LAPORAN TUGAS AKHIR

PERMAINAN RUBIK CUBE 3D MENGGUNAKAN OPENGL

Disusun sebagai salah satu tugas

mata kuliah Grafika Komputer



Patricia Joanne

140810160065

Dikumpulkan tanggal

30 November 2018

PROGRAM STUDI S-1 TEKNIK INFORMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS PADJADJARAN

2018

**BAB I**

**PENDAHULUAN**

* 1. Tujuan

Membuat sebuah project grafika komputer yang memiliki salah satu topik diantara:

1. Primitive Computer Graphics Simulation: menggambarkan garis menggunakan perintah sederhana titik menggunakan beberapa algoritma.
2. Primitive Circle Simulation: Menggambar lingkaran + filling menggunakan perintah sederhana titik menggunakan beberapa algoritma.
3. Primitive Polygon Simulation: Menggambar polygon + filling menggunakan perintah sederhana titik menggunakan beberapa algoritma.
4. Windowing, Clipping Simulation, Zoom
5. Graphics Translation: Rotate, Move
6. Drawing Software: Membuat software menggambar
7. Chart Application (Grafik): Pie, Bar, dll.
8. GIS
9. Games
10. Graphics Animation
11. 3D Model, rotate, rendering
12. VR Simulation
13. Dll.

Project yang saya buat menerapkan 3D model yang bisa dirotate.

* 1. Dasar Teori

OpenGL adalah sebuah program aplikasi interface yang digunakan untuk mendefinisikan komputer grafis 2D dan 3D. Program lintas-platform API ini umumnya dianggap ketetapan standar dalam industri komputer dalam interaksi dengan komputer grafis 2D dan juga telah menjadi alat yang biasa untuk digunakan dengan grafis 3D. Singkatnya, Open Graphics Library, OpenGL menghilangkan kebutuhan untuk pemrogram untuk menulis ulang bagian grafis dari sistem operasi setiap kali sebuah bisnis akan diupgrade ke versi baru dari sistem.

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.

Membuat perintah dalam OpenGL dapat terjadi dalam dua cara yang berbeda. Pertama, adalah mungkin bagi programmer untuk membuat dan menyimpan daftar perintah yang dapat dieksekusi secara berulang. Ini adalah salah satu cara yang lebih rutin untuk program interface yang digunakan. Seiring dengan berkembangnya kelompok perintah yang kurang lebih permanen, maka memungkinkan untuk membuat dan menjalankan salah satu perintah dalam batas-batas waktu dari komputer grafis.

Seiring dengan kemampuan interface dari sistem operasi, OpenGL juga menyediakan beberapa built-in protokol yang mungkin berguna bagi pengguna akhir. Di antaranya fitur alat seperti alpha blending, pemetaan tekstur, dan efek atmosfer. Alat ini dapat berinteraksi dengan sistem operasi yang sedang digunakan.

Untuk dapat membuat project OpenGL menggunakan C# di Visual Studio 2017 hal-hal yang harus dilakukan adalah terlebih dahulu install C# di Visual Studio melalui Tools > Get Tools and Features. Cari **.NET desktop development** dan klik install. Setelah instalasi selesai mungkin akan diminta untuk merestart Visual Studio. Secara umum C# sudah support untuk OpenGL namun kita dapat mengkustomisasi di Tools > NuGet Package Manager > Manage NuGet Packages for Solution. Cari SharpGL for WPF lalu install.

**BAB II**

**PEMBAHASAN**

2.1. Alat dan Bahan

Alat dan bahan yang dibutuhkan untuk membuat project ini adalah sebagai berikut:

1. Laptop dengan spesifikasi Windows 10, RAM minimal 4 GB, memori minimal 500 GB
2. Visual Studio 2017 yang telah terinstall di laptop
3. C# yang telah terinstall di Visual Studio 2017
4. Tao Framework (bisa diunduh di <https://sourceforge.net/projects/taoframework/>)

2.2. Cara Kerja

Dalam tahapan pengerjaan ini tahapan instalasi Visual Studio dan C# dilewati karena dianggap sudah diinstall. Untuk C# bisa dicek pada bab sebelumnya.

2.2.1. Instalasi dan Pengalamatan Tao Framework

Penginstalannya cukup mudah karena bentuk filenya sudah berupa .exe sehingga hanya perlu beberapa klik saja dan nantinya akan terextract sendiri dalam laptop. Tempat disimpannya Tao Framework juga bisa dikustomisasi namun defaultnya akan tersimpan di C:\Program Files.

