

APLIKASI VIRTUAL TOUR 3D PABRIK PT. X

Willy Nugraha Utomo¹, Liliana^{1*}, Kartika Gunadi¹

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra

Jl. Siwalankerto 121-131, Surabaya 60236

Telp (031) 2983455, Fax. (031) 8417658

E-mail: willymiiee@gmail.com, lilian@petra.ac.id, kgunadi@petra.ac.id

*Korespondensi penulis

Abstrak: PT. X merupakan sebuah perusahaan ternama yang memiliki karyawan berjumlah kurang lebih 1500 orang. Salah satu kegiatan rutin yang dilakukan PT. X yang ditujukan kepada para karyawan baru adalah orientasi, dimana terdapat sebuah acara yang mengajak para karyawan baru untuk berkunjung ke pabrik perusahaan. Kendala yang terjadi adalah lokasi pabrik perusahaan dengan kantor utama yang cukup jauh, sehingga kurang adanya efisiensi waktu dalam kegiatan tersebut. Aplikasi ini bertujuan untuk memfasilitasi PT. X dalam melakukan kegiatan orientasi karyawan baru sehingga lokasi pabrik dapat dilihat secara virtual. Hal yang dilakukan terlebih dahulu dalam proses pembuatan aplikasi adalah melakukan kunjungan ke lokasi pabrik untuk mengetahui bentuk dan posisi objek, sehingga diharapkan hasil yang ditampilkan dapat menyerupai kondisi aslinya. Objek 3d beserta teksturnya dibuat dengan menggunakan aplikasi Blender agar dapat diimpor ke dalam aplikasi yang dibuat. Pemberian fitur collision bertujuan agar user tidak dapat menembus objek. Posisi, bentuk dan objek yang terdapat pada aplikasi telah menyerupai keadaan asli yang terdapat pada lokasi pabrik. Hambatan utama dalam pembuatan aplikasi ini adalah waktu pembuatan, sehingga tidak dapat membuat peta pabrik secara detail. Gedung utama, *stage* dan *warehouse* telah dibuat, hanya objek-objek kecil yang terdapat di dalam gedung utama belum sempat terlengkapi.

Kata kunci: Objek 3d, *collision*, *WebGL*

Abstract: *Company X is a reputable company that has more than 1000 people employees. One of the routines that performed by company X is orientation for new employees, and there is an event that invites them to visit the factory. The location of the main office and the factory is far, so this activity lacks time efficiency. This application aims to facilitate company X in conducting new employee orientation so that the location can be seen in the virtual world. The first thing in the process of making the application is visiting the factory to determine the shape and position of the object, so that the expected results are shown to resemble its original condition. 3d object and its texture are created using Blender application that can be imported into the made application. Collision feature aims to prevent users from penetrating the objects. The position, shape and location of objects contained in the application already resemble the original plant site. The main obstacle in making this application is the creation time, so the detailed map of the plant cannot be completed. The main building, the production building and the warehouse have been made, only small objects contained in the main building have not been contained.*

Keywords: *3d object, collision, WebGL*

PENDAHULUAN

PT. X merupakan sebuah perusahaan yang bergerak di bidang struktur bangunan. Perusahaan tersebut telah berdiri sejak tahun 1986, dengan wilayah pabrik kurang lebih seluas 72.000 m² dan memiliki karyawan dengan jumlah total 1000 orang. Produk yang dihasilkan adalah struktur bangunan dengan kapasitas produksi total mencapai 100.000 ton/tahun. PT. X sangat peduli terhadap proses perekrutan karyawan, dan membekali karyawan-karyawannya dengan pengetahuan dasar tentang perusahaan seperti visi dan misi, filosofi, motto, dan

quality policy lewat kegiatan-kegiatan rutin, agar setiap karyawan dapat memiliki pandangan yang sama dan bersama-sama bergerak untuk mengembangkan dan memajukan perusahaan ke arah yang lebih baik.

