

ABSTRACT

A 2d graphics based game Robo bird is a great start for a student who starts learning computer graphics & visualization. The development of the game has a large scope to learn computer graphics from scratch. We used OpenGL utility toolkit to implement the algorithm, written it in c++ language. There is still scope left in the development of project like, after“game over” a list should show the top ten scorers, a need to embed a button “play again”. Welcome screen needs more modification there is the scope of embedding buttons like “about”, “how to play”, “configuration”, “profiles”, etc. In the future, we hope we would implement it in the source code for a better experience of playing this game. Finally, we could say by developing the game we have learned the basics of computer graphics and in the future, by developing it further we shall learn more. It will be our pleasure if we could develop in a 3d graphics package.

KEYWORDS

- Collision Condition
- Translation
- renderBitmapString()
- glutSwapBuffers()
- glPopMatrix()
- drawcopter()

INTRODUCTION

Computer Graphics:

Computer graphics is one of the most exciting and rapidly growing computer fields. It is also an extremely effective medium for communication between man and computer; a human being can understand the information content of a displayed diagram or perspective view much faster than he can understand a table of numbers or text containing the same information. Thus computer graphics are being used more extensively.

2D Robo-Bird:

This mini-project under Computer Graphics & Visualization Laboratory is an implementation of a kind popular Robo bird game using the OpenGL Graphics Library and GLUT Toolkit.

Scope:

The player inputs his/her name before starting the game & can control it either using keyboard or mouse at a time and at the end of the game it will show how much distance you covered as his/her score.

Objective:

The objective of the game is to fly a Robo bird in space with a restricted upward and downward motion using either mouse or keyboard, meanwhile, walls will move towards player's copter and player have to avoid a collision between them. The game will enter into next level as soon as player crosses next 50 unit distance and speed of the wall will increase by a certain amount each time player enter the next level. The game will be over if a collision occurring there.

Mouse function:-

On right click down, copter moves upward, On releasing it moves downward.

Keyboard function:-

Pressing UP key move copter up and down if press DOWN key.

ANALYSIS AND MODULE DESCRIPTION

The objective of the game is to fly a Robo bird in space with a restricted upward and downward motion using either mouse or keyboard, meanwhile, walls will move towards player's copter and player have to avoid a collision between them. The game will enter into next level as soon as the player crosses next certain fixed unit distance and speed of wall will increase by a certain fixed amount each time player enter the next level.

The basic feature of the 2D game was analyzed to be:-

1. A welcome screen which contains the following buttons: -

- i) PROFILE: - the player can make his/her own profile to save his level and score.
- ii) ABOUT: - a display about the game, its version, and owners of the game.
- iii) CONFIGURATION: - the player can choose the key for up and down action of the copter and can change the color of the copter.
- iv) START: - clicking on which game will start with the customized configuration.

2. Calculating distance traveled by copter and displaying and updating it continuously as the score of the player while the game is on.

3. Calculating level and continuously displaying and updating it as the game is on.

4. A game over the window which will show score and level of player and also a button "PLAY AGAIN" clicking on which game will start again.

5. If player score manages among top ten scorers after game over it should prompt for entering his name so it can store and display player's name among top ten scorers with the player's name in the list.

SYSTEM REQUIREMENTS

Software Requirements:

- Operating System: Ubuntu 10.10 / Windows 10
- Language: C
- Library: OpenGL(glut 3.7.6)

Hardware Requirements:

- Processor:
Intel Processor 100 MHz / Pentium Processor 100 MHz /AMD Processor 100 MHz
- RAM:
256 MB or more