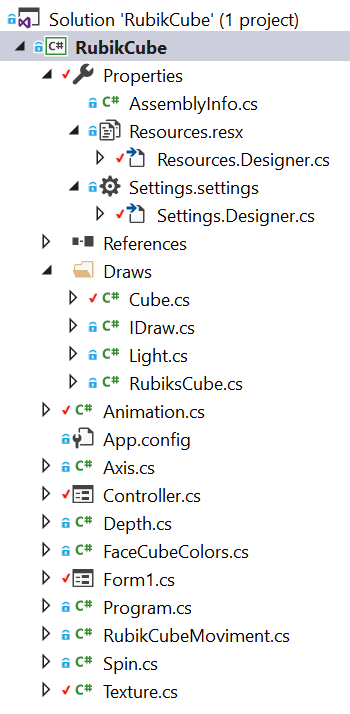
Selanjutnya kita harus mengalamatkan Tao Framework agar dapat dideteksi oleh Visual Studio. Pertama buat *environment variable* baru dulu dengan mengetikkan ‘env’ di Windows search. Masukkan path dari folder Tao yang telah terinstall diakhiri dengan /bin. Contohnya E:\TaoFramework\bin.

Kemudian kita buat project C# baru di Visual Studio. Pada *Project Explorer* klik kanan nama projectnya lalu pilih Add > References. Klik Browse dan arahkan ke file Tao.OpenGL.dll dan Tao.FreeGLUT.dll. Kedua file ini dapat ditemukan dalam folder bin dari TaoFramework.

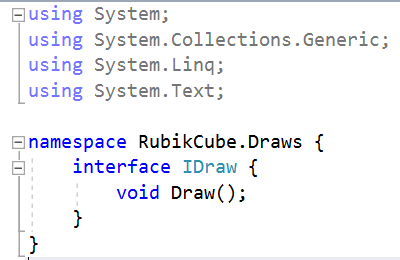
Sekarang Tao Framework sudah dapat digunakan ketika project dirun.

2.2.2. Pembuatan project

Untuk file-file yang harus dibuat terdiri dari:



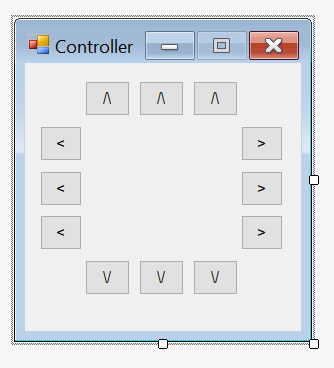
Bagian Properties dari project sudah *auto generated* ketika pertama kali membuat project. Untuk folder Draws dimana terdiri dari Cube, IDraw, Light, dan RubiksCube adalah dimana kita akan membuat grafiknya. Class yang bernama Cube, Light, dan RubiksCube dapat dipanggil jika class IDraw telah dibuat. IDraw semacam inisiasi dari class seluruh grafik. Code dari IDraw.cs dapat dilihat di bawah ini:

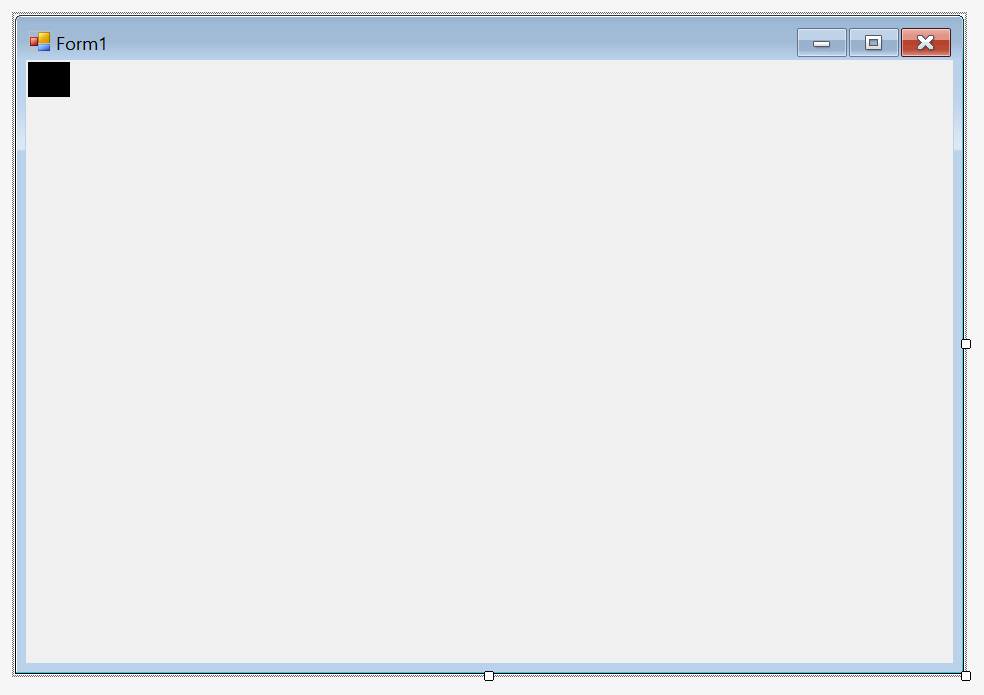


Untuk Cube.cs, Light.cs, dan RubiksCube.cs dapat dilihat langsung di project karena kodenya cukup panjang.

