Mutual Occlusions on Table-top Displays in Mixed Reality Applications

Daniel Kurz*
Bauhaus-Universität Weimar, Germany

Kiyoshi Kiyokawa†
Osaka University, Japan

Haruo Takemura‡
Osaka University, Japan

Abstract

This paper describes an approach to dealing with mutual occlusions between virtual and real objects on a table-top display. Display tables use stereoscopy to make virtual content appear to exist in 3 dimensions on or above a table top. The actual image, however, lies on the physical plane of the display table. Any real physical object introduced above this plane therefore obstructs our view of the display surface and disrupts the illusion of the virtual scene. The occlusions result between real objects and the display surface, not between real objects and virtual objects. For the same reason virtual objects cannot occlude real ones. Our approach uses an additional projector located near the user's head to project those parts of virtual objects that should occlude real ones directly onto the real objects. We describe possible applications and limitations of the approach and its current implementation. Despite its limitations, we believe that the proposed approach can significantly improve interaction quality and performance for mixed reality scenarios.

CR Categories: H.5.1 [INFORMATION INTERFACES AND PRESENTATION]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.3.7 [COMPUTER GRAPHICS]: Three-Dimensional Graphics and Realism—Virtual reality

Keywords: mutual occlusions, table-top displays, mixed reality

1 Introduction and Motivation

Today, table-top displays, which display interactive content on a table-top-like display surface, are used in many different applications. They enable users to interact with virtual content as if they were real objects located on a table and are as such predestined for applications where natural user interaction is paramount.

Touch screens are an intuitive and very natural way of interacting with two-dimensional applications. Users interact with content by touching the display surface with one or multiple fingers. A three-dimensional equivalent to the touch screen would be a stereoscopic display that allows the user to interact with virtual three-dimensional objects with their bare hands. An extension to such intuitive, natural user interfaces is the use of real objects that have a virtual counterpart. Such *props* are located above the table-top and the user manipulates them to interact with the virtual scene.

*e-mail: daniel.kurz@medien.uni-weimar.de
†e-mail: kiyo@ime.cmc.osaka-u.ac.jp
‡e-mail: takemura@ime.cmc.osaka-u.ac.jp

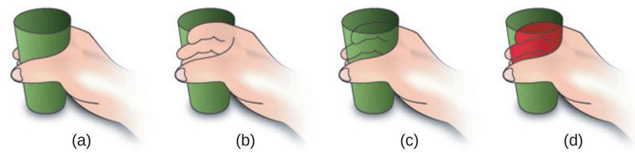

Figure 1: *Mutual occlusions between a hand and a cup in the real world (a). If the cup is a virtual representation on a table-top display, the cup no longer occludes the hand (b). Our approach achieves mutual occlusions (c) by displaying the virtual object (d) as a composite of two parts: the part produced by the table-top display (green) and the remainder (red) projected onto the real hand.*

Whether a prop or the user's hands, any real object located above the table-top display surface results in display inconsistencies that are caused by occlusions between the physical object and the representation of the virtual three-dimensional scene. The result is illustrated in figure 1b. Although the stereoscopic display creates the impression that the virtual object exists in 3D space on or above the table-top, its actual representation is physically located on the plane of the table-top and is therefore beneath the real object (prop or user's hands). This destroys the illusion of depth. Correct occlusions are therefore necessary as they provide essential depth cues.

To correctly handle occlusions in such situations, a second display is required that is located physically in front of the real objects and overlays them with the missing parts of the virtual scene. Possible solutions include augmented reality display setups such as head mounted displays (HMD), see-through displays or corrected projection directly onto the scene. We chose the latter approach and use a stereo projector mounted so that it projects onto the display surface from above (cf. figure 2). Using this projector, we can achieve consistent occlusions by projecting those parts of the virtual objects that should cover the real object directly onto them, as shown in red in figure 1d.

For this we need to know information about the shape, position and orientation of the real objects. Since we are dealing with a real-time application in which real objects move around unpredictably, offline calibration methods will not work here. Depending on what the *real object* is (a user's hand, prop, ...), different tracking techniques can be employed to ascertain this information. Once we have the necessary information, consistent occlusions between real and virtual objects can be displayed as a composite of the table-top stereo projection system and the top-projection.

