ICAT-EGVE (2018)
G. Bruder, S. Cobb, and S. Yoshimoto (Editors)

Adaptive Filtering of Physical-Virtual Artifacts for Synthetic Animatronics

Ryan Schubert, Gerd Bruder, and Gregory Welch

University of Central Florida

Abstract

Spatial Augmented Reality (SAR), e.g., based on monoscopic projected imagery on physical three-dimensional (3D) surfaces, can be particularly well-suited for ad hoc group or multi-user augmented reality experiences since it does not encumber users with head-worn or carried devices. However, conveying a notion of realistic 3D shapes and movements on SAR surfaces using monoscopic imagery is a difficult challenge. While previous work focused on physical actuation of such surfaces to achieve geometrically dynamic content, we introduce a different concept, which we call "Synthetic Animatronics," i.e., conveying geometric movement or deformation purely through manipulation of the imagery being shown on a static display surface. We present a model for the distribution of the viewpoint-dependent distortion that occurs when there are discrepancies between the physical display surface and the virtual object being represented, and describe a realtime implementation for a method of adaptively filtering the imagery based on an approximation of expected potential error. Finally, we describe an existing physical SAR setup well-suited for synthetic animatronics and a corresponding Unity-based SAR simulator allowing for flexible exploration and validation of the technique and various parameters.

CCS Concepts

• ***Computing methodologies*** → ***Rendering; Mixed / augmented reality; Perception;*** *Simulation support systems;*

1. Introduction

Spatial augmented reality (SAR) [BR05] has several advantages over traditional head-worn or hand-held AR systems that makes it particularly well suited for ad hoc group or multi-user content. SAR has been used for a wide range of applications including telepresence [LWN*11], entertainment (e.g., at theme parks), training [HDN*15], advertising, architectural design [MPWG05], and art. SAR does not encumber users with head-worn or carried devices and has benefits over using simple 2D display surfaces for group experiences where physicality and related viewpoint-dependent effects (e.g., eye-gaze) are important. SAR is typically achieved by displaying light on a physical surface using monoscopic projected imagery [RWLB01] or shaped OLED displays, as opposed to stereoscopic display systems based on active shutter glasses, passive stereo preserving polarization, or autostereoscopic displays, which have their own tradeoffs and downsides.

SAR content is generally static or constrained to motion that remains on or near to the physical display surface, and in some cases the augmented surface can be physically actuated to achieve larger motion or deformations [LWN*11,IOT*08]. However, physical actuation comes with several possible drawbacks: added cost and complexity, higher maintenance, unwanted noise, additional power requirements, larger footprint, and practical limits on the types of motion, such as degrees of freedom and deformations.

In contrast to physical actuation, *synthetic animatronics* refers to the concept of conveying geometric movement or deformation of a SAR object purely through manipulation of the imagery, while the physical display surface on which the imagery is being shown remains static. We use the term *virtual object* when referring to the *virtual* content (what we *want* users to perceive) and *physical display surface* to refer to the *physical* surface on which imagery of the virtual object (potentially distorted, re-projected, filtered, etc.) is shown.

When the geometry of the virtual object does not closely match the physical display surface, there will be apparent, viewpoint-dependent distortions, with the exception of some trivial cases (e.g., a single, head-tracked user). In this paper, we always assume that visual distortions should be minimized for multiple or non-tracked users, which means that no perfect computational solution to the challenge of arbitrary distortion-free viewpoints exists. However, there are several approaches for trying to minimize the overall misperception due to these distortions, including a one-time optimization of the physical display surface shape and optimization of how imagery from the virtual object is spatially mapped onto the physical display surface.

In this paper, we present a novel approach, utilizing dynamic filtering of the imagery being shown on the physical display surface to avoid incorrect shape cues or disturbing visual artifacts in areas of

high distortion. In this paper we also present an implementation of the dynamic filtering approach that uses components of the existing hardware accelerated rendering pipeline to downsample imagery in real-time, using a per-pixel approximation of the distortion based on the current pose of an arbitrarily animated virtual object.

2. Related Work

The basic concept of spatially augmented reality was first presented in 1998 by Raskar et al. [RWF98]. Bandyopadhyay et al. expanded on the idea of "lifting" the appearance of a virtual object to replace the appearance on a different physical object, including dynamic effects like relighting, under the term *shader lamps* [RWLB01]. While shader lamps have been used to recreate some dynamic appearance effects and some simple animations, the geometric motion was generally constrained such that the virtual object and physical display surface remained geometrically approximately co-located.

For projector-based SAR systems, a significant amount of work has investigated methods to account for diffuse and specular material properties, surface coloring, environmental lighting, etc. when projecting onto arbitrary display surfaces by computing a compensation image [GPNB04], performing non-linear optimization of color compliance [LAS*11], or leveraging the presence of environmental light [LA12].

