Rapid Scene Modelling, Registration and Specification for Mixed Reality Systems

Russell Freeman, Anthony Steed, Bin Zhou
Department of Computer Science, University College London
Gower Street, London, WC1E 6BT, United Kingdom
r.freeman@cs.ucl.ac.uk, a.steed@cs.ucl.ac.uk, b.zhou@cs.ucl.ac.uk

ABSTRACT
Many mixed-reality systems require real-time composition of virtual objects with real video. Such composition requires some description of the virtual and real scene geometries and calibration information for the real camera. Once these descriptions are available, they can be used to perform many types of visual simulation including virtual object placement, occlusion culling, texture extraction, collision detection and reverse and re-illumination methods.

In this paper we present a demonstration where we rapidly register prefabricated virtual models to a videoed scene. Using this registration information we were able to augment animated virtual avatars to create a novel mixed reality system. Rather than build a single monolithic system, we briefly introduce our lightweight modelling tool, the Mixed-Reality Toolkit (MRT) which enables rapid reconfiguration of scene objects without performing a full reconstruction. We also generalise to outline some initial requirements for a Mixed Reality Modelling Language (MRML).

Categories and Subject Descriptors
I.4.1 [**Image Processing and Computer Vision**]: Digitization and Image Capture – *camera calibration, imaging geometry.*

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis – *depth cues, object recognition, shape, surface fitting.*

General Terms
Algorithms, Experimentation.

Keywords
Mixed Reality, Modelling from Images, Camera Calibration.

1. INTRODUCTION
Computer systems which combine images of the real world with rendered images of virtual worlds are called Mixed Reality (MR) systems. Milgram and Kishino [11] proposed that all realities lie somewhere along the Virtuality Continuum (VC), which ranges from completely real through to completely virtual environments. Whether augmenting real images with rendered virtual models (Augmented Reality; AR), or extracting images of real objects to mix into virtual worlds (Augmented Virtuality; AV), a scene description will usually be required.

Figure 1. Fully configured view, with the real tables appearing to occlude the virtual avatars.

Milgram and Kishino also proposed [11] the Extent of World Knowledge (EWK) scale, which ranges from completely un-modelled through to completely modelled environments. For many situations it is not necessary to have a completely accurate description of the real world, but rather some simplified approximation that considers only those aspects that are pertinent to the MR systems operation. E.g. in Figure 1 the positions of the chairs and tables in the room are required to augment the virtual avatars to the real images.

Current photogrammetry and image analysis techniques enable the digital camera to be used as an accurate and fast sensing device [9][10], capable of reconstructing detailed geometric scene models. Techniques employed in some applications, such as Canoma [3], ImageModeler [13] or PFBarn [15], allow interactive registration of primitive modelling shapes to one or more still image views. Whilst these tools provide a flexible platform for multiple camera calibration and the recovery of registered object orientated scene models, their general focuses are on post production sequences, where large amounts offline time are available to produce accurate models and registrations. If, once a scene model has been produced the real scene is slightly changed or rearranged the entire model will often have to be rebuilt or tracked to a new position, which can be a time consuming and technically demanding process.

VRST'05, November 7–9, 2005, Monterey, California, USA.

In this paper we present a MR demonstration (Section 2) where we use a new lightweight tool, the Mixed Reality Toolkit (MRT), to rapidly export a scene description to an adapted version of the Distributed Interactive Virtual Environment (DIVE) platform [4] (Section 5) to create a novel MR system. Isolating two distinct components used in the production of an MR system: modelling of the geometry from live video (MR Modeller – Section 3) and production of the live animated runtime (MR Runtime – Section 5), allows us to avoid the temptation of building a completely new MR system; rather we wish to continue to exploit the significant investment in existing virtual reality and graphics systems.

Key to this separation is the description of the scene that is transferred between a MR Modeller and MR Runtime. Although there are currently many different formats and languages, a format does not yet exist which incorporates all of the appropriate information required by many MR systems. The requirements for a Mixed Reality Modelling Language (MRML) are therefore also considered in Section 4.

2. DEMONSTRATION OVERVIEW

To demonstrate the separate MR Modeller and MR Runtime components we created an MR adaptation of an existing Virtual Reality (VR) simulation of a meeting scenario [8]. Instead of the simulation's participants perceiving a virtual audience in a completely virtual environment (VE), they can observe the virtual audience members incorporated into live captured images of their surroundings. Figure 2 identifies the different processes and inputs that are involved.

