

II. TINJAUAN PUSTAKA

2.1 Permainan Edukatif

Bermain merupakan tugas pokok anak. Dalam usia dini anak-anak tidak diperbolehkan melakukan kegiatan belajar. Anak-anak hanya diperbolehkan melakukan kegiatan bermain, namun sebagai orang tua maupun tenaga didik harus jeli dalam menentukan jenis permainan anak, karena dalam bermain anak akan belajar mengendalikan diri dan dapat melakukan berbagai kegiatan untuk merangsang daya pikir dan keterampilan anak, karena bermain merupakan cermin perkembangan anak.

"Bermain adalah suatu kegiatan atau tingkah laku yang dilakukan anak secara sendirian atau berkelompok dengan menggunakan alat atau tidak untuk mencapai tujuan tertentu." (Antoso, 2002)

Permainan edukatif adalah semua jenis permainan yang bertujuan untuk menciptakan lingkungan dan jenis permainan yang bersifat edukatif demi kepentingan peserta didik. (Bahri, 1995)

Permainan disebut memiliki nilai edukatif jika permainan tersebut dapat merangsang daya pikir anak, meningkatkan konsentrasi dan dapat mengajarkan anak memecahkan suatu permasalahan. Selain itu juga mainan

edukatif tidak hanya sekedar membuat anak menikmati permainan tapi juga dituntut agar membuat anak untuk teliti dan tekun ketika mengerjakan mainan tersebut.

Permainan edukatif dapat mengembangkan aspek-aspek penting dalam pertumbuhan anak seperti mengembangkan aspek fisik, bahasa, aspek kognitif dan aspek sosial. Berikut adalah kategori permainan edukatif: (Zelva Group, 2008)

1. Multifungsi

Dari satu mainan bisa dibuat berbagai variasi mainan sehingga stimulasi yang didapat anak juga lebih beragam.

2. Melatih *problem solving*

Dalam memainkannya anak diminta untuk melakukan pemecahan masalah. Dalam permainan *puzzle* misalnya, anak diminta untuk menyusun potongan-potongannya menjadi utuh.

3. Melatih konsep-konsep dasar

Lewat permainan, anak dilatih untuk mengembangkan kemampuan dasarnya seperti mengenal bentuk, warna, besaran, juga melatih motorik halus.

4. Melatih ketelitian dan ketekunan

Dengan mainan edukatif, anak tidak hanya sekedar menikmati tetapi juga dituntut untuk teliti dan tekun ketika mengerjakannya.

5. Merangsang kreativitas

Permainan ini mengajak anak untuk selalu kreatif lewat berbagai variasi mainan yang dilakukan. Bila sejak kecil anak terbiasa untuk

menghasilkan karya, lewat permainan rancang bangun misalnya, kelak dia akan lebih berinovasi untuk menciptakan suatu karya. Melatih kemampuan motorik stimulasi untuk motorik halus diperoleh saat anak menjemput mainannya, meraba, memegang dengan kelima jarinya, dan sebagainya. Sedangkan rangsangan motorik kasar didapat anak saat menggerak-gerakkan mainannya, melempar, mengangkat, dan sebagainya.

6. Melatih konsentrasi

Mainan edukatif dirancang untuk menggali kemampuan anak, termasuk kemampuannya dalam berkonsentrasi. Saat menyusun *puzzle*, anak dituntut untuk fokus pada gambar atau bentuk yang ada di depannya, ia tidak berlari-larian atau melakukan aktivitas fisik lain sehingga konsentrasinya bisa lebih tergal. Tanpa konsentrasi, bisa jadi hasilnya tidak memuaskan.

7. Mengenalkan konsep sebab akibat

Contohnya, dengan memasukkan benda kecil ke dalam benda yang besar anak akan memahami bahwa benda yang lebih kecil bisa dimuat dalam benda yang lebih besar. Sedangkan benda yang lebih besar tidak bisa masuk ke dalam benda yang lebih kecil. Ini adalah pemahaman konsep sebab akibat yang sangat mendasar.

8. Melatih bahasa dan wawasan

Permainan edukatif sangat baik bila disertai dengan penuturan cerita. Hal ini akan memberikan manfaat tambahan buat anak,

yakni meningkatkan kemampuan berbahasa juga keluasan wawasannya.

9. Mengenalkan warna dan bentuk

Dari mainan edukatif, anak dapat mengenal ragam/variasi bentuk dan warna. Ada benda berbentuk kotak, segiempat, bulat dengan berbagai warna: biru, merah, hijau, dan lainnya.

