Drawing World

Project Proposal

Sidney Velado

Purpose:

The purpose of this program is to create a game based around drawing. The purpose of the game is to take user input to draw objects, and then allow the user to manipulate those objects. In order to ease the user experience, Drawing World will use voice commands to change “modes”. Modes include draw, move, enlarge, and others. First the program will be made to work on an empty canvas, using mouse input as the drawing tool. However, once established and if possible, the second phase of the program will be to implement it with a Kinect, displaying a picture or live video, which the user then will use their hands to edit or draw one. The reason this has not been in the initial phase is to establish feasibility, as well as hardware limitation that will be resolved after Black Friday. That’s another reason voice control was chosen to be used as the main tool for manipulating modes as opposed to key strokes, as gestures were being used as our main pointer tool.

Technologies that are or may be implemented(Modules):

* Pygame
* Pykinect
* Speech Recognition
* Tkinter
* Multithreading
* Time

Competitive Analysis:

In the market there are currently about 5 strong contenders in what would be competition for Drawing World. Among these are Microsoft Paint, MyPaint, Paint.net, GIMP and Inkscape and Artweaver. The main reason these programs are so popular are their ease of use while still maintaining a variety of features. Their uses range from very advanced projects to simple sketches. Drawing World will seek to replicate and even improve the ease of use, and due to its simplified interface.

Despite Drawing World’s similarity to these programs, it differs in one key aspect, that Drawing World is centered more around being a entertainment platform. Thus, one key attribute that has to inherit and improve on is the ease of use that most of these programs have. Microsoft Paint is notorious for having been small children’s favorite drawing tools for a huge generation. Drawing World aims to be equal if not more simple to use than paint. Assuming we are in phase 1 of the project, it will be easy to see and easy to feel if Drawing World is intuitive. Given the choice to use voice commands, it seems to suggest that it will come at a minimal effort. Instead of having a huge toolbar with unknown tools, a small included tutorial in Drawing World will be a lot better and faster for users to learn.

One caveat of Microsoft Paint however, that the other programs have, especially GIMP, and Drawing World will be looking to maintain will be its ability to process images. Drawing World will want to have a simple option to import pictures in order to draw over them. In the same way that you can draw on your Snapchat snap, Drawing World will have an option where you can upload an image to draw on. However, assuming that phase 2 of the project proceeds, the image processing will be especially important because the kindle will be the camera used to take picture or draw the camera live feed.

Finally, one difference Drawing World will have compared to the previously mentioned programs will be it’s identity as an entertainment platform. Paint.Net and Artweaver are known for their great variety of options and settings. But, Drawing World will not have a great variety of choice or settings, this is in order to keep it more simple and keep it fun for people who want to paint. The final image will be very much like the Snapchat feature where you can draw on whatever picture you took with a very basic UI, however with a little more functionality as the program will be completely centered around that.

Code Artifacts:

The code artifacts are inside a folder in the main zip directory. It includes a small demo demonstrating the voice recognition, a small program that will simply print out what it interprets. It uses the module as well as the google voice API. Then, it also includes a multithreading demo, where I compare the effectiveness of two different sorts, however, as opposed to the usual timing of both it demonstrates it live by tracking the starting and ending of the threads, meaning it runs both sorts at the same time for a better comparison. Finally, I bring the two concepts together by editing a in class demo, OOPY Dots to demonstrate how while running the program I am able to make another thread that not only runs effective speech recognition but that also can communicate with the main thread, via class attributes. It runs OOPY Dots and takes voice input which changes the dots colors while remaining completely self sufficient and usable.