AR-Stream: Two-way Streaming Infrastructure for Server-based AR systems

Background:
A larger project is being spearheaded by Dr. Mehmet Belviranli here on campus to use AR in clean room training. The project is planned to involve AR glasses connected to powerful servers to process user's video stream on the server, do object detection and data analysis and stream it back to the AR glass. As preparation for this effort, we need a specialized two-way streaming infrastructure to (1) have AR glass send captured video stream and (2) have the server transmit back set of bounding boxes and labels. The component to be developed during the field session includes only these two components and excludes other servers (e.g. object detection) or AR glass tasks. We (tentatively) plan to use Magic Leap 2 glasses (which will be provided by Belviranli) and do the processing on one of NVIDIA jetson devices. Since this project is part of a multi-discipline effort between Mines and University of Arizona, *modularity*, *portability* and *ease of use* are the primary requirements of the software being developed. Potential future development and research opportunities will be available if there is interest.

Project Description:
The project will involve two groups of students within the team to work collaboratively:

1. The AR team will be responsible for creating an AR app on the glass via the vendor's native C++ APIs (https://developer-docs.magicleap.cloud/docs/guides/native/mlsdk-native-overview/). This side involves the development of the following functionality:
    a. The *video stream* captured by the glass cameras will be streamed to the server.
    b. The *video annotation* (bounding boxes for the detected objects and corresponding text) coming from the server will be overlayed over the video captured earlier and projected onto the internal display of the glass.
2. The server team will be creating a portable Docker container that will be paired with the glass to perform offloaded computation. This side involves the development of the following functionality.
    a. The server will capture the video streamed by the AR-glass and write into a buffer for further processing by a future component called "Vir-Tech", which is excluded from the field session responsibilities but will be developed in parallel by another team.
    b. Vir-Tech will output the *video annotation* (bounding boxes and text) for each frame in the input stream. These outputs will be sent to the AR glass, in a streaming fashion.

The two groups will need to create and implement protocols/formats for both the video and information streams.

Project Goals:

1. Develop an AR-glass application and a Docker-based server application that implement the functionalities described above.
2. Design or designate a streaming protocol/format for the *video stream.*
3. Design or designate a streaming protocol/format for the *video annotation.*
4. Optimize both streams so that the round-trip communication latency is less than 30ms.
5. Fully document functionality and usability such that future work can take place.

Desired Skill Set:

- Experience with VR/AR platforms and programming
- Network programming
- C++

- Docker
- Backend Server Development
  - Python
  - Node.js
  - Etc.

Preferred Team Size: 4-5 Students

Location of Work: Mines Campus

Client Liaison:
Mehmet Belviranli
belviranli@mines.edu

Colter Snyder
csnyder1@mines.edu