2.2 Pendidikan Anak Usia Dini

Pendidikan Anak Usia dini (PAUD) adalah suatu upaya pelayanan pendidikan bagi anak usia dini (0-6 tahun) yang dilakukan di lingkungan keluarga, sekolah, lembaga atau tempat pengasuhan anak yang berpengaruh terhadap proses tumbuh kembang anak, agar dapat berkembang secara optimal dan memiliki kesiapan memasuki pendidikan dasar . (Dahasik, 2006)

Pendidikan Anak Usia Dini (PAUD) adalah jenjang pendidikan sebelum jenjang pendidikan dasar, yang merupakan suatu upaya pembinaan yang ditujukan bagi anak sejak lahir sampai dengan usia enam tahun yang dilakukan melalui pemberian rangsangan pendidikan untuk membantu pertumbuhan dan perkembangan jasmani dan rohani agar anak memiliki kesiapan dalam memasuki pendidikan lebih lanjut, yang diselenggarakan pada jalur formal, nonformal, dan informal. (Wikipedia, 2010)

Saat ini pendidikan anak usia dini menjadi sorotan penting bagi pemerintah dan masyarakat kita, mengingat pendidikan ini sangat

berpengaruh terhadap tumbuh kembang anak pada usia selanjutnya. Anak-anak akan mudah dibentuk dan diarahkan sesuai dengan bakat yang ada pada anak. Mengetahui perkembangan buah hati sejak dini membantu orang tua untuk lebih tepat memilih jalur pendidikan untuk anak selanjutnya.

Pendidikan anak usia dini diselenggarakan untuk membentuk anak Indonesia menjadi anak-anak yang kreatif dan memiliki kualitas yang baik. Selain itu pendidikan anak usia dini adalah tempat untuk anak-anak menyiapkan diri untuk memasuki jenjang pendidikan formal

Rentangan anak usia dini menurut Pasal 28 UU Sisdiknas No.20/2003 ayat 1 adalah 0-6 tahun. Sementara menurut kajian rumpun keilmuan PAUD dan penyelenggaraannya di beberapa negara, PAUD dilaksanakan sejak usia 0-8 tahun. (Direktorat Pendidikan Anak Usia Dini, 2006)

2.3 Pendekatan Sentra

Pendekatan Sentra adalah pendekatan penyelenggaraan Pendidikan Anak Usia Dini yang berfokus kepada anak yang dalam proses pembelajarannya berpusat di sentra main. (Direktorat Pendidikan Anak Usia Dini, 2006)

Ada beberapa jenis sentra main yang dapat diterapkan dalam metode pendekatan sentra. Beberapa diantaranya adalah sebagai berikut:

1. Sentra Bahan Alam
2. Sentra Main Peran

3. Sentra Balok
4. Sentra Persiapan
5. Sentra Seni
6. Sentra Memasak
7. Sentra Agama

Pendekatan Sentra berarti fokus, yakni anak-anak akan dituntun untuk mengikuti beberapa tahapan-tahapan dalam tema yang sama.

2.4 *Augmented Reality*

2.4.1 *Sejarah Augmented Reality*

Augmented Reality merupakan penggabungan dunia nyata dengan dunia virtual. Dimana obyek virtual dapat ditampilkan ke dalam dunia nyata, untuk itu obyek ditampilkan dengan format 3D agar terlihat lebih real dan benar-benar nyata. Sejarah tentang *Augmented Reality* dimulai dari tahun 1957-1962, seorang sinematografer bernama Morton Heilig menciptakan dan mematenkan sebuah simulator yang disebut Sensorama dengan visual, getaran dan bau. Kemudian pada tahun 1966, Ivan Sutherland menemukan *head mounted display* yang dia claim sebagai jendela ke dunia virtual. dapat dilihat pada gambar 2.1. *Head Mounted Display* (HMD) berfungsi layaknya webcam dan monitor. Pada umumnya berbentuk seperti kacamata digunakan untuk melihat obyek virtual.



