Three-Dimensional Text Applications with OpenGL

Alfian Erwinsyah1*, Yuliwati2, Liesna Andriany3 and Herman Fithra4
1IAIN Sultan Amai Gorontalo, Indonesia
2Department of Research and Educational Evaluation, STKIP Kusuma Negara Jakarta, Jakarta, Indonesia
3Faculty of Teacher Training and Education, Universitas Islam Sumatera Utara, Medan, Indonesia
4Department of Civil Engineering, Universitas Malikussaleh, Aceh, Indonesia

* alfian_erwinsyah@iaingorontalo.ac.id

Abstract. Text is one object that can be manipulated into three-dimensional shapes. The main effects in displaying three-dimensional text include the shadow effect, the depth of 3D effects, the type of bevel on the edges of the text or the addition of light effects and transformations such as rotating and flipping. In addition to the above its effects, it can also do 3D text rotation by doing animation on the text. One way to make text into 3D objects can use OpenGL which is an engine that is widely used for 3D applications. In addition to 3D effects OpenGL also provides features to give color, apply texture to objects, lighting, rotation, and scaling. The results of this study are software that can convert a text into three-dimensional text in the form of images by providing shadow effects and adding background graphics in the form of textures with the addition of several features such as rotation, scaling, and moving. The software also can print the results of 3D text and save the results in a bitmap image format.

1. Introduction
This application can manipulate objects into widely used three-dimensional shapes. 3D techniques almost used to make animations such as cartoons with special effects and for films. This application is run with a super computer and requires a fairly high processor performance. Of course, in terms of hardware and software, it requires a huge amount of money.

The results of the application are usually in the form of images that are widely used in web pages. Text is one object that can be manipulated to become a three-dimensional form [1][2][3]. The result may be used as a banner or decorator in desktop wallpaper on the Windows operating system. Three-dimensional applications for text processing can generally display text in the form of 3-D [4][5]. moreover it is also being able to display shadow (shadow) effects, depth of effect 3 be D, type of bevel on the edge of the text or the addition of light effects and carry out transformations such as rotating and flipping. Even in addition to the above effects, it can also do 3-D text rotation by doing animation in the text [6].

To produce an application that can be manipulate objects such as text into three-dimensional shapes with lighting and rotation techniques, using programming languages such as Visual Basic to produce simple 3D programs can use OpenGL. OpenGL programming 3D forms are easier to use and other advantages that the resulting program did not required high computer performance [7][8][9].
OpenGL is an interface application program used to define 2D and 3D computer graphics [10]. This cross-platform API program was generally considered a standard provision in the computer industry in interaction with 2D computer graphics and also became a common tool for use with 3D graphics. In short, Open Graphics Library, OpenGL eliminated the need for programmers to re-write the graphical part of the business operating system every time was upgraded into a new version of the system.

2. Related Works
OpenGL (Open Graphics Library) is a standard specification that defined a cross-language, cross-platform API for writing applications that produce 2D computers and 3D graphics [11]. The interface consists of more than 250 different function calls that can be used to draw complex three-dimensional scenes from simple primitives. OpenGL was developed by Silicon Graphics Inc. (SGI) in 1992 and is widely used in CAD, virtual reality, scientific visualization, information visualization and flight simulation. It is also used in video games, which compete with Direct3D on Microsoft Windows platforms. OpenGL is managed by a non-profit technology consortium, the Khronos Group.

3. Research Methodology
The design methodology of this system used the life cycle of developing the waterfall system where the stages are shown in Figure 1.

![Program Design Stages](image-url)
3.1. Study algorithms and OpenGL techniques
At this stage used an observation to review the literature that relating to the design of logos, algorithms and OpenGL techniques through books and websites. The results are then arranged and grouped to facilitate to understand.

3.2. Algorithm Analysis
The second stage the researcher used flow diagrams and process diagrams to describe the work processes of the program that will be designed with data / literature. In the work process the drawing program used algorithms to show the work process of the program[12].

3.3 Designing the Interface
Designed a system to facilitate in carrying out work process activities such as designing input forms directly in the Integrated Development Environment (IDE) Visual Basic 6.0 by arranging and arranging objects to form an interface that is easy to use.