A considerable amount of work has also been done looking at tracking and registration for SAR objects that may move [ZXT*16] or deform [PIS15, SLS*17] at runtime, but it is important to note that unlike these methods, which first and foremost attempt to minimize geometric discrepancies between the real and the virtual, synthetic animatronics is fundamentally doing something different: intentionally causing non-trivial geometric distances between the augmented surface and the virtual object. Because of this difference, traditional projection mapping techniques based on registering virtual content to a tracked or otherwise captured physical surface do not address the viewpoint-dependent distortions caused by synthetic animatronics.

Unlike SAR, projection-based VR systems display virtual content that is typically independent of the spatial shape of the display surface, e.g., viewing head-tracked perspective correct 3D content on a planar surface. However, similar to issues with distortion due to a mismatch between real and virtual surfaces in SAR, additional viewers in projection-baed VR see distorted imagery that is not viewpoint-correct. In an attempt to mitigate this problem for a specific projection-based VR system, de Haan et al. explored using an averaged perspective camera view to distribute the distortion over multiple users while maintaining user-specific visual interaction elements that appeared correct for the corresponding user [DHMKP07].

Moreover, researchers looked into stereoscopic projection-based SAR for multiple untracked or single tracked users as a promising technology for creating an illusion of virtual and real objects coexisting within the same space [AFT13]. For instance, Schmidt et al. performed an experiment with a perceptual matching task and showed that even geometrically and perspectively correct stereoscopic projections in SAR often do not lead to a consistent depth perception of the virtual and real scene content, and they found strong interpersonal differences depending on binocular vision [SBS17]. These results in SAR match observed misperception in depth and shape of 3D objects in a wide range of stereoscopic display systems [BST14, LK03, RVH13, BAOL16]. While most stereoscopic SAR is currently limited to a single user, future projector systems might enable multi-user head-coupled stereoscopic rendering in SAR [BKKF13]. However, as stated above, stereoscopic SAR and the related issues are beyond the scope of the work presented in this paper.

Physical interaction with and manipulation of SAR objects was described and explored by Raskar and Low [RL01] as well as Thomas et al. [TMS*14]. Hochreiter et al. studied a specific touch-based interaction paradigm leveraging the physicality of a SAR surface [HDN*15].

Much augmented reality (AR) and SAR work relies on assumptions that the model of the virtual object and the physical display surface are geometrically very close and that the imagery was properly registered on the display surface, however this is not always the case. For relatively small geometric features, such as wrinkles being viewed from some distance away, the viewpoint-specific changes in the correct imagery would be slight enough that projected imagery, such as virtual shading to convey fine details, provides an effective perception of the desired features [BBG*13]. AR systems will realistically always have some registration errors. These errors arise from multiple sources including tracking accuracy, latency, or geometric uncertainty. MacIntyre and Coelho explored the idea of *level of error* filtering in video-based AR to change how augmented content is presented given estimates of the current registration error [MC00]. MacIntyre et al. expanded on this idea for registering outlines of objects that were created by growing the visual hull of the object based on the current estimate of the registration error [MCJ02].

Our method is also informed by other work describing how we perceive shape and various static visual shape cues [Tod04, WF06], how our mental models affect shape perception [Gre97], and shape cues arising from motion [BWW94].

3. Distortion during Synthetic Animatronics

Geometric distance between the virtual object and the physical display surface on which it is being represented results in varying displacements for points on the virtual object and where they appear on the physical augmented surface. This displacement is dependent on the viewpoint, and while the mapping of virtual object imagery onto the physical display surface can be made to have effectively zero displacement for a single viewpoint (i.e., head-tracked, monocular perspective rendering), there would still be apparent feature displacements for other viewpoints (including between two eyes of a user), and for untracked users.

For a given point p on the physical display surface and a mapping between the current pose of the virtual object such that the imagery at p corresponds to point p' on the virtual object, we define the distance between the surfaces as $dG = |p - p'|$. At point p, for a given viewpoint v_0, the point p'_0 corresponds to the expected imagery (e.g., color RGB value) of the virtual object (see Figure 1(b)).

