

Laporan Tugas 2 IF3260 Grafika Komputer

3D WebGL Hollow Object



Oleh K03 - Kelompok 16:

Adzka Ahmadetya Zaidan	13520127
Steven Gianmarg H. Siahaan	13520145
Daffa Romyz Aufa	13520162

Program Studi Teknik Informatika - Institut Teknologi Bandung
Jl. Ganesha 10, Bandung 40132

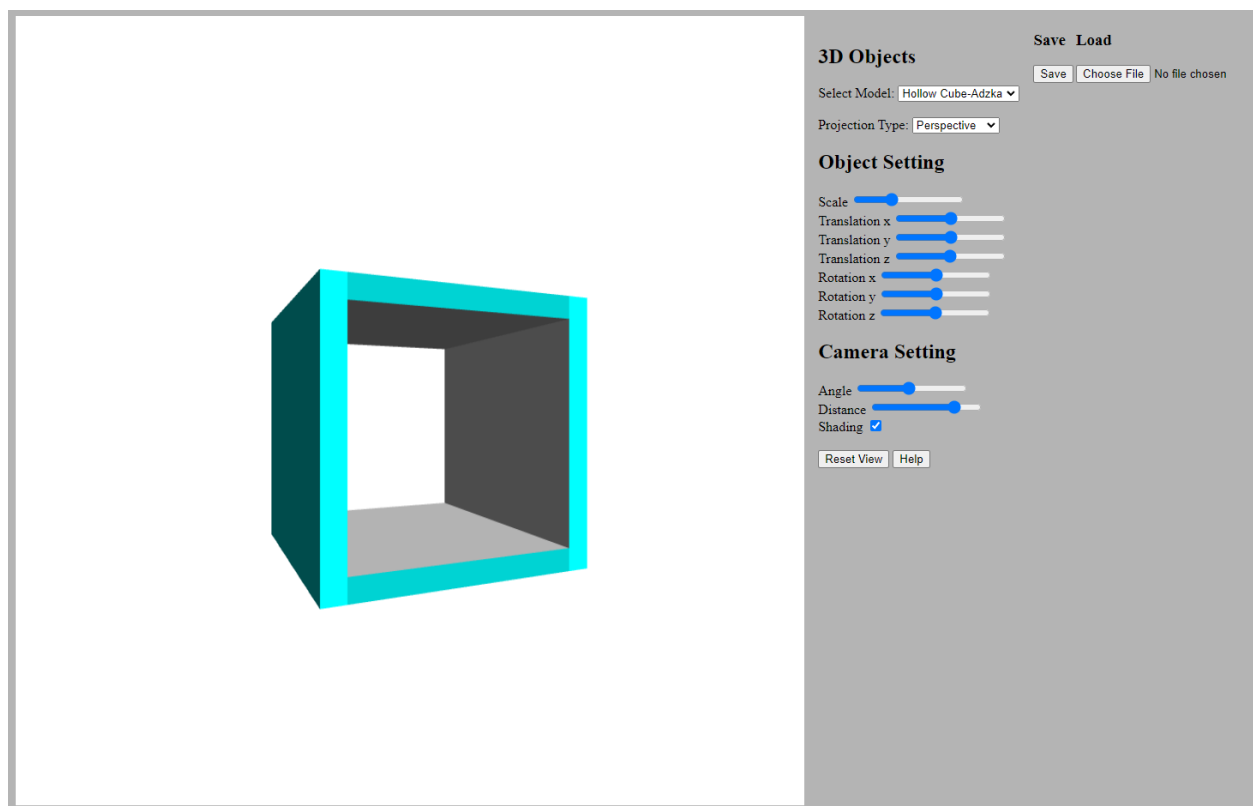
Daftar Isi

Daftar Isi	2
Deskripsi Program	3
Hasil Program	4
Manual/Contoh Fungsionalitas	6

Deskripsi Program

Program ini merupakan penggunaan WebGL 3D yang menampilkan objek berrongga 3D beserta proyeksinya. Program dapat melakukan transformasi objek *translation*, *rotation*, dan *scaling*, serta geometry dan projection. Model yang dibuat terdiri dari *hollow cube*, *pyramid*, dan prisma. Informasi model berisi nama, daftar koordinat vertex, vertex normal, dan warna setiap polygon. Program juga dapat membuka sebuah file berisi informasi model dalam bentuk `.json`.

Selain itu, program ini juga dilengkapi dengan interaksi untuk *view model* sehingga pengguna dapat mengubah jenis proyeksi untuk menampilkan semua objek. Proyeksi dapat dilakukan secara *orthographic*, *oblique* atau *perspective*. Selain itu, pengguna dapat mengubah jarak *view* kamera untuk mendekat atau menjauh dari model serta menggerakkan kamera untuk mengitari model. Program ini juga dilengkapi dengan fitur *reset* ke *default view* dan menu *help* yang memudahkan pengguna baru untuk dapat melakukan operasi di atas. Program ini juga memiliki teknik *shading* untuk menambahkan warna dasar pada model serta pencahayaan berupa *directional light*.