Salah satu kegiatan rutin yang diadakan oleh perusahaan yang ditujukan kepada para karyawan baru adalah orientasi, dan dalam kegiatan orientasi tersebut terdapat satu acara yang mengajak setiap karyawan baru untuk melakukan kunjungan ke pabrik perusahaan. Permasalahan yang terjadi adalah, dikarenakan lokasi pabrik yang cukup jauh maka untuk melakukan perjalanan dari lokasi kantor utama

ke lokasi pabrik membutuhkan waktu yang cukup lama. Hal ini menunjukkan kurang adanya efisiensi waktu dalam kegiatan orientasi karyawan baru, khususnya dalam rangkaian acara berkunjung ke lokasi pabrik perusahaan.

Dengan adanya aplikasi *virtual tour*, diharapkan agar PT. X dapat menggunakan aplikasi ini sebagai fasilitas dan sarana untuk memberikan gambaran peta pabrik secara virtual kepada para karyawan baru yang menjadi peserta orientasi, sehingga peserta orientasi tidak perlu berkunjung ke lokasi pabrik secara langsung untuk mengetahui kondisi pabrik. Hal ini dapat meningkatkan efisiensi waktu dari program orientasi karyawan baru, dan juga dapat meningkatkan nilai perusahaan di mata konsumen apabila terdapat konsumen yang melakukan kunjungan survei ke area pabrik.

DASAR TEORI

Virtual Reality

Virtual reality adalah sebuah teknologi yang dapat membuat pengguna merasa berada di dalam sebuah lingkungan *virtual* dengan menggunakan bantuan dari *hardware* dan *software* komputer. Pada umumnya, *virtual reality* hanya mengutamakan pengalaman visual yang ditampilkan lewat layar komputer atau melalui display stereoskopik khusus, tetapi beberapa simulasi juga memiliki beberapa informasi sensorik tambahan, contohnya seperti suara melalui *speaker* atau *headphone*. *Virtual reality* juga dapat meliputi komunikasi jarak jauh yang dapat menampilkan kehadiran *user* secara *virtual* melalui penggunaan dari perangkat *input* standar seperti *keyboard* dan *mouse*, atau melalui perangkat lainnya seperti *polhemus (wired glove)*. Lingkungan yang disimulasi oleh *virtual reality* dapat dibuat menyerupai dunia nyata, contohnya seperti simulasi untuk pilot atau pelatihan untuk pertempuran, atau dapat berbeda dari kenyataan secara signifikan, contohnya seperti di dalam game yang berbasis *virtual reality*. *Virtual reality* sering digunakan dalam pembuatan berbagai macam aplikasi, contohnya seperti aplikasi yang berfungsi sebagai terapi kecemasan, aplikasi untuk pelatihan seperti *flight simulator* dan *ship simulator*, *multiplayer online gaming* seperti *World of Warcraft* dan *Second Life*, *Nintendo Wii*, dan *virtual tour*.

Virtual Tour

Virtual tour adalah simulasi dari sebuah lokasi yang sesungguhnya, umumnya terdiri oleh *sequence* video atau kumpulan foto [2]. *Virtual tour* juga dapat

menggunakan beberapa elemen *multimedia* lain, contohnya seperti *sound effect*, musik, narasi, dan teks, dan sering digunakan untuk menggambarkan varietas dari media yang berbasis video dan fotografi [4].

Panoramic tour dan *virtual tour* sering dianggap memiliki arti yang sama, yaitu sebuah *virtual tour* yang dibuat dengan menggunakan bantuan kamera, terdiri dari sejumlah foto yang diambil dari satu posisi yang strategis dengan cara kamera dan lensa diputar mengelilingi satu lokasi yang sering disebut dengan istilah *no-parallax point* (titik tepat di belakang lensa dimana cahaya bertemu).