Setelah itu kita membuat file bernama Animation.cs, Axis.cs, Depth.cs, FaceCubeColors.cs, Program.cs, RubikCubeMoviment.cs, Spin.cs, dan Texture.cs. Sesuai namanya, Animation untuk membuat animasi project ini, Axis untuk menginisiasi sumbu X, Y, dan Z, Depth untuk menginisiasi baris dan kolom satu, dua, tiga (karena rubik punya tiga baris dan kolom), FaceCubeColors untuk menampilkan warna tampilan muka rubik (tapi bukan untuk inisiasi warnanya), Program untuk menjalankan project, RubikCubeMoviment untuk mengatur pergerakan rubik, Spin untuk membuat pergerakan berputar, dan Texture untuk menginisiasi bitmap.

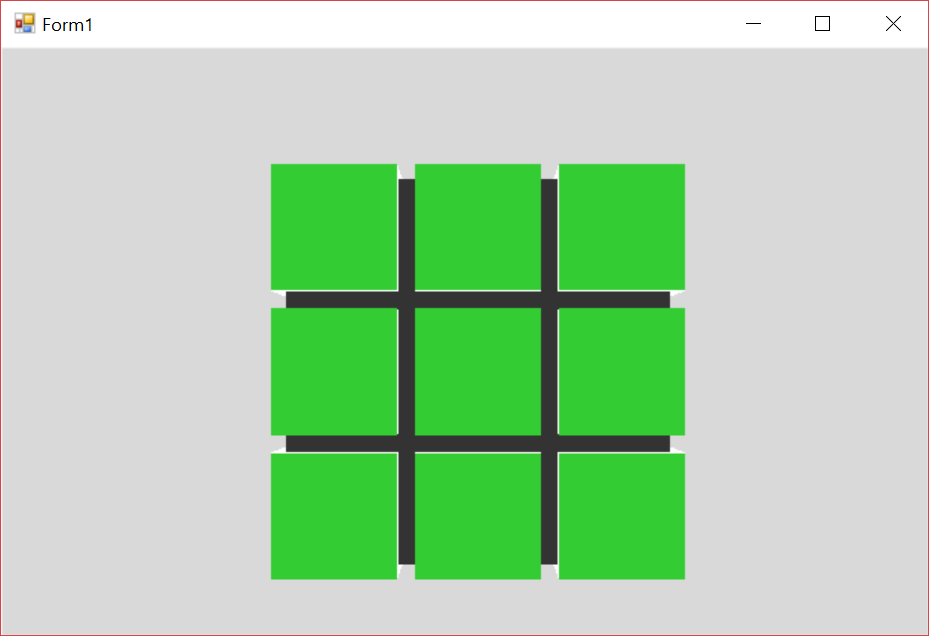
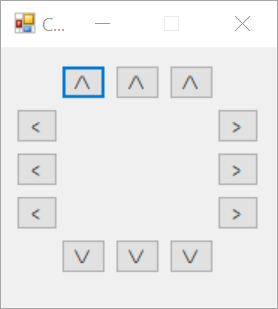
Terakhir kita mendesain Controller.cs dan Form1.cs. Kedua file ini adalah file yang berbentuk designer. Pada kedua file ini kita dapat menyusun desain project yang kita inginkan. Di sini saya memisahkan Form1 untuk rubiknya dan Controller untuk tombol-tombol yang akan menggerakan rubik sehingga lebih mudah dalam pengkodingannya. Setelah desain telah disusun, masing-masing object dapat kita kodingkan fungsinya sehingga seluruh class yang telah dibuat sebelumnya dapat dipanggil. Di bawah ini dapat dilihat desain dari Controller dan Form1. Dapat dilihat untuk Form1 mungkin hanya muncul seperti itu dikarenakan object yang dimasukkan adalah OpenGLControl sehingga hanya dapat diposisikan saja tidak bisa mengatur isinya di desain.