2 Related Work

Handling occlusions between real and virtual objects is a fundamental task in mixed and augmented reality. The technique employed is strongly dependent on the display device used.

As traditional optical see-through HMDs are only able to overlay computer graphics by adding it to the real world, it is impossible to occlude bright real surfaces with anything virtual. The real world always shows through. Approaches exist that add a spatial light modulator (e.g. an LCD panel) to the HMD to entirely mask particular regions of the real world so that they can be occluded by computer graphics [Kiyokawa et al. 2003].

[Bimber and Fröhlich 2002] have presented a projector-based approach for optical see-through displays. In this case those parts of the real objects that are supposed to be occluded are masked out by projecting occlusion shadows onto them. Here the illumination of the real object is entirely projector-based and only those parts that should be visible (i.e. not occluded) to the user are illuminated.

In this paper we present a first approach to handling mutual occlusions using table-top displays. Our proposed technique is similar to *Shader Lamps* [Raskar et al. 2001], where video projectors are applied to animate real objects using image-based illumination. Their approach also works for moving objects [Bandyopadhyay et al. 2001], however, their aim is to interactively change the appearance of real objects. In our case we wish to project different virtual objects that occlude real objects. This is an interesting and challenging task, particularly as commercially available table-top displays are gradually becoming increasingly widespread.

3 Approach

We use a video projector to project those parts of a virtual object that should occlude a real object directly onto the real object itself. The remainder of the scene is displayed by the table-top display. This approach is new and seems to be the most appropriate solution.

Our approach does, of course, have some limitations. When real objects are semi-transparent or reflective, projecting onto them leads to artifacts since refractions and reflections are not considered. In this version, we have considered only projections onto white surfaces but non-white surfaces could be catered for using radiometric compensation [Bimber et al. 2005]. Another issue is that the shape of some real objects may cause projection shadows on themselves. Shadows can be dealt with by using multiple projectors, as is the case with [Raskar et al. 2001], but this further increases the system's complexity by adding at least one more projector.

Any kind of display that is located between the user and the real object – be it an HMD or a see-through display – is a hindrance to natural user interaction. It is well-known that wearing an HMD will increase the likelihood of user fatigue and can cause motion sickness [Melzer and Moffitt 1996] while a hand-held display or spatial optical see-through display clearly limits elbow-room. Our approach does not physically limit the user in any way.

Note that we do still need the table-top display. If we were to use only a top-projector to project both onto the real object and the table-top surface, the real object would cast shadows onto the table-top surface resulting in unsightly artifacts. Similarly, multiple top-projectors do not represent a good alternative to rear-projection when considering the situation where a real object occludes the virtual scene. In our approach this occlusion happens automatically. With an exclusively top-projection-based approach, we would have to take care not to project onto the real object which is an additional potential source of artifacts due to tracking errors.

Tracking errors are, however, another issue that we will have to cope with. Inaccurate tracking and system latency causes misregistrations between the different coordinate systems. The degree of misregistration is strongly dependent on the tracking technique and system but totally independent of the display setup in use.

3.1 Setup and Calibration

Our prototypical display setup is illustrated in figure 2. The table-top display consists of a back-projection screen that is illuminated by two video projectors (7,8) located in the base of the unit running at 60Hz. A mirror (5) reflects the projection to reduce the size of the installation. LC shutters (6) are mounted in front of the projectors and shutter them at 120Hz to support active stereo (cf. figure 3). Note that passive stereo is not viable here since we are projecting onto real objects that do not preserve polarization. The top-projector (1) is mounted above the user's head on a wooden frame fixed to the table-top display. We use an InFocus DepthQ 3D Video Projector that natively supports active stereo at 120Hz. It is mounted so that it projects onto the table-top surface from above. We decided to mount it as high as possible to maximize the projection size and resulting interaction space.