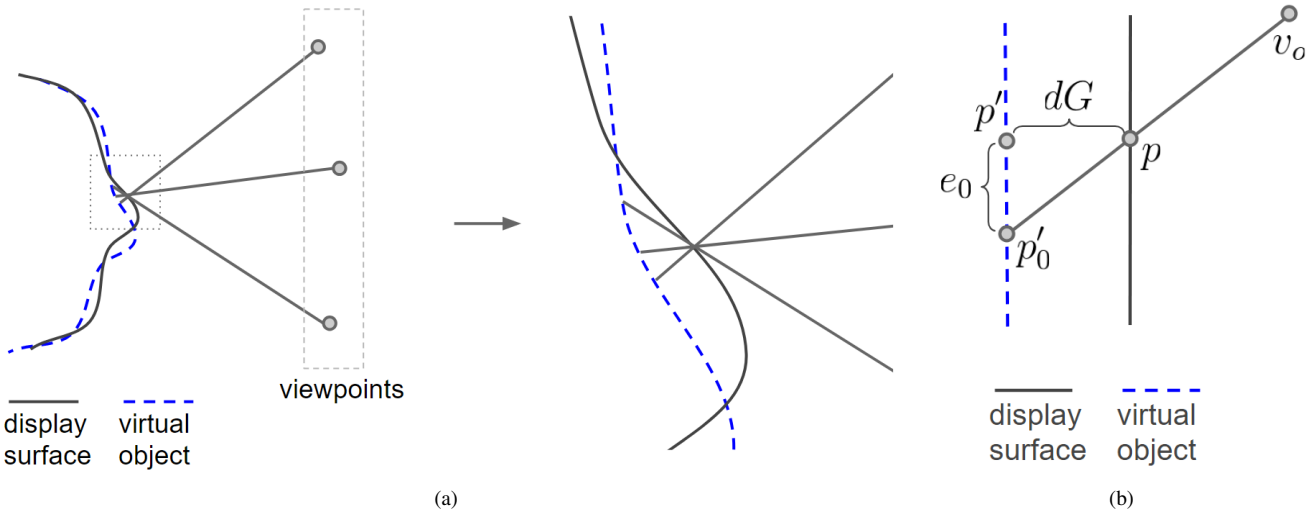

Figure 1: *Illustration of visual distortions in spatial augmented reality: (a) Viewpoint-dependence of incorrect imagery at example viewpoints when the distance between the virtual surface of the source object and the physical display surface increases. (b) The apparent displacement e_0 between the expected source object imagery p'_0 and the actual imagery from p', from viewpoint v_0.*

For small values of dG where both surfaces are approximately locally planar, the apparent displacement or error e_0 will be proportional to the distance between the surfaces: $e_0 \propto |p' - p'_0|$.

If all expected viewpoints or viewing regions are known ahead of time, the viewing information could be used to guide both the display surface shape as well as how the virtual object appearance is mapped onto the physical display surface. However, in the absence of any predetermined constraints, we make two generalizations that allow us to quickly filter the resulting imagery without needing any additional geometric information. The first is that $e_0 \propto |p' - p'_0|$ even for larger values of dG. The second is that across many possible viewpoints, the aggregate of all values of e is also approximately proportional to dG (see Figure 1(a)).

There is also a relationship between the *content* of the imagery on the virtual object and how apparent the resulting displacement is on the physical augmented surface. Specifically, lower visual structure (i.e., lower color or intensity gradient) in a local region of the virtual object means that the distortion from varying displacements due to geometric differences will be less apparent.

Figure 2: *Example of viewpoint-dependent distortion of nose and mouth imagery resulting from a small rotation of a virtual object. The left pane shows a simulated view of the virtual object while the right pane shows the same view of one possible augmentation resulting from that pose of the virtual object being displayed on a static physical display surface corresponding to the un-rotated state of the virtual object.*

4. Realtime Adaptive Filtering

Although the aforementioned distortions can be significant and cannot be avoided (except in very contrived or trivial cases), their negative effects on observers *can* be reduced. Using our approximate model of the potential expected displacement we can calculate, in real-time, for surface points the amount of possible "distortion", and we can thus try to mitigate it. Here, we describe our approach, which we call "adaptive filtering".

Given our model where the view-dependent distortion at a particular local patch on the physical augmented surface is directly proportional to the geometric distance (dG) between the virtual object and the physical display surface, we developed a realtime rendering process for adaptively filtering the displayed imagery on a spatially augmented object based on the value of dG, as computed at points on the physical display surface. This technique uses low-pass filtering to reduce the visual salience, where the magnitude of the filtering is proportional to the potential distortion (corresponding to dG). This maintains visually salient imagery in areas where it will be correctly perceived as part of the current pose, while reducing or preventing the perception of the incorrect shape cues of the underlying physical display surface.

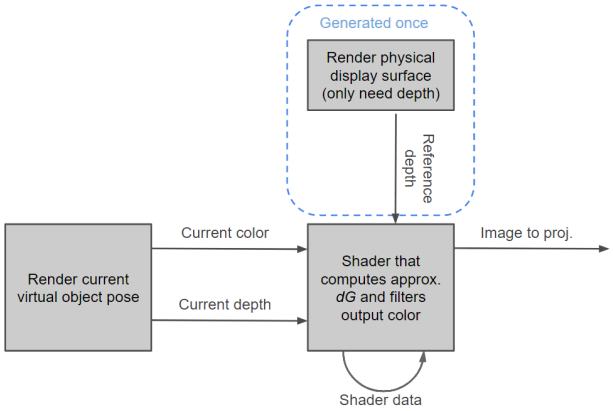

Figure 3: *The rendering process and data flow. The two run-time rendering passes are represented by the boxes on the lower left (first pass) and lower right (second pass).*

