

## PERTEMUAN 13

### PEMROGRAMAN KOMPUTER GRAFIK MENGGUNAKAN API

#### A. TUJUAN PEMBELAJARAN

Setelah menyelesaikan materi pada pertemuan ini, mahasiswa mampu merancang pemrograman komputer grafik menggunakan API.

Pada pertemuan ini akan dijelaskan mengenai :

1. OpenGL
2. Pengenalan Pengolahan Citra Digital

#### B. URAIAN MATERI

##### 1. Pengertian OpenGL

OpenGL awalnya disebut **Open Graphics Library** adalah Antarmuka *pemrograman* aplikasi (API) grafik 2D dan 3D yang dikembangkan oleh Silicon Graphics Inc. (SGI) untuk workstation Unix-nya. Meskipun versi OpenGL SGI telah ada sejak lama, spesifikasi OpenGL standar pertama muncul pada tahun 1992. Sekarang diadopsi secara luas di semua sistem operasi, standar OpenGL membentuk dasar dari sebagian besar permainan, desain berbantuan komputer (CAD) , dan bahkan industri virtual reality (VR).

OpenGL dirancang sebagai antarmuka yang efisien dan tidak bergantung pada perangkat keras untuk di terapkan pada banyaknya platform perangkat keras untuk diterapkan pada banyak platform perangkat keras yang berbeda. Untuk menciptakan kualitas ini, tidak ada perintah untuk melakukan tugas windowing atau mendapatkan input pengguna yang disertakan dalam OpenGL. Sebaliknya, anda harus bekerja melalui system windowing apa pun yang mengontrol perangkat keras tertentu yang anda gunakan. Demikian pula, OpenGL pula, OpenGL tidak menyediakan perintah tingkat tinggi untuk mendeskripsikan modal objek 3D. Perintah semacam itu memungkinkan anda untuk menentukan bentuk relative rumit seperti mobil, bagian tubuh, pesawat terbang, atau molekul.

Dengan OpenGL, anda harus membangun modal yang diinginkan dari sekumpulan kecil geometris primitive, titik, garis dan poligon

**API** adalah singkatan dari antarmuka pemrograman aplikasi ( API ) adalah antarmuka komputasi yang mendefinisikan interaksi antara beberapa perantara perangkat lunak. Ini mendefinisikan jenis panggilan atau permintaan yang dapat dibuat, bagaimana membuatnya, format data yang harus digunakan, konvensi yang harus diikuti, dll. Ia juga dapat menyediakan mekanisme ekstensi sehingga pengguna dapat memperluas fungsionalitas yang ada dengan berbagai cara dan untuk berbagai tingkat. API dapat sepenuhnya dibuat khusus, spesifik untuk suatu komponen, atau dirancang berdasarkan standar industri untuk memastikan interoperabilitas. Melalui penyembunyian informasi, API mengaktifkan pemrograman modular, memungkinkan pengguna menggunakan antarmuka secara independen dari implementasi. Karena OpenGL adalah API grafik dan bukan platformnya sendiri. Fungsi dasar dari OpenGL adalah untuk mengeluarkan koleksi perintah khusus atau executable ke sistem operasi. Dengan demikian, program ini bekerja dengan perangkat keras grafis yang ada yang berada pada hard drive atau sumber tertentu lainnya. Setiap perintah dalam dirancang untuk melakukan tindakan tertentu, atau memulai efek khusus tertentu yang terkait dengan grafis.

a. Langkah-langkah OpenGL

- 1) Install Microsoft Visual Studio.NET pada komputer anda.
- 2) Siapkan file OpenGL95.exe dan glut-3.7.6.zip pada direktori sementara
- 3) Masukan :
  - a) GL.H, GLAUX.H, GLU.H dan glut.h ke drive:\Program Files\Microsoft Visual Studio .NET 2003\vc7\include\gl
  - b) GLAUX32.LIB, GLU32.LIB, OPENG32.LIB dan glut32.lib ke drive:\Program Files\Microsoft Visual Studio .NET 2003\vc7\lib
  - c) OPENG32.DLL, GLU32.DLL dan glut32.dll ke drive:\Windows\System

b. Membuat Program OpenGL Sederhana

- 1) Jalankan Visual Studio .NET, buatlah sebuah proyek dengan tipe Visual C++ Projects dan template Win32 Console Project. Beri nama sesuai selera anda.
- 2) Pada Application Settings pilih Console Application dan Empty Project

