

# Grimage: 3D Modeling for Remote Collaboration and Telepresence

Benjamin Petit\*  
 INRIA  
 Grenoble Universities

Jean-Denis Lesage†  
 INRIA  
 Grenoble Universities

Jean-Sébastien Franco‡  
 INRIA  
 Université Bordeaux 1

Edmond Boyer§  
 INRIA  
 Grenoble Universities

Bruno Raffin¶  
 INRIA  
 Grenoble Universities

## Abstract

Real-time multi-camera 3D modeling provides full-body geometric and photometric data on the objects present in the acquisition space. It can be used as an input device for rendering textured 3D models, and for computing interactions with virtual objects through a physical simulation engine. In this paper we present a work in progress to build a collaborative environment where two distant users, each one 3D modeled in real-time, interact in a shared virtual world.

**CR Categories:** I.3.2 [Computer Graphics]: Graphics Systems—Distributed/network graphics I.4.5 [Image Processing and Computer Vision]: Reconstruction

**Keywords:** Telepresence; Collaborative 3D Interactions; Markerless 3D Modeling; Multi-cameras; PC Cluster

## 1 Introduction

Telepresence is of central importance in virtual and augmented reality applications. It refers to the ability to embed a person in a given location into virtual environments at other locations. While several technologies already allow visualization at different locations, the creation of virtual environments where persons in remote sites can interact with each other, as well as with virtual objects, is still challenging. One of the difficulty is to build virtual user models that carry enough information for such applications.

There are different applications of virtual immersive and collaborative environments such as video games, teleconferencing with the capability to manipulate and interact together with a CAD model, learn and train from remote sites or manage emergency crisis, among others. Traditional technologies for such applications are generally limited in several aspects. The main reasons are to be found in the virtual representations of the user which are often coarse and hence prevent people from interacting in a natural way. Most of the existing systems uses 2D representations obtained with mono-camera systems. While giving a partially true representation of the user, they are not immersive enough for 3D virtual worlds. Other systems, which are to some extent more suitable for 3D virtual worlds, use avatars, e.g multi-player games like *Second Life*.

\*e-mail: benjamin.petit@inrialpes.fr

†e-mail: jean-denis.lesage@imag.fr

‡e-mail: jean-sebastien.franco@labri.fr

§e-mail: edmond.boyer@inrialpes.fr

¶e-mail: bruno.raffin@imag.fr

Permission to make digital or hard copies of all or part 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. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. VRST'08, October 27-29, 2008, Bordeaux, France  
 Copyright 2008 ACM ISBN 978-1-59593-951-7/08/10 ...\$5.00

But avatars only carry partial information about users. Though real-time motion capture environments can improve such models and allow for animation, avatars do not yet provide a realistic representation.

One way to improve the sense of presence and realism is to compute real-time 3D models based on both photometric and geometric data. They yield more faithful lifelike representations that can include user appearance, motions and even facial expressions. Different strategies can be followed to obtain the 3D model of a person. The most common one is to use multi-camera systems. Such systems are already used for telepresence [Gross et al. 2003]. Nevertheless, existing 3D person representations, in real-time systems, often have limitations such as imperfect or coarse geometric models and low resolution textures. This as a result of the method used to reconstruct in 3D, e.g. stereo-vision or visual hull, and of the number of cameras and the algorithm computation time.

We present works in progress on using multi-camera real-time 3D modeling between remote sites. Such works are based on our previous experience with the Grimage multi-camera platform that enables one or several users present in the same acquisition space to interact with virtual objects animated by a physical simulation. We consider here the extension to multi-site for telepresence and collaborative environments.

The remainder of the paper is as follows. Section 2 presents the existing Grimage platform while section 3 describes how this environment has evolved to support multi-site interactions, remote collaboration and telepresence.

## 2 The Single-site Environment

The Grimage platform consists in a multi-camera environment, a PC cluster providing acquisition, computing and rendering resources.

We compute 3D geometric models and extract multi-view textures from a set of calibrated cameras without using intrusive markers or assumptions about the objects being filmed. We compute the visual hull of the objects by reconstructing their shape from the silhouettes extracted from the video streams. The algorithm we developed and use, called the Exact Polyhedral Visual Hull algorithm [Franco and Boyer 2003], computes in real-time the complete and exact visual hull polyhedron with regard to silhouette inputs.

The 3D models are injected into the SOFA physical simulation software [Sof]. SOFA animates virtual objects taking into account the presence of the real-objects through their 3D models. The scene is rendered after texturing the 3D models for an improved realism.

This single-site platform was presented at the Siggraph 2007 Emerging Technologies show by Allard & al. [2007]. Photos and videos are available at [Gri].

### 3 Remote Collaboration and Telepresence

We extend Grimage to a distributed context. We connect different clusters together inside the same telepresence application. Each cluster hosts a site with an acquisition area and renders the virtual world independently. A central node of the application gathers the 3D models from all sites and runs the physical simulation (Figure 1). This simulation detects collisions and computes the effect of each user on the virtual world. It is of course impossible to change the state of the input models themselves as there are no force-feedback device on the platform. Physical interactions between participants are also impossible for the same reason.