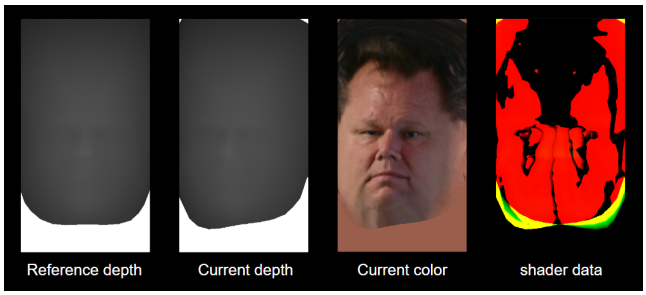

Figure 4: *Example of the textures used as inputs in the shader that generates the final filtered image. Note that the data is encoded using separate channels of a single RGB texture.*

4.1. Rendering Process

Rather than determining the actual shortest distance between the virtual object and the physical display surface, we instead compute dG as an approximation of the distance, using depth maps that are already generated as part of the normal rendering process (e.g., as would be used for standard 2D display renderings of 3D geometric content). By manually allocating and specifying color and depth texture render targets, we can easily use them within the adaptive filtering shader without needing to ever copy the color or depth buffer data off of the GPU. A static reference depth map is generated corresponding to a viewpoint positioned behind the physical display surface ("inside" or behind the virtual object). At run-time, a second depth map is produced while rendering the current pose of the virtual object from the same viewpoint. The difference between the two depth maps can easily and quickly be calculated in a shader, resulting in a per-pixel approximation of the geometric distance dG between the two surfaces. The same shader can then also sample the color buffer from the rendering of the current pose of virtual object with a low-pass filtering kernel where the sampled area is directly proportional to dG as calculated at each pixel.

An initial rendering of a digitized model of the physical display surface is only needed to be done once. The depth buffer can then be saved to a texture as a reference depth for use later by the adaptive filtering shader. At run-time there are two rendering passes required:

1. Render the view of the current pose of the virtual object from same viewpoint as the reference depth image. This results in both a color RGB texture and a depth texture.
2. The adaptive filtering shader computes the difference between the reference and current depths and samples the current color texture with a low-pass kernel to generate the resulting filtered image that can then be projected onto the physical display surface.

See Figure 3 for a diagram of the rendering process and data flow.

Persistence and Decay

Certain types of virtual object movement will result in cyclical error at local areas on the display surface. For example, virtual head rotation during a "no" head shaking animation may result in the larger dG corresponding to the tip of the nose on the virtual object passing back and forth past a point on the physical display surface in a very repetitive fashion. This rapid, cyclical transition between heavy filtering and little to no filtering is very visually distracting. To deal with this phenomenon we introduce temporal terms for *persistence* and *decay* of the magnitude of the filtering.

Persistence refers to a configurable length of time during which the magnitude of the adaptive filtering will not decrease. For example, persistence values longer than the period of a cyclical animation result in a constant (maximal) amount of filtering during the entire duration of the animation. When the animation has finished and after the persistence delay, rather than having the filtering immediately snap back to the current values, the decay rate defines how quickly the magnitude will drop back down to the current level (as defined by dG). There are inherent trade-offs between persistence and decay values related to avoiding distractions from noticeable changing in the amount of filtering at the cost of having less salient imagery overall that remains downsampled for a longer amount of time.

To store the per-pixel persistence information, we use an additional texture buffer, which can be passed to our rendering shader through standard multitexturing bindings. Decay can either be statically defined inside the shader or passed to the shader as a simple uniform variable.

Using the Gradient of the Geometric Distance

There may be cases where there are geometric differences between the virtual object and physical display surface that are relatively constant temporally (e.g., while the virtual object is not moving) or that do not cause significant distortion related to shape cues of the physical display surface (e.g., for large regions of the physical display surface with very low curvature). In these cases, while the apparent displacement of a single feature may vary significantly across disparate viewpoints, from each relatively static viewpoint the displacement will be approximately the same for spatially local collections of nearby features, preserving the salience of lo-

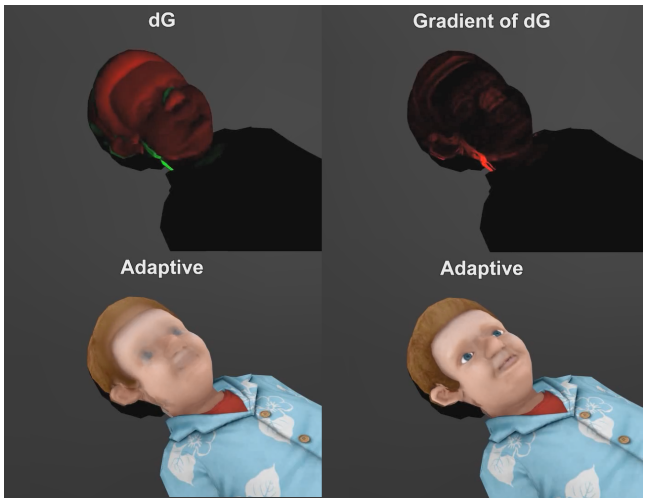

Figure 5: *A comparison of using absolute approximate distance and a locally computed gradient of dG during one pose of a head nod animation. The top row shows a visualization of dG or ∇dG on the physical display surface and the bottom row shows a simulated view of the final adaptively rendered physical augmented surface.*

cal visual content. Depending on the specific application and content, it may be more important to preserve the additional detail at the cost of global displacements at different viewing locations. To achieve this, we can compute the local gradient of the geometric distance between surfaces ∇dG and filter proportional to the gradient rather than the absolute value (see Figure 5). In our current implementation we compute a rapid local approximation of $\nabla dG = < \frac{\partial dG}{\partial u}, \frac{\partial dG}{\partial v} >$ in (u,v) texture space and then filter according to $|\nabla dG|$. In the future a more sophisticated approach could filter more specifically using temporal information such as the gradient as computed along the current velocity of p' given the movement of the virtual object between rendering frames.

