Collaborative Geometry-Aware Augmented Reality with Depth Sensors

Yuhao Ma
University of California, Irvine
6 Old Barge Road
Simsbury, CT 06070 USA
ypma@uci.edu

Kyle Boos
University of California, Irvine
710 Pascali Ct.
San Marcos, CA 92069 USA
kyle.boos21@gmail.com

Joshua Ferguson
University of California, Irvine
1512 Vallejo Drive
San Jose, CA 95130 USA
jfergusone@gmail.com

Donald Patterson
University of California, Irvine
5084 Donald Bren Hall
Irvine, CA 92617 USA
djp3@ics.uci.edu

Kevin Jonaitis
University of California, Irvine
2543 N. Dunbar St
Orange, CA 92865 USA
kevitis117@gmail.com

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.
Copyright is held by the owner/author(s).
UbiComp Adjunct *'14, September 13-17, 2014, Seattle, WA, USA*
ACM 978-1-4503-3047-3/14/09.
http://dx.doi.org/10.1145/2638728.2638765

Abstract
Augmented reality (AR) has progressed to the point where geometry-aware real-time applications involving multiple users are now possible. We present an approach for a collaborative augmented reality environment using an RGB-D camera and KinectFusion, collecting visual and depth data from a static environment that is used as a fiducial for multiple users. This allows for collaborative augmented reality environments where digital data and real world objects can appear to interact. We anticipate that the combination of technologies that we used in our prototype will soon be available in mobile devices and will support our approach to collaborative augmented reality.

Author Keywords
Augmented Reality; Wearable Technology; Collaboration; Kinect;

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Introduction

Improvements in both cameras and computing power have led to a large increase in the use of various augmented reality systems. These systems offer the ability to show auxiliary information to the user. Examples of this include overlaying text, graphics, audio, or digital objects onto a camera feed of the real world. This extra information has many practical applications: from showing a user what to do or where to go, to providing additional information about objects, to discovering nearby points of interest.

Although existing demonstrations of augmented reality require sensor data from accelerometers, gyroscopes, and RGB cameras viewing 2D fiducial markers, depth sensing cameras allow augmented reality systems to localize data without requiring specialized 2D markers. Multi-user interaction in an augmented reality setting is a step towards a new form of interacting with digital information that is well suited to wearable computers.

Background

Previous research done on collaborative work environments has shown that augmented reality increases productivity in certain types of collaborative environments [1]. These collaborative systems allow remote users to collaborate around digital content. However, these collaboration environments are often limited in their scope, or in the format of digital content that can be displayed. There have also been advances in collaborative augmented realities that enable users to interact with and modify digital objects simultaneously within the same space [2]. However, these implementations are often limited by their technological capabilities, often requiring 2D fiducials to track digital objects in physical space [3]. Recent advances in AR allow for real-time geometry-aware tracking and placement of objects [4][5].

Kinect Fusion

KinectFusion is a framework that can reconstruct a 3D model from the physical world in real-time by stitching together data from a depth camera as it is moves around a scene. The framework uses the data from the depth camera to perform Simultaneous Localization and Mapping (SLAM) [5]. KinectFusion accomplishes this mapping by using the Kinect sensor to collect depth data about the environment, and matching it to an internal model of the physical world using its implementation of the Iterative Closest Point (ICP) algorithm. This algorithm is run on a GPU to allow the process to be parallelized, maximizing its speed [6].

Architecture

Introduction

In order to simultaneously coordinate multiple users, we use multiple KinectFusion powered clients, each which communicates with a central server. Each user is given a prototype head-mounted Kinect-computer rig that is used to display the user's perspective and to keep track of the user's position with respect to a global origin. The server is used to synchronize views, to allow users to interact with digital objects, and to modify what other users see based on these interactions.

Camera Tracking

Our approach requires a primarily static environment in order to track the user's position and orientation, or pose. In order to do so, we utilize the KinectFusion framework to build a reconstruction of a given

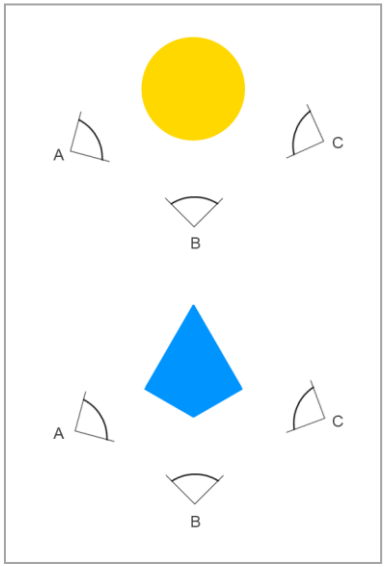

Figure 1: The top figure shows a radially symmetrical fiducial. For viewers A, B and C, the depth image is essentially identical. The bottom figure shows a radially asymmetrical fiducial. Viewers A, B, and C all have unique depth images.

environment. To build our scanned model, we move a Kinect sensor around the environment, allowing the KinectFusion framework to construct a model by integrating captured depth data into a scanned model. With our scanned model, we are now able to track the user's pose by calculating the 6 degrees of freedom (6DOF) transformation between the user's view and the view of the scanned environment model. The user's pose is calculated by KinectFusion's camera tracking and its parallelized implementation of the ICP algorithm. By applying the same transformation to other models that belong to the view of the statically scanned environment, we can successfully display them in the correct relative positions in the user's view.

