# AR Monitoring System

Design Document

Team Number: Dec1714
Client: Andrew Gui
Advisor: Dr. Johnny Wong
Arbaaz: Team Leader
Ben: Key Idea Holder
Sam: Key Idea Holder
Nipun: Communication
Patrick: Webmaster
Dheeraj: Communication
Dec1714@iastate.edu
Dec1714.sd.ece.iastate.edu

March 7, 2017
Version 1

# 1 Introduction

## 1.1 Project Statement

With the advancement in technology the ability to simplify life has become easier. With the possibilities of AR, we can minimize the human labor and make the tracking/oversight of autonomous vehicles easier. The intention of this project is to design an augmented reality monitoring system that will be integrated into the cab of an autonomous tractor.

## 1.2 Purpose

This project aims to provide an easy-to-use and vertical solution to remotely monitor and/or control autonomous vehicles. With enough refinement, the solution could be used for efficient oversight of a work force or autonomous fleets.

## 1.3 Goals

Upon completion of the project, we hope to have a complete set of software and hardware needed to remotely manage, store, and visualize a wide array of devices and machinery. This would include but not be limited to a lightweight device mounted on a monitored machine, a centralized server, and software written for the Microsoft HoloLens. We also hope to gain a greater understanding and appreciation for AR in general and to improve our abilities to be effective team members.

# 2 Deliverables

By the end of the spring semester we plan to have a flow chart and wiring diagram for all hardware integration as well as a rough prototype for our application which demonstrates the primary functionalities which we plan to bring to completion by the end of the second semester. By the end of the fall semester of 2017 we plan to have the capability to both create and save user profiles, use head tracking technology to control a 2-axis camera gimbal, view a 2D map in 3D, and show where monitored machines are on Google Earth in real time.

## Server

- Authenticate connecting devices
- Handles and sets up all communication from gimbal setup and hololens

## Raspberry Pi/Remote Device

- Have an assembled gimbal
- Research controlling the gimbal with streamed accelerometer data from Hololens - implement as proof of concept if have time
- Transmit GPS data to server (Raspberry Pi)
- Transfer video output to server (Raspberry Pi)
- NOTE: The Hololens will not communicate directly with the Raspberry Pi nor vice versa - instead all communication will pass through the server for both directions

## Hololens App

- Draw a scrollable 2D map in visual space
- Display simplistic 3D models to represent monitored vehicles on 2D map
- User authentication against server

# 3 Design

## 3.1 System Specifications

The specifications for this project are outlined in the deliverables section. Our goal for this semester of the project is to have an app that is able to act as an intermediary between the gimbal and the HoloLens. The specification that we have identified revolves around the three main components of this project: HoloLens app, server(s), and gimbal. What we have identified for the app are to be able to login and store multiple users. This ties into the specifications for the server. The specifications that we have for the gimbal are mainly related to the type of microprocessor that we choose to control the gimbal with.

### 3.1.1 Non-functional

- The software should be able to be adapted for many different purposes and inputs.
- The software should be secure. Unauthorized devices should not be able to feed false data nor should they be able to view device data.

### 3.1.2 Functional

- Ability to view monitored machine on map using Hololens app
- Ability to view a custom area map on the Hollens app
- Ability to asses machine status from Hololens app
- Ability to control gimbal/camera with Hololens app

### 3.1.3 Standards

Workers who are being observed may feel like they are being intruded upon. This feeling of intrusion may come from the fact that there is a camera in cabin and also data on their workday is being recorded. However we don’t believe that this will be a critical concern because data will only be stored only in emergency situations i.e. car crashes.

## 3.2 Proposed Design/Method

As we have discussed earlier, the main three components for this project are the: Hololens app, the server, and the gimbal. Each of these systems are meant to communicate with each other and work in unison.

The Design Method that we have chosen to pursue for this project is to have a 3 axis gimbal, and use a third party microprocessor to connect to the HoloLens, and to use the accelerometers of the hololens to control the motors of the gimbal. Additionally the microprocessor should also oversee the video streaming of the camera. It should be taking the view from the camera and sending that data to the server so the the HoloLens can access and view what the camera sees.

The server will be the backbone of communications between all the devices in the system. Since having devices detect each other among a sea of devices or perhaps behind a firewall isn't optimal, this is the best solution. Acting as a bridge between the observer and the observee, there must be a degree of differentiation between the two. Because the observed devices are located in remote locations that may not be physically secure, these devices should have limited access to the system and should only "offer" data to the server. The server should only consume the data for the purposes of passing it on to the HoloLens.

The HoloLens application will be set up using Unity. Unity offers wide-ranging support for the HoloLens being the officially recommended engine upon release. Installing third-party software along with integrated versioning makes it an ideal candidate for development. Along the lines of third-party software, some excellent sub-solutions for features are only available via Unity. Using the engine, we will display a map on a physical surface and use it to pinpoint the locations of various remote devices.

## 3.3 Design Analysis

We have created a basic class diagram for the server. This diagram is likely to be revised as we progress through our project. We have also generated a basic wiring diagram for our camera-to-raspberry pi setup as well as a list of the parts needed for this. Finally, we have set up a server and a basic gui.

# 4 Testing/Development

We have not begun any testing for our project yet as we do not have our three primary components set up yet.  We plan to begin completing the milestones for our project which constitute the groundwork of our main systems once we return from Spring Break.  Once the components for our system reach a testable state, then we will begin concurrent testing in order to guide our development process.

## 4.1 Interface Specifications

The main application will be run on the Microsoft Hololens. A corresponding unity project and C# application will provide the 3D graphics engine that renders the gui and map application correctly from the user's point of view. This application also makes API calls to the server which will primarily serve to establish a relay of information to/from the raspberry pi. The Raspberry Pi will interface with several external hardware components. An external GPS module will be connected in order to gather accurate GPS data to be sent to the server. The Raspberry Pi will also communicate to the gimbal motors their position during runtime through an external motor controller. A camera will be used to stream video through the server via the Raspberry Pi. Finally, a bluetooth module will be used in the event that CAN bus information is integrated into the project plan next semester.

## 4.2 Hardware/Software

The software and hardware segments will be tested independent of each other for initial tests because the planned milestones to be accomplished by then do not support a full system. The software tests will consist of only the server and HoloLens software while the hardware tests will consist of the Raspberry Pi and the gimbal's motor controller.

## 4.2 Process

We have not begun testing for our project as we are not yet at a testable stage as explained under the *Testing/Development* heading. The planned method is to run a basic script on the Raspberry Pi to locally control the gimbal and run a script on the server to randomly move a dummy device. The expected results are the gimbal moving according and the HoloLens displaying a moving marker.

# 5 Results

As mentioned above we have not done any testing. Thus we do not have any information to communicate as of now.

# 6 Conclusions

The work that we have accomplished so far has laid out the ground for us to start building and testing systems to see what will work best for our project. These systems will be parts of the goals that we have set for ourselves: a server that connects our mounted monitoring system and our Hololens app. The best way to build and test our system is to build our 3 main components separately and test to see if they work on their own and then work to connecting them together.

# 7 References

Server Class Diagram: http://dec1714.sd.ece.iastate.edu/docs/ServerClassDiagram.pdf