Hal. 2.

<sup>3</sup> Bertalya, 2005, Universitas Gunadarma, <http://www.scribd.com/doc/2815662/Representasi-Citra> diakses 20 November 2010.

<sup>4</sup> Rafael C. Gonzales dan Richard E. Woods, Digital Image Processing 2<sup>nd</sup> Edition (New Jersey : Prentice Hall, 2002) Hal. 1.

<sup>5</sup> Munir, Rinaldi, Pengolahan Citra Digital Dengan Pendekatan Algoritmik (Bandung : Informatika, 2004) Hal. 18.

### 2.1.2 Jenis – Jenis Citra Digital

1. Binary Image (Monochrome Image)
2. Grayscale Image
3. Color Image

### 2.1.3 Elemen – Elemen Citra Digital

1. Kecerahan (Brightness)
2. Kontras (*Contrast*)
3. Kontur (*Contour*)
4. Warna (*Color*)
5. Bentuk (*Shape*)
6. Tekstur (*Texture*)

## 2.2 Interpolasi

Interpolasi adalah metode dasar yang digunakan secara ekstensif dalam melakukan *zooming*, *shrinking*, *rotating*, dan *geometric corrections*. Karena penggunaannya yang cenderung pada bidang *image resizing*, sehingga interpolasi sering disebut juga sebagai *resampling* <sup>[6]</sup>. Pada dasarnya, interpolasi bekerja dengan menggunakan data yang diketahui untuk memperkirakan nilai – nilai pada titik – titik yang tidak diketahui. <sup>[7]</sup>

Interpolasi citra bekerja dalam dua arah, dan mencoba untuk mencapai pendekatan yang terbaik dari sebuah *pixel* yang warna dan intensitasnya didasarkan pada nilai – nilai di sekitar *pixel*. Semakin banyak yang diketahui tentang informasi di sekitar *pixel*, maka semakin baik dalam proses interpolasi. Oleh karena itu, semakin citra diperbesar, maka semakin banyak pula informasi yang harus diketahui <sup>[8]</sup>.

Algoritma interpolasi dapat diklasifikasikan menjadi dua yaitu :

7. Algoritma non-adaptif
8. Algoritma adaptif

### 2.2.1 Algoritma Nearest Neighbor

Algoritma *Nearest Neighbor* adalah algoritma yang implementasinya paling mudah dan paling sederhana dalam operasi penskalaan (*scalling*). *Nearest Neighbor* merupakan algoritma yang sering digunakan dalam melakukan *zooming image* untuk *editing* atau untuk *thumbnails preview*. Algoritma ini digunakan apabila kecepatan dalam mengeksekusi adalah hal yang diutamakan. Algoritma *Nearest Neighbor* memiliki

---

<sup>6</sup> Rafael C. Gonzales dan Richard E. Woods, Digital Image Processing 2<sup>nd</sup> Edition (New Jersey : Prentice Hall, 2002) Hal. 87.

<sup>7</sup> Purnomo, Seno, 2009, <http://senosuke.wordpress.com/2009/12/22/interpolasi/> diakses 08 Oktober 2012.

<sup>8</sup> Ibid

kelemahan pada penggunaannya, yaitu terkadang citra yang dihasilkan menjadi bergerigi (*Jaggy Image*).

Prinsip dasar algoritma *Nearest Neighbor* adalah menunjuk sebuah citra digital asli sebagai dasar untuk membangun citra digital skala baru. Citra digital skala baru yang dihasilkan bisa berukuran lebih kecil, lebih besar atau sama dengan citra asli tergantung pada rasio skala. Ketika melakukan operasi penskalaan, sebenarnya citra digital membuat ruang – ruang kosong (*empty pixels*) pada citra digital asli yang digunakan. Pada teknik *Nearest Neighbor*, titik – titik pada *empty pixels* akan diganti dengan *pixel* warna yang sesuai dengan *pixel* warna pada tetangga terdekatnya, oleh karena itulah disebut *Nearest Neighbor*.

