

ClonAR: Rapid Redesign of Real-World Objects

Michael Csongei*

Liem Hoang †

Ulrich Eck‡

Christian Sandor§

Magic Vision Lab
University of South Australia

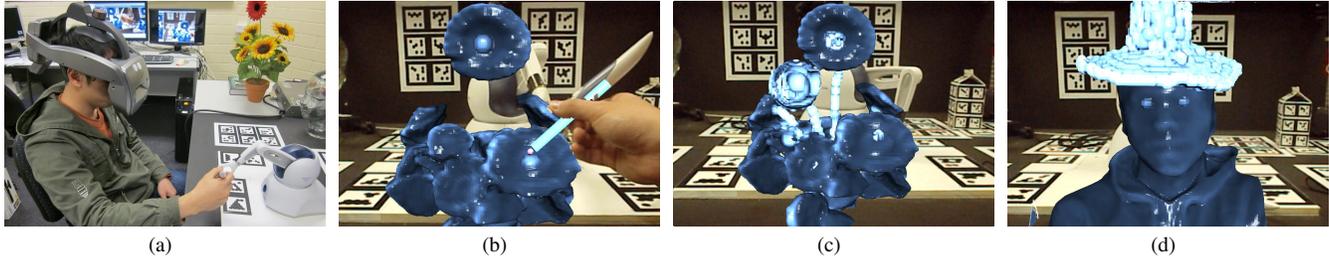


Figure 1: Users of our system can rapidly clone and edit real-world objects in our visuo-haptic Augmented Reality environment: (a) outside view. (b) view through the head-worn display; cloned flowers are being edited through a haptic device. (c) final model after deformation. (d) further example: a virtual hat was added to a cloned person.

ABSTRACT

ClonAR enables users to rapidly clone and edit real-world objects. First, real-world objects can be scanned using KinectFusion. Second, users can edit the scanned objects in our visuo-haptic Augmented Reality environment. Our whole pipeline does not use mesh representation of objects, but rather Signed Distance Fields, which are the output of KinectFusion. We directly render Signed Distance Fields haptically and visually. We do direct haptic rendering of Signed Distance Fields, which is faster and more flexible than rendering meshes. Visual rendering is performed by our custom-built raymarcher, which facilitates realistic illumination effects like ambient occlusions and soft shadows.

Our prototype demonstrates the whole pipeline. We further present several results of redesigned real-world objects.

Index Terms: H.5.1. [Information Interfaces and Presentation]: Multimedia Information Systems—[Artificial, augmented and virtual realities] H.5.2. [Information Interfaces and Presentation]: User Interfaces—[Haptic I/O] H.1.2. [Information Systems]: Models and Principles—[Human factors]

1 INTRODUCTION

Our vision was to enable users to rapidly clone and edit real-world objects through our prototype ClonAR. Cheaply scanning real-world objects has become possible through Microsoft's KinectFusion [4]. Users can then edit the scanned objects in our visuo-haptic Augmented Reality (AR) environment: cloned objects can be observed through a head-worn display (HWD) and edited with haptic feedback through the haptic device. The ability to rapidly deform real-world objects has many applications, including artistic works, home improvement, plastic surgery previews; potentially any modelling task currently performed by 3D computer modeling software can be augmented by our method. Our prototype could also be applied to medical training. With the ability to clone objects in real

time, medical students could be given the ability to perform surgery training on spontaneously scanned body parts.

AR applications traditionally render polygon meshes; however convincing interactive deformations of meshes are difficult to achieve. Instead, our whole pipeline does not use mesh representation of objects, but Signed Distance Fields (SDFs), the output of KinectFusion. SDFs are directly rendered haptically, for faster and greater flexibility over rendering meshes, and visually facilitating realistic illumination effects.

