Pervasive 2006 Workshop Proceedings, TSI 2006 (3rd Int'l Workshop on the Tangible Space Initiative), Dublin, Ireland, May 7, 2006, pp. 513–523

INVITE: 3D-Augmented Interactive Video Teleconferencing

Tobias Höllerer and Chris Coffin

Department of Computer Science
University of California,
Santa Barbara, CA 93106

{ccoffin,holl}@cs.ucsb.edu

Abstract. We propose a novel approach to mixed-reality teleconferencing that focuses on 3D augmentations of multi-camera 2D video streams. We exchange information about the geometric layout of all participating meeting sites, as well as the camera parameters for surveying these spaces. We can then correctly overlay 3D graphics on top of the video feeds, representing meeting content as well as highlight annotations and interaction tools for scene manipulation and bookkeeping of meeting contributions and decisions. Our primary contribution is a standard for the delivery and interaction with such data, which will allow for immersive meeting participation on a diverse set of devices, ranging from special-purpose 3D immersive environments to ultra-mobile platforms, such as cell phones and portable video players.

Keywords: augmented video, teleconferencing, tele-immersion, interactive TV

1 Introduction

We distinguish three common types of teleconferencing systems: First, there are multi-window desktop videoconferencing applications that provide audio and video feeds of meeting spaces together with shared 2D GUI applications, and sometimes also instant messaging functionality. This is the most common category, in which we find most commercial desktop video conferencing and remote collaboration systems. On the other end of the spectrum, there are efforts in tele-immersion, featuring full 3D reconstruction of meeting spaces that aims to establish a virtual reality experience in which all collaborating partners or their avatars are virtually co-located [16][19][6]. In between, there are quite a few research projects that look at the integration of 3D spaces and 2D videos and applications [3][8][2][13][18][1].

Our project shares the ambitious goals of the tele-immersion category, but is situated in the third (hybrid) category, where it introduces a new approach: instead of bringing 2D videos into a virtual 3D space, we propose to bring 3D graphics into 2D video space. We plan to make 3D-augmented video streams the main medium of the tele-meeting experience. From a task-requirements perspective there are two partially conflicting goals for the kind of remote conferencing system we envision: First, in line with the goals of the Tangible Space Initiative [7], we want to have as immersive and

engaging a representation of the meeting space as possible – this would point in the direction of full virtual tele-immersion. On the other hand, we also want to support highly mobile users with limited infrastructure for rendering and interaction. In particular, a popular current form factor for communication and entertainment exists in the form of cell phones or miniature music and video players (such as the Apple video iPod). On those platforms, all that is usually available in terms of output devices is a set of headphones and usually a small hand-held display. Cell phones also typically have pager motors for vibrations which can be used for rough haptic sensations.

In terms of making good use of limited graphics screen estate, live video is an advantageous medium for information exchange. The human brain is very capable in filling in missing detail to match an overall anticipated image if it resembles a natural scene. Videos are recognizable even at very low resolutions, and video telephony [14] is becoming more established with the arrival of third generation cell phone technology and global high-speed internet access in business and home markets.

We propose InViTe, the *In*teractive *Vi*deo *Te*leconferencing system, which tackles integration of real and virtual imagery in a video-based augmented reality approach. Our goal is to provide a highly interactive tangible experience with augmented videos.

Our system makes use of multiple camera feeds, as is fairly common for videoconferencing. Apart from videos of each meeting participant, we capture overview videos of the participating collaboration spaces. We also share information about the 3D geometry of those spaces and about the location of the cameras within them. We propose to provide participants with the ability to create augmentations directly on top of the video stream(s). Participants are then enabled to interact with these 3D annotations directly in the videos or create local proxies or duplicates to be viewed and manipulated in separate display spaces, potentially using more immersive 3D interaction hardware.

This paper reports on work in progress, and many of the more advanced concepts have not been implemented yet. However, we feel that it is important to share the design rationales with the TSI community [7] at this relatively early stage.