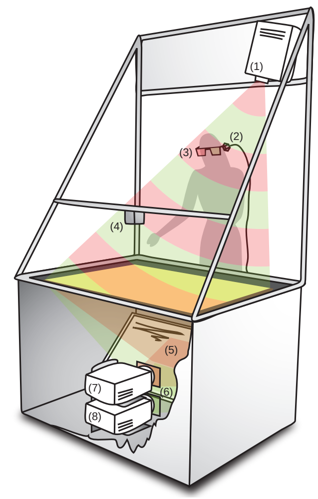

Figure 2: *Proposed display setup; active stereo is illustrated using color coding (red/green).*

The user wears shutter glasses (3) that shutter the eyes alternately in synch with the shutters of both the table-top display and the top-projection. This leads to proper separation of the stereo image pairs for both displays. Head tracking is necessary in order to be able to render properly. We use a Polhemus FASTRAK tracking system. A receiver (2) is attached to the shutter glasses. The emitter (4) defining the world coordinate origin is attached to the wooden frame close to the user. The whole setup is powered by a single PC. In order to drive three displays at once we use a Matrox Dualhead2GO in addition to an NVIDIA Quadro FX 3450 graphics card.

The system has to be calibrated to align the different coordinate systems of the three projectors and the tracking device. First of all, the two rear-projections are aligned manually using their built-in options. Since they only project onto the surface of the table-top, we do not need any information on their intrinsic and extrinsic parameters. All we need to know is the four corners of the projection which are obtained using measuring tape. For the top-projector we need to know its intrinsic and extrinsic parameters. The estimation of these parameters follows the principle described in [Raskar et al. 2001]. A set of three-dimensional points and their correspondences in the screen coordinate system of the projector are acquired by manually clicking points in a calibration rig in the top-projection with the mouse. This provides us with the corresponding 2d points in the screen coordinates system which we need in order to estimate the projector's parameters numerically.

In order to be able to render properly, we need a model of the real object. Depending on the complexity of the shape, the model can be defined either manually for simple shapes such as the wooden tool shown in figure 4 or scanned offline for more complex shapes

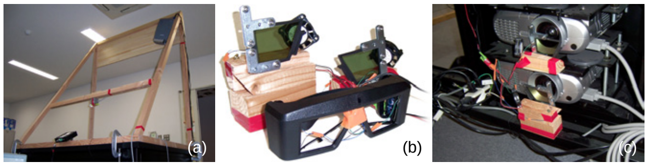

Figure 3: *Photographs of our prototype (a); the rear-projection is shuttered using a disassembled pair of shutter glasses (b,c).*

such as the shard in figure 5. Non-rigid objects, such as a hand in the case of bare-hand interaction, are considerably more complex as the shape has to be obtained online as discussed in section 4.2.

3.2 Rendering

During rendering, the virtual content is shown to the user as a composite of parts that are displayed by the table-top display and parts that are projected onto real objects by the top-projector. Since both displays run in stereo, each of those two parts has to be rendered twice – once for each eye. Accordingly, four images have to be generated for every single frame. Since our setup consists of three displays running at different frame rates, we need to handle two render contexts. We use GLWUT [Grosse 2007], a library that extends OpenGL to easily handle multiple windows. One window is located in the viewport of the top-projector and runs in stereo. The second window covers the whole horizontal span of the Dualhead2GO and therefore both back-projectors.

The images rendered for display on the table-top only show the virtual objects as here the real objects do not influence the rendering in any way. For each eye, an off-axis frustum is defined according to the position of the eye relative to the table-top. All virtual content is then rendered and the result is displayed. This is a typical procedure used to render stereoscopic images for head-tracked users.

The top-projector displays only those parts of the virtual scene that should occlude real objects. These parts need to be geometrically distorted so that they appear correctly from the user's viewpoint when projected onto the real object. To achieve this, multiple render passes are needed. As mentioned above, the following render passes have to be carried out twice – once for each eye.

In the first off-screen rendering pass, the virtual objects are rendered from the user's viewpoint to show proper occlusions with the real objects. The position of the virtual camera is therefore that of the eye facing in the user's viewing direction. The real objects are first rendered to the depth buffer only. The virtual objects are then rendered to the color buffer using z-buffering. The resulting color buffer contains only those parts of the virtual objects that occlude real ones. As these are rendered from the viewpoint of the user but projected from elsewhere, they also have to be transformed for projection into the coordinate system of the top-projector. The pass described above is therefore rendered to a framebuffer object with a depth and a color buffer so that it can be used as an input for the next rendering pass. Furthermore, the current camera transformation (i.e. modelview matrix and projection matrix) is also stored.