Gambar 2.1 *Head Motion Display*

Tahun 1975 seorang ilmuwan bernama Myron Krueger menemukan *Videoplace* yang memungkinkan pengguna dapat berinteraksi dengan obyek virtual untuk pertama kalinya. Tahun 1989, Jaron Lanier, memperkenalkan *Virtual Reality* dan menciptakan bisnis komersial pertama kali di dunia maya dan tahun 1992 mengembangkan *Augmented Reality* untuk melakukan perbaikan pada pesawat boeing. Pada tahun yang sama, LB Rosenberg mengembangkan salah satu fungsi sistem AR, yang disebut *Virtual Fixtures*, yang digunakan di Angkatan Udara AS Armstrong Labs, dan menunjukkan manfaatnya pada manusia, dan pada tahun 1992 juga, Steven Feiner, Blair MacIntyre dan Dorée Seligmann, memperkenalkan untuk pertama kalinya *Major Paper* untuk perkembangan *Prototype AR*. (Vedca, 2010)

2.4.2 Perkembangan *Augmented Reality*

Pada tahun 1999, Hirokazu Kato, mengembangkan ArToolkit di HITLab dan didemonstrasikan di SIGGRAPH, pada tahun 2000, Bruce.H.Thomas, mengembangkan ARQuake, sebuah Mobile Game AR yang ditunjukkan di *International Symposium on Wearable Computers*.

Pada tahun 2008, Wikitude AR Travel Guide, memperkenalkan Android G1 Telephone yang berteknologi AR. Tahun 2009, Saqoosha memperkenalkan FLARToolkit yang merupakan perkembangan dari ArToolkit. FLARToolkit memungkinkan kita memasang teknologi AR di sebuah website, karena output yang dihasilkan FLARToolkit berbentuk Flash. Di tahun yang sama, Wikitude Drive meluncurkan sistem navigasi berteknologi AR di Platform Android. Tahun 2010, Acrossair menggunakan teknologi AR pada I-Phone 3GS. (Vedca, 2010)

2.4.3 *Augmented Reality* dan *Virtual Reality*

Augmented Reality mengaburkan batasan antara dunia nyata dan dunia maya.(Vanestorrz, 2010)

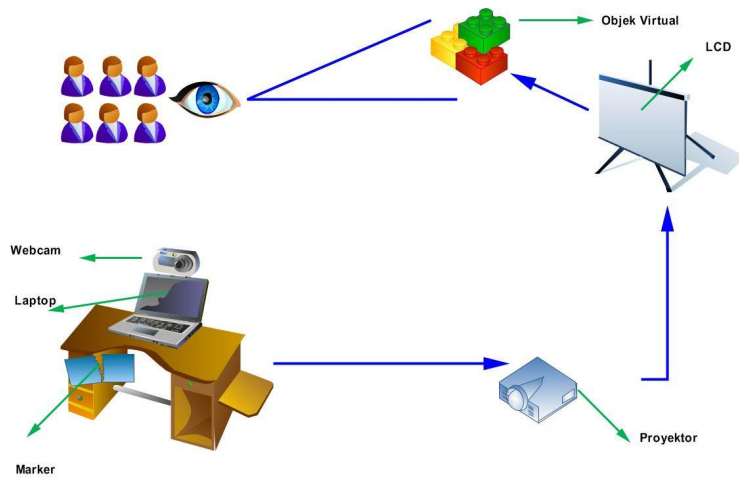
Augmented Reality (AR) berada di antara dunia nyata (*real environment*) dengan dunia maya (*virtual environment*). Keindahan dunia nyata tidak pernah bisa ditandingi oleh kehebatan teknologi yang menawarkan tampilan gambar, secanggih apa pun teknologi

tersebut. Setinggi apa pun resolusi gambar di layar monitor maupun pada rekaman film DVD (*Digital Versatile Disc*) yang ditampilkan pada layar datar, atau bahkan ketajaman gambar di bioskop dengan layar lebarnya, tetap saja semuanya tidak bisa menggantikan kenikmatan pemandangan dunia yang sebenarnya.

Dunia maya tidak pernah bisa seindah dunia nyata, tetapi dunia nyata terkadang memasang batasan-batasan yang tidak bisa ditembus oleh mata normal manusia. Karena itulah keduanya kini disatukan dalam teknologi AR yang mulai merebak di awal abad ke-20. Dengan AR kita bisa melihat hologram di dunia nyata, bukan hanya di laboratorium. Saat kita sedang berjalan-jalan di ruang terbuka untuk menikmati pemandangan alam, kita bisa melihat panduan-panduan berupa gambar-gambar semacam hologram.

