

# PEMBUATAN PERANGKAT LUNAK EDITOR DENAH RUANG 2D UNTUK DIVISUALISASIKAN SECARA 3D<sup>1</sup>

**Rudy Adipranata, Liliana, Yohanes Pribadi**

Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra  
Jl. Siwalankerto 121-131, Surabaya  
Email: rudya@petra.ac.id, lilian@petra.ac.id

**ABSTRAK:** Untuk merancang denah ruang pada sebuah bangunan, biasanya dilakukan penggambaran sketsa ruang yang diinginkan pada kertas. Tidak semua orang, terutama orang yang awam terhadap bidang arsitektur, membayangkan sebuah gambar denah sebagai ruangan akan sangat menyulitkan. Untuk menjembatani antara perancang dengan pelanggan, antara rancangan denah dalam bentuk 2D dengan visualisasi dalam bentuk 3D, maka dalam penelitian ini dikembangkan sebuah perangkat lunak yang mampu menghasilkan visualisasi bentuk sebuah ruang dari inputan berupa gambar denah 2D tampak atas. Dengan adanya fasilitas editor pada perangkat lunak ini, maka perancang dapat melakukan perubahan denah ataupun ukuran yang sudah tergambar dan langsung menampilkan hasil visualisasinya. Oleh karena itu, pada penelitian ini, dikembangkan pembuatan perangkat lunak editor denah ruang secara 2D yang dapat langsung divisualisasikan dalam bentuk 3D dengan ukuran yang sama dengan yang diinputkan pada gambar denahnya. Pada visualisasi yang berbentuk 3D, pengguna dapat melakukan penjelajahan ruang seperti berada pada ruang secara nyata. Pada perangkat lunak ini, bangunan dapat dibuat bertingkat dan bagian ruang yang disediakan adalah lantai, dinding, pintu, jendela serta tangga. Perangkat lunak yang dihasilkan mampu menampilkan visualisasi dalam bentuk 3D secara nyata. Kamera dapat diletakkan di sembarang tempat dan kemudian menggeser-geser posisinya dengan menggerakkan kursor.

**Kata kunci:** editor denah ruang, 2D, 3D, visualisasi, *OpenGL*

**ABSTRACT:** Usually we use paper to draw sketch of room for designing building. Using paper can cause a problem especially for changing the sketch or size of the room. To solve that problem, we can use software to design sketch of the room. Nowadays there are many software can be used but many of those can only view in two dimension, top side. Using only two dimensions viewing, user have difficulties to imagine the real room in three dimensions. Also there are software which can view in three dimensions, but unfortunately the price of those software is so high. In this paper, we implement software to help user to design the sketch of the room in two dimensions, but the result can be viewed in three dimensions. In three dimensions view, user can do movement like view the real world. This software also can be used to create building have multi level, and the accessories that can be used are floor, wall, door, window and stair. To implement this software we use *OpenGL* library.

**Keywords:** room sketch editor, 2D to 3D transformation, *OpenGL*

## PENDAHULUAN

Saat ini dalam dunia teknologi informasi telah diterapkan berbagai macam teknologi yang dapat digunakan untuk membantu menyelesaikan pekerjaan manusia. Salah satu teknologi tersebut adalah komputer grafis. Komputer grafis banyak digunakan untuk membuat model dari suatu benda yang ada dalam dunia nyata, sehingga benda tersebut dapat ditampilkan di monitor komputer. Salah satu penggunaan komputer grafis adalah untuk perancangan denah ruang.

Untuk merancang denah ruang pada sebuah bangunan, biasanya dilakukan penggambaran sketsa ruang yang diinginkan pada kertas. Hal ini dapat merepotkan terutama jika melakukan perubahan denah ataupun ukuran yang sudah tergambar. Untuk mengatasi hal tersebut, dapat digunakan perangkat lunak untuk menggambar denah ruang. Saat ini banyak perangkat lunak yang ada di pasaran yang menyediakan editor pembuatan denah ruang secara 2D, yaitu tampak atas. Akan tetapi kebanyakan perangkat lunak tersebut tidak mempunyai fasilitas untuk memvisualisasikan denah yang sudah dibuat

---

<sup>1</sup> Dibiayai proyek Pengkajian dan Penelitian Ilmu Pengetahuan Terapan dengan surat perjanjian pelaksanaan penelitian nomor: 236/SP3/PP/DP2M/II/2006 Direktorat Pembinaan Penelitian dan Pengabdian pada Masyarakat Direktorat Jenderal Pendidikan Tinggi Departemen Pendidikan Nasional

tersebut ke dalam bentuk 3D, yang menyerupai bentuk ruang secara nyata. Dengan hanya menampakkan denah secara dua dimensi, pengguna akan mengalami kesulitan untuk membayangkan bentuk ruang sesungguhnya yang merupakan ruang tiga dimensi. Untuk itu pada penelitian ini dibuat perangkat lunak editor denah ruang dua dimensi yang dapat langsung ditampilkan visualisasinya dalam bentuk tiga dimensi dengan ukuran yang diinputkan pada gambar denah 2D nya.

