COSC363 Assignment 1 – A Robots World

Jack Wilkie – ID: 31481060 (jwi123)

# Introduction

This scene was inspired by an interpretation of the phrase “Mobile robot”, which may have meant to refer to a robot that moves, but which was instead interpreted as a robot that makes mobiles (cell phones). This scene consists of a production line for mobile phones, where robotic arms grab parts out of boxes, and assemble them into completed mobile phones. The parts have randomized colors to demonstrate that the phones are distinct objects that move along the line.

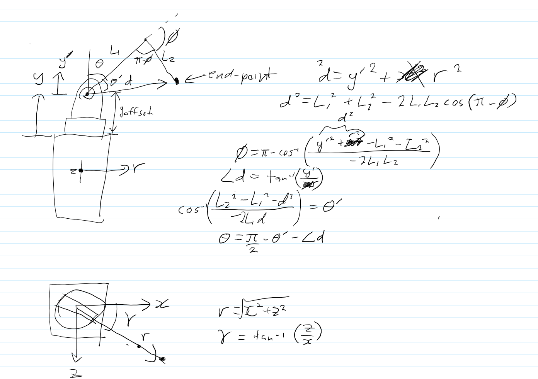


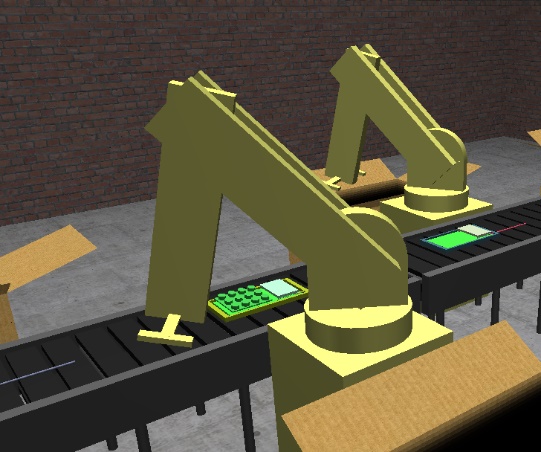
# Key bindings

* WASD: Movement
* QE: Up/down motion
* Arrows: Rotation
* +/-: Increase/decrease speed multiplier (High speeds may cause glitches)
* \*: Reset to default speed
* 1: Set to freelook camera
* 2: Set to controlled/animated camera

# Notable Features

## Robotic arm

I developed the robotic arm myself, and formulated some equations that can convert a point relative to the base of the arm into the required angles of the arm required for it to reach that point. One challenge to overcome was using arctan to find the rotation on the ‘core’ axis, as arctan(x/z) generally only maps to ±90°, as -x/z is indistinguishable from x/-z, this was solved by adding PI to the result in the cases where arctan normally got it wrong.

The original working is shown to the left, and the final equations are (separated for readability):

,

,

,

,

Where gamma, theta, and phi are the angles of the core axis, arm 1, and arm 2, respectively (as per the diagram).

The robot arm was constructed entirely from built-in primitives.

## Variable Frame Timing (delta-time), Key Hold Handling, Smooth Camera

I decided to implement my own system to wrap and improve on the glut callback system that we were taught for animations. At the end of the display function, a timer is set for 0ms to call the animation/ processing function as soon as possible. The processing function uses chrono::high\_resolution\_clock to find the exact time elapsed since the last frame, dt, and then passes this to all animation functionality. This means that no matter what framerate the program runs at, the animations should always take the same time, and run at the same speed.

In addition, I created my own wrapper functionality around “glut[Keyboard/Special]Func” and glut[Keyboard/Special]UpFunc, so that the keys currently being held are stored in an array, and can be queried anywhere in the application. This was used to give smooth camera control, by detecting if certain camera motion buttons were down at each frame, and then moving the camera in the appropriate way, by an amount equal to a preset value per second, multiplied by the dt value from the variable frame timing system. This gave smooth camera movement at a speed independent of framerate.

## Mobile model

The mobile phone model used in the application was developed in blender as several parts. The meshes were loaded separately, and can be separately rendered. Any mobile phone part in the scene is an instance of the Mobile class, with a ‘state’ variable (which is a enum that also doubles as a bitfield of which parts to render) describing what kind of part it is, or in what stage of assembly the phone is in. The meshes are only loaded once for all the mobile models in the scene by keeping track of the number of mobiles created, and only loading them on the first constructor, and deleting them on the last destructor. The mobile object has an array that stores the color values of all of its parts.

The mobile object contains a combine method, that combines it and another mobile object, normally preserving the colors of the two original part objects, this is used in my animation to combine the parts from the robots into the mobile on the assembly line, preserving the existing colors of both objects.

# Resources

Cardboard texture: [www.textures.com/download/cardboardplain0008/28990](http://www.textures.com/download/cardboardplain0008/28990)

Brick texture: [www.textures.com/download/bricksmallbrown0270/66235](http://www.textures.com/download/bricksmallbrown0270/66235)

Concrete texture: [www.textures.com/download/concretebare0433/108718](http://www.textures.com/download/concretebare0433/108718)

loadTGA, and LoadBMP from the labs were included, but only loadTGA was utilized.