### 2.2.2 Algoritma Bilinear

Algoritma *Bilinear* adalah algoritma yang hampir sama dengan algoritma *Nearest Neighbor*, hanya saja berbeda pada metode interpolasinya. Metode interpolasi pada *Nearest Neighbor* dengan cara melakukan penggandaan *pixel* berdasarkan *pixel* tetangga dekatnya, sedangkan interpolasi pada *Bilinear* melakukan penggandaan *pixel* berdasarkan *pixel* disekelilingnya sehingga algoritma *Bilinear* menghasilkan citra digital yang lebih baik dibandingkan dengan hasil dari algoritma *Nearest Neighbor*.

Sesuai dengan namanya, algoritma ini adalah algoritma yang terbentuk berdasarkan kombinasi dari dua *Linear Interpolation*. *Linear Interpolation* merupakan metode untuk menaksir sembarang titik di antara kedua titik yang telah ditentukan (diketahui). Selain keterkaitan metode *Linear Interpolation* terhadap algoritma *Bilinear*, ada juga hal sederhana yang sama pentingnya dalam algoritma *Bilinear*, hal sederhana tersebut adalah Tekstur. Dalam *Bilinear scaling* yang paling sederhana, dimensi dari tekstur terdiri dari empat buah *pixels* yang dibagi menjadi 2x2. Tekstur tersebut diinisialisasikan terdiri dari A, B, C, dan D.

### 2.3 Image Viewer

Image Viewer adalah program komputer yang dapat menampilkan citra digital dari format citra digital yang telah disimpan. *Image Viewer* mempermudah dalam melakukan proses *viewing*, *cataloging*, dan pengelolaan citra digital. Beberapa *Image Viewer* juga menawarkan fitur untuk melakukan pengeditan citra digital seperti *crop*, *rotate*, *color balance adjustment*, *brightness*, and *contrast controls*. Ada berbagai macam *Image Viewer* seperti **ACDSee**, **Gwenview**, **iPhoto**, **IrfanView**, **Windows Photo Viewer**, dan lain – lain.

Meskipun untuk menampilkan citra digital dapat menggunakan *Image Editor*, tetapi terkadang jika menggunakan *Image Editor* saat menampilkan citra digital terasa cukup lambat dibandingkan dengan *Image Viewer*, selain itu juga dalam *Image Editor* tidak mampu menampilkan citra digital secara *slideshow*.

## 2.4 Android

Android ada sistem operasi yang dikembangkan untuk perangkat *mobile* berbasis Linux. Pada awalnya sistem operasi ini dikembangkan oleh Android Inc, yang kemudian dibeli oleh **Google** pada tahun 2005.<sup>[9]</sup> Android dikembangkan untuk telepon seluler dan juga tablet PC. Fungsinya sama seperti sistem operasi Symbian pada **Nokia**, iOS pada **iPhone**, dan BlackBerry OS pada **BlackBerry**. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti Android tidak terkait pada satu merek *handphone* saja, beberapa vendor terkenal yang sudah memakai Android antara lain **Samsung**, **Sony Ericsson**, **HTC**, **Nexus**, **Motorola**, dan lain – lain.

### 2.4.1 Versi Android

### 2.4.2 Fitur Android

### 2.4.3 Arsitektur Android

## 2.5 Java

Java adalah bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan – rekannya seperti Patrick Naughton, Chris Warth, Ed Frank, dan Mike Sheridan di suatu perusahaan perangkat lunak yang bernama Sun Microsystems, pada tahun 1991. Bahasa pemrograman ini mula – mula diinisialisasi dengan nama “Oak”, namun pada tahun 1995 diganti dengan nama “Java”.