In the remainder of this paper, we present the proposed approach in more detail. First, in Section 2, we will focus on the main philosophy behind our shared video spaces and present our proposed system architecture and initial client interface. In Section 3, we classify potential 3D augmentations into several semantic categories, highlighting a few usage scenarios. We also discuss direct interaction with the video feeds in Section 3.2. Our goal is a tangible experience that is scalable to hardware platforms of varying complexity, which is discussed in Section 4. We present conclusions and give an overview of future work in Section 5.

2 Shared Augmented Videos: Approach and Architecture

Our overall approach to tangible tele-conferencing can be summarized as follows:

- **Video-based medium**: Leveraging user familiarity with traditional 2D videos, we use video streams as the main medium of exchange, which the participants augment and directly interact with.

Fig. 1. Initial client interface with main augmented video feed and interaction (left), supplementary video feeds and snapshots (center column), and a separate 3D interaction

- **Augmentation and interaction**: We overlay 3D computer graphics on top of the videos, representing both meeting content and annotation/communication tools.
- **Platform scalability**: 3D-augmented video streams can be viewed and interacted upon on a wide variety of computational platforms ranging from special purpose immersive environments to simple mobile phone platforms.
- **Separation of remote spaces**: Groups of remote collaboration partners maintain separate meeting spaces, each one accessible to all participating parties. We decided against one overall virtual shared space in order to keep the meeting scalable and the physical arrangement of local meeting spaces flexible, while still enabling all parties to understand and navigate the involved remote spaces.

Our work can be seen as complementary to tele-immersive approaches that aim to reconstruct virtual 3D scenes of the respective meeting spaces in real-time and present them to the remote partners on 3D immersive displays [16][19][6]. While we share the ultimate goal of presenting a highly realistic immersive impression of the remote spaces, our approach differs substantially in that it tries to sidestep today's technical difficulties in pursuing real-time 3D scene reconstruction and remote rendering with current computer and networking technology. Instead, we suggest an incremental approach. We start with 2D video streams and knowledge about the 3D layouts of the respective physical meeting spaces, enabling the user to switch back and forth between various cameras covering different viewpoints on the remote environments. By raising the number of available cameras, and by leveraging image-based rendering techniques, we will eventually be able to allow for increasing levels of virtuality, providing novel views and 3D interaction possibilities not provided by straight camera feeds. Until then, however, we plan to improve the augmentation of – and interaction with – 2D video streams and to bring in *just enough* 3D information to reap the benefits of tele-immersion while keeping interaction and navigation simple and straightforward.

Towards this end, we implemented simple augmentation of video streams with virtual data. Currently, we are working on a desktop-based client interface (Figure 1) as both a simple prototype as well as a method for testing the usability of the underlying framework. Our goals on platform scalability are summarized in Section 4.

2.1 Architecture

Our initial architecture follows a client-server model with one local server per participant site, acting as the sequencer for update requests. The server coordinates all incoming video streams for an environment, manages control of objects, holds the scene graph of the local environment, and distributes selected data and video to client applications. The server is also responsible for establishing connections to remote servers. Outgoing streams from a client must be first transferred to the local server, from where they are distributed to other clients.

As mentioned, we are maintaining a separation of remote spaces. Any *shared* virtual object must virtually occupy some physical space, either on the client side or at a remote location. The physical location of the object will determine ownership. That is, any requests to grab or manipulate objects will take place on the owner's server. Requests for control of an object (e.g., for any changes in orientation, position, scale, texture etc.) are sequentialized at the home server, and changes are propagated to all participants that view that object.

It may be that an object is introduced by a member of party A and a member of party B is interested in examining the object from all angles or close up, but does not wish to disturb the virtual object for the members of party A. B may then create a *copy instance*, choosing either to display the copied object and subsequent changes to all parties, or to keep it private. In the case of a private copy the client alone has control and the server is not aware of the instance's existence. Note that creating a local instance of an object does not mean that the original object needs to remain in the client's view – the client may choose to identify and cull the shared instance and only display the local copy allowing for both shared and local instances to reside in the same space. Wireframe or transparent rendering styles could be preferable alternatives to culling.