Virtual World

Virtual world adalah lingkungan simulasi berbasis komputer yang identik dengan lingkungan virtual 3D yang interaktif, dimana *user* dapat mengambil bentuk sebagai *avatar* yang dibuatnya dan dapat dilihat oleh orang lain [3]. Umumnya konsep *virtual world* digunakan di dalam berbagai macam game, tetapi *virtual world* tidak terbatas pada game saja. Dalam bidang bisnis, *virtual world* telah mulai diaplikasikan dalam pelatihan untuk karyawan sehingga dapat mengatasi kendala-kendala seperti jarak, infrastruktur, biaya akomodasi dan jadwal yang ketat. Sebagai contoh, aplikasi *Second Life* telah digunakan di sekolah-sekolah bisnis. Isi dari pelatihan virtual tersebut umumnya menyerupai tutorial sederhana dan pengujian dari pengetahuan pengguna. Beberapa perusahaan dan tempat-tempat umum memperbolehkan akses virtual secara bebas pada fasilitas-fasilitas yang mereka miliki sebagai suatu alternatif untuk video maupun gambar.

Collision Detection

Collision biasanya mengacu pada proses mendeteksi persimpangan antara dua objek atau lebih. Topik yang sering dikaitkan dengan penggunaan fitur *collision detection* dalam *video game* dan simulasi fisik lainnya, juga dapat diaplikasikan dalam robotika. Selain menentukan apakah dua benda telah bertabrakan, fitur *collision detection* juga dapat menghitung waktu dari terjadinya *collision* tersebut [7].

WebGL

WebGL (Web Graphics Library) adalah API *JavaScript* untuk rendering grafis 3D interaktif dan grafis 2D dalam setiap *web browser* yang kompatibel tanpa menggunakan *plug-in* [9]. *WebGL* dirancang dan dikelola oleh perusahaan *non-profit*, Khronos Group [6].

WebGL terintegrasi ke semua *web browser* standar yang memungkinkan pengolahan gambar sebagai bagian dari halaman web. Elemen *WebGL* dapat dicampur dengan elemen *HTML* lain dan digabungkan dengan bagian lain dari sebuah halaman web atau latar belakang dari halaman web^[9]. *Scene WebGL* dapat dibuat tanpa pemrograman dengan menggunakan aplikasi pembuatan konten, contohnya seperti *Blender*. *Scene* yang telah dibuat kemudian diekspor ke dalam *WebGL*.

Babylon.js

Babylon.js adalah *library* JavaScript yang digunakan untuk membuat dan menampilkan animasi grafika komputer 3D di *Web browser* tanpa membutuhkan adanya *plug-in* tambahan. *Babylon.js* berjalan di semua *Web browser* yang didukung oleh *WebGL* dan mempunyai beberapa fitur utama yang menjadi faktor penting dalam pembuatan animasi 3D di *Web browser* dan akan digunakan dalam pembuatan skripsi ini, antara lain *Babylon engine*, *scene*, kamera, *light*, dan *renderer*.

Babylon engine merupakan komponen yang berperan untuk mengaktifkan *Babylon.js* ke dalam sebuah *canvas* yang telah didefinisikan terlebih dahulu. *Scene* merupakan objek utama di dalam *Babylon.js*, karena apabila kita menginginkan sebuah objek dapat ditampilkan di layar, maka kita harus menambahkan objek tersebut ke dalam *scene*. Untuk dapat *render* sesuatu, hal yang perlu dilakukan adalah menambahkan kamera ke dalam *scene*, sehingga *renderer* dapat mengetahui dari sudut pandang mana *renderer* tersebut harus *render* objek. Terdapat dua macam kamera di dalam *Babylon.js*, yaitu *Arc Rotate Camera* dan *Free Camera*. *Light* merupakan komponen *Babylon.js* yang berperan untuk memberikan efek cahaya pada *scene*. Terdapat tiga macam tipe *light* yang dimiliki oleh *Babylon.js*, yaitu *Point Light*, *Spot Light*, dan *Directional Light*. *Scene* tidak terbatas oleh jumlah, tetapi satu *renderer* hanya dapat menggambar satu *scene* dalam waktu yang bersamaan. *Renderer* tersebut menggambar segala sesuatu yang terdapat di dalam *scene* ke *canvas WebGL*.

DESAIN SISTEM

Aplikasi ini berperan sebagai *software* pendukung yang bertujuan untuk memperlihatkan peta pabrik PT. X secara nyata kepada *user*. Sebelum dilakukan perancangan aplikasi, diperlukan penelitian terhadap objek-objek dan gedung yang akan dibuat sehingga hasil yang ditampilkan dapat menyerupai keadaan aslinya. Desain sistem akan

membahas mengenai analisa dari permasalahan yang terjadi, arsitektur dari aplikasi, cara kerja aplikasi dan isi dari sistem yang dibuat.