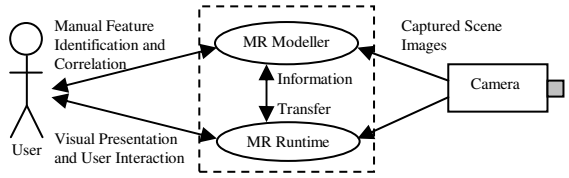

Figure 2. Integration diagram of a generic MR system.

The MR Modeller includes those techniques required to perform 3D model recovery and registration from 2D images along with extrinsic camera calibration. The MR Runtime process encapsulates those other parts of an MR system, including simulation, visualisation, interaction, manipulation and animation. Some examples of other MR Runtimes include [2][7][18], where limited geometric models of the scene are used to enable realistic relighting and illumination effects.

For this demonstration a new lightweight MR Modeller, known as the MR Toolkit (MRT) is used to register prefabricated 3D virtual furniture models to real imaged furniture observed from a fixed live camera view. The furniture registration process provides a method to calibrate a camera's relative extrinsic parameters, place virtual audience avatars within the imaged scene and define any occluding surfaces.

Prior to starting the modelling process, the camera's intrinsic parameters, such as focal length, principal point and image dimensions, are pre-calculated using the OpenCV toolkit [6][19], and are loaded to configure both the MR Modeller and MR Runtime components at startup.

3. MR MODELLER

Modelling a scene as a cloud or mesh of 3D points, as they appear in an image, requires depth estimations be made for each of the corresponding observed image points. This gives rise to a problem of potentially very large dimensions. However many scenes contain regular structured shapes, for which we only need estimates for corner positions whilst the surfaces can be mathematically interpolated between them.

Debevec, Taylor and Malik [12] demonstrate primitive shape modelling techniques, and manage to reduce the dimensions of the modelling problem still further. If, instead of describing each individual vertex of a shape in terms of (*x, y, z*), it is described in terms of a collection of variable parameters, such as *width*, *height* and *depth*, the problem dimensions are significantly reduced. By constraining the interrelations between individual primitives they [12] also suggested ways to construct a scene graph to describe the internal arrangements of a complete scene model, with greatly reduced dimensions. Based on these techniques we created the lightweight MRT to perform the registration of prefabricated VRML [16] models by bounding volumes.

The process of estimating the parameters of the scene model may proceed by first identifying common features in the captured images and on the virtual primitive shapes. An estimation process can then be applied to find the optimum positions, orientations and parameters of the model. The MR Modeller MRT used in the presented demonstration, uses an adapted two step initial guess estimator, proposed by Taylor and Kriegman [1]. An advantage of using this particular two step process is that it requires only a simple linear estimation algorithm.

Taylor and Kriegman [1] use a non-linear algorithm as a final stage to their estimation process; their initial guess estimator was not required to give an exact result, but rather a coarse approximation for the non-linear estimator to refine. The first step of the two step guess estimator is unfortunately therefore limited in its capability as it is unable to discriminate between primitives which are periodically rotated about an axis of symmetry. This causes estimations to become unstable for many shapes.

To remove these ambiguities we extended the work of [1] by applying an additional constraint that enabled the two step linear estimation algorithm be used independently of any final stage non-linear algorithm. The constraint requires that two vectors $\underline{d}_r$ and $\underline{d}_v$, measured from a cameras centre of projection to corresponding points on the real scene object and its virtual model edge counterparts respectively, be aligned and made parallel with one another as follows:

$$\left|\underline{d}_r^T\right| \cdot \left|M(\check{R}_c(\check{R}_p \underline{d}_v - \underline{T}_p) - \underline{T}_c)\right| = 1$$

$$\min \sum_{j=0}^{N} \sum_{i=0}^{M} \left(\left[\left|\underline{d}_{r_{ij}}^T\right| \cdot \left|M_j(\check{R}_{c_j}(\check{R}_{p_i} \underline{d}_{v_i} - \underline{T}_{p_i}) - \underline{T}_{c_j})\right|\right] - 1\right)^2$$

Where: $\check{R}_c$ *is the camera's estimated rotation matrix.*
$\check{R}_p$ *is the primitive's estimated rotation matrix.*
$\underline{T}_c$ *is the camera's translation vector.*
$\underline{T}_p$ *is the primitive's translation vector.*
M *is the camera's projection matrix.*
$|\ldots|$ *is vector normalisation.*
j*=0...N are the number of cameras.*
i*=0...M are the number of primitive shapes.*

Figure 3. Virtual furniture registered to live video images.

Incorporating this additional constraint helps the estimation process iteratively find the correct local minima. Model parameters and rotations are not initially known; therefore the estimation process must be applied iteratively until an error metric falls below a predefined value. The initial predictions for the parameters and rotations can be randomly selected or set to some predefined starting condition.