2.2 Transferred Data

At the initialization stage, we will exchange several vital bits of information. For the later stages of development we assume some knowledge about the geometry of the environment in which the teleconferencing is taking place. We will begin by transferring the data for the geometry of the room, as well as at least rough estimates of any physical geometry such as tables or desks. These can be transferred during the initial stages of establishing a connection. Several types of data will also need to be transmitted dynamically. For example, models may be loaded by a client and associated data such as texture and material information need to be transferred. We are also interested in providing for a larger variety of information such as annotations, billboards, or user avatars.

During the connection stage we will also establish information about the cameras in each environment. We will transmit not only camera positions and orientation, but also their intrinsic parameters. This will allow us more accuracy in representing the views from each camera as well as the placement of the virtual objects in the scene.

2.3 Implementation Details

One of the central goals of this project is to incorporate several video streams as well as a potentially high number of virtual objects into one central application. Several methods for transfer of the video, audio, data, and control streams have been explored. Currently we support a modified version of H.323 networking with optional support for the MPEG4 part 10 standard. In our current prototype, video streams are read from MPEG2 and h261 encodings. At this point, we are considering latency to be a larger issue than bandwidth. Varying levels of service will be explored later as we begin to expand on the client base.

To decrease latency, any updates to models will be transmitted from the owner's server. In other words, initially, changes are restricted to clients that "own" the virtual object. This is to prevent the need for a client to contact remote servers with changes and then to have those changes distributed from there. In the future, we may choose to implement data replication strategies as for example discussed in [12]. Note that the owner of an instance, that is the client allowed to make changes, is determined by the environment server in which the object resides in the case of a shared instance. All private instances are administered solely by the local client. Note also that with shared billboard objects and some annotations, while the position may be specified in shared virtual coordinates, the orientation will be determined by the respective client.

2.4 Initial Client Interface

Our initial client interface is designed for use with a desktop system and serves as a simple testing environment for augmented-video-based interactions and collaboration. The current interface is a precursor to more immersive future client environments and is subject to ongoing iterative changes. This section refers to the implementation depicted in Figure 1. Our current client-server implementation has not yet combined multiple input streams and our initial client interface displays the view (including virtual annotations) from one camera stream at a time. For bandwidth and screen real estate purposes, only one central stream occupies a large portion of the application window. Any other available streams are displayed as smaller lower fps windows off to the side of the main window, grouped by the physical environment they are part of. The client can select one of these alternate streams by simply clicking on the low resolution image. We believe as an early starting point that providing video feeds of even low resolution allow the user to have a greater understanding of not only the various view points available, but also the advantages each camera has in viewing virtual objects. Eventually we plan to integrate the video streams such that there is a smoother and more natural transition between camera views. At that point, video quality for the separate feeds could be driven by automatically determined awareness rather than user control [17].

The main camera view is surrounded by a border region which is used to represent a 3D space in front of the camera image. The camera view serves as a portal into the augmented meeting space. 3D objects that live in the area in front of it are not part of the shared augmented environment but can be easily moved into this space. We are currently experimenting with virtual distance cues and 3D interaction techniques to provide good 3D understanding for this space and support the transition operation. As depicted on the right side in Figure 1, we currently also provide an additional more spacious environment for 3D viewing, to and from which the client can drag and drop private instances of objects he wishes to manipulate in a setting removed from the active teleconferencing environments.

The camera feeds potentially represent separate environments which are physically far apart. We decided not to simulate one common joint virtual environment but instead aim to provide a good understanding of whatever different augmented physical spaces the meeting organizers and participants decided to register as part of the meeting. There may be just one physical augmented environment that all participants interact with, or several spaces that can each be selected for focus and interaction. In Figure 1, the session shows three camera views in one single registered environment.

3 3D Annotations

Even though 2D videos are the main base medium for the telecommunication in our project, 3D graphics plays an important role. In the long run, a multitude of cameras could give access to navigation and realistic interaction with a 3D image space [19][16]. In the short to medium term, we are interested in perfecting 3D augmentations of 2D video streams. Since we exchange all pertinent geometry and camera viewing parameters in the initialization procedure, we can correctly embed 3D graphics into the video material. We have a choice to make the augmentations photorealistic, employing, e.g., the real-time lighting and filtering techniques from [4], or to rely on stylized graphics to emphasize the artificial nature of certain types of annotations.