OUTPUT SNAPSHOTS

1. Welcome Window Screen



Figure 1: Welcome Window Screen

2. Game Running Window

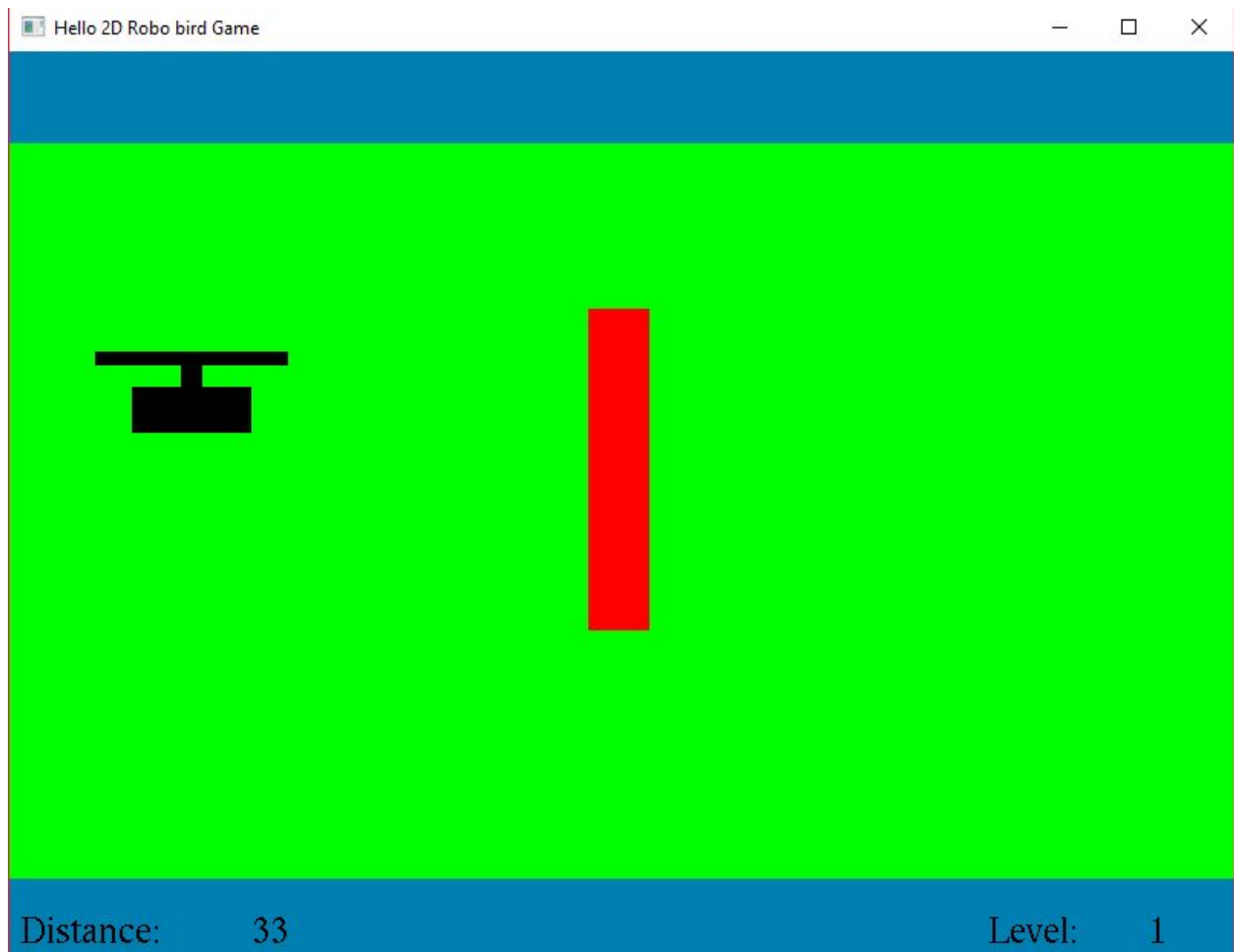


Figure 2: Game Running Window

3. Game Over

You Scored: 33

CONCLUSION

We have attempted to design and implement a "2D RoboBird". OpenGL supports enormous flexibility in the design and the use of OpenGL graphics programs. The presence of many built-in classes methods take care of much functionality and reduce the job of coding as well as makes the implementation simpler. The project was started

with the designing phase in which we figured the requirements needed, the layout design then comes the detail designing of each function after which, was the testing and debugging stage. We have tried to implement the project making it as user-friendly and error-free as possible. We regret any errors that may have inadvertently crept in.

REFERENCES

1. **Computer Graphics** – Principals And Practice (Foley, Van Dam, Fenier, and Hughes) helped me to understand graphics generation algorithms, user interface, and dialogue design.
2. **OpenGL Programming Guide** (Addison-Wesley Publishing Company) helped me to get through all OpenGL functions and Commands and understandings of all aspects of them.
3. **www.cplusplus.com**: - provided references regarding all c++ functions and their uses.
4. **www.stackoverflow.com**: - help to get rid of all types of error occurred regarding uses of OpenGL functions.
5. **www.lighthouse3d.com** - OpenGL tutorial for implementing the OpenGL functions in Source code.