Alasan utama pembentukan bahasa pemrogramna Java adalah untuk membuat aplikasi – aplikasi yang dapat diletakan di berbagai macam perangkat elektronik, seperti *microwave oven* dan *remote control*, sehingga Java harus bersifat *portable* atau yang sering disebut dengan *platform independent* (tidak tergantung pada *platform*)<sup>[10]</sup>. Itulah yang menyebabkan dalam dunia pemrograman Java, dikenal adanya istilah “Write once, run everywhere”, yang berarti kode program hanya ditulis sekali<sup>[11]</sup> namun dapat dijalankan dibawah *platform* manapun, tanpa harus melakukan perubahan kode program.

### 2.5.1 Arsitektur Java

### 2.5.2 Java 1

### 2.5.3 Java 2

## 2.6 Aplikasi yang digunakan

### 2.6.1 Eclipse

Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent).

---

<sup>9</sup> Hermawan, 2011, Hal. 2.

<sup>10</sup> <http://www.scribd.com/doc/82-71224/6-unikom-t-t> (Hal. 63) diakses 27 April 2012.

<sup>11</sup> Ibid hal. 27.



### **2.6.2 ADT (Android Development Tools)**

*Android Development Tools* (ADT) adalah *plug-in* untuk Eclipse IDE yang dirancang untuk memberikan lingkungan yang *powerful* dan terpadu untuk membangun aplikasi Android.

### **2.7 UML**

*Unified Modeling Language* (UML) adalah bahasa spesifikasi standar untuk mendokumentasi, menspesifikasi, dan membangun sistem perangkat lunak.

## **3. ANALISIS**

### **3.1 Analisis Sistem**

Analisis sistem adalah teknik pemecahan masalah yang menguraikan tentang bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka<sup>[12]</sup>.

#### **3.1.1 Analisis Kelemahan Sistem**

Untuk mengidentifikasi masalah, maka harus melakukan analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi dan pelayanan.

#### **3.1.2 Identifikasi Masalah**

Mengidentifikasi masalah merupakan tahap yang dilakukan dalam analisis sistem. Masalah sendiri dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan.

Berdasarkan hasil penelitian dan pengamatan pada aplikasi *Image Viewer*, dapat didefinisikan bahwa terkadang jika ingin melakukan *zoom in* menggunakan *image viewer* terjadi penurunan kualitas dari citra digital. Penurunan kualitas dari citra digital tersebut cukup mengganggu untuk sebagian pengguna, terutama pengguna yang menginginkan informasi yang lebih dari citra digital tersebut.

#### **3.1.3 Identifikasi Penyebab Masalah**

Permasalahan yang terjadi pada suatu sistem tidak akan muncul dengan sendirinya, pastinya ada suatu penyebab yang menimbulkan permasalahan. Penyebab terjadinya penurunan kualitas citra digital pada *Image Viewer* dikarenakan oleh ketidakmampuan citra digital dalam membentuk atau memproduksi *pixels* yang tepat pada saat terjadi perubahan ukuran atau resolusi dari citra digital tersebut.

#### **3.1.4 Alternatif Solusi**

Alternatif solusi dibuat untuk memecahkan masalah yang ada. Alternatif solusi yang ada harus dapat memenuhi atau menyediakan agar penyebab masalah yang ada pada sistem lama tidak menjadi masalah pada sistem baru yang akan dibuat. Berdasarkan identifikasi masalah diatas, dapat ditarik keputusan untuk membuat sebuah solusi dengan memberikan fitur interpolasi pada aplikasi *Image Viewer*. Interpolasi adalah

---

<sup>12</sup> Fatta, Sepetember, 2007, hal. 49

metode dasar yang digunakan secara ekstensif dalam melakukan *zooming*, *shrinking*, *rotating*, dan *geometric corrections*. Pada dasarnya, interpolasi bekerja dengan menggunakan data yang diketahui untuk memperkirakan nilai – nilai pada titik – titik yang tidak diketahui<sup>[13]</sup>.