In any case, a complete photorealistic mediated virtual re-enactment of the remote meeting spaces is not our goal. Instead, we believe that we can create more powerful communication tools by embracing the opportunities that tangible interaction with general 3D computer graphics offers. 3D graphics that are situated in a physical environment can be used efficiently in a photorealistic or stylized way to guide the attention of meeting participants to important elements of the meeting content and interpersonal interaction.

3.1 Types of Annotations

Examples for 3D augmentations include:

- 3D content that participants are discussing in the virtual space between them (e.g., CAD models, architectural scenes, or the layout of a new 3D display environment)

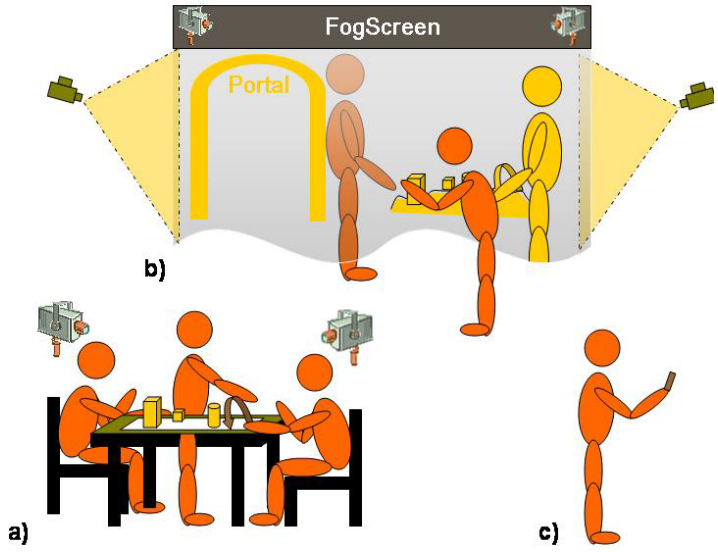

Fig. 2. Usage scenarios exemplifying the need for scalable UI technology: a) meeting room, b) Two-sided interactive FogScreen as an example 3D immersive environment, c) Portable device (e.g. cell phone)

- Highlights in the form of outlines or spotlight effects, and
- Tool-specific annotations such as speech bubbles that could maintain a history of important contributions to the meeting discussion.

Apart from the actual video augmentation, much of the benefit of our approach is based on the technology to interact with the augmented video stream (currently via mouse/pen gestures) to grab these 3D annotations from the video selectively and drag and drop them to a suitable 3D viewer that enables tangible collaboration on this data.

3.2 Interaction with videos

Client users create annotations by directly gesturing onto the currently active video stream, using mouse or pen input. 3D objects can be loaded or modeled in helper applications (in our initial client interface in the optional 3D window, cf. Figure 1) and then dragged and dropped into the video. Since the 3D scene and camera position are known, 3D elements will populate the video in correct perspective wherever the 2D pointer hits the first modeled surface.

As a usage example, imagine that a team of architects in the U.S. wishes to communicate with a group of their peers in Korea. The Americans load up a virtual model of a city surrounding the building they are currently constructing and place the model on their meeting table. The model is referred to often by both parties, and although the view for the remote clients can be changed, because of the nature and physical location of the model it is difficult to find a good viewing angle. One of the Korean architects wishes to move the object and examine it from several angles,

possibly also eliminating some of the outer buildings to get a better view of the new construction site. However, he wants to avoid moving the city model or, worse, eliminating buildings while the Americans are still actively referring to specific locations. To avoid disturbing the flow of conversation, the Korean architect creates a private instance of the object and begins culling buildings. Noticing a problem with the architecture of the new structure he loads his private instance into his own environment as a shared copy and begins pointing out the flaw to her fellow colleagues.

4 Scalable Tangible Experience

As seen, alternatively to manipulating 3D annotations directly on top of the video streams, the client may choose to perform manipulation of an object off to the side in a completely virtual environment.

