Ben Morehead
Personal Project Portfolio

Up to date as of December 9th, 2021
(WORK IN PROGRESS)

Table of Contents

1. Automatic Player Tracking Camera

1.1 Motivation:

1.2 Design Details:

1.2.1 Person Tracking
1.2.2 Court Segmentation

1.3 Debugging:

1.2.1 Person Tracking
1.2.2 Court Segmentation

1.4 Summary:

Summary Point	Information	Notes
Languages Used	Python	
Equipment/Tools Used	Ultralytics YoloV5, Coco Annotator	
Project Length	TBD	
Project Difficulty (1-10)	10	

2. Drone Network in Democratic Republic of Congo

1.1 Motivation:

1.2 Design Details:

1.2.1 LoRA Access Points

1.3 Debugging:

1.3.1 LoRA Access Points

1.4 Summary:

Summary Point	Information	Notes
Languages Used	C/Arduino	
Equipment/Tools Used	ESP8266 WiFi Module, LoRA V2 Modules	
Project Length	2 Semesters	
Project Difficulty (1-10)	9	

3. Augmented Reality Climbing Wall

1.1 Motivation:

1.2 Design Details:

1.2.1 Pose Tracking
1.2.2 Game Logic
1.2.3 Broadcasting Game State

1.3 Debugging:

1.3.1 Pose Tracking
1.3.2 Game Logic
1.3.3 Broadcasting Game State

1.4 Summary:

Summary Point	Information	Notes
Languages Used	Python	
Equipment/Tools Used	Kinect V2, Touch Designer	
Project Length	3 Months	
Project Difficulty (1-10)	10	

4. Wildfire Prediction Model

1.1 Motivation:

1.2 Design Details:

1.2.1 Architecture
1.2.2 Results/Success

1.3 Debugging:

1.3.1 Architecture
1.3.2 Results/Success

1.4 Summary:

Summary Point	**Information**	**Notes**
Languages Used	Python, SQL	
Equipment/Tools Used	Pytorch	
Project Length	2 Months	
Project Difficulty (1-10)	5	

5. Cascading Stair Lights

1.1 Motivation:

1.2 Design Details:

1.2.1 Analog Light Circuit
1.2.2 Digital Logic
1.2.3 Sensors and Detection

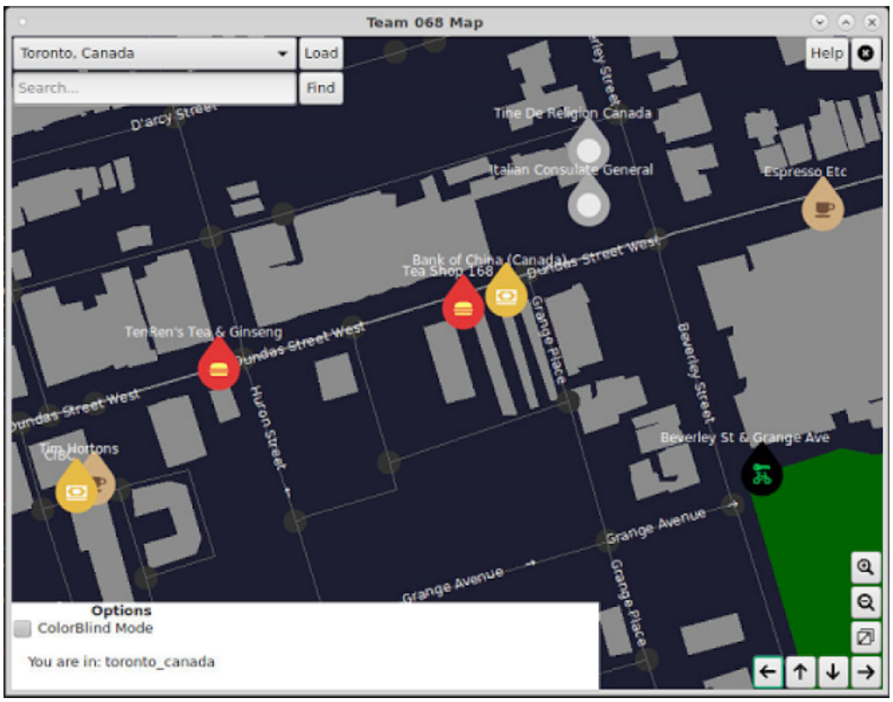
1.3 Debugging:

1.3.1 Analog Light Circuit
1.3.2 Digital Logic
1.3.3 Sensors and Detection

1.4 Summary:

Summary Point	Information	Notes
Languages Used	C/Arduino	
Equipment/Tools Used	Analog Circuits, Arduino	
Project Length	1.5 Years	
Project Difficulty (1-10)	6	

6. Custom GIS Software



1.1 Motivation:

In my second year of Electrical and Computer Engineering I took a course dedicated to designing a custom GIS software in C++. The engine for every project in the course was the same, and we were supposed to differentiate ourselves by the UI features we could add, and our approach to classic mapping algorithms such as the Travelling Salesman Problem. Overall this project was a great introduction to a larger scale software project.