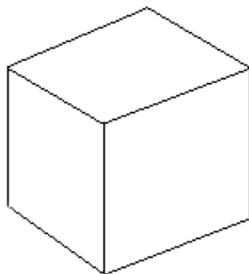
## KOMPUTER GRAFIS

Secara sederhana komputer grafis adalah gambar yang dibuat dengan menggunakan komputer [4]. Pada saat ini dapat dilihat banyak contoh pembuatan gambar dengan menggunakan komputer dalam berbagai bidang pekerjaan, seperti pada majalah dan televisi. Dengan komputer juga dapat dibuat desain-desain yang diperlukan dalam bidang permesinan, perindustrian dan juga arsitektur. Biasanya gambar tersebut nantinya akan dibuat bentuk realnya, dalam bentuk benda 3D.

Komputer grafis juga menunjuk pada alat-alat yang digunakan untuk membuat gambar. Alat-alat tersebut dibedakan menjadi dua jenis yaitu perangkat keras (contoh: *monitor, mouse, keyboard*) serta perangkat lunak (contoh: sistem operasi, *editor, compiler*). Akhirnya komputer grafis seringkali juga berarti semua bagian yang termasuk alat-alat untuk menghasilkan gambar dan gambar itu sendiri [4]. Bidang ini diyakini oleh banyak orang bermula sejak awal tahun 1960, yaitu pada tesis doctoral oleh Ivan Sutherland yang berjudul *Sketchpad* di MIT (*Massachusetts Institute of Technology*).

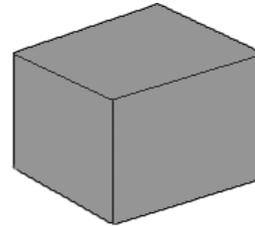
### Tiga Dimensi

Istilah tiga dimensi atau 3D berarti sebuah obyek yang dideskripsikan mempunyai tiga dimensi ukuran yaitu: lebar, tinggi dan kedalaman [8]. Contoh benda 3D adalah benda-benda yang ada dalam dunia nyata. Dalam komputer grafis, 3D ditampilkan dalam bentuk 2D ke layar monitor dengan perhitungan proyeksi perpektif dan pewarnaan. *Perspective* menunjuk pada sudut antar garis yang dapat menghasilkan efek 3D [8]. Gambar 1 menunjukkan gambar hasil *perspective*.



Gambar 1. Gambar *Perspective*

Untuk lebih memantapkan persepsi terhadap gambar 3D, dapat ditambahkan warna untuk membuat obyek menjadi kelihatan *solid*. Dengan menambahkan warna pada obyek maka sisi permukaan obyek tidak tampak berlubang.



Gambar 2. Benda Berwarna

### Teknik *Sphere Collision*

Jika sebuah obyek digerakkan, maka ada kemungkinan obyek tersebut akan menabrak obyek lain. Untuk mengetahui apakah sebuah obyek telah menabrak obyek lain diperlukan teknik *collision*. Tanpa menggunakan teknik *collision*, maka obyek tersebut akan menembus obyek lain. Teknik *collision* ada bermacam-macam, dibedakan berdasarkan obyek yang digerakkan, di antaranya adalah *sphere collision*. Teknik *sphere collision* adalah teknik *collision* yang mengasumsikan obyek yang digerakkan berbentuk bola (*sphere*). Obyek berbentuk bola inilah yang nantinya akan dideteksi pada setiap gerakan apakah menabrak obyek lain atau tidak.