### **3.2 Analisis Kebutuhan Sistem**

Untuk dapat mempermudah analisis sistem dalam menentukan keseluruhan kebutuhan-kebutuhan secara lengkap, maka analisis membagi kebutuhan sistem menjadi dua bagian, yaitu kebutuhan fungsional (*functional requirement*) dan kebutuhan nonfungsional (*nonfunctional requirement*). Kebutuhan fungsional adalah jenis kebutuhan yang berisikan proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisikan informasi-informasi apa saja yang harus ada dan dihasilkan oleh sistem yang berjalan. Kebutuhan nonfungsional adalah kebutuhan yang berisi properti yang dimiliki oleh sistem.

#### **3.2.1 Kebutuhan Fungsional**

#### **3.2.2 Kebutuhan Non Fungsional**

### **3.3 Analisis Kelayakan Sistem**

### **3.4 Perancangan Sistem**

Perancangan sistem merupakan tahapan setelah analisis dari siklus pengembangan sistem pendefinisian dari kebutuhan – kebutuhan fungsional dan persiapan – persiapan untuk rancangan – rancangan sistem. Tahap perancangan sistem dilakukan oleh analisis sistem dan personil – personil teknik yang lainnya.

Perancangan sistem dapat diartikan sebagai berikut :

1. Tahapan setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian dari kebutuhan – kebutuhan fungsional.
3. Persiapan untuk rancang bangun implementasi.
4. Penggambaran bagaimana suatu sistem dibentuk.
5. Dapat berupa penggambaran, perencanaan, dan pembuatan sketsa ataupun pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi.

#### **3.4.1 Diagram Konteks (Context Diagram)**

Diagram konteks adalah suatu diagram alir yang menggambarkan seluruh jaringan, masukan, dan keluaran. Diagram konteks digunakan untuk menggambarkan sistem yang sedang berjalan, mengidentifikasi awal dan akhir data yang masuk dan keluar sistem.

---

<sup>13</sup> Purnomo, Seno, 2009, <http://senosuke.wordpress.com/2009/12/22/interpolasi/> diakses 08 Oktober 2012.

### 3.4.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan gambaran sistem secara logika, yang akan mempermudah seorang pengembang sistem dalam mengembangkan sistem ke dalam basis data. DFD digunakan untuk mendokumentasikan proses berjalannya suatu sistem yang sedang dibangun, termasuk entitas sumber – sumber masukan, hasil, dan keluaran.

### 3.4.3 Sistem Flowchart

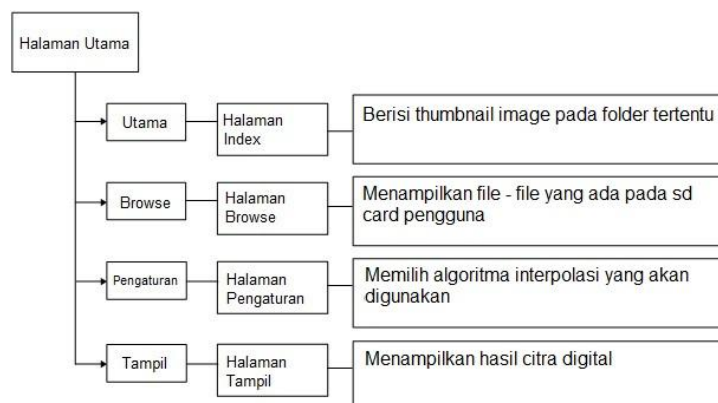
*Flowchart* merupakan diagram alir yang menggambarkan suatu sistem peralatan yang digunakan untuk suatu proses pengolahan serta hubungan antara peralatan tersebut. Tujuan utama sistem *flowchart* adalah untuk menyederhanakan rangkaian proses ataupun prosedur agar mudah untuk dipahami oleh pengguna informasi tersebut.