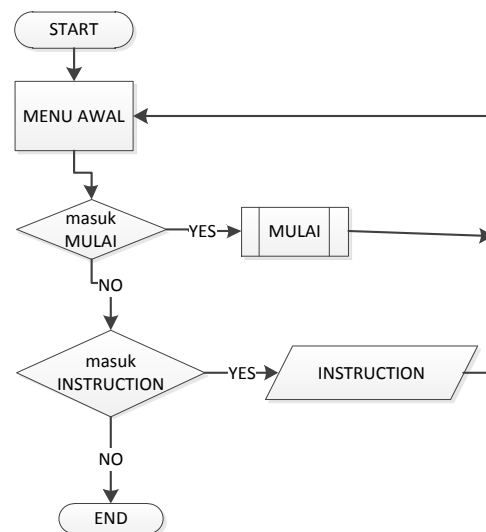
Analisa

Permasalahan utama dalam pembuatan aplikasi adalah proses pembuatan objek-objek yang terdapat di dalam *virtual tour* agar objek-objek tersebut dapat ditampilkan semirip mungkin dengan aslinya. Dilakukan penelitian terhadap pabrik PT. X, mulai dari denah pabrik, bangunan apa saja yang terdapat di lokasi, interior dari gedung utama dari pabrik PT. X, dan pengukuran terhadap gedung dan objek-objek yang terdapat di lokasi. Setelah hal-hal tersebut dilakukan, langkah selanjutnya adalah proses pembuatan objek-objek 3D menggunakan aplikasi pendukung, yaitu *Blender*. Terdapat beberapa permasalahan yang muncul pada saat proses pembuatan aplikasi *virtual tour*, antara lain:

1. Menentukan objek mana yang akan dibuat dan diberi tekstur menggunakan aplikasi *Blender* dan objek mana yang dapat dibuat dan diberi tekstur langsung menggunakan *Babylon.js*.
2. Mengetahui jenis file yang cocok dengan *Babylon.js* sehingga objek yang telah dibuat dengan aplikasi *Blender* dapat diimpor ke dalam *virtual tour*.
3. Memberi *collision detection* untuk setiap objek agar objek-objek tersebut tidak dapat ditembus.

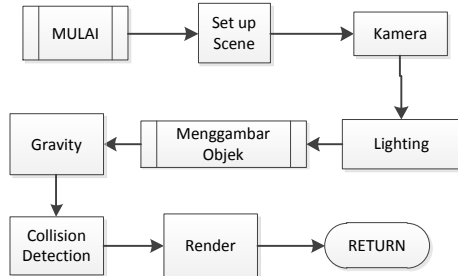
Flowchart sistem

Aplikasi yang dibuat memiliki beberapa tahapan berupa proses-proses yang berkelanjutan. Di bagian ini dibuatlah *flowchart* untuk memperjelas tahapan atau proses-proses yang terdapat di dalam aplikasi.



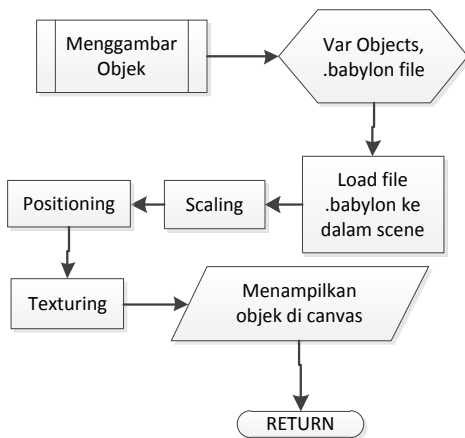
Gambar 1. Garis besar aplikasi

Ketika user menjalankan aplikasi, user dihadapkan pada menu awal yang dibagi menjadi 2 bagian, yaitu Mulai dan Instruction. Bagian Instruction menjelaskan tentang cara menggunakan aplikasi ini, yaitu tombol-tombol apa saja yang digunakan untuk kontrol user. Bagian Mulai digunakan untuk memulai aplikasi.