Aplikasi paling hebat bisa dirasakan dalam dunia kesehatan. Para dokter bisa melihat jaringan-jaringan otak pasiennya tanpa memerlukan kamera dan layar monitor tambahan. (Surya, 2000)

Alur Augmented Reality dapat dilihat pada gambar 2.2



Gambar 2.2 Alur Augmented Reality

2.4.4 Penerapan *Augmented Reality*

Saat ini telah banyak di jumpai taknologi-ternologi berbasis *Virtual Reality* maupun *Augmented Reality* pada kehidupan sehari-hari. khususnya negara-negara maju telah banyak menciptakan teknologi ini untuk berbagai kebutuhan. Beberapa diantaranya adalah sebagai berikut:

1. Kedokteran (*Medical*): Teknologi pencitraan sangat dibutuhkan di dunia kedokteran, seperti misalnya, untuk simulasi operasi, simulasi pembuatan vaksin virus, dll. Untuk itu, bidang kedokteran menerapkan *Augmented Reality* pada visualisasi penelitian mereka.
2. Hiburan (*Entertainment*): Dunia hiburan membutuhkan *Augmented Reality* sebagai penunjang efek-efek yang akan dihasilkan oleh hiburan tersebut.

3. Latihan Militer (*Military Training*): Militer telah menerapkan *Augmented Reality* pada latihan tempur mereka. Sebagai contoh, militer menggunakan *Augmented Reality* untuk membuat sebuah permainan perang, dimana prajurit akan masuk kedalam dunia game tersebut, dan seolah-olah seperti melakukan perang sesungguhnya.
4. *Engineering Design*: Seorang *engineering design* membutuhkan *Augmented Reality* untuk menampilkan hasil design mereka secara nyata terhadap klien. Dengan *Augmented Reality* klien akan tahu, tentang spesifikasi yang lebih detail tentang desain mereka.
5. *Robotics* dan *Telerobotics*: Dalam bidang robotika, seorang operator robot, menggunakan pengendali pencitraan visual dalam mengendalikan robot itu. (Vedca, 2010)

2.5 ARToolkit

2.5.1 Pengertian ARToolkit

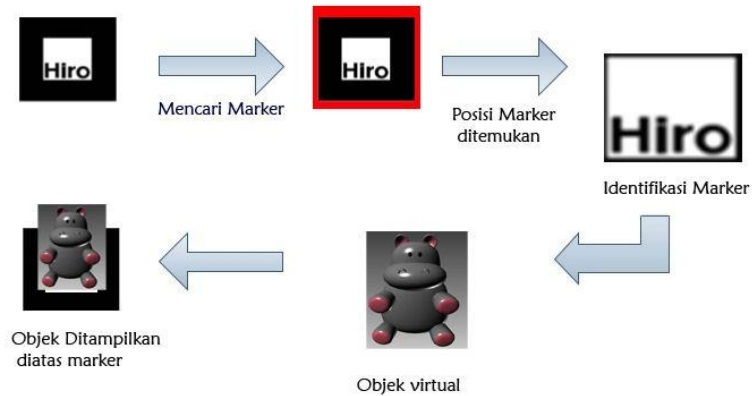
ARToolkit adalah sebuah *software* yang digunakan untuk membangun dan menjalankan aplikasi AR (*Augmented Reality*). ARToolkit akan melacak marker yang tertangkap oleh kamera sebagai perintah untuk *load* obyek yang akan ditampilkan, sehingga kamera seolah-olah menangkap sebuah gambar atau obyek dalam dunia nyata.

ARToolKit adalah *tracking system library* yang bersifat *open-source* yang memungkinkan programmer dengan mudah mengembangkan aplikasi *Augmented Reality*. (Bilinghursts M dan Poupyrey, 2000)

ArToolkit menggunakan pelacakan video, untuk menghitung posisi kamera yang nyata dan mengorientasikan pola pada kertas marker secara realtime. Marker adalah pola yang akan dibaca oleh webcam. Setelah posisi dari kamera telah diketahui, maka *virtual camera* dapat diposisikan pada titik yang sama, dan obyek 3D akan digambarkan di atas marker.

2.5.2 Sistematika Kerja ARToolkit

ARToolkit melacak marker yang terekam oleh webcam sebagai perintah untuk meload gambar atau obyek virtual kedalam dunia nyata. Marker akan terdeteksi apabila disekeliling marker terdapat blok berwarna hitam. Setelah kamera menangkap marker. ARToolkit akan menidentifikasi marker tersebut. Jika marker sebelumnya telah dikenalkan pada ARToolkit maka ARToolkit akan mencari obyek virtual yang sesuai dengan perintah pada data VRML. Setelah itu obyek akan diload dan muncul sebagai obyek nyata. Sistematika ARToolkit dapat dilihat dalam gambar 2.3.