2 IMPLEMENTATION

An overview of our software architecture is provided in Figure 2. For marker-based computer vision we used a Canon VH2007 HWD. The framework used is MR Platform [7] providing hand masking and preprocessing of video data. For haptic rendering, we used a Phantom Omni. A Kinect, using KinectFusion [4], is used to capture models. The resulting Truncated SDFs (TSDFs) is then converted into two SDFs buffers; one optimized for haptic rendering, the other for visual rendering. Deformations including addition and subtraction of shapes can be performed upon implicit surfaces, through solid constructive geometry operations [5]. Our system is running on a single computer with an Intel i7 3.20 GHz, 6 Gigabytes of RAM running Windows XP 64bit. We utilize two 1 Gigabyte memory GTX 560 graphics cards, one for model capture, and the other for the visual rendering.

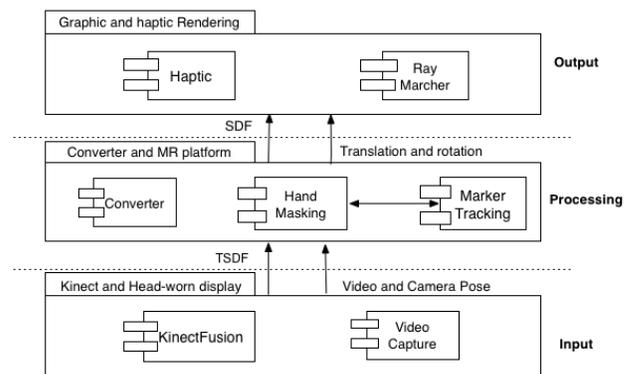


Figure 2: Overall system architecture (UML Package Diagram).

*e-mail: michael.csongei@gmail.com

†e-mail: hoald001@mymail.unisa.edu.au

‡e-mail: ueck@net-labs.de

§e-mail: chris.sandor@gmail.com

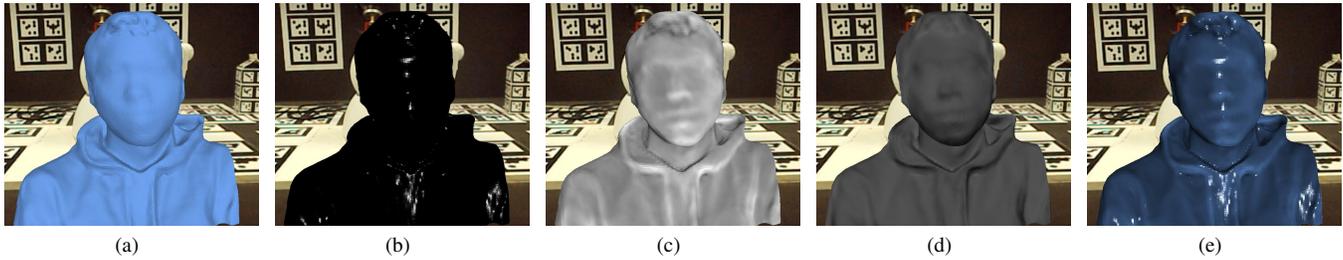


Figure 3: The different lighting components used in our raymarcher: (a) diffuse, (b) specular, (c) soft shadows, (d) ambient occlusion, and (e) the final result.

2.1 Haptics

Our haptic rendering method is based on the god object algorithm [8]. Salisbury and Tarr [6] have optimized the concept of Surface Contact Point (SCP) for implicit surface functions. The concept of a seed point is introduced for finding the closest point on the surface. They applied the Lagrange multiplier to find the more precise SCP, which reduces the distance between the seed point and the point on the surface of volume.

In our implementation for SDFs, firstly we calculate the distance from tool tip position to the surface that is determined by the SDFs. The gradient is computed by sampling the SDF for six points around the contact point. Applying the solution from Salisbury and Tarr [6] for implicit surfaces, we derive the closest point, or SCP, of the tool tip on the surface. Once the SCP is found, the spring-damper model is used to compute the force vector. The viscosity coefficient has been used to minimize the oscillation of force vector:

$$F = k * (p_c - p_t) - b * V$$

where F is the force vector, k is the stiffness, p_c is the surface contact point coordinate, p_t is the tool tip coordinate, b is the viscosity coefficient, and V is the tool tip velocity.