- 3) Pada Solution Explorer, klik kanan Source Files lalu Add - Add New Item, lalu pilih template C++ File (.cpp), beri nama menurut selera anda, klik Open. Pada layar akan terlihat halaman kosong
- 4) Masukkan program yang berikut ini:

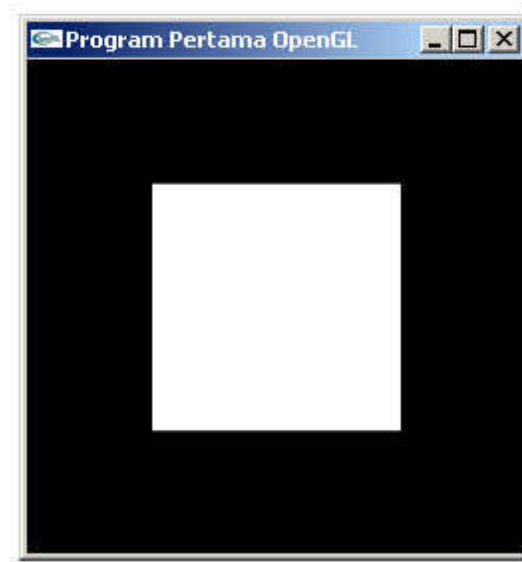
```
#include
#include void display(void)
{
    glClear (GL_COLOR_BUFFER_BIT);
    glColor3f (1.0, 1.0, 1.0);
    glBegin(GL_POLYGON);
    glVertex3f (0.25, 0.25, 0.0);
    glVertex3f (0.75, 0.25, 0.0);
    glVertex3f (0.75, 0.75, 0.0);
    glVertex3f (0.25, 0.75, 0.0);
    glEnd();

}

int main(int argc, char** argv)
{ glutInit(&argc, argv);
  glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
  glutInitWindowSize (250, 250);
  glutInitWindowPosition (100, 100);
  glutCreateWindow ("Program Pertama OpenGL");
  glClearColor (0.0, 0.0, 0.0, 0.0);
  175 glOrtho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0); glutDisplayFunc(display);
  glutMainLoop();
  return 0;
}
```

- 5) Modifikasi Project Properties dengan memilih Project – Properties – All Configurations – Linker, masukkan opengl32.lib glu32.lib glut32.lib pada textbox Additional Dependencies f.

- 6) Jika diperlukan, agar console windows tidak dibuka pada saat menjalankan program (console output akan disabled), masukan `/SUBSYSTEM:WINDOWS /ENTRY:mainCRTStartup` pada Linker – Command Line – Additional Options g.
- 7) Lakukan kompilasi terhadap program tersebut, hasilnya adalah sebagai berikut



Gambar 13. 1 Program Pertama OpenGL

## 2. Pengenalan Pemrosesan Citra Digital

### a. Pengertian Citra Digital

**Citra** adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi 2 yaitu citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog

adalah citra yang bersifat *continue* seperti gambar pada monitor televisi, foto sinar X, dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer. Citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom. Dengan kata lain, sampling pada citra menyatakan besar kecilnya ukuran *pixel* (titik) pada citra, dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam nilai tingkat keabuan

(*grayscale*) sesuai dengan jumlah bit biner yang digunakan oleh mesin, dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra.

b. Jenis – jenis Citra Digital

Ada banyak cara untuk menyimpan citra digital di dalam memori. Cara penyimpanan menentukan jenis citra digital yang terbentuk. Beberapa jenis citra digital yang sering digunakan yaitu :

1) Citra Biner (Monokrom)

Banyaknya dua warna, yaitu hitam dan putih. Dibutuhkan 1 bit di memori untuk menyimpan kedua warna ini.