*Flowchart* merupakan langkah awal dalam pembuatan program. Dengan adanya sistem *flowchart*, maka urutan kegiatan menjadi lebih jelas. Setelah sistem *flowchart* disusun, maka selanjutnya *programmer* menerjemahkannya ke dalam bentuk program dengan bahasa pemrograman.

## 3.5 Perancangan Antarmuka

Perancangan antarmuka adalah suatu tampilan dimana pengguna atau user berinteraksi dengan sistem. Karena terdapat beberapa tingkat pengguna, untuk mendesain suatu antarmuka diasumsikan pengguna yang menggunakannya adalah pengguna terakhir. Tujuan dibuatnya suatu antarmuka adalah untuk memungkinkan pengguna menjalankan setiap tugas dalam kebutuhan pengguna (*user requirement*).

### 3.5.1 Perancangan Struktur



Gambar 3.6 Perancangan Struktur.

### 3.5.2 Perancangan Layout

1. Halaman Index
2. Halaman Browse
3. Halaman Pengaturan
4. Halaman Tampil

## **4. HASIL PENELITIAN DAN PEMBAHASAN**

### **4.1 Implementasi**

Implementasi merupakan tahapan setelah melakukan analisis dan perancangan sistem pada siklus rekayasa perangkat lunak, dimana aplikasi siap dioperasikan pada keadaan yang sebenarnya sehingga akan diketahui apakah program yang kita buat benar – benar dapat menghasilkan sebuah keluaran yang sesuai dengan tujuan yang diinginkan.

Tujuan implementasi adalah untuk mengkonfirmasi modul perancangan pada para pelaku sistem sehingga pengguna dapat memberi masukan kepada pengguna sistem.

Implementasi dan pembahasan penggunaan metode interpolasi pada studi kasus aplikasi *Image Viewer* ini menggunakan bahasa pemrograman Java untuk *Android Platform*. Batasan implementasi yang terdapat pada aplikasi *Image Viewer* ini adalah bahwa aplikasi ini hanya melakukan implementasi penggunaan metode interpolasi *Nearest Neighbor* dan interpolasi *Bilinear* pada saat pembentukan citra.

#### **4.1.1 Lingkungan pengembang**

##### **1. Perangkat Lunak**

Implementasi perangkat lunak merupakan proses instalasi perangkat lunak sehingga aplikasi dapat menjalankan operasi dengan benar dan sesuai dengan yang diinginkan.

##### **2. Perangkat Keras**

Implementasi perangkat keras merupakan realisasi dari sebuah analisis dan perancangan kebutuhan perangkat keras. Implementasi perangkat keras merupakan salah satu syarat dalam implementasi perangkat lunak yang akan dilakukan tahapan – tahapan berikutnya.

#### **4.1.2 Pengujian program**

Pengujian terhadap suatu program digunakan untuk mengetahui lebih dini tentang kesiapan program dalam melakukan input data, proses pengolahan data, dan output dari data yang dihasilkan. Disamping itu, juga dimaksudkan untuk mengetahui lebih jauh masih adakah kesalahan – kesalahan dan kekurangan dari program yang telah dibuat.

#### **4.1.3 Antarmuka**

##### **1. Menu Utama**

Halaman menu utama berisi tampilan thumbnail citra berdasarkan folder tertentu dari SD Card. Pengguna secara langsung dapat memilih citra mana yang ingin ditampilkan atau bisa juga memilih melalui file explorer yang dapat diakses dari menu utama ini.

## 2. File Eksplorer

Dari halaman file eksplorer ini pengguna dapat menelusuri file – file yang ada pada SD Card dan memilih citra mana yang ingin ditampilkan, selain itu pengguna juga dapat menentukan folder mana yang akan menjadi folder untuk menampilkan thumbnail pada halaman menu utama.

## 3. Halaman Pengaturan

Dalam halaman pengaturan pengguna dapat melakukan kustomisasi sesuai kebutuhan pengguna. Salah satu kustomisasi yang dapat dilakukan adalah memilih jenis metode interpolasi apa yang akan digunakan saat menampilkan citra. Secara *default* metode interpolasi yang digunakan adalah *No Interpolation*.