5. Physical SAR Use-Case Setup and Application

One existing applied SAR use case is a physical-virtual child patient, consisting of a roughly child-shaped, rear-projected plastic shell designed for studying applications related to teaching or training medical practitioners or healthcare providers (see Figure 6).

5.1. Application

Throughout history people have modeled the human body [MMŽ10] for education and training purposes. Currently healthcare educators/triners use a combination of *standardized patients* (human actors), *physical* patient simulators, and *virtual* patient simulators.

Standardized patients can be very powerful emotionally, however there are limits to anyone's acting ability, particularly when it comes to simulating appearance or behavior changes that are impossible or dangerous for a healthy person to mimic. Virtual patients are dynamic computer graphics renderings of humans, typically on 2D displays. Such 2D systems [Hea15, Cll15, iHP15] can readily exhibit a broad range of variations in appearance including medical symptoms, race, and gender; and visual/verbal behaviors. However they are inherently virtual—you cannot touch them and they appear in the context of their own virtual environment. Physical patient simulators include human-sized robotic *manikins* with realistic skin and computer-controlled electro-mechanical simulation of many physiological characteristics. In addition to the general benefit of physicality [Li15], such systems can simulate a wide range of medical conditions, and offer the benefit of a very "hands on" experience for individuals or teams training together [Lae15, CAE15b, CAE15a]. However the manikins are primarily static in terms of their visual appearance—it is difficult or impossible to simulate the visual aspects of conditions such as vomiting (common), a heart attack, a stroke, jaundice, wound infection, and fever. Furthermore the patient's "humanity" is limited to a disembodied voice and blinking eyes.

The physical-virtual patient simulator combines the *physicality* of manikins and standardized patients with the *flexibility* of virtual humans for the training and assessment of healthcare providers. It can (seem to) converse with providers while exhibiting relevant physiological and behavioral cues including internal body sounds, visual sweat & breathing, tactile pulse, and a tactile sense of warm/cold head and extremities. The SAR-based approach supports changes in demographics (e.g., race and gender) and sophisticated emotional complexity through a combination of dynamic computer animation and computer/human-controlled speech and behaviors.

5.2. Apparatus

The physical-virtual nature of the SAR-based patient simulator shown in Figure 6 is motivated by a desire to combine the physicality of conventional patient manikins, with the flexibility of virtual humans, and the humanity of real human control. The current prototype uses a rear-projection paradigm with interchangeable body "shells" for a lightweight and simple (no electronics) approach to changing the simulated physical-virtual patient for different genders, ages, and medical conditions. The prototype includes controllable temperature over the surface of the body using forced air on the underside of the shells; a sense of pulse to appropriate places on the body using TECHTILE toolkit units [MKN*12]; and a sense of a breathing and other sounds emanating from inside the body using internal audio transducers and signal processing.

5.3. Unity-based SAR Simulator

The framework used for controlling and rendering on this setup uses the Unity graphics engine, which comes with several advantages: ease of modifying the spatial layout (lighting, projector/surface positions, etc.); ease of swapping in new virtual models that can have arbitrary complex animations/deformations (e.g., via skeletal animation or blend shapes); and access to a wide range of existing third party packages for controlling a character (e.g., mouth movements and shapes dynamically generated at run-time corresponding to arbitrary speech audio).

To facilitate adaptive filtering on the physical apparatus described above, we first created a Unity-based application for simu-

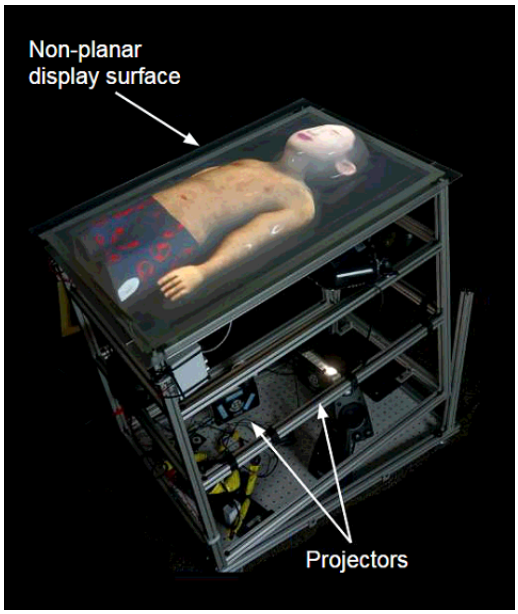

Figure 6: *A spatially augmented display surface setup designed for teaching or training medical students or professionals.*

lating the results of projecting virtual character poses onto the physical display surface, including adaptively filtered imagery during synthetic animatonics. The simulator virtually replicated the physical two-projector setup, using an additional projective texturing rendering pass to simulate the projected imagery on a digital model of the physical plastic shell to generate arbitrary interactive views of the augmented surface.