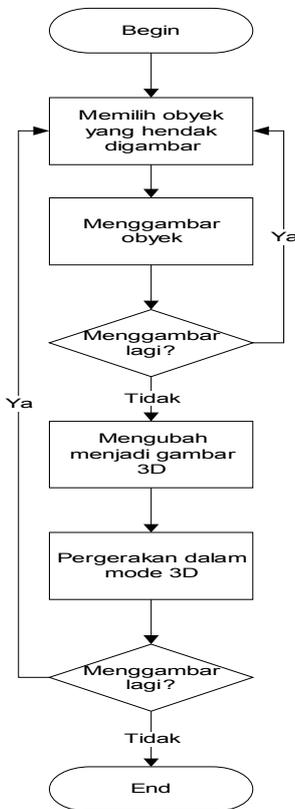
Secara garis besar teknik *sphere collision* mempunyai empat langkah yaitu :

1. Langkah pertama adalah mendeteksi apakah bola menabrak *plane* poligon. *Plane* bersifat tak terbatas, sehingga meskipun obyek terletak sangat jauh dari *plane* tetap membutuhkan langkah ini. Langkah ini akan menghasilkan letak bola terhadap *plane* apakah berada di depan, memotong, atau di belakang *plane*. Apabila bola memotong *plane*, maka dilanjutkan ke langkah kedua.
2. Langkah kedua digunakan untuk mendapatkan titik perpotongan bola dengan *plane* tepat di depan bola. Untuk menentukan titik tersebut, pertama dilakukan pencarian vektor normal dari *plane* poligon yang sedang dideteksi. Vektor normal berfungsi untuk memberitahu ke arah mana poligon menghadap. Kemudian dilakukan perkalian antara vektor normal dengan jarak antara titik pusat bola terhadap poligon untuk mendapatkan offset. Setelah itu dilakukan pengurangan antara letak titik pusat bola dengan *offset*, dan menghasilkan titik perpotongan bola dengan poligon tepat di depan bola dengan arah *plane*. Hal yang dilakukan ini seringkali disebut memproyeksikan titik pusat bola pada *plane*.

3. Setelah mendapatkan titik perpotongan, titik tersebut dicek apakah berada di dalam atau di luar poligon. Apabila titik terletak di luar poligon, maka dilanjutkan ke langkah keempat.
4. Meskipun pada langkah sebelumnya telah didapat bahwa titik perpotongan bola berada di luar poligon, namun masih ada kemungkinan poligon tersebut bertabrakan dengan bola karena radius bola. Pengecekan terakhir ini digunakan untuk menemukan titik pada setiap *edge* poligon yang terdekat dengan titik pusat bola. Apabila poligon berupa segitiga, untuk setiap sisinya dapatkan vektor *edge*, dan dicari titik terdekat pada garis tersebut terhadap titik pusat bola. Setelah mendapatkan titik terdekat, dilakukan perhitungan antara jarak titik tersebut dengan pusat bola. Jika jarak kurang dari radius bola, maka terjadi tabrakan.

**PERANCANGAN PERANGKAT LUNAK**

Sebelum melakukan pembuatan perangkat lunak, terlebih dulu dilakukan perancangan dengan menggunakan *flowchart*. Secara keseluruhan perangkat lunak mempunyai empat tahap utama. Tahap-tahap tersebut yaitu: memilih obyek yang hendak digambar, menggambar obyek, mengubah menjadi gambar 3D dan pergerakan dalam mode 3D. Berikut ini adalah *flowchart* untuk sistem secara keseluruhan.



**Gambar 3. Flowchart Sistem Keseluruhan**

Pada tahap pemilihan obyek, pengguna memilih obyek yang hendak digambar. Pemilihan obyek dapat dilakukan dengan menekan tombol menu yang telah tersedia. Obyek yang dapat digambar adalah: dinding, lantai, pintu, jendela dan tangga. Setelah memilih obyek yang akan digambar, pengguna dapat menggambar pada *form* yang tersedia. Masing-masing obyek mempunyai properti yang berbeda.

Pada obyek dinding serta lantai, properti yang dimasukkan adalah titik awal, panjang, ketebalan serta ketinggian dinding. Titik awal dibuat dengan mengklik pada bidang gambar. Kemudian untuk menentukan panjangnya dengan menarik mouse secara vertikal dan untuk ketebalan dengan menggeser mouse ke arah horisontal. Ketinggian di input dengan menggunakan *form* inputan.

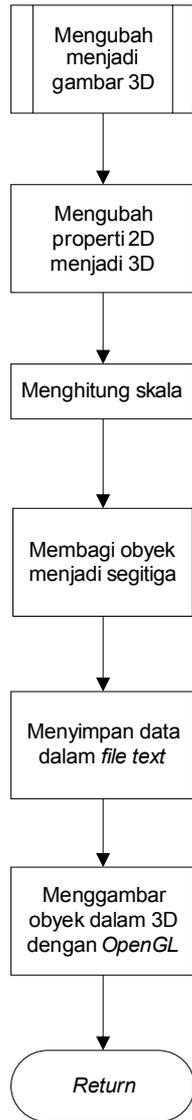
Pintu dan jendela, selain mempunyai properti seperti dinding dan lantai, terdapat syarat lain yaitu harus terletak pada dinding. Untuk membuat obyek jendela, pengguna juga harus memasukkan data ketinggian jendela dari lantai melalui *form* inputan.