Gambar 2.3 Sistematika ARToolkit

2.6 Desain Grafis 3D Blender

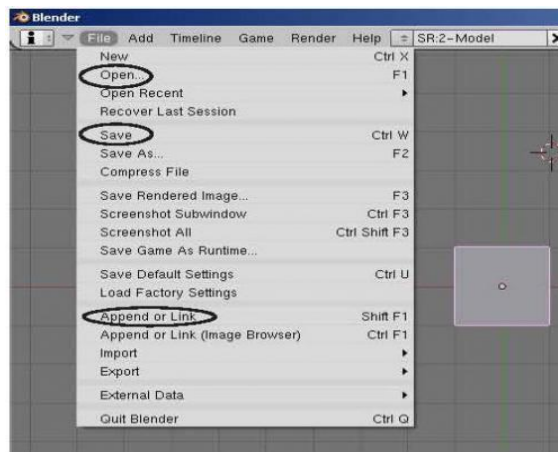
Menggambar Obyek 3D merupakan hal yang menarik untuk dipelajari. Selain tutorial pembuatannya mudah dijumpai, hasil dari gambarnya pun seolah mendekati gambar aslinya, karena 3D memiliki ruang dimensi yang hampir sama dengan dunia nyata. Tidak hanya berupa gambar 3D, dapat pula diaplikasikan pada gambar bergerak atau animasi. (Khoirul, 2009)

Salah satu kendala dalam memproduksi obyek 3D adalah mahalnya biaya yang dipakai untuk pembelian software. Tapi dengan 'Blender', software 3D berbasis open source, semua masalah biaya bisa diatasi. Blender memiliki banyak sekali kemampuan mulai dari pemodelan 3D, rendering, shading, animasi 3D, sampai pembuatan game 3D secara utuh.

Blender 2.49 memiliki fitur baru seperti :

1. Video Texture
2. Real-time GLSL Material
3. Game Logic
4. Bullet SoftBody
5. Python Editor
6. Multilayer Textures
7. Physics
8. Render Baking dan Normal Mapping dan masih banyak yang lainnya.

Adapun kekurangan pada Blender, untuk penguasaannya sangat membutuhkan waktu lama karena memang agak sulit dipahami terutama pada GUI nya. gambar 2.4 menampilkan menu *open*, *saving* dan *appending files*.



Gambar 2.4 *Open, Saving dan Appending Files*

Blender menggunakan perintah mirip dengan program lain untuk menyimpan dan membuka pekerjaan dengan beberapa pengecualian.

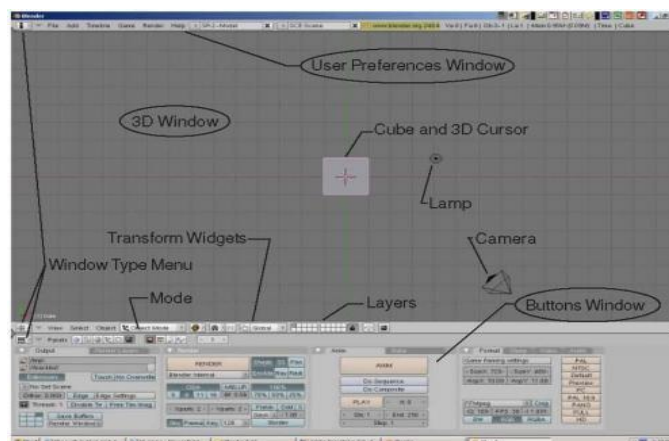


Gambar 2.6 *Append dan Link*

Pilihan Link memungkinkan untuk link ke *file* Blender lain. Pilihan ini memungkinkan untuk menghubungkan perubahan pada *file* yang akan diperbarui secara otomatis ketika *file* lain dibuka.

Pengaturan yang penting untuk saat ini dalam jendela Preferensi User

- View and Controls – tips dan pengaturan mouse
- Edit Methods – bagaimana hal-hal yang terkait dan terduplikasi, pengaturan undo
- Language and Fonts – gaya dan ukuran teks
- System and OpenGL – mengubah beberapa pengaturan sistem
- File Paths – memberitahu File Blender di mana harus mencari hal-hal tertentu



Gambar 2.7 *Interface Blender*

2.7 Bahasa Pemrograman C++

C++ adalah salah satu bahasa pemrograman komputer. Dibuat pada tahun 1980-an oleh Bjarne Stroustrup sebagai pengembangan dari Bahasa pemrograman C. Salah satu perbedaan yang paling mendasar dengan bahasa C adalah dukungan terhadap konsep pemrograman berorientasi obyek (Object Oriented Programming). (Alivian, 2010)