## 4. Halaman Tampil

Halaman dimana citra yang telah dipilih oleh pengguna ditampilkan. Sebelum citra tersebut ditampilkan, citra tersebut terlebih dahulu diolah menggunakan metode interpolasi yang telah di pilih pengguna pada halaman pengaturan.

## 4.2 Pembahasan

### 4.2.1 Pengujian Hasil Keluaran

Pengujian hasil keluaran citra menggunakan sepuluh sampel citra digital, dimana dua dari sepuluh sampel akan diambil untuk pembahasan pada pengujian hasil keluaran. Kedua sampel citra ini memiliki ukuran resolusi yang berbeda. Sample citra yang pertama memiliki resolusi sebesar 1280x1024 pixel, sedangkan sampel kedua memiliki resolusi sebesar 633x950 pixel. Pada saat menampilkan citra, untuk menghindari terjadinya *Out of Memory* yang mengakibatkan kegagalan dalam menampilkan citra, maka perlu dilakukan *resampling size*<sup>[14]</sup>. Citra yang mengalami *resampling size* akan menghasilkan citra dengan ukuran resolusi yang lebih rendah untuk menghemat penggunaan *memory* pada saat menggunakannya (menampilkan).

Apabila citra telah siap maka langkah selanjutnya adalah melakukan interpolasi. Metode interpolasi yang dilakukan akan sesuai dengan jenis interpolasi yang telah ditentukan pengguna dalam halaman pengaturan. Pada aplikasi ini metode *No Interpolation* merupakan *default* interpolasi. Pengguna dapat menggunakan metode interpolasi *Nearest Neighbor*, maupun juga dapat menggunakan metode interpolasi *Bilinear*. Metode interpolasi *Bilinear* memiliki proses yang hampir sama dengan metode interpolasi *Nearest Neighbor*, hanya saja berbeda pada saat pembentukan *pixel* dari citra yang sedang diproses.

Setelah citra melewati proses – proses yang terjadi diatas, maka citra tersebut telah siap untuk ditampilkan. Berikut tampilan dari keluaran citra secara *fullscreen* :

---

<sup>14</sup> <http://developer.android.com/training/displaying-bitmaps/load-bitmap.html> diakses 17 April 2013.



Gambar 4.5 *Fullscreen* Citra I tanpa Interpolasi.



Gambar 4.6 *Fullscreen* Citra I dengan Interpolasi *Nearest Neighbor*.



Gambar 4.7 *Fullscreen* Citra I dengan Interpolasi *Bilinear*.



Gambar 4.8 *Fullscreen* Citra II tanpa Interpolasi.



Gambar 4.9 *Fullscreen* Citra II dengan Interpolasi *Nearest Neighbor*.



Gambar 4.10 *Fullscreen* Citra II dengan Interpolasi *Bilinear*.

Berikut tampilan dari hasil keluaran citra yang telah mengalami *zoom in* sebesar 400% :



Gambar 4.11 Perbesaran Citra I tanpa Interpolasi.



Gambar 4.12 Perbesaran Citra I dengan Interpolasi *Nearest Neighbor*.



Gambar 4.13 Perbesaran Citra I dengan Interpolasi *Bilinear*.



Gambar 4.14 Perbesaran Citra II tanpa Interpolasi.



Gambar 4.15 Perbesaran Citra II dengan Interpolasi *Nearest Neighbor*.





Gambar 4.16 Perbesaran Citra II dengan Interpolasi *Bilinear*.

Dari hasil keluaran tersebut bisa dilihat bahwa citra yang menggunakan metode interpolasi *Bilinear* menghasilkan citra lebih halus daripada citra yang menggunakan metode interpolasi *Nearest Neighbor*. Selain melakukan pengujian pada hasil keluaran citra disini juga melakukan pengujian kecepatan dalam melakukan proses interpolasi.