1.2 Design Details:

1.2.1 Back End Engine

The problem for this project was separated into two parts: how can we acquire geographical information, and how can we visualize it. The acquisition of information was done through a GIS C++ library provided to us by the course, with which we could make api calls to get a large variety of data points on a provided mapping area. These data points contained information such as intersections, points of interest and environmental features.

1.2.2 Front End UI/Graphics

The front end was done using the EasyGL library. This UI library allowed us to visualize all of the data points mentioned earlier, as well as add some features to our software that made us stand out. Most significantly, I designed a search bar with automatic completion and directions provided on the selection of a location. Although fairly basic in concept, it took a lot of experimentation to get fully accustomed to EasyGL, but the skill of figuring out how a software library works is one I've carried through in my career since.

1.3 Summary:

Summary Point	**Information**	**Notes**
Languages Used	C++	This was my first real experience utilizing external software libraries, so it was a good chance for exploration and experimentation
Equipment/Tools Used	N/A	Pure software project
Project Length	1 Semester	Project for my ECE297 Communications course
Project Difficulty (1-10)	7	A little challenging, but overall most of the work was in fixing the fine details rather than major design decisions

7. Table Top Arcade Basketball

Finished: November 2019

1.1 Motivation:

In the fall semester of 2018, I was taking a digital systems course and my partner Lisa and I were tasked with designing a custom project that uses the DE1-SOC FPGA Board (the main tool used within the class). Being a fan of basketball, I saw a way in which a small scale basketball arcade game could be implemented using the FPGA Board, leading to a functioning game that incorporates some basic electronics, digital game logic, and display output.

1.2 Design Details:

The project itself was split into 3 parts:

1. The physical structure which included the actual basketball elements as well as the ball sensor
2. The FPGA-implemented game logic
3. VGA FPGA-output

<u>1.2.1 Physical Structure</u>

The physical game was made out of some scrap 2x2s and thin plywood. The frame exists as a slanted box so that after every shot the ball can roll back to the player unprompted. The hoop was purchased from the local dollar store and attached to the wood base, and some masonite and wood glue were used to create a compartment to limit the ball from bouncing too much outside of the box.

Attached from the back in the compartment is a photoresistor. This allowed us to actually track when a ball went effectively through the hoop, as the photoresistor would experience a darkened environment. The photoresistor was then attached to a simple mosfet circuit that allowed a strong signal to be sent to the GPIO port of the FPGA board, which in turn lead to a logic signal to increase the score.

<u>1.2.2 FPGA-Game Logic</u>

The game state-system was implemented with Verilog. With basic on board accessories such as the GPIO and on-system clock, we were able to create a time based game that would reset the score at the end of the timer, and update the high score respectively. The game was started with a button located on the side of the physical structure, that then fed a signal to the GPIO of the FPGA.

1.2.3 VGA Output

The VGA Output was implemented through a process we learned in class. The work involved creating bitmap representations of the different elements we'd want to show on screen. We then used those files in our pre-existing VGA output module to properly display information acquired from the sensor and timer.

1.3 Debugging:

1.3.1 Physical Structure

The major issues that arose with the physical structure surrounded tracking the ball. A score was initially tallied using a trip-wire style laser and photoresistor combo, but there were inconsistencies with where the ball would fall through the hoop, leading to no interruption of the photoresistor setup. To fix the consistency issue, I changed the environment of the photoresistor to be in a shaded box where the ball needed to fall, and as the ball passed by the resistor the circuit was able to detect that change in light.

1.3.2 FPGA-Game Logic

Game logic was fairly standard. An on-board timer determined how long the analog circuit would track the ball for, highscore was saved in local random access memory, and there would be a full reset on the completion of the game. Given the simplicity of the system there was not a lot of debugging to be done.

1.3.3 VGA Output

The VGA had limited bugs as it used a module acquired from the course.

1.4 Summary:

Summary Point	**Information**	**Notes**
Languages Used	Verilog	Was my first large project with Verilog, gained more experience later on in follow-up courses.
Equipment/Tools	DE1-SOC FPGA Board, Simple BJT circuit	A lot of experimenting with

Used	with Photoresistor, Basic woodworking	the physical product, the software and logic was more straightforward.
Project Length	1 Month	Was the project for our digital systems course so a month of course time was allocated for it
Project Difficulty (1-10)	7	It was my first experience using very low-level digital hardware.

8. (Bonus) Two Floor Connect Four

Summary:

I designed and constructed, with the help of my peers, a multi-floor game of Connect Four. With fully functional piece input slide, and piece catching system. Due to the risk involved with this project, designing a foolproof mechanism for both catching and inserting the pieces was essential, as well as ensuring that the proper support system was in place to prevent any chance of the structure failing and hurting someone.