Each iteration of ICP requires a prior transformation estimate from which a local search is initiated. For most runs, a sufficient guess is the last successful transformation computed by ICP as there is relatively little movement between consecutive frames; however, in order to obtain an initial guess for the first time we run the algorithm, we require that users check in at any of several "starting positions". This starting position is a location relative to the scanned object where the transformation between the statically scanned model's view and user's view is known. We also require that the scanned model be asymmetric to prevent ambiguity in pose estimation. ICP is more likely to succeed in estimating the position relative to the model if the depth image is unique to only one view. If the model has symmetric views, the depth image can represent one of several possible positions that may cause ICP to give an incorrect transformation. It is relatively easy to adjust ambiguous environments by making minor modifications to existing fixtures or introducing furniture into a scene.

Occlusion

One effect that adds to the realism of the augmented reality effect is occlusion of the digital content by parts of the real environment. Occlusion is the visual effect of hiding certain parts of a scene that should not be displayed such that only the elements that are shown are those that are closest to the user. Our system supports two types of occlusion: one where the statically scanned environment itself occludes the digital data and one where the real environment as a whole, including unregistered objects, can occlude the data.

Physics Application

With our approach, we can run physics simulations that take into account both digital and real-world objects. Research with KinectFusion shows the capabilities of performing physics simulations with digital particles that can interact with the real world environment using the Kinect sensor [6]. The digital data interacts with the real-world by responding to collisions with the statically scanned environment.

Consistent, Real-time Physics Collaboration

Applications such as a physics engine can be extended for multiple users so that they can collaborate with the system in real time. One collaborative example that we have prototyped is for multiple users to throw digital balls at a scene. Each user will have an instance of the client application, including a physics engine, running locally. However, individual user's physics simulations can diverge as they interact with the environment. Our solution to this problem is to use a synchronization server to make sure user views of the augmented environment to not diverge. We use a server scheduler because the user's actions are discrete. The user will

Figure 2: Example of occlusion. The real chair is part of the static environment. The chair hides part of the digital globe behind it.

request to throw a digital ball, this request is sent to the server and is scheduled to happen at a future time, the server then broadcasts this action to all of the clients and each client updates its physics simulation at the scheduled time. If the scheduling and notification occur within a small enough time, the action is sufficiently responsive to appear to be rendered in real-time.

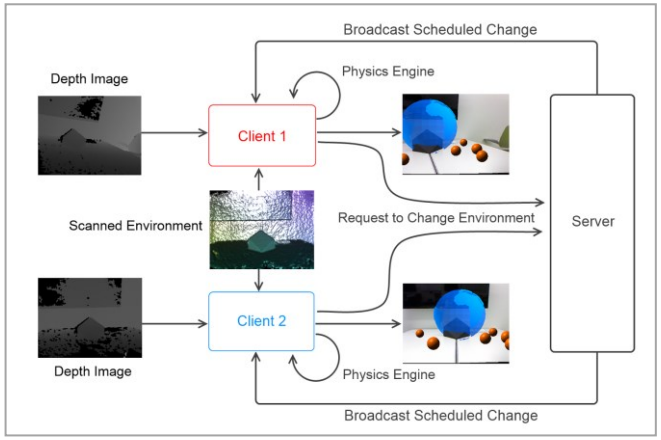

Figure 3: Depth images and scanned environment are fed in as input to the KinectFusion clients. Users are placed into the same coordinate space. A server synchronizes client physics engines so that views remain consistent.

Conclusion

In this paper we have introduced the architecture of a new system that we have built from existing and novel technology for real-time, collaborative augmented reality environments. We utilize the KinectFusion framework to create a static 3D model of the environment. Upon creating the scanned model, users check in to an initial position and their reference to both the model and all digital overlays is updated in real time on the server.

Our research explores potential benefits of collaborating in a system in which interacting with both the environment and digital content is possible. These forms of interactive, collaborative, augmented reality environments have the potential to change the way we users process information in computer-mediated environments.

References

[1] Gauglitz, Steffen, Matthew Turk, and Tobias Hollerer. "Augmented Reality-based Remote Collaboration." *The Eighth Annual Graduate Student Workshop on Computing* (2013): 17-18. Web. 11 May 2014.

[2] Benko, H.; Ishak, E.W.; Feiner, S., "Collaborative mixed reality visualization of an archaeological excavation," *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on,* (2004) 132-140. Web. 11 May 2014.

[3] Billinghurst, Mark, and Hirokazu Kato. "Collaborative Augmented Reality."*Communications of the ACM* 45.7 (2002): n. pag. Web. 11 May 2014.

[4] Marchand, E., Comport, A.I., and Chaumette, F. "A Real-time Tracker for Markerless Augmented Reality." *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on* (2003): 36-45. Web. 11 May 2014.

[5] Durrant-Whyte, H., and T. Bailey. "Simultaneous Localization and Mapping: Part I." *IEEE Robotics & Automation Magazine* 13.2 (2006): 99-110. Web.

[6] Newcombe, Richard, Shahram Izadi, David Kim, and Otmar Hilliges. "KinectFusion: Real-time Dense Surface Mapping and Tracking."*Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on* (2011): 127-36. Web.