2009-01-01

Forked:A Demonstration of Physics Realism in Augmented Reality

David Beaney
Technological University Dublin

Brian Mac Namee
Technological University Dublin, brian.macnamee@tudublin.ie

Recommended Citation

Beaney, D. &,MacNamee, B.,(2009) Forked:A Demonstration of Physics Realism in Augmented Reality. *The International Symposium on Mixed and Augmented Reality* (ISMAR 2009)Orlando, Florida, USA, 19 - 22 October. doi:10.1109/ISMAR.2009.5336479

Forked! A Demonstration of Physics Realism in Augmented Reality

David Beaney*and Brian Mac Namee†

DIT AI Group, School of Computing
Dublin Institute of Technology, Ireland

Abstract

In making fully immersive augmented reality (AR) applications, real and virtual objects will have to be seen to physically interact together in a realistic and believable way. This paper describes *Forked!* a system that has been developed to show how physical interactions between real and virtual objects can be simulated realistically and believably through appropriate use of a physics engine. The system allows users control a robotic forklift to manipulate virtual crates in an AR environment. The paper also describes a evaluation experiment in which it is shown that the physical interactions between the forklift and the virtual creates are realistic and believable enough to be comparable with the physical interactions between a forklift and real crates.

1 Introduction

One of the most important challenges facing the augmented reality (AR) research community is how best to create environments in which real and virtual objects physically interact together in a realistic and believable way. As a step towards meeting this challenge this paper describes *Forked!*, an AR application in which a real robotic forklift is controlled by a user to manipulate virtual crates. The forklift and the crates are physically simulated so that they interact together in a real and believable way. The system has been evaluated in a user trial in which participants were asked to perform the same tasks in a fully real environment (real robot and real crates) and in an AR environment (real robot and virtual crates) so as their performance in each could be compared.

The contributions of this paper are a description of a system in which real and virtual objects are involved in subtle and sophisticated physical interactions, and an evaluation of this system.

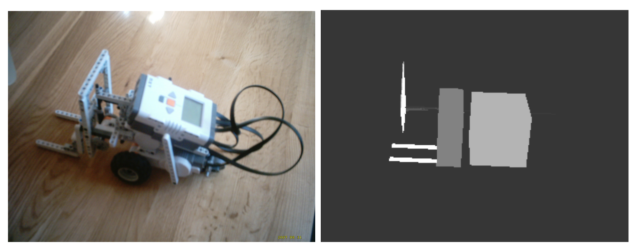
2 Implementing *Forked!*

The robotic forklift used in *Forked!* (shown in Figure 1(a)) was constructed using a Lego Mindstorms NXT robotics kit (mindstorms.lego.com) and controlled with a Microsoft XBox controller (www.windowsgaming.com) through a Bluetooth connection from a desktop computer. In order to create the illusion that the forklift interacted with the virtual crates, three key tasks must be achieved: AR registration; physics simulation; and rendering. The remainder of this section will describe each of these in turn.

AR Registration: *Forked!* uses fiducial-marker-based registration [2] using the ARToolkit API (www.hitl.washington.edu/artoolkit/). Fiducial markers are used for three purposes: to recognise the ground on which the robotic forklift operates; to define the reference frame within which the physics simulation takes place; and to determine the position of the robotic forklift. In order to recognise the ground on which the robotic forklift operates several fiducial markers that together define a single coordinate reference frame are used. These ground fiducial markers are also used as the initial reference frame for the physics simulation. Finally, to determine the position of the forklift within the scene, a fiducial marker is attached to it and the position of the forklift relative to the viewing camera is calculated.

Physics Simulation: The key to creating a believable and satisfying experience for users of *Forked!* is ensuring that the real forklift and the virtual crates appear to inhabit the same physical space, and exhibit behaviours that are consistent with this. In *Forked!* the Havok physics engine (www.havok.com) is used to simulate the physical interactions between real and virtual objects. Virtual crates are represented within the physics simulation as rigid body cubes and are simulated appropriately.

To allow the robotic forklift interact with the virtual crates it is represented within the physics simulation using a *physics proxy*, which is a rigid body based representation of the forklift as shown in figure 1(b). The position, orientation and movement of the physics-proxy forklift are synchronised with the position, orientation and movement of its real counterpart using *key-framing*, a feature of the physics engine. A key-framed physics entity is entirely under the control of the programmer, and has an effectively infinite mass. Once given a new position, the physics simulator calculates the path that the key-framed object must take to get there, moves the object there (over a specified amount of time), and simulates interactions with other objects that are along this path. Each time the AR registration system notices a new position for the forklift, the physics system is given a new key-framed position for it and updates accordingly.



(a) (b)

Figure 1: (a) The robotic forklift and (b) its physics proxy.

One challenge in achieving physics simulation within an AR application is integrating the AR registration engine with a physics engine - in this case ARToolkit and Havok. The main issues in achieving this are ensuring that both engines use a common coordinate reference frame, and that representations of the same object in both engines are not allowed to drift apart. To achieve this the camera position is used as common ground between the two engines. All positions, orientations, and transformations from Havok are transformed through the camera position, before bringing them back to a representation that ARToolkit and the rendering engine can use; and vice versa. Furthermore, because both the the initial coordinate reference frame of the virtual crates and the position of the forklift are based on the positions of fiducial markers, drift between the AR and physics engines in the system, a perennial problem for such systems, is reduced. The use of key-framing also helps

*e-mail:David.Beaney@student.dit.ie

†e-mail:Brian.MacNamee@comp.dit.ie

to reduce drift within the system.