Tampilan program 3D WebGL Hollow Object

Hasil Program

Program yang kami buat berupa file `index.html`, `style.css`, dan `main.js`. File `index.html` berfungsi sebagai halaman web yang menampilkan objek 3D berbentuk hollow dengan teknologi WebGL. File ini berisi struktur dasar HTML beserta tag dan atribut yang digunakan untuk menampilkan halaman web tersebut. File tersebut menampilkan hasil model pada *canvas* di html serta menerima input dari pengguna. File `index.html` menggunakan `style.css` untuk mengatur tampilan visual dari halaman web. Selain itu, `index.html` juga memiliki `main.js` sebagai script untuk menangani proses keseluruhan program 3D WebGL.

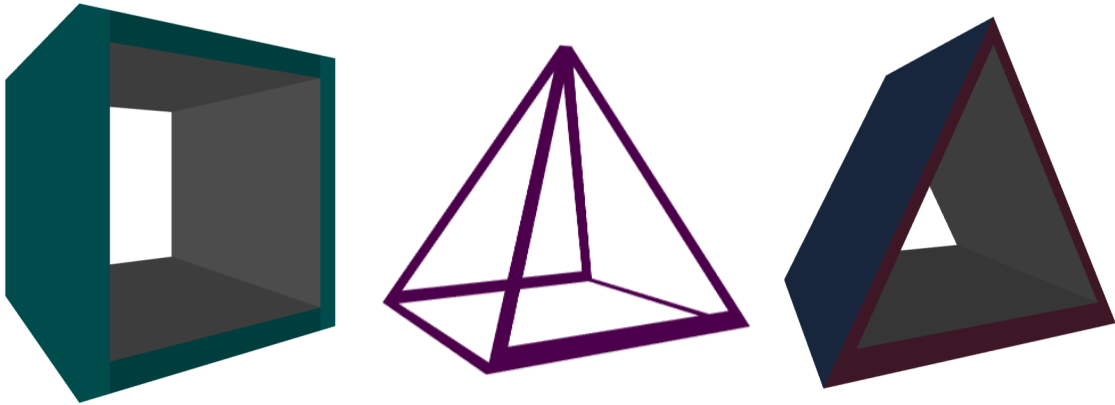
File `main.js` digunakan untuk mengatur dan menjalankan proses rendering objek 3D dalam lingkungan WebGL. Dalam file ini terdapat beberapa fungsi dan variabel yang digunakan untuk menginisialisasi dan mengatur properti *canvas*, data buffer objek 3D, program WebGL, posisi kamera, proyeksi objek 3D, serta proses menggambar objek 3D. Input dan pengaturan objek dan camera berupa translasi, rotasi, scaling, reset, save, load, camera distance, camera angle ditangani oleh `main.js`.

Di awal, file `main.js` akan menginisialisasi dan mengatur properti *canvas* dan WebGL context dengan memanggil fungsi `check` dan mengatur ukuran *canvas* sesuai ukuran layar perangkat pengguna. Selanjutnya, file akan membuat dan mengatur data buffer yang berisi posisi, warna, dan normal dari objek 3D yang akan dirender. Kemudian, program WebGL akan dibuat dengan mengkompilasi dan menghubungkan vertex dan fragment shader.

Selain itu, file `main.js` juga akan mengatur beberapa properti WebGL seperti depth test, clear color, dan viewport. Terdapat juga fungsi untuk menggambar objek 3D dengan memasukkan matriks proyeksi, matriks model, serta index start dan end dari buffer yang akan digambar. Untuk mengatur posisi kamera, file ini membuat variabel `distance` dan `view_matrix` yang akan digunakan dalam proses rendering objek 3D. File `main.js` juga mengatur proyeksi objek 3D dengan matriks *perspective*, *orthographic*, atau *oblique* tergantung dari jenis proyeksi yang dipilih oleh pengguna. Dalam proses menggambar objek 3D, fungsi `draw` akan dipanggil dengan memasukkan matriks proyeksi, matriks model, serta index start dan end dari buffer yang akan digambar.

File `index.html`, `style.css`, dan `main.js` disimpan pada folder `src`. Terdapat juga folder `test` yang berisi file contoh model dalam bentuk `.json` yang dapat di-load pada program kami.