In the second rendering pass, the image that has been rendered from the user's viewpoint has to be transformed to the viewpoint of the top-projector that displays the final image. The camera parameters are set to those of the intrinsic and extrinsic parameters of the top-projector. We use projective texture mapping [Everitt 2005] to virtually project the resulting texture of the first render pass onto the real objects as if they derived from a slide projector located in the user's eye. This requires the camera transformation matrix previously stored. The real objects are then rendered from the viewpoint of the top-projector, textured with the occluding parts of the virtual objects and finally displayed.

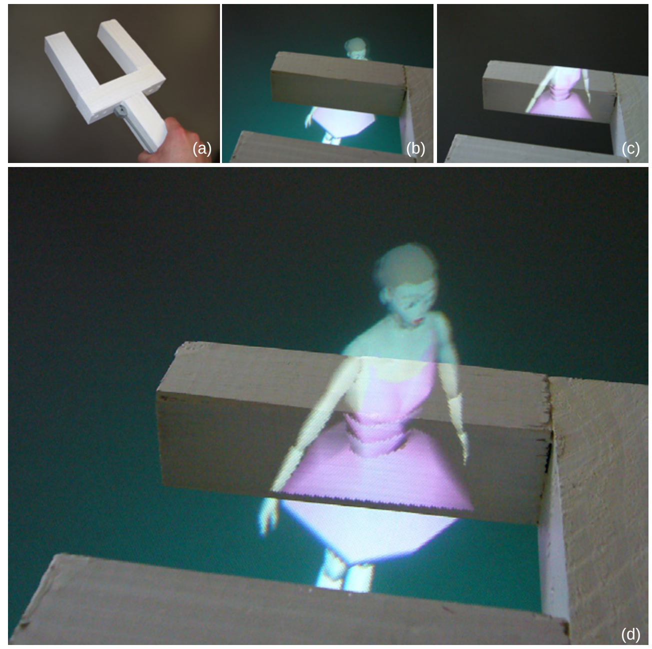

Figure 4: *Photographs of the simple occlusion testing application: the tool in the user's hand (a), rear projection only (b), front projection only (c) and composite (d).*

4 Applications and Limitations

What up to now we have abstractly termed *real objects* can in reality be nearly anything, opening up considerable potential for ideas for possible applications. Here we briefly discuss three possible applications.

4.1 Simple Occlusion Testing

To prove that our system works we start with a very simple application. The user holds a Y-shaped wooden tool (cf. figure 4a) which he or she can use to test for occlusions with virtual content displayed by our setup (in this case an image of a danseuse). This application can be seen in figure 4. Figure 4 (b) shows the rear projection only, which is how the application would look like on an ordinary table-top display. By adding the top-projection (c) it is clearly visible, that the danseuse is located in-between the prongs of the tool, not behind it (d). This simple application shows that the proposed system works and gives an idea of its potential.

4.2 Bare-hand Interaction

The "killer application" and initial idea for the display configuration proposed would be bare-hand interaction. Here, no real objects are used, only the user's hands which interact with purely virtual objects in 3d space. We believe that the visual perception of consistent occlusions will significantly assist the user in understanding the spatial relations between his or her hands and the virtual content being manipulated. In such a scenario the user does not wear any hardware on his or her hands (e.g. data gloves) so there is no tactile feedback. Occlusions provide the only cue to the user that he or she is actually touching a virtual object.

Figure 5: *Augmented shards: a scanned shard and possible context (a); choice of different possible contexts to evaluate (b); augmentation of the context of the shard while held in the user's hand (c).*

The main problem with this application is that the currently achievable quality of bare-hand tracking is quite poor. Obtaining a real-time model of one's hands is a very complex problem and the level of accuracy currently possible is not yet sufficient for proper rendering and projecting onto real hands. Promising approaches already exist so we can assume that the quality will approve in future.