2) Citra *Grayscale* (Skala Keabuan)

Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna ini. Citra 2 bit mewakili 4 warna, citra 3 bit mewakili 8 warna, dan seterusnya. Semakin besar jumlah bit warna yang disediakan di memori, semakin halus gradasi warna yang terbentuk.

3) Citra Warna (*True Color*).

Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = *Red Green Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte*, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak  $255 \times 255 \times 255 = 16.777.216$  warna lebih. Itulah sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam.

c. Elemen- elemen Citra digital

Berikut adalah elemen-elemen yang terdapat pada citra digital yaitu sebagai berikut :

1) Kecerahan (*Brightness*).

*Brightness* merupakan intensitas cahaya yang dipancarkan piksel dari citra yang dapat ditangkap oleh sistem penglihatan. Kecerahan pada sebuah titik (piksel) di dalam citra merupakan intensitas rata-rata dari suatu area yang melingkupinya.

2) Kontras (*Contrast*).

Kontras menyatakan sebaran terang dan gelap dalam sebuah citra. Pada citra yang baik, komposisi gelap dan terang tersebar secara merata.

3) Kontur (*Contour*).

Kontur adalah keadaan yang ditimbulkan oleh perubahan intensitas pada piksel-piksel yang bertetangga. Karena adanya perubahan intensitas inilah mata mampu mendeteksi tepi-tepi objek di dalam citra.

4) Warna (*Colour*).

Warna sebagai persepsi yang ditangkap sistem visual terhadap panjang gelombang cahaya yang dipantulkan oleh objek.

5) Bentuk (*Shape*).

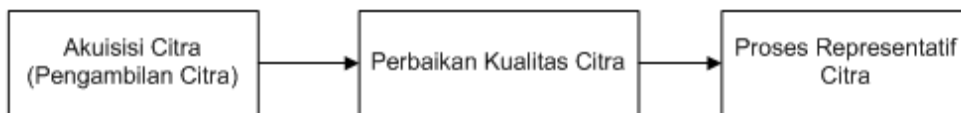
*Shape* adalah properti intrinsik dari objek 3 dimensi, dengan pengertian bahwa bentuk merupakan properti intrinsik utama untuk sistem visual manusia.

6) Tekstur (*Texture*).

*Texture* dicirikan sebagai distribusi spasial dari derajat keabuan di dalam *isekumpulan* piksel-piksel yang bertetangga. Tekstur adalah sifat-sifat atau karakteristik yang dimiliki oleh suatu daerah yang cukup besar, sehingga secara alami sifat-sifat tadi dapat berulang dalam daerah tersebut. Tekstur adalah keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital. Informasi tekstur dapat digunakan untuk membedakan sifat-sifat permukaan suatu benda dalam citra yang berhubungan dengan kasar dan halus, juga sifat-sifat spesifik dari kekasaran dan kehalusan permukaan tadi, yang sama sekali terlepas dari warna permukaan tersebut.

d. Pengertian Pemrosesan Citra Digital

Pemrosesan atau pengolahan citra digital (*digital image processing*) merupakan proses mengolah piksel-piksel di dalam citra digital untuk tujuan tertentu. Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi yang memungkinkan manusia dapat mengambil informasi dari suatu citra. Pemrosesan citra digital secara diagram pemrosesan dimulai dari pengambilan citra, perbaikan kualitas citra, sampai dengan pernyataan representatif citra yang dicitrakan sebagai berikut:



Gambar 13. 2 Diagram pemrosesan citradigital

Dalam perkembangan lebih lanjut, *Digital image processing* dan *computer vision* digunakan sebagai mata manusia, dengan perangkat *input image capture* seperti kamera dan *scanner* dijadikan sebagai mata dan mesin komputer (dengan program komputasinya) dijadikan sebagai otak yang mengolah informasi. Sehingga muncul beberapa pecahan bidang yang menjadi penting dalam *computer vision*, antara lain: *pattern recognition* (pengenalan pola), *biometric* pengenalan identifikasi manusia berdasarkan ciri-ciri biologis yang tampak pada badan manusia), *content based image and video retrieval* (mendapatkan kembali citra atau video dengan informasi tertentu), video editing, dan lain-lain