Obyek tangga mempunyai properti sama dengan dinding yaitu titik awal tangga, panjang tangga di input dengan menggeser *mouse* secara horisontal, ketebalan setiap anak tangga serta tinggi tangga secara keseluruhan diinput melalui *form* inputan.

Tahap ketiga adalah memvisualisasikan gambar dan data dari denah 2D menjadi ruang 3D. Tahap ini mempunyai beberapa langkah yaitu:

- a. Mengubah properti 2D menjadi 3D dengan memperhitungkan ketinggian yang sudah diinputkan pada saat pembentukan obyek. Pada tahap ini properti obyek yang meliputi data posisi dalam bentuk 2D (x,y), diubah menjadi bentuk 3D (x,y,z) dengan cara koordinat y pada 2D menjadi koordinat z pada 3D, ketinggian pada 2D menjadi koordinat y pada 3D, sedangkan koordinat x tetap.
- b. Menghitung skala. Pada tahap ini nilai koordinat yang semula dalam satuan *centimeter* diubah menjadi satuan unit dalam *OpenGL*.
- c. Membagi obyek menjadi segitiga. Untuk menggambar suatu obyek tidak digambar apa adanya, melainkan dibentuk dari segitiga-segitiga. Perubahan menjadi bentuk segitiga ini dimaksudkan untuk mempermudah dalam melakukan teknik *collision* nantinya.
- d. Menyimpan dalam *file* teks. Semua data *vertex* obyek kemudian disimpan dalam *file* teks.
- e. Menggambar obyek 3D dengan *OpenGL*. Obyek digambar menggunakan perintah-perintah *OpenGL*. Penggambaran didasarkan pada *file* teks yang berisi data *vertex* semua obyek.

Pada Gambar 4 terdapat *flowchart* untuk memvisualisasikan denah berupa data 2D menjadi 3D.

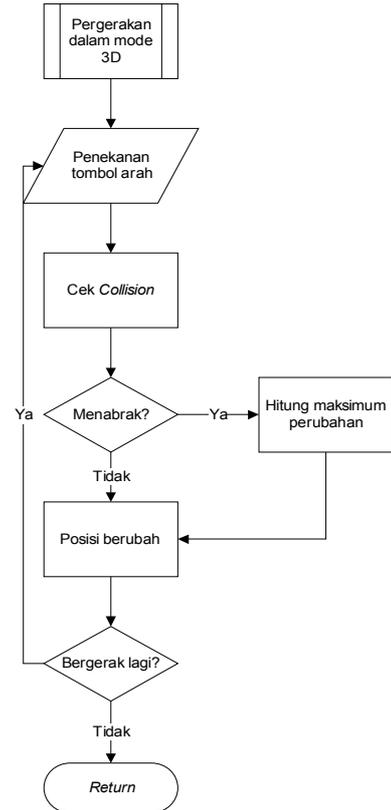


Gambar 4. Flowchart Perubahan Data 2D ke 3D

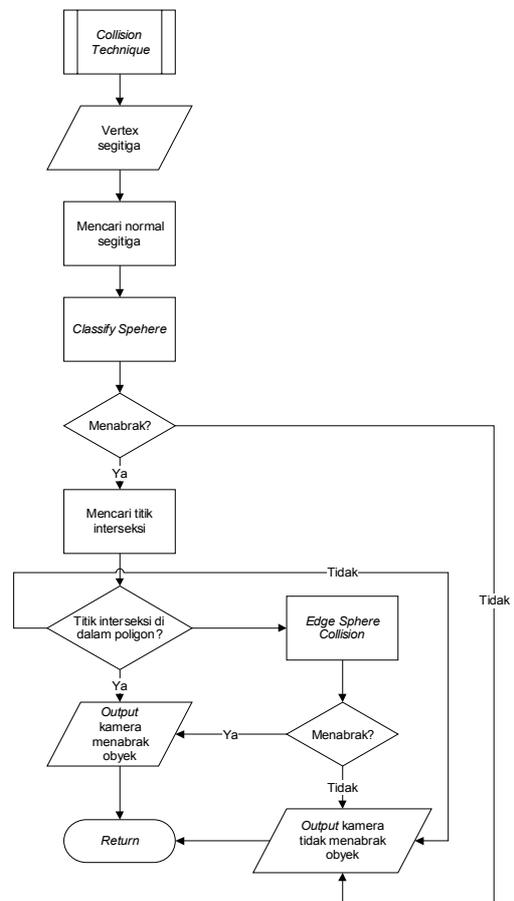
Tahap terakhir yaitu tahap pergerakan dalam mode 3D. Adapun tahap ini juga terbagi menjadi beberapa langkah yaitu :