Berikut adalah tabel hasil pengujian kecepatan proses interpolasi :

Sampel citra	Resampling size	Ukuran citra baru	Interpolasi <i>Nearest Neighbor</i>	Interpolasi <i>Bilinear</i>	Pembentukan pixel awal	Pembentukan pixel baru	Total waktu eksekusi
Citra I 1280x1024 px	Ya	640x512 px	✓	-	47 ms	34 ms	81 ms
Citra I 1280x1024 px	Ya	640x512 px	-	✓	75 ms	186 ms	261 ms
Citra II 633x950 px	Tidak	633x950 px	✓	-	59 ms	180 ms	239 ms
Citra II 633x950 px	Tidak	633x950 px	-	✓	78 ms	494 ms	572 ms

Berdasarkan pada tabel hasil pengujian kecepatan proses interpolasi, maka bisa dilihat bahwa citra yang menggunakan metode interpolasi *Nearest Neighbor* menghasilkan kecepatan proses pembentukan citra yang relatif lebih cepat dibandingkan kecepatan menggunakan metode interpolasi *Bilinear*. Selain itu, semakin rendah ukuran resolusi citra maka semakin cepat dalam melakukan pembentukan citra.

## 5. KESIMPULAN

### 5.1 Kesimpulan

Berdasarkan uraian – uraian yang telah penulis jelaskan pada pembahasan sebelumnya dan hasil dari perancangan serta implementasi metode interpolasi pada aplikasi *Image Viewer*, maka dapat diambil kesimpulan sebagai berikut :

1. Dengan adanya implementasi metode interpolasi pada aplikasi *Image Viewer*, maka citra digital mampu untuk membentuk pixel – pixel yang tepat pada saat terjadi perubahan ukuran/resolusi pada citra yang ditampilkan.
2. Metode interpolasi *Nearest Neighbor* menghasilkan kecepatan proses pembentukan citra yang lebih cepat dibandingkan kecepatan menggunakan metode interpolasi *Bilinear*.
3. Metode interpolasi *Bilinear* menghasilkan hasil keluaran citra yang lebih halus dibandingkan dengan hasil keluaran citra menggunakan metode interpolasi *Nearest Neighbor*.

## **5.2 Saran**

Dalam penyusunan skripsi ini, masih banyak terdapat kekurangan. Untuk itu apabila anda ingin membuat aplikasi ini menjadi lebih baik lagi, penulis memberi saran :

1. Pada saat thumbnail citra ditampilkan, terkadang masih mengalami *Memory Leak*. Untuk itu diperlukan *Memory Management* yang lebih baik agar mampu menghindari keadaan *Out of Memory*.
2. Dalam metode interpolasi *Bilinear*, perhitungannya masih menggunakan bilangan *Floating point*. Sehingga agar mampu menghasilkan kecepatan dalam proses interpolasi yang lebih, maka perlu melakukan konversi pada bilangan tersebut menjadi bilangan *Fixed point*.

## DAFTAR PUSTAKA

Gonzales, Rafael C. dan Richard E. Woods. 2002. *Digital Image Processing 2<sup>nd</sup> Edition*. New Jersey : Prentice Hall.

Munir, Rinaldi. 2004. *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*. Bandung : Informatika.

T. Sutoyo, dkk. 2009. *Teori Pengolahan Citra Digital*. Semarang : ANDI.

Anonim. 2011. <http://www.scribd.com/doc/82-71224/6-unikom-t-t> (Hal. 63). diakses tanggal 27 April 2012.

Bertalya. 2005. *Representasi Citra*. <http://www.scribd.com/doc/2815662/Representasi-Citra>. diakses tanggal 20 November 2010.

Team Android Developer. 2013. Loading Large Bitmaps Efficiently. <http://developer.android.com/training/displaying-bitmaps/load-bitmap.html>. diakses tanggal 17 April 2013.