3.4. Program Manufacturing
In this stage coding was done on a number of previously designed interfaces so the software can produced and can performed the calculation process according to the implementation of a predetermined algorithm[13].

3.5. Program Testing
This stage is useful for testing the programs that have been made whether it has been running as intended and if there are errors, it will be corrected and re-evaluated[14].

3.6 Techniques for creating 3D Text
The basic technique in this three-dimensional text program is actually copying the text up to several times according to the depth of the shadow that will be formed [15]. The front part was called foreground image is text that has a color, while the background image can consist of several pieces according to the desired depth. Background image is a copy of the foreground image, usually colored according to the color chosen by the user (by default it is black). The technique of producing three-dimensional text effects is done by inserting a background image from the foreground image [16]. The illustration describes how the image background was inserted as a shadow foreground image.

In simple terms the shadow effect above can be obtained by adjusting the position of each object at a certain coordinate and giving a color effect. For more details, see the following example. Suppose the text "3D Text" if expressed in position in the Cartesian coordinates is in the upper left position (2, 6).

![Figure 2](attachment:image.png)

**Figure 2.** Background Image on XY coordinates

Figure 2 above is a picture of a text at the position of coordinates (2, 6). The text is black and will be used as a background image which is a shadow image. To create a shadow’s effect the next step is to duplicate the text of the same size except that the color is different. Next put the duplicated image
overwriting the background image and shifting the position of the coordinates of several units to the position of the background image. [17].

4. Result and Discussion

4.1. Design
This section will explain about the design of the program. The program was designed using Visual Basic version 6.0 Professional Edition and can also be opened and compiled with Visual Basic version 5.0. The interface of this program is a window shaped form. The form designed consists of five forms, namely the form input text, the color logo input form, the background color input form, the rotation form, the load texture bitmap form, and the save as form, and the Main form. [18]

4.2. Form Design
The first part of the form designed was the input text form (Figure 3). This form serves to request input from the user regarding the text that will be made into a 3D form. 3D form made from label and command button objects. The label section will be created and the caption will contain the text "Enter Logo Text". While the two command buttons are "OK" if the user presses it, the inputted text will be converted into 3D. The selected of the "Cancel" button means canceling the input text. [11]

![Figure 3. Design of Text Input Form](image)

Next is the second form is the logo color input form (Figure 3). This section was used to change the color of the text. Users can use a combination of three colors, namely RGB (Red Green Blue) by sliding on the slider button. The slider is set from a minimum value of 0 (zero) and a maximum of 100. The combination of the three colors mixed will be shown in the box below. If the user presses the "OK" button then the color of choice will be directly applied to the text or logo.

![Figure 4. Form Design Input Color Logo](image)

The next part is designing the background of color input form (Figure 4). This form is used to select the background color (back color) of the text. There are three slider buttons available as in the color
logo input form. The results of mixing these three colors will be displayed at the bottom. And if the user presses the "OK" button, the chosen color will be applied as the background color on the logo.

![Image of color mixing form](image1)

**Figure 5.** Form Design to Input Background Color

In this program there are also facilities to rotate text or logos with a rotation angle up to 360°. This must be made and a form for this facility. This form was named rotation form (Figure 4) with three sliders which indicate the rotation performed on the x, y, and z axes. The minimum value for these three slider values is -360° and the maximum value is 360°. If the selected angle is positive, the rotation will be carried out with the direction of rotation opposite the clockwise direction and vice versa if the selected angle is negative then the direction of rotation will be in the direction of rotation clockwise.

The results of the selection angle for all three axes were displayed on the label below the slider. If the user clicks on the section marked with an arrow on the slider then the rotation angle changes 1° but if the user shifts to the rectangular button on the slider then the rotation angle changes to 5°. If the user pushed the "OK" button then the three rotation angles will be applied to the text or logo.

![Image of rotation form](image2)

**Figure 6.** Rotation Form Design

The program designed also has the facility to move 3D text along the x, y, and z axes. For that purpose, a form was designed for this requirement. This form was called the Rotation Form (Figure 6).
This form used three sliders for the x, y, and z axes respectively. This slider has a minimum value of -5 and a maximum of 5. If the user presses the "OK" button, the position of the text will be moved according to the values of x, y, and z. [19].

