

Visible Surface Detection: Warnock’s Algorithm

3D Computer Graphics and Animation Programming Assignment 16



Ida Bagus Bhaskara (001201500016)

Vera Debora Vitamas (001201500076)

Jonathan Surya Sandjaya (001201500055)

CIT 2 2015

**Table of Contents**

Cover Page 1

**Table of Contents 2**

[Chapter 1 Introduction 2](#_Toc506154720)

[Chapter 2 Basic Theory 2](#_Toc506154721)

[2.1 Transforming 3D Object into 2D Image 2](#_Toc506154722)

[2.2 Parallel Projections 2](#_Toc506154723)

[2.3 Axonometric and Oblique Projection and Its Differences 5](#_Toc506154724)

[2.4 Oblique Projection Transformation 11](#_Toc506154725)

[Chapter 3 Implementation 14](#_Toc506154726)

[3.1 Main Interface of the Application 14](#_Toc506154727)

[3.2 Features of the Applicaton 15](#_Toc506154728)

[Chapter 4 Design 15](#_Toc506154729)

[4.1 Variables Used in the Program 15](#_Toc506154730)

[4.2 Representation of Edge and Point in the Application 16](#_Toc506154731)

[4.3 Initializing the Cube 17](#_Toc506154732)

[4.4 Setting the Edge 18](#_Toc506154733)

[4.5 Setting the Vertex 18](#_Toc506154734)

[4.6 Drawing the Cube 18](#_Toc506154735)

[4.7 Hiding the Cube 19](#_Toc506154736)

[4.8 Setting the Matrix 19](#_Toc506154737)

[4.9 Multiplying the Matrix 19](#_Toc506154738)

[4.10 Rotation Tick Method 20](#_Toc506154739)

[Chapter 5 Evaluation 21](#_Toc506154740)

[5.1 Display the Oblique Projection with Different Value of Phi 21](#_Toc506154741)

[5.2 Display the Oblique Projections with Different Values of Alpha 22](#_Toc506154742)

[Chapter 6 Work Log 23](#_Toc506154743)

[Chapter 7 Conclusion and Remarks 24](#_Toc506154744)

# **Introduction**

1. About the Application

This program uses Microsoft Visual Studio as the programming platform and Visual Basic as the programming language. This report covers the basic theory, implementation, design, evalution, work log, and conclusion.

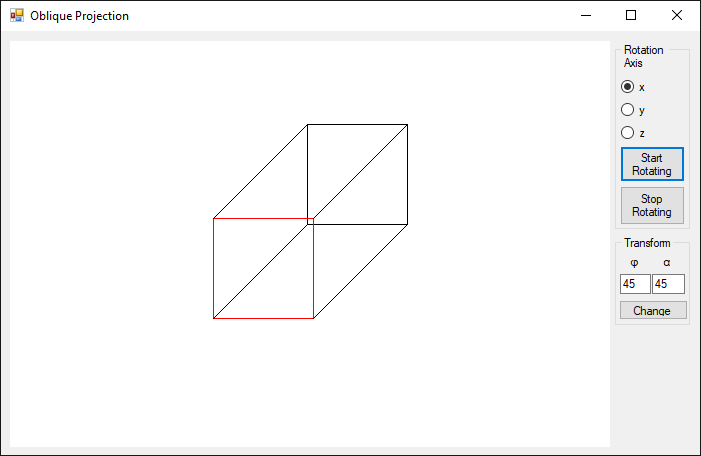
# **Basic Theory**

## I’m a Placeholder

# **Implementation**

## Main Interface of the Application

This application has one interface, which it uses as its main interface. The interface is pictured below:



7

6

5

4

3

2

1

The interface consists of the following components:

1. Canvas Picture Box

The canvas is used to display the cube using the application.

1. Rotation Axis Radio Buttons

The radio buttons are used to change the , and coordinate of cube rotation.

1. Start Rotating Button

The button is used to rotate the cube in the canvas.

1. Stop Rotating Button

The button is used to stop the rotation of the cube.

1. Phi Textbox

The input box is used to determine the value of the phi.

1. Alpha Textbox

The input box is used to determine the value of the alpha.

1. Change Button

The change button is used to change the value of alpha and theta of the cube based on the value entered on the textboxes.

## Features of the Applicaton

The application has several features, namely:

1. Rotating the 3D Cube Around the , and Axes

To rotate the cube, the user must choose 1 out of 3 axes shown in the radio button and then click the start rotating button to rotate the 3D cube.

1. Changing the Values of Alpha () and Phi ()

To change the value of alpha and theta the user must input the values in the textboxes and click change button.

1. Stopping the Rotation of the Cube

To stop the cube rotation the user must click the stop rotating button.

# Design

## Variables Used in the Program

The following variables are used in this program:

* Dim graphics As Graphics  
  This variable is used to pass the canvas variable to the Graphics class so functions from the Graphics class, for example the DrawLine function, can be used to draw the cube’s edges in PictureBox1, which acts as the canvas.
* Dim canvas As Bitmap  
  This variable is used to initialize the canvas as a New Bitmap with PictureBox1’s width and height as its properties.
* Dim vertex(7) As Point

This array is used to store the cube’s vertices.

* Dim edges(12) As Edge

This array is used to store the cube’s edges.

* Dim view(3, 3), screen(3, 3) As Single

These 2D arrays are used to store the and matrices.

* Dim VR(7), VS(7) As Point

These 2D arrays are used to store the result of multiplying the cube’s vertex with the and matrices respectively, where and . will be the displayed vertex.

## Representation of Placeholders

# Evaluation

To ensure that the application works as it is designed, test cases are conducted. These test cases are as follows:

## Placeholders

# Work Log

The work log is extracted directly from Visual Studio’s Git Log History, which is also available publicly at <https://github.com/bakanui/WarnocksAlgorithm/commits/master>.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Author | Date | Time | Commit Message |
| 1 | Bhaskara Ida Bagus | 3/15/2018 | 9:09:29 PM | Initializing data structures |
| 2 | Jonathan Surya | 3/18/2018 | 8:11:14 PM | Interface and try to changed the vpolygon using list of |
| 3 | Jonathan Surya | 3/18/2018 | 8:33:10 PM | debug some mistype |
| 4 | Bhaskara Ida Bagus | 3/21/2018 | 5:41:07 PM | 2 Pyramids Now Present |
| 5 | Vera Debora Vitamas | 3/23/2018 | 6:10:46 PM | Add backface culling algorithm |
| 6 | Bhaskara Ida Bagus | 3/24/2018 | 4:41:00 PM | Add rotation and move with keypresses, interface redesign |
| 7 | Bhaskara Ida Bagus | 3/24/2018 | 5:14:23 PM | Forgot to actually put the rotation in keypress |
| 8 | Bhaskara Ida Bagus | 3/24/2018 | 5:35:50 PM | Add report with placeholders |

# Conclusion and Remarks

Shy asking lost in road.