4.3 Augmented Archaeological Finds

Another possible application is the augmentation of archaeological finds. On making a find, archaeologists then attempt to reconstruct the fragment's or shard's original context. Even though it might be easy to tell that a fragment must have been part of, say, a pot, there are still many different possibilities of what this pot may have looked like. In order to illustrate the possible former context, it is common to model it physically (e.g. in clay). This helps provide a good understanding of what this shard could once have been part of but is very static as it does not allow one to evaluate a choice of possible contexts. Figure 5 shows an application that employs the prototype display setup to virtually augment possible contexts around an archaeological find whilst it is held in the user's hand.

We scanned the shard using a laser scanner and modelled possible contexts around the scan using a 3d modelling software (cf. figure 5a). The final application allows the user to select and evaluate a choice of possible contexts. Figure 5c shows the user holding the shard in his hand. A tracking receiver attached to the shard tracks its current position and orientation and allows one to move it around freely and to rotate it to examine the possible context from all sides. Since the shard is rigid and its shape has been scanned offline, we have all the information necessary to render it properly.

Figure 5c reveals that there are problems with this technique. On the one hand there are misregistration artifacts that are due to slight errors in tracking and calibration and system latency. All systems are subject to some delay – here the time passed between the real object being at a particular position and the display of occlusion corrections for that position – so this is independent of the actual application. In our current prototype, the latency lies at around 50ms, which results in significant offset during fast movements.

In this particular case of projecting onto a shard, we also have to deal with specular reflections and shadows cast within the shard. Further work is needed to investigate how to deal with surfaces of this kind that are not natively suited to being projected onto.

5 Conclusions and Future Work

In this paper we have presented an approach for enabling mutual occlusions on table-top displays when used in mixed reality scenarios. To the best of our knowledge, this is the first approach that deals with this interesting problem that will play an important role in future systems. We have developed a first proof-of-concept prototype that demonstrates how it works and shows the potential of our technique. Virtual objects are able to occlude real objects by being projected directly onto the real objects. To achieve this we use an additional stereo projector mounted close to the user's head.

Our setup and implementation works, however, some limitations, problems and challenges remain. Inaccurate calibration and tracking can lead to misregistrations between the displays and real objects. We believe these problems will diminish in the near future as the quality of tracking improves. Similarly, state-of-the-art projector calibration will also help improve this. The quality needed to realistically assist a user in his or her task is not far off. We believe that users will be able to cope with slight errors as the benefits it offers for understanding spatial relationships far outweigh the negative effect of artifacts. In particular, applications which primarily display visual feedback to support a user interface stand to benefit from consistent occlusions. For instance, displaying the fact that a user touches a virtual object represents a significant improvement of an interface, even if there are misregistrations. Part of our future work will focus on investigating how the display of consistent occlusions supports the user's performance of their task and how distracting misregistrations and other artifacts are for the user.

References

BANDYOPADHYAY, D., RASKAR, R., AND FUCHS, H. 2001. Dynamic shader lamps: Painting on movable objects. In *Proc. IEEE and ACM Int. Symposium on Augmented Reality*, 207–216.

BIMBER, O., AND FRÖHLICH, B. 2002. Occlusion shadows: Using projected light to generate realistic occlusion effects for view-dependent optical see-through displays. In *Proc. IEEE and ACM Int. Symposium on Mixed and Augmented Reality*, 186.

BIMBER, O., WETZSTEIN, G., EMMERLING, A., AND NITSCHKE, C. 2005. Enabling view-dependent stereoscopic projection in real environments. In *Proc. IEEE and ACM Int. Symposium on Mixed and Augmented Reality*.

EVERITT, C., 2005. Projective Texture Mapping. http://developer.nvidia.com.

GROSSE, M., 2007. LibGLWUT – OpenGL Window Utility Toolkit. http://code.ioctl.eu/wiki/LibGLWUT.

KIYOKAWA, K., BILLINGHURST, M., CAMPBELL, B., AND WOODS, E. 2003. An occlusion-capable optical see-through head mount display for supporting co-located collaboration. In *Proc. IEEE and ACM Int. Symposium on Mixed and Augmented Reality*, 133.

MELZER, J., AND MOFFITT, K. 1996. *Head-Mounted Displays: Designing for the User*. McGraw-Hill Professional.

RASKAR, R., WELCH, G., LIM LOW, K., AND BANDYOPADHYAY, D. 2001. Shader lamps: Animating real objects with image-based illumination. In *EG Workshop on Rendering*.