Gambar 2. Diagram alir proses mulai

Setelah user memilih menu Mulai, proses pertama yang dijalankan adalah proses pembuatan scene, dilanjutkan dengan proses penambahan kamera dan lighting. Langkah selanjutnya adalah menggambar objek-objek yang telah dipersiapkan sebelumnya ke dalam scene, dan menambahkan fitur gravity dan collision detection pada objek-objek tersebut.



Gambar 3. Diagram alir proses menggambar objek

Dalam proses menggambar objek, langkah pertama yang dilakukan adalah mempersiapkan objek yang akan dipanggil ke dalam scene dalam format .babylon. Setelah itu, dilakukan proses scaling dan positioning, yaitu menyesuaikan ukuran objek sehingga sesuai dengan yang diinginkan dan meletakkan objek tersebut pada koordinat yang telah ditentukan.

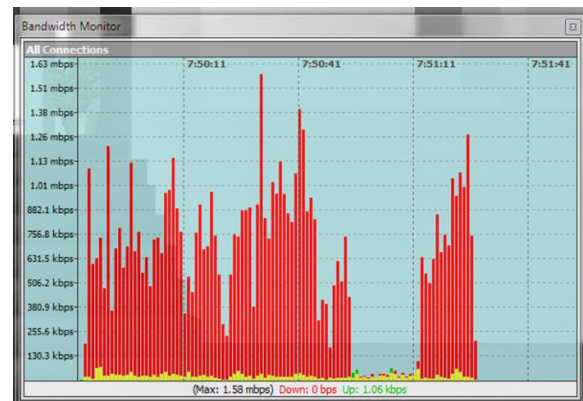
PENGUJIAN SISTEM

Pada bab ini dilakukan uji coba mengenai fitur-fitur yang disediakan pada aplikasi yang telah dibuat. Uji coba dilakukan untuk memastikan aplikasi dapat

berjalan dengan baik dan tanpa adanya bug atau error. Pada bab ini juga dilakukan perbandingan dengan foto-foto yang telah diambil di lokasi pabrik untuk menguji kemiripan aplikasi dengan keadaan nyata lokasi. Skripsi ini dibuat berdasarkan data-data yang diperoleh setelah melakukan penelitian terhadap pabrik PT. X.

Pengujian penggunaan bandwidth

Pengujian ini menggunakan bantuan dari aplikasi lain yaitu Bandwidth Monitor. Langkah pertama adalah mengunggah semua file yang merupakan bagian dari aplikasi ke dalam sebuah situs, lalu membuka situs tersebut menggunakan sebuah browser sambil mengaktifkan aplikasi Bandwidth Monitor. Detail dari aplikasi tersebut dapat dilihat pada Gambar 4.



Gambar 4. Detail tampilan aplikasi Bandwidth Monitor

Berdasarkan dari data yang ditampilkan oleh aplikasi Bandwidth Monitor, grafik berwarna merah mencapai kurang lebih 1.63 mbps ketika membuka situs tersebut, dengan catatan situs tersebut belum pernah dibuka sebelumnya sehingga tidak ada cache yang tersimpan di dalam browser, dan grafik berwarna merah merupakan jumlah data download.

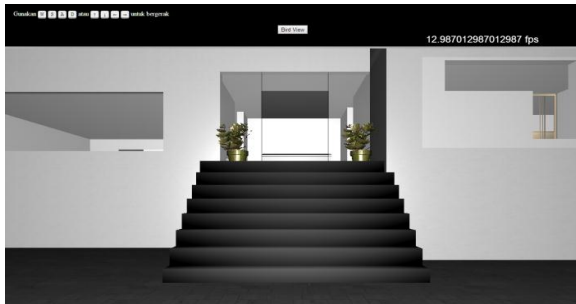
Pengujian frame per second (fps) rate

Untuk dapat melakukan pengujian fps rate, disediakan dua perangkat komputer yang memiliki spesifikasi yang berbeda, yaitu seperti Tabel 1.