Additional rotational constraints were also applied using the MRT to those primitives that rest on a flat surface, and therefore must have a common single perpendicular free axis of rotation. This rotational constraint can be achieved by defining a primitive's rotational matrix $\underline{R_p}$ in terms of a 3D axis of rotation $\underline{n} = (n_x, n_y, n_z)$ and a rotation angle θ using the Rodrigue's Formula [10]:

$$R_p = I + \sin(\theta) X(\underline{n}) + (1 - \cos(\theta)) X(\underline{n})^2$$

Where: I is a 3x3 identity matrix.
$X(\underline{n})$ is the antisymmetric matrix formed from $\underline{n}$

$$X(\underline{n}) = \begin{bmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{bmatrix}$$

Until enough features are identified and pinned into place the estimation process will not be able to maintain a consistent registration result. To reduce fluctuations in the intermediate iterative results the MRT incorporates down-weighted predictions for unpinned features of the model. The predictions for the unpinned features are made by back projecting their current positions into the camera's image plane, and are updated after each estimation epoch.

As previously described, modelling of complex scenes can be achieved through subdividing the scene into primitive subcomponent shapes. As many scenes contain multiple instances of similar classes of object, such as furniture, books, people, etc, we can collect together these basic primitive shapes into contiguous groups, which might also be semantically useful in a MR Runtime. The requirements for image modelling are therefore reduced to one of deforming and registering particular generic classes of object to fit with individual imaged instances.

Figure 3 shows the MRT software that uses these techniques. The tables and chairs in an office space are being progressively registered using simple cubed primitive shapes representing their bounding volumes. The cubes are each of known dimensions, but are free to move across and rotate about an axis perpendicular to a ground plane.

The left image shows the first primitive being registered to the image. This primitive is used to calibrate the camera and define the ground plane. This progressive process enables the MRT to be used to rapidly find both the furniture and camera calibrations within the office space.

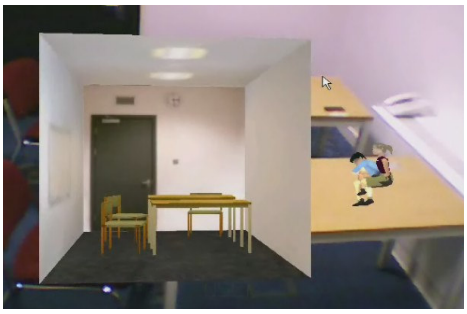

Figure 4. Scene imported into MR DIVE environment prior to camera and object

4. MR SCENE SPECIFICATION

In the presented demonstration the scene description is exported from the MR Modeller to the MR Runtime, containing registered prefabricated chair, table and room models, along with a camera viewpoint. The scene description is specified via a collection of files, including several VRML [16] 3D model files for the chairs, tables and room models and 3D DIVE [4] models files for the avatars. Also included is a scene description file, which brings together the prefabricated model files and specifies their individual absolute placements and orientations within the scene. The format used for the scene description file is the DIVE '.vr' format [4], which allows for registration specification and inline inclusion and combination of external 3D object model files. It also provides a mechanism for specifying camera viewpoints.

The scene description in this demonstration example contains two major types of object: real objects such as the tables and proxy objects in this case chairs and a room. The chairs are used to indicate where the virtual audience members will sit, and are not required for the occlusion purposes required of the tables. For those objects that are used to occlude the virtual audience it is important that the structural geometry and positioning be as precise as possible. However for objects that act only as proxies, the exact structure of the object is not as important as the positioning and choice of object itself.

The presented MR DIVE demonstration highlighted the need for extra attributes of the scene description to be combined into a single common format. The ability to describe a hierarchical object based scene graph, complete camera model along with primitive registration constraints and parameters is not currently possible using existing formats. The final stages of the modelling processes were therefore to indicate inside DIVE which objects were real and proxy, to select the virtual camera that matches the video and to seat the avatars on their chairs (Section 5). Figure 4 shows the initial scene description imported into DIVE.

There are other aspects of a MR system that could also be usefully combined into a single Mixed Reality Modelling Language (MRML) format, including lighting and model-based tracking descriptions. Tracking of both cameras and scene objects are important components of many MR systems. Model-based tracking methods have been shown [17] to be very capable of both wide-baseline initialisation and incremental tracking in real-time. Some such tracking methods use pre-calculated feature descriptors which could also be usefully integrated into a MRML format.

5. MR RUNTIME

DIVE [4][5] is a platform for the development of collaborative virtual environments, and has previously been extended to support immersive displays. To support mixed-reality displays and the demonstration as discussed in Section 4, three small modifications were required.

