Demo: Practical Mobile Augmented Reality

Puneet Jain
Duke University

Justin Manweiler
IBM Research

Romit Roy Choudhury
UIUC

ABSTRACT

We present demonstration of real-time indoor mobile augmented reality (MAR) on off-the-shelf smartphones. Our proposed system OverLay allows random physical object tagging from camera's viewfinder. Later, these tags can be seen by others from different angles, locations, and times. Our approach does not require changes to infrastructure, localization schemes, specialized cameras, or modification to phone's operating system. Designed and developed for current generation smartphones, our experiments in an indoor setup of 100 objects and 10 volunteers, OverLay shows promising 95% precision, 85% recall, and 480ms end-to-end latency in the median case. If made commercially available, OverLay can immediately apply to city tourism, PoI discovery, infrastructure maintenance, and object privacy.

Demonstration Setup

We expect to demonstrate our live system on Android smartphones in HotMobile 2015 workshop. The users shall be able to experience OverLay on provided smartphones (option of installing on attendees personal devices is under-consideration). The participants shall be able to tag or retrieve (subjected already tagged) random objects in the workshop venue by pointing camera at them. We shall allow simultaneous tagging and retrieval from multiple users to demonstrate that same object can be retrieved from different locations, angles, and phones. Moreover, a video demonstration of OverLay featuring first hand experience of a volunteer inside Coordinated Science Lab at UIUC will be shown during the demo session.

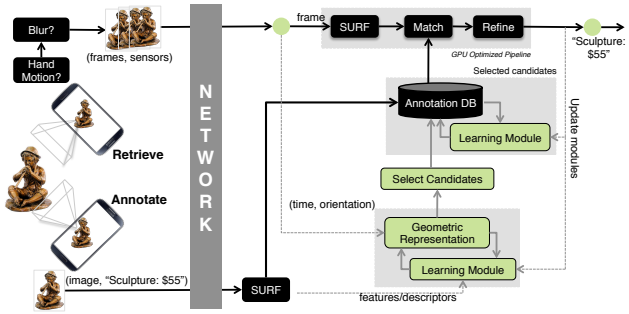

Figure 1: Schematic diagram of OverLay system showing tagging and retrieval process.

HotMobile'15, February 12–13, 2015, Santa Fe, New Mexico, USA.
ACM 978-1-4503-2793-0/14/06.
http://dx.doi.org/10.1145/2594368.2601472 .

System Details

Real-time MAR is a challenging problem because of trade-offs between accuracy, desired user experience, and practical deployment assumptions. Some of the well known challenges are as follows:

1. MAR requires indoor localization to reduce candidate search space. Unfortunately, best known practical schemes perform in the order of meters. Similarly, precise attitude estimation is also not possible because of noisy onboard sensors.
2. MAR requires heavyweight computer vision to run on device but current generation smartphones have limited processing power.
3. MAR may be enabled by offloading computation to the cloud but end-to-end network latency is too high for per frame offloading resulting unacceptable end-user experience.

Our exhaustive experimentation with *vision* and *sensing* based approaches in *isolation* conclude that neither of the two can solve this problem in real-time with acceptable accuracy. The vision based approaches require matching present camera view with existing set of tags in the database. Due to sheer computation requirement on a practical database size, this operation is not possible in real-time. Opportunities needs to be identified in leveraging lightweight sensors to reduce the search database size before applying heavyweight vision algorithms for the accurate matching.

Figure 1 shows architecture diagram of OverLay. Our core research contribution in the design of OverLay is in building "location-free" geometric model of the surroundings using inertial sensors. This model consists of relative positioning of objects in space and time — learnt from the natural use of our application. We utilize this geometric model and a past series of retrievals to reduce the candidate tags for future retrievals. Putting together this optimization on a GPU based pipeline, we are able to reduce processing latency from tens of seconds to 480ms without compromising on the accuracy. OverLay client is an Android app running on Samsung Galaxy S4 phones. OverLay server is a remote GPU equipped desktop computer satisfying our data storage and computation offloading requirements. An example retrieval from OverLay application is shown in Figure 2.

Figure 2: Live object retrieval using OverLay.