We implemented the depth-map based adaptive filtering described in Section 4 in the Unity version 2017.2.0 graphics engine. Unity has built-in mechanisms for rendering color and depth to textures that we can use as shader inputs. At the core of the filtering is a low-pass filtering kernel. There is an implementation trade-off between doing the filtering in a single pass versus multiple, iterative passes. For a single pass implementation, the framerate is largely consistent while larger amounts of filtering either require a large kernel (very slow), introduce sampling artifacts (e.g., multiple distinct copies of the same underlying feature), or are simply impossible. Multi-pass filtering on the other hand can achieve better results and performance for significant magnitudes of downsampling, but at an overall worse best-case framerate and requiring additional complexity to be able to ping-pong between intermediate buffer results.

For a rear projection SAR character, color and shape for the virtual object should correspond to the "outer-most" surface of the virtual object. However, unrestricted source content with arbitrary animations is likely to have complex internal or layered geometry (e.g., eyeballs, mouth, tongue, etc.). The surface that we want to convey on the physical display surface actually corresponds to the outer-most surface of the source object, ignoring the internally modelled content except in places where it is actually visible from the front. To get the correct color and corresponding depth values for adaptive filtering we do reverse-depth rendering—instead of keeping the color/depth values closest to the camera, we clear the depth buffer to a near value and keep the farthest values for each pixel. In Unity, for example, this simply involves modifying the z check in the rendering shader to use "ZTest GEqual". Because the depth buffer could not be cleared to a custom value in Unity, we instead place a geometric plane at the near plane of the camera, effectively "clearing" to that depth value.

6. Assumptions and Limitations

Although the practical application described in Section 5 utilizes projector-based SAR, the adaptive filtering technique described in this paper is to a large degree independent of the underlying display technology. Alternative methods for displaying imagery on a shaped surface, e.g., flexible OLED displays, could also be used. Because the core concept is abstracted from the display modality and hardware implementation, in this paper we do not address issues related to surface material properties, environmental lighting affects, projector intensity fall-off, display element size and shape, or blending between multiple overlapping display or projector areas. There is a large body of active and previous work in solving or dealing with these issues for specific display modalities.

In general, an important limitation of any kind of visual-only geometric SAR manipulations is that there is no perfect solution, i.e., in most cases the virtual object presented for a single head-tracked user will always look better than a filtered representation for multiple or non-tracked users, even if it avoids large amounts of geometric misperception as proposed in this paper. Adaptive filtering provides a tuneable tradeoff, sacrificing some visual detail to avoid distracting artifacts that may provide unwanted shape cues for the underlying physical display surface. The usefulness of adaptive filtering is largely application-specific—if maintaining the overall perception of the animation or motion is more important than the specific areas of detail that may be "lost," then the end result can be more effective in achieving the application's goals. Knowing the task ahead of time can also allow for content-aware optimizations that shift the location of the filtering (perhaps even at the cost of *more* overall filtering), to try to preserve the specific details that are most important for the specific task.

We performed preliminary subjective analyses with domain experts in the described application field using the physical apparatus, and we identified the potential of the described techniques to reduce unwanted visual distractions. However, formal experiments should be performed to determine and verify optimal values for the parameters related to persistence and decay and evaluate the visual quality for specific applications or tasks.

7. Conclusion and Future Work

In this paper we presented a method for achieving realtime adaptive filtering of dynamic SAR content where the filtering is based on a viewpoint-agnostic abstraction of the expected distortion at every point. We presented a depth-map-based approximation to the geometric distance between a virtual object and the physical display surface, and we described how we leverage existing hardware accelerated rendering processes to do the adaptive filtering in real

time for arbitrary virtual object animation that does not need to be known ahead of time. We presented an example application based on our adaptive filtering method.

While adaptive filtering can be utilized when adding synthetic animatronics to existing static SAR installations where the physical display surface is already defined, there is additional future work in a one-time optimization of the physical shape of the display surface. Such future optimizations should take into account known or expected viewpoints or viewing regions, known or predicted animations or motion statistics, as well as task-specific importance of local regions, features, or higher level perceptual cues.

Acknowledgements

This material includes work supported in part by the National Science Foundation (NSF) under Grant Number 1800961 (Dr. Tonya Smith-Jackson, IIS) and 1564065 (Dr. Ephraim P. Glinert), as well as the Office of Naval Research (ONR) under Grant Number N00014-17-1-2927 (Dr. Peter Squire, Code 30). We also acknowledge Florida Hospital for their support of Prof. Welch via their Endowed Chair in Healthcare Simulation.