Such a 3D viewer (currently simulated by the client-integrated 3D window on the right of Figure 1) can be implemented using the FogScreen [5] (at UCSB) or stereo projection in a "3D Smart Studio" (at KIST). Annotated videos will be able to be used on any video-playback platform, including small handheld devices such as cell phones. Depending on the computational and interaction affordances of the respective video-playback platforms, increasingly sophisticated interaction with the 3D augmented videos can be implemented. We will explore options for level of service, including for devices such as cell phones, on which the modeling of 3D objects or complex 3D environment navigation is not feasible. One option for such a client platform will be for the server to encode the augmentations as part of the video feed, while optionally providing simple 2D data (perhaps bounding boxes) for picking (object selection) and object manipulation. On a cell phone or PDA viewer, users will at least be able to watch the annotated video, point to the computer-graphics overlays, and transfer them to more suitable graphics and interaction environments.

Figure 2 illustrates three different meeting environments with varying user contexts, all enabling the respective participants to take part in the tele-collaboration: A common meeting room and table, the UCSB two-sided interactive FogScreen [5], and a single participant with a camera cell phone.

5 Conclusions and Future Work

We have introduced InViTe, a 3D-augmented interactive video teleconferencing system. While the system is still in its infancy, we have already verified the general feasibility and potential value of our approach.

One interesting observation is the similarity and applicability of the developed video augmentation and interaction mechanisms for interactive TV applications [9][10]. In our view, tele-meetings with their person-to-person video feeds present the first serious testbed and application domain for future gesture-based interactive TV technology, but instead of acting on pre-authored content, the participants interact with each other via live augmented videos. As in any two-way interaction system, it is a

good idea to first get the human-to-human interfaces right, before one can tackle the problem of interacting with an automated system.

In terms of the networking infrastructure, our initial architecture is a basic client-server architecture with regional servers that act as sequencers for all local objects. We will run experiments to determine the latency incurred by our system in different use cases, and may consider a more flexible replication-based approach in a future version if performance permits it.

Porting our system to multiple platforms to realize the scalable tangible infrastructure from Section 4, is an important agenda item once the current prototype has passed several milestone tests in real transpacific communication settings.

Allowing the client movement of a virtual camera has not yet been implemented, but is an upcoming step in this project. In order to reduce load on the servers and the network, we plan to perform these image-base calculations on the client side. Camera streams will be paused during camera navigation and will resume streaming from the server once the camera view has changed to take input from another camera view being streamed from the server. Streams of rendered views may be considered later as possible extensions to support video-only clients.

Future work may also include more extensions into photorealistic rendering, including support for correct lighting between the physical and virtual imagery.

We do not plan at this point to provide a complete backbone for animation or physical simulation. However, we plan to allow for animation and interaction with non-static (e.g. deformable) models, by providing an API for the server through which an application can update object(s) and if need be respond to stimuli on an object. Such an API will allow for a wider (unforeseen) variety of client interfaces to be developed. In many respects, an application to control an object would act as an automated client, performing arbitrary transformations on the object via the API, leasing ownership of the object when necessary.

Acknowledgements

This work is supported in part by a research contract with the Korea Institute of Science and Technology (KIST) through the Tangible Space Initiative Project; by research funds from the University of California, Santa Barbara; and by an equipment donation from Microsoft.

References

1. Bekins, D., S. Yost, M. Garrett, J. Deutsch, W. Htay, D. Xu, and D. Aliaga, "Mixed-Reality Tabletop (MRT): A Low-Cost Teleconferencing Framework for Mixed-Reality Applications", *IEEE Virtual Reality*, to appear, short paper, March 2006
2. Billinghurst, M. Shared space: An Augmented Reality Approach for Computer Supported Collaborative Work. In: *Virtual Reality: Systems, Development and Applications*, 3(1), 1998, pp. 25-36.