e. Teknik – teknik pengolahan citra digital

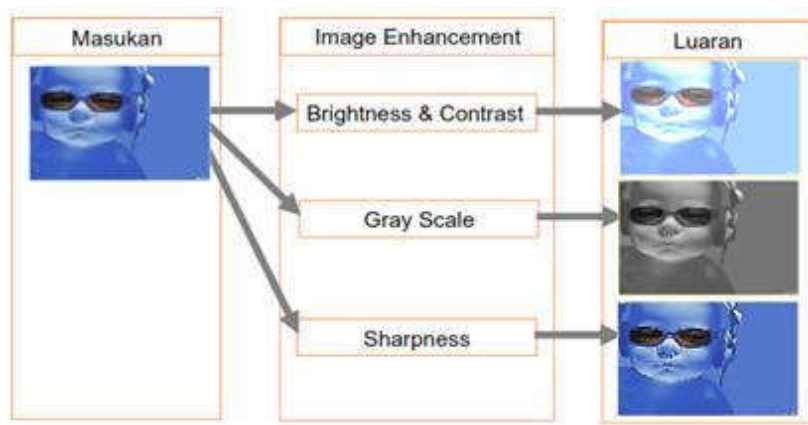
Secara umum, teknik pengolahan citra digital dibagi menjadi tiga tingkat pengolahan, yakni sebagai berikut:

- 1) Pengolahan Tingkat Rendah (*Low-Level Processing*). Pengolahan ini merupakan operasional-operasional dasar dalam pengolahan citra, seperti pengurangan noise (*noise reduction*), perbaikan citra (*image enhancement*) dan restorasi citra (*image restoration*).

- 2) Pengolahan Tingkat Menengah (*Mid-Level Processing*). Pengolahan ini meliputi segmentasi pada citra, deskripsi objek, dan klasifikasi objek secara terpisah.
- 3) Pengolahan Tingkat Tinggi (*High-Level Processing*). Pengolahan ini meliputi analisis Citra.

Dari ketiga tahap pengolahan citra digital di atas, dapat dinyatakan suatu gambaran mengenai teknik-teknik pengolahan citra digital dan macam-macamnya, antara lain sebagai berikut :

- 1) **Image enhancement**, berupa proses perbaikan citra dengan meningkatkan kualitas citra, baik kontras maupun kecerahan.



Gambar 13. 3 Image enhancement

- 2) **Image restoration**, yaitu proses memperbaiki model citra, biasanya berhubungan dengan bentuk citra yang sesuai.



Gambar 13. 4 Image restoration



- 3) **Color image processing**, yaitu suatu proses yang melibatkan citra berwarna, baik berupa *image enhancement*, *image restoration*, atau yang lainnya.



Gambar 13. 5 Color image processing

- 4) **Wavelet dan multiresolution processing**, merupakan suatu proses yang menyatakan citra dalam beberapa resolusi.



Gambar 13. 6 Wavelet dan multiresolution processing

- 5) **Image compression**, merupakan proses yang digunakan untuk mengubah ukuran data pada citra.



Gambar 13. 7 Image compression

- 6) **Morphological processing**, yaitu proses untuk memperoleh informasi yang menyatakan deskripsi dari suatu bentuk pada citra.



Gambar 13. 8 Morphological processing

- 7) **Segmentation**, merupakan proses untuk membedakan atau memisahkan objek-objek yang ada dalam suatu citra, seperti memisahkan objek dengan latar belakangnya.



Gambar 13. 9 Segmentation

- 8) **Object recognition**, yaitu suatu proses yang dilakukan untuk mengenali objek-objek apa saja yang ada dalam suatu citra.



Gambar 13. 10 Object recognition

**C. SOAL LATIHAN/TUGAS**

Latihan	Petunjuk Pengerjaan Tugas
<b>Latihan Pertemuan 13</b>	1. Carilah aplikasi pengolah citra di internet, kemudian analisislah aplikasi tersebut berdasarkan teori diatas!

**D. REFERENSI**