References

[AFT13] ADCOCK M., FENG D., THOMAS B.: Visualization of off-surface 3D viewpoint locations in spatial augmented reality. In *Proceedings of the ACM Symposium on Spatial User Interaction* (2013). 2

[BAOL16] BRUDER G., ARGELAGUET SANZ F., OLIVIER A.-H., LECUYER A.: CAVE Size Matters: Effects of Screen Distance and Parallax on Distance Estimation in Large Immersive Display Setups. *Presence: Teleoperators and Virtual Environments 25*, 1 (2016), 1–16. 2

[BBG*13] BERMANO A., BRÜSCHWEILER P., GRUNDHÖFER A., IWAI D., BICKEL B., GROSS M.: Augmenting physical avatars using projector-based illumination. *ACM Trans. Graph. 32*, 6 (Nov. 2013), 189:1–189:10. doi:10.1145/2508363.2508416. 2

[BKKF13] BECK S., KUNERT A., KULIK A., FROEHLICH B.: Immersive Group-to-Group Telepresence. *IEEE Transactions on Visualization and Computer Graphics (TVCG) Special Issue on IEEE Virtual Reality (VR) 19*, 4 (2013), 616–625. doi:10.1109/TVCG.2013.33. 2

[BR05] BIMBER O., RASKAR R.: *Spatial augmented reality: merging real and virtual worlds*. AK Peters/CRC Press, 2005. 1

[BST14] BROECKER M., SMITH R. T., THOMAS B.: Depth perception in view-dependent near-field spatial AR. In *Proceedings of the Australasian User Interface Conference* (2014). 2

[BWW94] BOUFAMA B., WEINSHALL D., WERMAN M.: Shape from motion algorithms: a comparative analysis of scaled orthography and perspective. In *Proceedings of the third European conference on Computer vision (vol. 1)* (Secaucus, NJ, USA, 1994), ECCV '94, Springer-Verlag New York, Inc., pp. 199–204. URL: http://portal.acm.org/citation.cfm?id=189359.189393. 2

[CAE15a] CAE HEALTHCARE: Caesar. http://caehealthcare.com/eng/patient-simulators/caesar-simulator, August 2015. (Visited on 08/16/2015). 5

[CAE15b] CAE HEALTHCARE: Hps human patient simulator medical simulator. http://www.caehealthcare.com/eng/patient-simulators/hps-human-patient-simulator, August 2015. (Visited on 08/18/2015). 5

[Cll15] CLLINISPACE: Virtual sim center. http://virtualsimcenter.clinispace.com/, August 2015. (Visited on 08/16/2015). 5

[DHMKP07] DE HAAN G., MOLENAAR R., KOUTEK M., POST F. H.: Consistent viewing and interaction for multiple users in projection-based vr systems. In *Computer Graphics Forum* (2007), vol. 26, Wiley Online Library, pp. 695–704. 2

[GPNB04] GROSSBERG M., PERI H., NAYAR S., BELHUMEUR P.: Making one object look like another: controlling appearance using a projector-camera system. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on* (june-2 july 2004), vol. 1, pp. I–452 – I–459 Vol.1. doi:10.1109/CVPR.2004.1315067. 2

[Gre97] GREGORY R. L.: Knowledge in perception and illusion. *Phil. Trans. R. Soc. 352* (August 1997), 1121–1127. doi:10.1098/rstb.1997.0095. 2

[HDN*15] HOCHREITER J., DAHER S., NAGENDRAN A., GONZALEZ L., WELCH G.: Touch sensing on non-parametric rear-projection surfaces: A physical-virtual head for hands-on healthcare training. In *Proceedings of IEEE Virtual Reality (VR)* (2015), pp. 69–74. 1, 2

[Hea15] HEALTH S.: Shadow health web site. http://www.shadowhealth.com/, August 2015. (Visited on 08/16/2015). 5

[iHP15] I HUMAN PATIENTS: i-human patients web site. http://www.i-human.com/, August 2015. (Visited on 08/16/2015). 5

[IOT*08] ITOH K., ONISHI Y., TAKAHASHI S., AOKI T., HAYASHI K., TAKANISHI A.: Development of face robot to express various face shapes by moving the parts and outline. In *Proceedings of the IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)* (2008), pp. 439–444. 1

[LA12] LAW A., ALIAGA D.: Spatial augmented reality for environmentally-lit real-world objects. In *IEEE Virtual Reality Workshops (VR)* (2012), pp. 7–10. doi:10.1109/VR.2012.6180867. 2

[Lae15] LAERDAL: SimMan™ 3G. http://www.laerdal.com/us/SimMan3G, August 2015. (Visited on 08/16/2015). 5

[LAS*11] LAW A. J., ALIAGA D. G., SAJADI B., MAJUMDER A., PIZLO Z.: Perceptually based appearance modification for compliant appearance editing. *Comput. Graph. Forum 30*, 8 (2011), 2288–2300. 2