The simulation sends the 3D scene, i.e. the virtual objects, to all rendering nodes of each site. Each acquisition platform sends its 3D textured models to the other sites. The rendering nodes of each cluster merge these data sent by different sources and provide a consistent view of the scene. Rendering can be done on virtually any display device with different levels of immersion (e.g stereoscopic display, HMD, display wall, Cave).

Coupling the different software components involved into this project and distributing them on the nodes of a PC cluster for reaching real-time executions is performed using FlowVR [Flo ; Lesage and Raffin 2008], a hierarchical component oriented middleware based on a data-flow paradigm. FlowVR enforces a modular programming that leverages software engineering issues while enabling high performance executions on multi-cluster architectures.

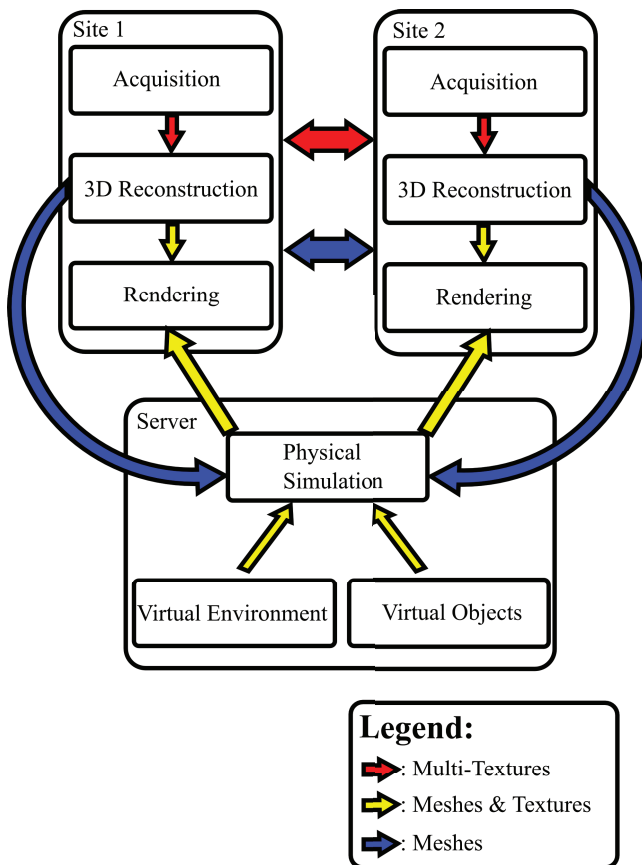


Figure 1: Application architecture for 2 multi-camera acquisition spaces.

### 3.1 Practical Setup

The demonstration is conducted on two platforms located at the VRST conference site. Each platform has an acquisition space of one squared meter at table height with five firewire cameras. It is suitable for hands/arms interactions. Each platform is equipped with a PC cluster composed of five PCs used for camera acquisition, one PC for computation and rendering and a laptop for supervision. The two clusters are connected by a gigabyte ethernet network using one PC as gateway between the two platforms. This PC runs the physical simulation.

### 3.2 Results

We succeeded in broadcasting the 3D models between sites at eleven frames per second while the physical simulation was running at 30 iterations per second. Performances reached during experiments enabled two users to interact with a common physical simulation without significant latency.

### 4 Conclusion

We presented an on-going work for telepresence and collaborative interaction using two multi-camera and real-time 3D modeling platforms and a physical simulation to handle the virtual interaction space. First results are very positive. From the user point of view the sense of presence is strong. We also noticed that users do not need to learn the interaction paradigms based on the ones people experience in the real world.

This work also stresses a number of weak points that we will address in the future. Texturing 3D model is a key component of telepresence, but improving texture quality tends to overload the communication network and leads to extra-latency in the application. Interactions using a 3D model and physical simulation do not allow some actions like object preemption.

### Acknowledgment

This work was partly funded by Agence Nationale de la Recherche, contract ANR-06-MDCA-003.

### References

ALLARD, J., MÉNIER, C., RAFFIN, B., BOYER, E. AND FAURE, F. 2007. Grimage: Markerless 3D Interactions. In *SIGGRAPH '07: ACM SIGGRAPH 2007 emerging technologies*, ACM, New York, NY, USA, 9.

FlowVR. <http://flowvr.sf.net>.

FRANCO, J., AND BOYER, E. 2003. Exact Polyhedral Visual Hulls. In *Proceedings of BMVC2003*.

Grimage. <http://www.inrialpes.fr/grimage>.

GROSS, M., WÜRMLIN, S., NAEF, M., LAMBORAY, E., SPAGNO, C., KUNZ, A., KOLLER-MEIER, E., SVOBODA, T., GOOL, L. V., 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, 819–827.

LESAGE, J.-D., AND RAFFIN, B. 2008. A Hierarchical Component Model for Large Parallel Interactive Applications. *Journal of Supercomputing* (July).

Sofa. <http://www.sofa-framework.org>.