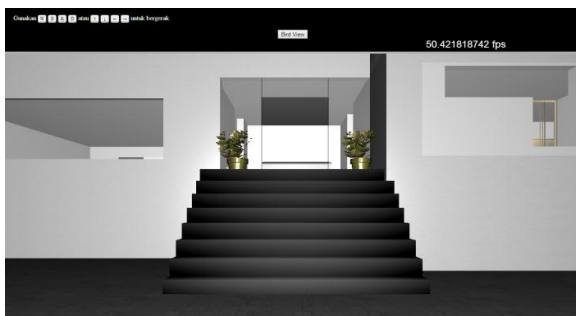
Tabel 1. Perbandingan perangkat A dan perangkat B

	Perangkat A	Perangkat B
OS	Windows 7 Home Premium (Service Pack 1)	Windows XP Professional Version 2002 (Service Pack 2)
Processor	Intel ® Core™ i5 CPU M 430 @ 2.27 GHz	Intel ® Core™ i3 CPU M 540 @ 3.07 GHz
VGA	Nvidia GeForce® 310M 1 GB	Onboard
RAM	1.86 GB	2.86 GB

Berikut adalah tampilan dari hasil pengujian *fps rate* di masing-masing perangkat seperti pada Gambar 5 dan 6.



Gambar 5. Pengujian *fps rate* pada perangkat B

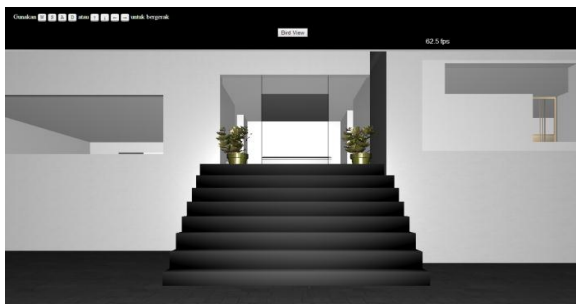


Gambar 6. Pengujian *fps rate* pada perangkat A

Seperti yang dapat dilihat pada Gambar 5 dan Gambar 6, dapat dikatakan bahwa *fps rate* di perangkat A yang memiliki nilai 50.42 *fps*, lebih baik dari perangkat B dengan nilai 12.98 *fps*.

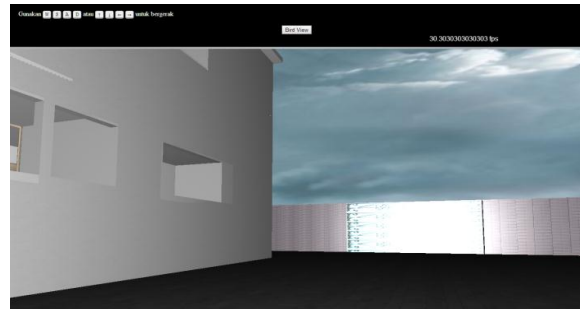
Pengujian navigasi

Ketika user masuk ke dalam menu *Start*, maka visualisasi dalam aplikasi ini akan dimulai. Tampilan pertama yang dilihat oleh user adalah seperti yang dapat dilihat pada Gambar 7, yaitu lokasi tepat di depan gedung utama.



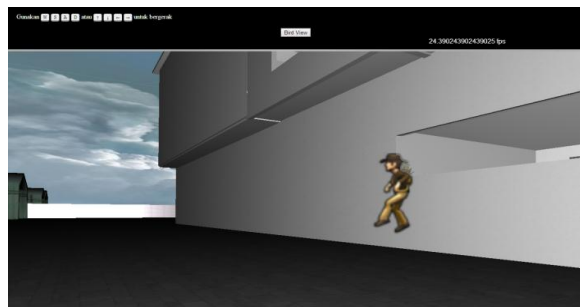
Gambar 7. Tampilan awal visualisasi

Ketika user memasuki bagian visualisasi, user dapat menggerakkan *mouse* sebagai sarana navigasi di dalam aplikasi ini. Gambar 8 menunjukkan apabila user menggerakkan *mouse* ke arah kanan.