3. Breiteneder, C.J., Gibbs, S.J., and Arapis, C., Teleport – An Augmented Reality Teleconferencing Environment, In: *Proc. of Eurographics Workshop on Virtual Env. and Scientific Vis*., 1996, pp. 41-49.
4. Candussi, A., T. Höllerer, and N. Candussi, Real-Time Rendering of Realistic Trees in Mixed Reality, In: *Proc. IEEE/ACM ISMAR '05 (Intl. Symp. on Mixed and Augmented Reality)*, 5-8 October 2005, Vienna, Austria, pp.204-205.
5. DiVerdi, S., I. Rakkolainen, T.Höllerer, and A. Olwal, A Novel Walk-through 3D Display, In: *Proc. SPIE Vol. 6055, 605519, Stereoscopic Displays and Virtual Reality Systems XIII*, Jan. 30, 2006.
6. Gross, M., Würmlin, S., Naef, M., Lamboray, E., Spagno, C., Kunz, A., Koller-Meier, E., Svoboda, T., Van Gool, L., Lang, S., Strehlke, K., Moere, A. V., and Staadt, O. 2003. blue-c: A Spatially Immersive Display and 3D Video Portal for Telepresence. *ACM Trans. Graph.* 22, 3 (Jul. 2003), 819-827.
7. Ha, S. and C.K. Ahn. Introduction to TSI Project in KIST. In: Proc .2nd Intl. Workshop on the Tangible Space Initiative, ICAT 2004 (14th Intl. Conference on Artificial Reality and Telexistence), Nov.30 – Dec. 2, 2004, Coex, Korea.
8. Han, J. and B. Smith, CU-SeeMe VR Immersive Desktop Teleconferencing, In: *Proc. 1996 ACM conference on Multimedia (MM'96)*, pp. 199-207.
9. InformITV – Interactive TV, http://informitv.com/
10. Interactive TV Today, http://www.itvt.com/
11. Kim, L., Ko, H., Park, M., and Byun, H. 1999. Interactive virtual studio and immersive televiewer environment. In: *Proc. ACM VRST '99 (Symposium on Virtual Reality Software and Technology)*, London, United Kingdom, December 20-22, 1999, ACM Press, New York, NY, 172-173.
12. MacIntyre, B., Exploratory Programming of Distributed Augmented Environments, Ph.D. Thesis, Columbia University, 1999.
13. Miwa, Y. and C. Ishibiki, Shadow Communication: System for Embodied Interaction with Remote Partners. In: *Proc. 2004 ACM Conference on Computer Supported Cooperative Work, CSCW 2004*, Chicago, Illinois, USA, November 6-10, 2004, pp. 467-476.
14. Myers, D.J. Mobile Video Telephony for 3G Wireless Networks, McGraw-Hill, 2004.
15. Panzic I.S., Çapin T.K., Magnenat-Thalmann N., Thalmann D., "MPEG-4 for Networked Collaborative Virtual Environments", *Proc. VSMM'97*, Geneva, Switzerland, 1997
16. Raskar, R., G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The Office of the Future: A United Approach to Image-Based Modeling and Spatially Immersive Displays. In: *Proc. ACM SIGGRAPH 1998*, pp. 179-188, 1998.
17. Reynard, G., Benford, S., Greenhalgh, C., and Heath, C. 1998. Awareness driven video quality of service in collaborative virtual environments. In *Proc. SIGCHI Conference on Human Factors in Computing Systems* (ACM CHI'98), Los Angeles, CA, April 18 - 23, 1998, 464-471
18. Schafer, W. A. and Bowman, D. A. 2005. Integrating 2D and 3D views for spatial collaboration. In *Proc. 2005 Int. ACM SIGGROUP Conference on Supporting Group Work* (GROUP '05), Sanibel Island, Florida, USA, November 06 - 09, 2005, pp. 41-50.
19. Yang, R., C. Kurashima, A. Nashel, H. Towles, A. Lastra, and H. Fuchs. Creating Adaptive Views for Group Video Teleconferencing – An Image-Based Approach. In: *International Workshop on Immersive Telepresence (ACM ITP 2002)*, 2002.
20. Yankelovich, N., Walker, W., Roberts, P., Wessler, M., Kaplan, J., and Provino, J., Meeting Central: Making Distributed Meetings More Effective. In: *Proc. 2004 ACM Conference on Computer Supported Cooperative Work, CSCW 2004*, Chicago, Illinois, USA, November 6-10, 2004. ACM 2004, pp. 419-428.