[Li15] LI J.: The benefit of being physically present: A survey of experimental works comparing copresent robots, telepresent robots and virtual agents. *International Journal of Human-Computer Studies 77* (2015), 23–37. doi:http://dx.doi.org/10.1016/j.ijhcs.2015.01.001. 5

[LK03] LOOMIS J. M., KNAPP J. M.: *Virtual and adaptive environments*. Mahwah, 2003, ch. Visual perception of egocentric distance in real and virtual environments, pp. 21–46. 2

[LWN*11] LINCOLN P., WELCH G., NASHEL A., STATE A., ILIE A., FUCHS H.: Animatronic shader lamps avatars. *Virtual Real. 15* (2011), 225–238. doi:http://dx.doi.org/10.1007/s10055-010-0175-5. 1

[MC00] MACINTYRE B., COELHO E.: Adapting to dynamic registration errors using level of error (loe) filtering. In *Augmented Reality, 2000. (ISAR 2000). Proceedings. IEEE and ACM International Symposium on* (2000), pp. 85–88. doi:10.1109/ISAR.2000.880927. 2

[MCJ02] MACINTYRE B., COELHO E., JULIER S.: Estimating and adapting to registration errors in augmented reality systems. In *Proceedings of IEEE Virtual Reality (VR)* (2002), pp. 73–80. doi:10.1109/VR.2002.996507. 2

[MKN*12] MINAMIZAWA K., KAKEHI Y., NAKATANI M., MIHARA S., TACHI S.: Techtile toolkit: A prototyping tool for design and education of haptic media. In *Proceedings of the 2012 Virtual Reality International Conference* (New York, NY, USA, 2012), VRIC '12, ACM, pp. 26:1–26:2. URL: http://doi.acm.org/10.1145/2331714.2331745, doi:10.1145/2331714.2331745. 5

[MMŽ10] MARKOVIĆ D., MARKOVIĆ-ŽIVKOVIĆ B.: Development of anatomical models—chronology. *Acta Medica Medianae 49*, 2 (2010), 56–62. 5

[MPWG05] MATKOVIC K., PSIK T., WAGNER I., GRACANIN D.: Dynamic texturing of real objects in an augmented reality system. In *Virtual Reality, 2005. Proceedings. VR 2005. IEEE* (march 2005), pp. 257 –260. doi:10.1109/VR.2005.1492784. 1

[PIS15] PUNPONGSANON P., IWAI D., SATO K.: Projection-based visualization of tangential deformation of nonrigid surface by deformation estimation using infrared texture. *Virtual Reality 19*, 1 (2015), 45–56. 2

[RL01] RASKAR R., LOW K.-L.: Interacting with spatially augmented reality. In *Proceedings of the 1st international conference on Computer graphics, virtual reality and visualisation* (2001), ACM, pp. 101–108. 2

[RVH13] RENNER R. S., VELICHKOVSKY B. M., HELMERT J. R.: The perception of egocentric distances in virtual environments - a review. *ACM Comput. Surv.* (2013), 1–38. doi:10.1098/rstb.1997.0095. 2

[RWF98] RASKAR R., WELCH G., FUCHS H.: Spatially augmented reality. In *First IEEE Workshop on Augmented Reality (IWARâĂŹ98)* (1998), Citeseer, pp. 11–20. 2

[RWLB01] RASKAR R., WELCH G., LOW K.-L., BANDYOPADHYAY D.: Shader lamps: Animating real objects with image-based illumination. In *Rendering Techniques 2001*. Springer, 2001, pp. 89–102. 1, 2

[SBS17] SCHMIDT S., BRUDER G., STEINICKE F.: Moving Towards Consistent Depth Perception in Stereoscopic Projection-based Augmented Reality. In *Proceedings of the International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments (ICAT-EGVE)* (2017), pp. 161–168. 2

[SLS*17] SIEGL C., LANGE V., STAMMINGER M., BAUER F., THIES J.: Faceforge: Markerless non-rigid face multi-projection mapping. *IEEE transactions on visualization and computer graphics 23*, 11 (2017), 2440–2446. 2

[TMS*14] THOMAS B. H., MARNER M., SMITH R. T., ELSAYED N. A. M., VON ITZSTEIN S., KLEIN K., ADCOCK M., EADES P., IRLITTI A., ZUCCO J., ET AL.: Spatial augmented reality—A tool for 3D data visualization. In *Proceedings of the IEEE VIS International Workshop on 3DVis* (2014), pp. 45–50. 2

[Tod04] TODD J. T.: The visual perception of 3d shape. In *Trends in Cognitive Science* (2004), Elsevier, p. 2004. 2

[WF06] WHITE R., FORSYTH D.: Combining cues: Shape from shading and texture. In *Proceeding sof the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2006), vol. 2, pp. 1809–1816. 2

[ZXT*16] ZHOU Y., XIAO S., TANG N., WEI Z., CHEN X.: Pmomo: projection mapping on movable 3d object. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (2016), ACM, pp. 781–790. 2