Beberapa hal yang perlu diingat mengenai bahasa C:

1. Komentar dalam C menggunakan `/* ... */` dan bisa memuat beberapa baris sekaligus.
2. Sebuah program dalam C harus memiliki fungsi bernama `main()`, fungsi ini adalah fungsispesial karena di situlah main program kita.
3. Dalam C sebenarnya tidak ada *procedure*, *procedure* didefinisikan sebagai *function* yang mengembalikan nilai *void* (*void* = sesuatu yang tidak ada).
4. Setiap baris perintah harus diakhiri dengan titik koma ‘;’ kecuali yang diawali dengan kres ‘#’.
5. Blok program diawali kurung-kurawal-buka ‘{’ dan diakhiri kurung-kurawal-tutup ‘}’.
6. Bahasa C adalah bahasa yang *case-sensitive*, jadi perhatikan dalam penamaan variabel, type, maupun pemanggilan fungsi-fungsi yang sudah ada.
7. Perintah terakhir di dalam `main()` yaitu `return 0` digunakan untuk mengindikasikan bahwa program tersebut sukses berjalan. Pada

beberapa kontes pemrograman, hal ini merupakan salah satu syarat diterimanya program. (Program yang tidak me-*return exit code* 0 dianggap mengalami runtime *error*.)

contoh gambaran umum program C++ dapat dilihat pada program 2.1

```
//C++:  
//ini bagian 'header' program, berisi library-library yang  
//diperlukan  
#include <iostream>  
#include <fstream>  
#include <string>  
  
//berikut adalah bagian deklarasi kostanta, type bentukan,  
//dan variabel global yang akan digunakan  
const float pi = 3.1415926;  
  
pedef int angka;  
  
int i,j;  
  
//berikut adalah bagian deklarasi fungsi-fungsi  
//dan prosedur yang akan dipakai  
longint kuadrat(int x)  
{  
return (x*x);  
}  
void tukar(int *a, int *b);  
int temp;  
{  
temp := *a;  
*a := *b;  
*b := temp;  
}  
  
// main program  
int main()  
{  
std::cout << "Selamat datang!" << endl;  
return 0;  
}
```

Program 2.1 Gambaran Umum Program C++

2.8 Open GL

OpenGL adalah suatu spesifikasi grafik low-level yang menyediakan fungsi untuk pembuatan grafik primitif termasuk titik, garis, dan lingkaran. OpenGL digunakan untuk keperluan-keperluan pemrograman grafis. OpenGL bersifat Open-Source, multi-platform dan multi-language serta digunakan mendefinisikan suatu obyek, baik obyek 2 dimensi maupun obyek 3 dimensi.

Untuk dapat menggunakan OpenGL dalam Visual C++, diperlukan library OpenGL. Sebelum menggunakan open Gl terlebih dahulu harus memasukan *file* “opengl32.lib”, “glu32.lib”, dan “glaux.lib” ke dalam folder “..\vc\lib”. Kemudian *file* header “gl.h”, glaux.h”, dan “glu.h” ke dalam folder “..\vc\include\gl”

OpenGL juga merupakan suatu antarmuka pemrograman aplikasi (*Application Programming Interface* (API)) yang tidak tergantung pada piranti dan platform yang digunakan, sehingga OpenGL dapat berjalan pada sistem operasi Windows, UNIX dan sistem operasi lainnya. OpenGL pada awalnya didesain untuk digunakan pada bahasa pemrograman C/C++, namun dalam perkembangannya OpenGL dapat juga digunakan dalam bahasa pemrograman yang lain seperti Java, Visual Basic, Delphi, maupun Fortran. OpenGL di-*package* secara berbeda-beda sesuai dengan bahasa pemrograman yang digunakan.

OpenGL lebih mengarah pada prosedural daripada sebuah deskriptif API grafis. Untuk mendeskripsikan *scene* dan bagaimana penampilannya, programmer dituntut untuk menghasilkan efek yang diinginkan. Langkah

tersebut termasuk memanggil banyak perintah OpenGL, Perintah tersebut digunakan untuk menggambarkan grafis primitif seperti titik, garis dan poligon dalam tiga dimensi. Sebagai tambahan, OpenGL mendukung lighting, shading, texture mapping, blending, transparency dan banyak efek khusus lainnya. (Chafied, 2010)