Model yang telah kami buat tersimpan sebagai informasi dalam variabel di main.js berupa koordinat setiap vertex, vertex normal, serta vertex color. Model yang dibuat tersebut berupa kubus berrongga, piramid, serta prisma berrongga sebagai berikut:

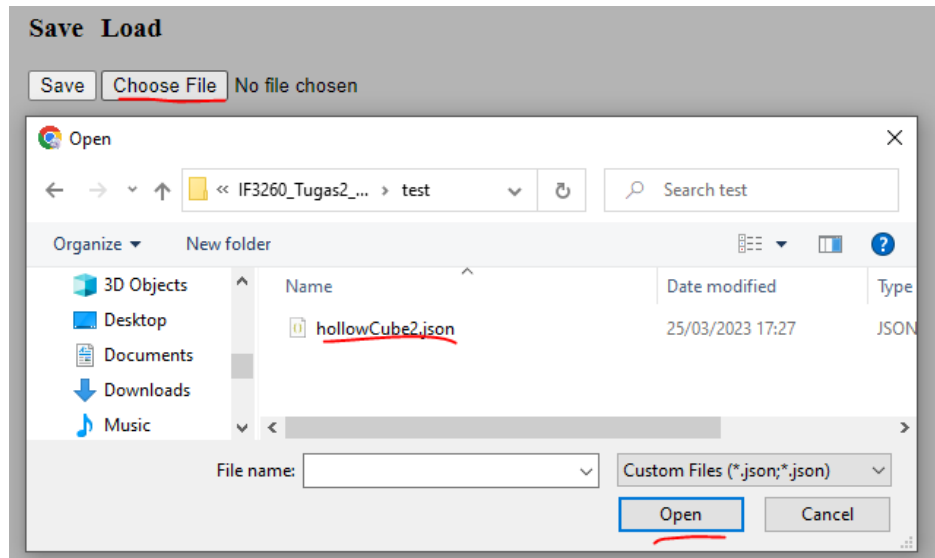


3 model 3D *hollow object* yang telah dibuat

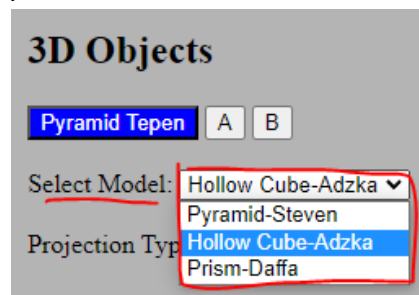
Manual/Contoh Fungsionalitas

Secara umum, berikut merupakan deskripsi manual untuk melakukan fungsionalitas yang telah diimplementasikan:

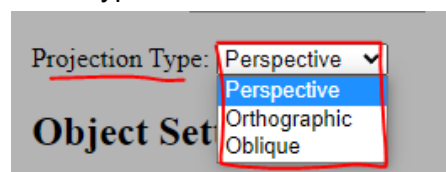
- Load file model atau hasil save: Klik tombol "Choose File" pada bagian "Load" untuk membuka file model yang sudah tersimpan dalam format JSON.



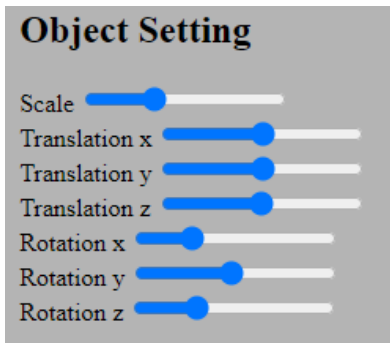
- Memilih model yang ingin ditampilkan: Pilih model yang ingin ditampilkan dengan menggunakan menu *drop-down* "Select Model".



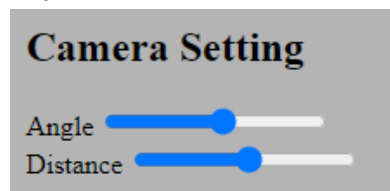
- Mengubah jenis proyeksi: Pilih jenis proyeksi yang diinginkan dengan menggunakan menu *drop-down* "Projection Type".



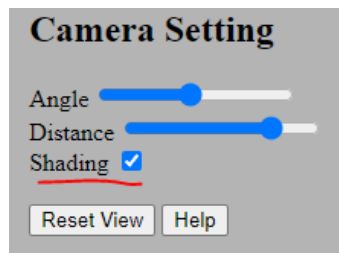
- Melakukan rotasi, translasi dan *scaling* dari objek: Gunakan *slider* "Scale", "Transition x", "Transition y", "Transition z", "Rotation x", "Rotation y", dan "Rotation z" pada menu control bagian "Object Setting" untuk melakukan rotasi, translasi dan *scaling* dari objek yang dipilih.



- Mengubah jarak kamera view: Gunakan slider "Angle" pada menu *control* bagian "Camera Setting" untuk menggerakkan kamera mengitari scene, dan gunakan slider "Distance" untuk mengubah jarak dari kamera.



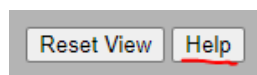
- ON/OFF shading: Ceklis *checkbox* "Shading" untuk menyalakan atau mematikan shading *directional light* pada objek.



- Me-reset ke default view: Tekan tombol "Reset View" untuk me-reset posisi objek dan kamera ke default view.



- Menu help: Tekan tombol "Help" untuk menampilkan penjelasan cara menggunakan aplikasi.



- Melakukan save setting model dan camera: Tekan tombol "Save", kemudian file akan tersimpan dengan nama "model.json" yang dapat di-*load* kembali pada program.