- Penekanan tombol arah. Pada tahap ini pengguna diminta untuk melakukan penekanan tombol arah (atas, bawah, kiri, kanan).
- Pengecekan collision. Tahap ini melakukan pengecekan, apakah kamera menabrak obyek atau tidak.
- Penghitungan perubahan maksimum. Setelah diketahui bahwa akan terjadi tabrakan, program akan menghitung batas maksimum kamera dapat maju.
- Perubahan posisi. Merubah posisi kamera sesuai dengan arah yang dimasukkan.

Berikut ini ditampilkan flowchart untuk pergerakan dalam mode 3D pada Gambar 5 serta pengecekan collision pada Gambar 6.



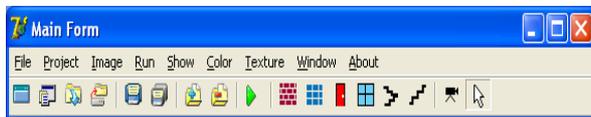
Gambar 5. Flowchart Pergerakan Dalam Mode 3D



Gambar 6. Flowchart Teknik Collision

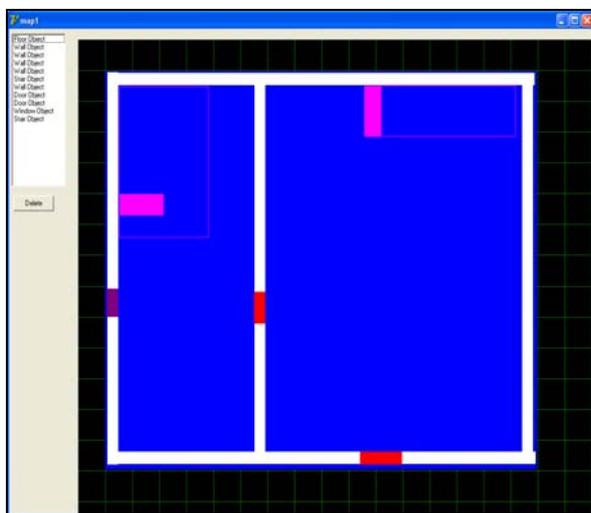
## PEMBUATAN DAN HASIL PERANGKAT LUNAK

Untuk pembuatan perangkat lunak digunakan bahasa pemrograman Delphi 7 dengan bantuan *library OpenGL*. *OpenGL* adalah sebuah *library* yang digunakan sebagai suatu *interface* untuk *hardware* grafis [8]. Pada intinya, *OpenGL* adalah sebuah grafis 3D dan *modelling library* yang *portable* dan sangat cepat. Dengan *OpenGL* dapat dibuat grafis 3D yang indah dan hampir mendekati kualitas visual *ray tracing*. *OpenGL* menggunakan algoritma-algoritma yang dibuat dan dioptimasi oleh Silicon Graphics Inc. (SGI), yang dikenal sebagai pemain utama dalam komputer grafis dan animasi. Berikut ini ditampilkan hasil pembuatan perangkat lunak.



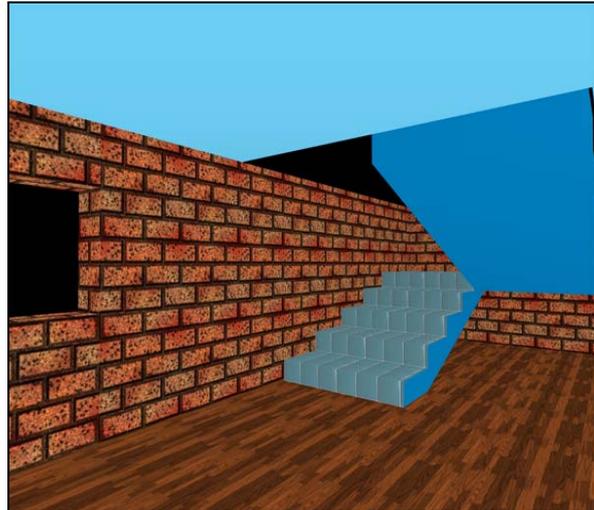
Gambar 7. Menu Utama Perangkat Lunak

Pada menu utama, dapat dibuat *project* yang merupakan denah ruang yang akan di-desain. Tampilan desain denah ruang adalah secara dua dimensi, dapat terlihat seperti Gambar 8 di bawah ini.