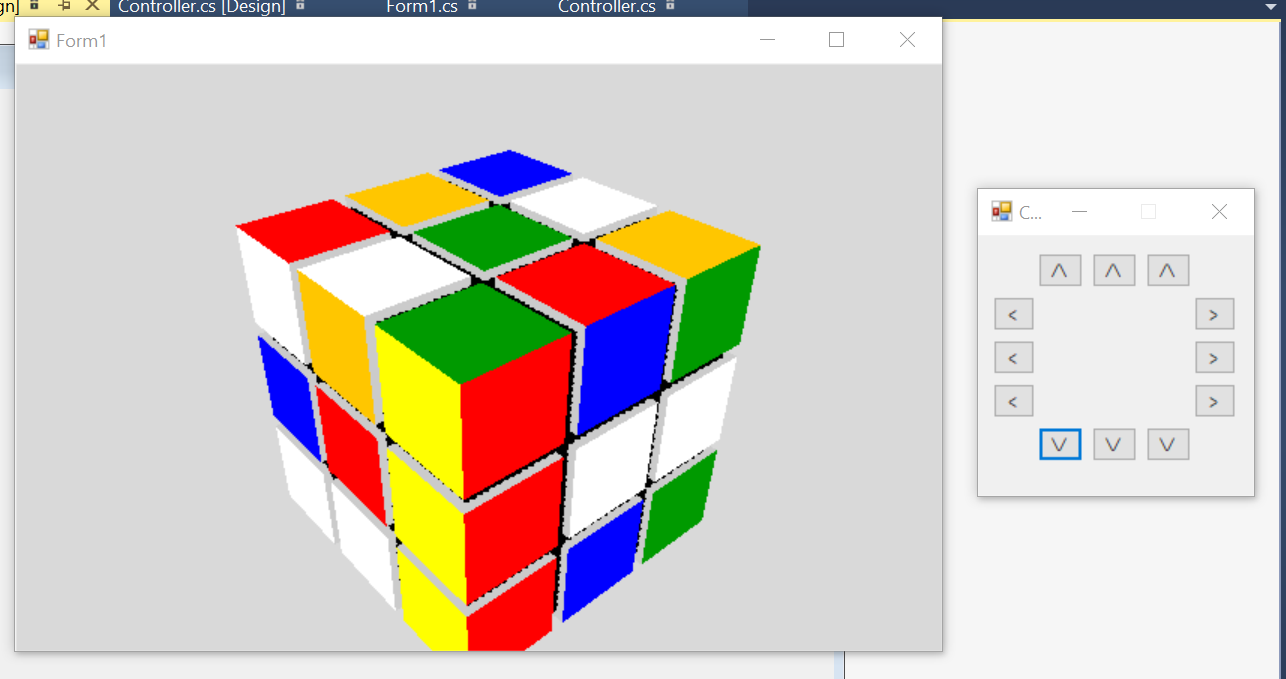


2.2.3. Demo

Setelah semua project telah siap, start projectnya. Di bawah ini terdapat screenshot dari hasil demo project.

Permainan rubik ini bisa dimainkan dan juga bisa memutarnya untuk melihat dari sudut pandang lain menggunakan tombol W, A, S, dan D.



2.3. Kesimpulan

Program yang telah dibuat masih banyak kekurangannya dikarenakan masih belum luasnya pengetahuan saya mengenai grafik komputer. Rubik masih belum bisa memiliki nilai maupun waktu agar terlihat seperti permainan asli. Controllernya juga masih terpisah dan belum dapat caranya agar bisa dijadikan satu window saja (dua window kurang nyaman dipakai). Framework yang digunakan yaitu Tao Framework juga sudah tidak digunakan lagi oleh orang-orang namun saya baru mendapatkan pengetahuan sebatas ini saja. Namun di balik seluruh kekurangan itu setidaknya rubik sudah berjalan dengan baik mulai dari controller pemutar baris dan kolomnya maupun rotasi untuk melihat sudut pandang lainnya.

**DAFTAR PUSTAKA**

<http://agussale.com/penjelasan-mengenai-apa-itu-opengl>

<https://www.opengl.org/discussion_boards/showthread.php/200790-How-To-Install-OpenGL-In-Visual-Studio-2017-Community-Edition>

<https://codeyarns.com/2013/06/18/how-to-get-started-with-tao-framework-using-c/>

<https://stackoverflow.com/questions/2795442/is-the-tao-framework-dead>

<https://www.youtube.com/watch?v=1KgFS9PAVEE>

<https://www.youtube.com/watch?v=5XMKBtqbEog>

<https://www.youtube.com/watch?v=BfXWc4tCkjQ>