5. Conclusion
Based on the discussion of the previous chapters that have been done, some conclusions can be taken as follows:
1. This three-dimensional text processing or logo program can be used as an alternative program to produce three-dimensional text or logos that have perspectives such as other commercial 3D text programs. OpenGL is an API that facilitate for developers to create 3D programs.
2. Programs created have the facility to add textures to text or 3D logos, lighting, rotation, adjusting the depth of 3D effects and transferring position and font selection.
3. The output results of the application designed can be used as an image that can be used on web pages as a banner or logo.
4. The results of this program design can be used as a reference that can be used to explain the technique of producing 3D text and the three-dimensional text produced can be used as a banner for web pages.

References
[1] A. E. Megahed, “Method and apparatus for data compression for three-dimensional graphics.” Google Patents, Jul-2002.
[2] A. Muhson, “Pengembangan media pembelajaran berbasis teknologi informasi,” Jurnal Pendidikan Akuntansi Indonesia, vol. 8, no. 2, 2010.
[3] A. Iskandar, M. Rizal, N. Kurniasih, D. U. Sutikso, and A. Purnomo, “The Effects of Multimedia Learning on Students Achievement in Terms of Cognitive Test Results,” in Journal of Physics: Conference Series, 2018, pp. 1–7.
[4] G. G. Robertson, J. D. Mackinlay, and S. K. Card, “Cone trees: Animated 3d visualizations of hierarchical information.,” in CHI, 1991, vol. 91, pp. 189–194.
[5] D. Putra, Pengolahan citra digital. Penerbit Andi, 2010.
[6] J. Simarmata, Rekayasa web. Penerbit Andi, 2010.
[7] D. H. Eberly, 3D game engine design: a practical approach to real-time computer graphics. CRC Press, 2006.
[8] D. Sudrajat, “Pembuatan Simulasi Pergerakan Objek 3D menggunakan OpenGL,” Jurnal Teknik Elektro dan Komputer, vol. 1, no. 2, 2012.
[9] D. Sudrajat, I. Mulyaninglish, C. Kurniawan, I. N. Sara, and E. P. Permana, “Computer assisted instruction model for mathematics education,” Journal of Advanced Research in Dynamical and Control Systems, vol. 10, no. 13 Special Issue, pp. 1613–1616, 2018.
[10] A. Kokalj, “Computer graphics and graphical user interfaces as tools in simulations of matter at the atomic scale,” Computational Materials Science, vol. 28, no. 2, pp. 155–168, 2003.
[11] A. S. Riyadi, “IMPLEMENTASI OPEN GL32 UNTUK MEMANIPULASI GAMBAR SEGITIGA DAN SEGEMPAT,” Komputer Teknologi Informasi, vol. 1, no. 1, 2016.
[12] D. Sudrajat et al., “Expert system application for identifying formalin and borax in foods using the certainty factor method,” Eurasian Journal of Analytical Chemistry, vol. 13, no. 6, pp. 321–325, 2018.
[13] R. Rahim et al., “Hashing Variable Length Application For Message Security Communication,” ARPN Journal of Engineering and Applied Sciences, vol. 14, no. 1, pp. 259–264, 2019.
[14] A. Iskandar et al., “Web based testing application security system using semantic comparison method,” in IOP Conference Series: Materials Science and Engineering, 2018, pp. 1–6.
[15] J. D. Foley et al., Computer graphics: principles and practice, vol. 12110. Addison-Wesley Professional, 1996.
[16] I. Binanto, Multimedia digital-dasar teori dan pengembangannya. Penerbit Andi, 2010.
[17] and E. Ongko, O. S. Sitompul, and E. B. Nababan, and D. Abdullah, “Hybrid Approach Redefinition (HAR) Method with Loss Factors in Handling Class Imbalance Problem,” in 2018 International Symposium on Advanced Intelligent Informatics (SAIN), 2018, pp. 56–61.

[18] S. G. Fashoto, O. Amaonwu, and A. Afolorunsho, “Development of A Decision Support System on Employee Performance Appraisal using AHP Model,” JOIV: International Journal on Informatics Visualization, vol. 2, no. 4, pp. 262–267, Aug. 2018.

[19] D. Abdullah et al., “Super-Encryption Cryptography with IDEA and WAKE Algorithm,” J. Phys.: Conf. Ser., vol. 1019, p. 012039, Jun. 2018.