Gambar 8. Tampilan ketika user menggerakkan *mouse* ke arah kanan

Apabila user menggerakkan *mouse* ke arah kiri, maka tampilan aplikasi menjadi seperti yang dapat dilihat pada Gambar 9.

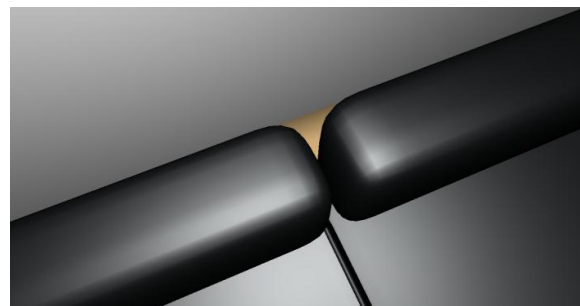


Gambar 9. Tampilan ketika user menggerakkan *mouse* ke arah kiri

Selain menggunakan *mouse*, user juga dapat menggunakan *keyboard* untuk sarana navigasi, yaitu tombol W, A, S, D atau tombol arah panah atas, bawah, kiri, dan kanan.

Pengujian *Collision Detection*

Pada bagian ini akan dilakukan pengujian terhadap fitur *collision detection* yang diberikan untuk masing-masing objek di dalam aplikasi. Seperti yang dapat dilihat pada Gambar 10, objek yang akan dijadikan contoh adalah objek sofa, dan dalam gambar tersebut fitur *collision detection* belum diaktifkan sehingga kamera dapat menembus objek tersebut.



Gambar 10. Kamera dapat menembus sofa

Setelah fitur *collision detection* diaktifkan, tampilan dapat dilihat seperti pada Gambar 11. Kamera tidak dapat menembus objek sofa tersebut.



Gambar 11. Kamera tidak dapat menembus sofa

KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan beberapa hal sebagai berikut:

- Engine yang cocok dan mudah digunakan untuk pembuatan aplikasi virtual 3D adalah *Babylon.js*.
 - Ukuran dari aplikasi dapat dimasukkan ke dalam kategori ringan, dengan total sebesar 14.5 MB.
 - Untuk dapat membuka aplikasi, dibutuhkan *browser* yang *men-support* komponen *WebGL*.
 - Diperlukan spesifikasi VGA dan RAM yang cukup baik untuk perangkat komputer yang digunakan untuk membuka aplikasi, agar *fps rate* yang didapatkan ketika membuka aplikasi dapat mencapai nilai yang baik (kurang lebih 50 *fps*).
- Terdapat beberapa kekurangan pada detail objek di dalam gedung utama yang terdapat di dalam aplikasi, contohnya pada bagian pigura foto yang terdapat di dalam ruangan *meeting*.
 - Fitur pencahayaan yang masih kurang sempurna, tidak terlihat adanya bayangan dari objek yang lebih detail.

DAFTAR PUSTAKA

- [1] Catuhe, David; Rousset, David. *Babylon.js* <<https://github.com/BabylonJS/Babylon.js>>, 2013.
- [2] Colasante, Meg. *Nicola Building Virtual Tour; Considering simulation in the equity of experience concept*. 2011.
- [3] Cook, A.D. *A case study of the manifestations and significance of social presence in a multi-user virtual environment*. 2009.
- [4] Jan, Dusan; Rogue, Antonio; Leuski, Anton; Morie, Jacki; Traum, David. *A Virtual Tour Guide for Virtual Worlds*. 2009.
- [5] Elliot, Ian. *Babylon.js - A WebGL Game Engine From Microsoft*. 2013.
- [6] Khronos.org. *WebGL - OpenGL ES 2.0 for the Web*. 2011.
- [7] Mohler, B.; Raffin, B. Saito, H.; Standt, O. *Ray-Traced Collision Detection: Interpenetration Control and Multi-GPU Performance*. 2013.
- [8] Parisi, Tony. *WebGL: Up and Running - Building 3D Graphics for the Web*. 2012.
- [9] Tavares, Gregg. *WebGL Fundamentals*, 2012.