2.2 Raymarching

Raytracing algorithms mimic the traversal of light rays from the camera into a scene. Performing raytracing in real time requires efficient ray-surface intersection computations [2]. Traditional raytracing uses intersect comparisons against scenes described with primitive 3D objects. Raymarching or sphere tracing, in contrast, traverses into the scene by the distance to the closest object, reducing the intersection computations. The distance marched is calculated by solving implicit surface functions that describe the scene [2].

Raymarching implicit surface volumes have been used for displacement mapping [1] achieved in real time using parallelization. Using implicit surface volumes provides the ability to deform and alter the models in real time through solid constructive geometry [5].

Raytracing algorithms have computational independence for each ray, making them perfect for parallelization. Our approach is based upon parallelized GPU based raymarching algorithms for SDFs [5]. Visual effects such as ambient occlusion, reflections, refractions, and dynamic soft shadows, are difficult to simulate using other rendering algorithms, but are a natural result of raytracing algorithms.

An overview of the different lighting components used in our system are shown in Figure 3. Figure 3 (a) diffuse and (b) specular, show standard lighting effects. Figure 3 (c) soft shadows and (d) ambient occlusion, show advanced lighting effects, which require raymarching to be performed in real time. Using advance lighting, including dynamic shadows, has shown an increase in immersion for mixed reality applications [3].

3 CONCLUSIONS AND FUTURE WORK

We have presented our prototype for the rapid clone and redesign of real-world objects. After models have been scanned, they are visually and haptically rendered. Using SDFs enables us to deform our reconstructed models in real time. Direct haptic rendering of SDFs results in faster and increased flexibility over mesh rendering. Our raymarcher has the ability to produce advanced lighting effects, improving immersion.

For future work we plan to extend our application to capture models in real time. We also plan to increase the detail of captured models within our system, by using color textures and deduce haptic material properties from the visual appearance of the model. We propose porting our raymarcher to NVidia Optix to achieve increased performance with larger volumes, higher model resolution, and boost rendering quality. Further plans for our system include using multiple deformation objects, primitive shapes, or customized shapes from scanned objects. We also consider printing the deformed model with a 3D printer.

Real time deformation of real-world objects can be applied to many applications. We intend to focus on extending 3D modeling packages with our technique.

ACKNOWLEDGEMENTS Canon Inc., for HWD and MR Platform. eResearch SA, for loan of the Phantom Omni.

REFERENCES

- [1] W. Donnelly. Per-pixel displacement mapping with distance functions. In *GPU Gems 2*, pages 123–136. Addison-Wesley Boston, MA USA, Mar. 2005.
- [2] J. C. Hart, D. J. Sandin, and L. H. Kauffman. Ray tracing deterministic 3-d fractals. In *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 289–296, Boston, MA USA, July 1989. ACM New York, NY USA.
- [3] M. Knecht, C. Traxler, O. Mattausch, W. Purgathofer, and M. Wimmer. Differential instant radiosity for mixed reality. In *Proceedings of the 2010 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 99–107, Seoul, South Korea, Oct. 2010. IEEE Computer Society Washington, DC, USA.
- [4] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 99–107, Basel, Switzerland, Oct. 2011. IEEE Computer Society Washington, DC, USA.
- [5] I. Quilez. Rendering worlds with two triangles. In *NVScene*, pages 289–296, San Jose, CA USA, Aug. 2008.
- [6] K. Salisbury and C. Tarr. Haptic rendering of surfaces defined by implicit functions. In *Proceedings of the ASME 6th Annual Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 61–68, Dallasm, TX USA, Nov. 1997.
- [7] S. Uchiyama, K. Takemoto, K. Satoh, H. Yamamoto, and H. Tamura. Mr platform: A basic body on which mixed reality applications are built. In *Proceedings of the 1st International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 246–320, Darmstadt, Germany, Sept. 2002. IEEE Computer Society Washington, DC, USA.
- [8] C. B. Zilles and J. K. Salisbury. A constraint-based god-object method for haptic display. In *Proceedings of the International Conference on Intelligent Robots and Systems-Volume 3 (IROS)*, pages 146–151, Pittsburgh, PA, USA, Aug. 1995. IEEE Computer Society Washington, DC, USA.