Open Source VPS for Indoor Enterprise AR Use Cases

Categories: Interoperability/Standards

Background: Visual Positioning Systems or Services (VPS) determine a device's location and position/orientation based on features in images captured by the user's device rather than GPS signals. VPS systems create a map of any space by taking a series of images that have a known location and analyzing them for key visual features to create a fast, searchable index of those visual features.

To localize the user, the VPS compares the features received from the device in real time with those in the VPS' feature index. The accuracy of localization through a VPS is greatly affected by the quality of both the imagery indexed and sent when locating, and the location associated with it.

Executive Summary: This project will develop an Open Source Visual Positioning Service for indoor enterprise use cases. The question of how to manage spatial data (e.g., limiting the spatial map to only those spaces relevant to the user, and whether a solution can be deployed by an enterprise IT group on premises) can or should be in scope.

Problem statement

- Today AR experiences must often be manually initiated by the user and are not sensitive to the user's location/position and orientation. This is in part due to the fact that the device is unable to accurately position itself (and to detect visual anchors) in a user's indoor environment.
- It is not possible to use GPS for reliable user positioning in most enterprise AR workplaces. Beacons and other technologies (e.g., markers) are not always easy to install and maintain or are unsuitable in some settings.
- Having an indexed visual map of an indoor space on a user's AR display device requires expensive memory and may be computationally complex for the device, resulting in low battery life and/or latency.

Desired outcomes of the research project/problem resolution
Successful completion of this project will result in the following benefit for AREA members:

- The AREA members and the AR ecosystem will be able to use a basic Open Source VPS to build their indoor experiences and develop new value added services.
- Provider segment members will be able to build improvements (optimization, new features) upon the Open Source VPS project.

Technologies of interest/focus

- 3D mapping for visual positioning/localization of user's device
- Image processing pipeline
- Recognition of visual features for localization
- Management of spatial data (on device or in cloud)
- Premise-based VPS (deployment entirely on premise, but not on device)
- Cloud-based VPS

Inputs

The following inputs will be required for the successful completion of this project:

- Hierarchical Localization (Hloc), an Open Source library that builds 3D maps with Structure-from-Motion and localizes users based on images
- Access to AR-capable devices such as HoloLens, Magic Leap, Meta Quest, iPhone/iPad, Android devices for testing and development of demonstration videos
- At least one, ideally two, spaces representative of indoor enterprise AR workplaces.
- Example navigation/route for testing the VPS accuracy, latency, bandwidth or computational load requirements and other metrics.
- AREA member interview identifying any concerns or questions they may need to be answered about the use of VPS platforms or services.
-

Deliverables

Successful completion of this project will result in the following deliverables for use by the entire enterprise AR ecosystem:

- A basic Open Source Visual Positioning Service (or System) suitable for enterprise indoor use cases based only on standards or Open Source libraries.
- Interactive or multimedia content that demonstrates the Open Source VPS service running in at least two enterprise spaces/use cases.
- A step-by-step methodology for AREA members to follow to install and evaluate the Open Source VPS from a public repository.
- Recommendations for deployers of VPS for indoor localization.

A webinar and executive summary for public audiences are two additional deliverables.

Starting point and direction, including components of the final project below:

1. Basic Hloc functionality is working and, in a couple weeks (quick and dirty) can probably pull it apart into three parts:
    a. model training,
    b. server, and
    c. client components.
2. Fully implementing the GeoPose protocol would be next step.
3. Still needs to support many other aspects such as:
    a. geo-alignment,
    b. movable objects,

c. collaborative mapping,
d. sensor fusion,
e. mapping API,
f. common acquisition pipeline,
g. streaming and/or broker integration,
h. robust deployment etc.