The first is support for an optional video underlay using DSVideoLib [14]. Second was functionality to assist with the alignment of objects to one another, to allow the arrangement of virtual objects to fit proxy objects. Functions to do this were written in DIVE /TCL and then added to the default menu system. They included functions to align the bases of two objects, or fit one object into the bounding box of another.

The final modification was the ability to selectively change the properties of objects in the scene graph to indicate that they are real objects and thus only occluders or are proxies and thus are invisible. To support this we extended DIVE's subjective view facility [5].

The avatar simulations shown in Figure 1 use the avatar libraries developed for a number of experiments on social reaction to avatar behavior [8]. Their behavior is scripted in DIVE/TCL and aside from pre-programmed animations the avatar can be instructed to fidget, look at the user or look at named objects in the scene. No change to this functionality was needed to support this demonstration.

6. CONCLUSIONS

Using simple primitive modelling techniques to register prefabricated VRML [16] models to live video images, we were able to rapidly describe a scene and calibrate a camera for use in a MR system. Using this scene description we presented a demonstration MR system that uses an adapted version of the DIVE [4] platform to augment virtual avatars to a real imaged scene.

The final, fully configured MR view in Figure 1 shows the real imaged room with two virtual audience avatars augmented, correctly occluded by the tables and seated on the chairs.

The demonstration highlighted the need for extra attributes of the scene description to be combined into a single format (MRML) that could be readily understood by other MR Runtime systems and components.

7. ACKNOWLEDGMENTS

Russell Freeman is funded by EPRSC and the Equator IRC. Thanks to Mel Slater & David Pertaub for access to the virtual human software.

8. REFERENCES

[1] C. Taylor and D. Kriegman, Structure and Motion from Line Segments in Multiple Images, IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 17 No. 11, pp. 1021-1033, 1995.

[2] A. Fournier, A. S. Gunawan and C. Romanzin, Common Illumination between Real and Computer Generated Scenes. In Proc. of Graphics Interface, pp. 254-262, 1993.

[3] Canoma: http://www.canoma.com/, accessed on 1st September 2005.

[4] Distributed Interactive Virtual Environment: http://www.sics.se/dive/, accessed on 1st September 2005.

[5] E. Frécon, G. Smith, A. Steed, M. Stenius and O. Stahl, an Overview of the COVEN Platform. Presence: Teleoperators and Virtual Environments, 10(1), pp. 109-27, 2001.

[6] Intel Research Lab OpenCV Toolkit: http://www.intel.com/technology/computing/opencv/, accessed on 1st September 2005.

[7] K. Jacobs, C. Angus, C. Loscos, J. Nahmias, A. Reche and A. Steed, Automatic Shadow Detection and Shadow Generation for Augmented, Proc. of Graphics Interface, 2005.

[8] M. Slater, D. Pertaub and A. Steed (1999) Public Speaking in Virtual Reality: Facing and Audience of Avatars, IEEE Computer Graphics and Applications, 19(2), pp. 6-9, 1999.

[9] M. Trucco, & A. Verri, Introductory Techniques for 3-D Computer Vision, Prentice Hall, 1998.

[10] M. Shah, Fundamentals of Computer Vision: http://www.cs.ucf.edu/courses/cap6411/book.pdf, accessed on 1st September 2005.

[11] P. Milgram and A. F. Kishino, Taxonomy of Mixed Reality Visual Displays, IEICE Transactions on Information and Systems, E77-D(12), pp. 1321-1329, 1994.

[12] P. Debevec, C. Taylor and J. Malik, Modelling and Rendering Architecture from Photographs, Proc. of SIGGRAPH 96, Computer Graphics Proceedings, pp. 11-20, 1996.

[13] RealViz ImageModeler & MatchMover: http://www.realviz.com/, accessed on 1st September 2005.

[14] T. Pintaric, http://www.ims.tuwien.ac.at/~thomas/dsvideolib.php/, accessed on 1st September 2005.

[15] The Pixel Farm PFBarn, PFTrack & PFMatch: http://www.thepixelfarm.co.uk/, accessed on 1st September 2005.

[16] The Virtual Reality modelling Language (VRML): http://tecfa.unige.ch/guides/vrml/vrml97/spec/, accessed on 1st September 2005.

[17] V. Lepetit, L. Vacchetti, D. Thalmann and P. Fua, Fully Automated and Stable Registration for Augmented Reality Applications, Proc. of the second IEEE and ACM International Symposium on Mixed Reality, 2003.

[18] Y. Yu and J. Malik: Recovering Photometric Properties of Architectural Scenes from Photographs. In Proc. of the 25th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, 1998.

[19] Z. Zhang. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, 1998