Rendering: In *Forked!* OpenGL (www.opengl.org) is used for rendering, and the display hardware used is a Trivisio ARVision 3D HMD (www.trivisio.com). It is worth noting the use of shadowing and occlusion of virtual objects by real objects in the rendering system. The virtual crates cast simple shadows underneath where they stand, as it was found in informal experiments that this made it much easier for users to accurately determine the positions of the crates when trying to pick them up using the forklift. Similarly, realistic occlusion of the virtual crates by the forklift is important in creating the illusion of coexistence within the same physical space. The *dark matter* occlusion technique [3] is used. In this approach proxy representations of real objects are drawn to the graphics engine's z-buffer before virtual objects are drawn, resulting in virtual objects being suitably cropped as if they are being occluded.

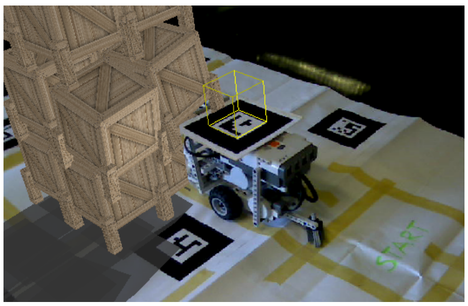


Figure 2: The real robotic forklift interacting with virtual crates.

Figure 2 shows a screenshot of *Forked!* running. The fiducial markers, the interaction between virtual and real objects, the use of shadowing, and the effect of occlusion can all be seen.

3 Evaluation

Evaluating the *realism* and *believability* of simulations is notoriously difficult due to the subjectivity of these issues [5]. For this reason the evaluation of *Forked!* was based primarily on the performance of participants in a series of tasks completed using the robotic forklift. For each task participants were asked to perform a *reality-based* version in which the forklift interacted with small cardboard crates, and a *mixed-reality-based* version in which the forklift interacted with virtual crates. The evaluation was designed in this way based on the expectation that if user performance in the reality-based and mixed-reality-based versions of the tasks was similar, this would support the conclusion that the interactions between the forklift and crates in both versions was also similar.

The first task given to participants, *pick-up-one*, asked them to drive the forklift to a suitable position and pick up a single crate. In the second task, *forklift football*, participants had to pick up a crate and move it between a set of goal posts. The reality-based and mixed-reality-based versions of each task were performed by each of the 11 participants in the study in a random order. The time taken and number of attempts required for each participant to complete each task were recorded. Participants were also asked to complete a survey detailing their experience completing the tasks.

The average times taken for participants to complete each task, with standard deviations, are shown in Figure 3. While the average times taken by participants to complete the mixed-reality-based tasks are slightly longer than the times taken to complete the reality-based ones, a paired two-tailed student's t-test does not indicate a statistically significant difference. The results of the survey tell a similar story. When asked to rate the ease with which participants could complete the different versions of the tasks they rated the reality-based versions as only slightly easier to complete than the mixed-reality-based versions. The reasons participants gave for finding the mixed-reality-based versions of the tasks more difficult to complete were that it was difficult to tell exactly where the virtual crates sat on the ground, and that the HMD was bulky and uncomfortable. The first issue can be addressed through better rendering, while the second can be removed in future evaluations by making participants wear the HMD in both versions of the tasks.

Overall the results of this evaluation, and the expectation that comparable performance in the two versions of the tasks suggests that the interactions between the forklift and the crates are similar in both versions, indicate that the interactions between the forklift and the virtual crates in *Forked!* are realistic and believable.

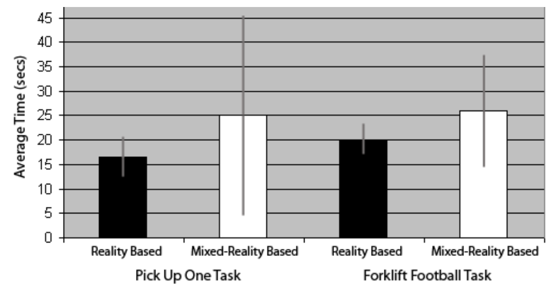


Figure 3: Average times taken, and standard deviations, for participants to complete the tasks in the evaluation study.

4 Conclusions & Future Work

This paper describes an AR application in which users control a real robotic forklift to manipulate virtual crates. The purpose of this application was to demonstrate that through appropriate use of a physics engine AR applications in which real and virtual objects interact together in a physically realistic and believable way can be created. This application uses *physics proxies* and *key-framing* to allow subtle and sophisticated interactions between these different classes of objects. The fact that the application responded realistically and believably enough for evaluation participants to complete reality-based and mixed-reality-based versions of different tasks in similar times, and that they were generally satisfied with the ease with which they could complete the tasks, shows that the approach taken was suitable.

Planned directions for future work include: the use of non-fiducial based registration methods (e.g. [4]; the use of more sophisticated rendering techniques and possibly 3D viewing; making use of advanced features of the physics engine; and allowing virtual objects to physically impact real ones (similarly to [1]).

References

[1] T. Aoki, T. Matsushita, Y. Iio, H. Mitake, T. Toyama, S. Hasegawa, R. Ayukawa, H. Ichikawa, M. Sato, T. Kuriyama, K. Asano, T. Kawase, and I. Matumura. Kobito: virtual brownies. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) - Emerging technologies*, 2005.

[2] M. Billinghurst, H. Kato, and P. I. The magicbook: A transitional ar interface. *Computers and Graphics*, 25(5):745–753, 2001.

[3] M. Fiala. Dark matter method for correct augmented reality occlusion relationships. In *IEEE International Workshop on Haptic Audio Visual Environments and their Applications, 2006. HAVE 2006*, pages 90 – 93, 2006.

[4] J. Mooser, S. You, and U. Neumann. Real-time object tracking for augmented reality combining graph cuts and optical flow. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–8, 2007.

[5] M. Slater. How colourful was your day? why questionnaires cannot assess presence in virtual environments. *Presence: Teleoperators and Virtual Environments*, 2003.