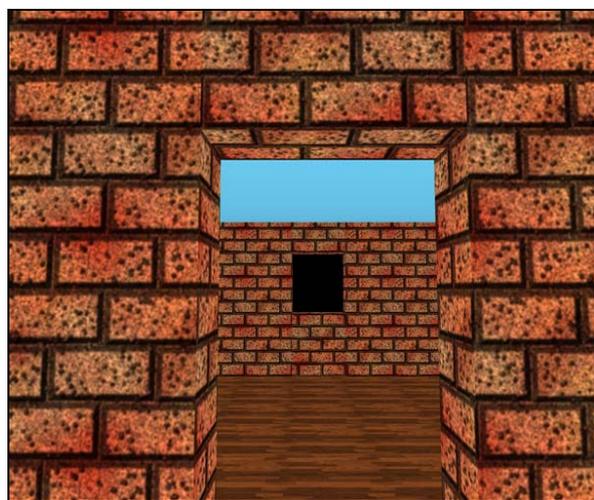


Gambar 8. Denah Ruang 2D

Setelah selesai melakukan desain ruang, maka hasil desain dapat divisualisasikan secara tiga dimensi dan pada visualisasi ini pengguna dapat melakukan pergerakan dengan menggunakan tombol *keyboard* sehingga seolah-olah pengguna bergerak memasuki ruangan tersebut. Sebagai contoh hasil visualisasi tiga dimensi ditampilkan pada Gambar 9.



Gambar 9. Visualisasi dari Denah 2D pada gambar 8



Gambar 10. Beberapa Model 3D Dari Denah 2D

## KESIMPULAN

Perangkat lunak yang dibuat telah dapat memenuhi tujuan pembuatan yaitu untuk membuat denah ruang secara dua dimensi dan ditampilkan dalam bentuk tiga dimensi yang memungkinkan pengguna menjelajah ruang yang dibuat.

Faktor utama yang menentukan kecepatan perangkat lunak ini adalah *graphic card*, dimana makin baik *graphic card* yang digunakan, maka semakin halus gambar yang dihasilkan. Lebih baik digunakan *graphic card* yang mempunyai fasilitas *anti-aliasing*. Faktor penentu kecepatan yang lain adalah jenis prosesor serta kapasitas memori.

Agar sesuai dengan denah dua dimensi yang dibuat, diperlukan perhitungan dengan menggunakan skala agar semua obyek dalam bentuk tiga dimensi yang dihasilkan mempunyai ukuran yang proporsional.

Teknik *collision* penting untuk diaplikasikan guna mendeteksi keberadaan obyek, sehingga dapat terjadi efek tabrakan terhadap obyek yang bersifat *solid*.

Sebagai pengembangan lebih lanjut, dapat dilakukan penambahan obyek-obyek lain seperti perabot ruangan.

## DAFTAR PUSTAKA

1. Angel, E., *OpenGL: a primer*. New York: Addison-Wesley. 2002.
2. Anton, H., *Aljabar Linear Elementer 5<sup>th</sup> ed.* Jakarta: Erlangga. 1987.
3. Borland Software Corporation. *OpenGL with Delphi (<http://www.borland.com/>)*. February 2, 2005.
4. Hill, F. S., Jr., *Computer Graphics Using OpenGL 2<sup>nd</sup> ed.* New Jersey: Prentice Hall. 2001.
5. Harrington, S., *Computer Graphics a Programming Approach 2<sup>nd</sup> ed.* Singapore: McGraw-Hill. 1987.
6. Humphrey, B., *Camera World Collision (<http://www.gametutorials.com>)*. February 2, 2005.
7. Neider, J., Davis, T., *OpenGL Programming Guide (<http://www.aw.com/devpress/>)*. February 1, 2005.
8. Wright, R. S., & Lipchak, B., *OpenGL Superbible 3<sup>rd</sup> ed.* Indianapolis: Sams Publishing. 2005.
9. Xerox Palo Alto Research Center. *Graphics Gems*. San Diego: Academic Press, Inc. 1990.