A General Approach for Closed-Loop Registration in AR

Feng Zheng and Ryan Schubert*
The University of North Carolina at Chapel Hill

Greg Welch†
The University of Central Florida and UNC-CH

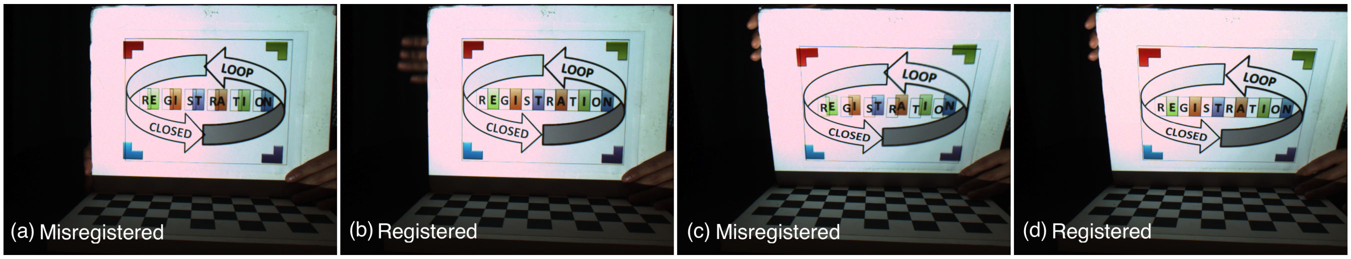


Figure 1: Two misregistered frames due to user motion in a projector-based AR experiment are shown in (a) and (c). Our approach observes the augmented imagery and corrects any visible misregistration. The corresponding registered frames are shown in (b) and (d).

Abstract

The typical registration process in augmented reality (AR) consists of three independent consecutive stages: static calibration, dynamic tracking, and graphics overlay. The result is that the real-virtual registration is “open loop”—inaccurate calibration or tracking leads to misregistration that is seen by the users but not the system. To cope with this, we propose a general approach to “close the loop” in the displayed appearance by using the visual feedback of registration for pose tracking to achieve accurate registration. Specifically, a model-based method is introduced to simultaneously track and augment real objects in a closed-loop fashion, where the model is comprised of the combination of the real object to be tracked and the virtual object to be rendered. This method is applicable to paradigms including video-based AR, projector-based AR, and diminished reality. Both qualitative and quantitative experiments are presented to demonstrate the feasibility and effectiveness of our approach.

Keywords: Closed-loop registration, visual feedback, tracking, projector-based AR, video-based AR, diminished reality.

Index Terms: H.5.1 [INFORMATION INTERFACES AND PRESENTATION (e.g., HCI)]: Multimedia Information Systems—Artificial, augmented, virtual realities; I.4.8 [IMAGE PROCESSING AND COMPUTER VISION]: Scene Analysis—Tracking

1 Introduction

Accurate registration between virtual content and real objects is a critical aspect of most Augmented Reality (AR) systems. The conventional method for achieving registration is a three-step process in which independent mechanisms are used first to do a one-time calibration of system parameters, then to track the object to be augmented, and finally to generate the appropriate virtual content to be overlaid on the real object using the pose information from the tracking step. However, such an open-loop system has no mechanism for observing registration error—it simply generates the virtual content that should be consistent with the estimated pose data, assuming there are no sources of error in the tracking or calibration. In practice, there are a number of static and dynamic error sources [5], resulting in misregistration that goes “unseen” by the system.

Other researchers have explored specific cases of using the augmented imagery (the combination of real and virtual) as feedback into the tracking step to achieve a closed-loop registration system [2, 3]. We propose a related closed-loop approach, with the novelty of being suitable for multiple AR paradigms, that employs a registered real-virtual model.

This real-virtual model-based registration offers several advantages. It embodies a closed-loop system that is continuously adjusting parameters to maintain the desired augmented appearance. It does so without the explicit detection and use of features or points in the camera imagery, instead optimizing the parameters directly using any misregistration manifested in the augmented imagery. In addition to simplifying the closed-loop registration, this approach can use information implicit in the augmented imagery, such as misregistration manifested in the form of T-junctions or other features that do not exist in either the real or the virtual imagery, but arise as a result of interactions between the real and virtual imagery. Our approach can be used by itself in cases where inter-frame movement is relatively small (where the misregistration is correctable by an iterative optimization), or in combination with a conventional open-loop approach by using the open-loop tracking for a coarse pose estimate prior to closed-loop optimization. Finally, the approach can be used with projector-based AR as well as video-based AR such as on hand-held or head-worn devices.

2 Related Work

2.1 Registration Error Analysis

There are several instances of prior work on registration error analysis and correction. Holloway [5] identified a number of error sources including system delay, tracker error, calibration error and misalignment of the model. Bajura and Neumann [3] measured registration error in the 2D augmented imagery and using that information to correct the 3D registration. They did so by using explicitly chosen/detected features in the scene, including features intentionally added to the scene to close the loop (e.g., via LEDs placed in the scene).

Similar to [3], our method uses the 2D error in the augmented imagery to measure registration accuracy. However, rather than choosing/detecting specific features in the scene we implicitly incorporate the error measure into the tracking procedure.

*e-mail: {zhengf, res}@cs.unc.edu
†e-mail: welch@ucf.edu

IEEE Virtual Reality 2013
16 - 20 March, Orlando, FL, USA
978-1-4673-4796-9/13/$31.00 ©2013 IEEE

2.2 Vision-Based Tracking

Template-based tracking methods are widely used in video-based AR. Since the seminal work of Lucas and Kanade (LK) [7], many related methods [8, 10] have been proposed. Baker and Matthews [4] presented a unifying framework to understand and categorize many variants of the LK method. However, these methods fall short in projector-based AR due to the strong interference caused by projector light, and since they do not consider feedback from augmentation, they are still "open loop" for AR applications.

For projector-based AR, vision-based methods typically either avoid using the projected imagery or explicitly use it, as summarized by Audet *et al* [2]. In one example of the latter case, Audet *et al* [2] proposed considering the projected content as useful information that could be incorporated into an image alignment formulation. This method was novel in the way it closed the loop, however it only looked at projector-based AR and not other AR paradigms. Our approach is applicable to both projector- and video-based AR.

3 Real-Virtual Model-Based Registration

3.1 Cost Function and Optimization

As human observers, we expect to see the correct combined appearance of the real and virtual, i.e., the appearance we observe should match a goal appearance. This suggests a natural formulation of the cost function:

$$\arg\min_{\mathbf{p}} \|\hat{C}(\mathbf{u}) - \hat{G}(W(\mathbf{u};\mathbf{p}))\| \tag{1}$$

where $\hat{C}$ is the image we observe, i.e., the combined appearance of the real and virtual, called the *combined image*, while $\hat{G}$ represents the expected appearance, called the *goal image*. The goal image $\hat{G}$ is the 2D view of the goal model, which is the registered combination of the real object to be tracked and the virtual object to be rendered. To minimize the 2D image difference, the goal image is acquired by transforming the goal model using the warping function $W(\mathbf{u};\mathbf{p})$, where $\mathbf{u} = (u,v)^T$ is a 2D column vector containing the pixel coordinates, and $\mathbf{p} = (p_1, \cdots, p_n)^T$ is a vector of parameters for arbitrary spatial transformation, e.g., a 2D homography or a 3D pose. If the goal model is planar, i.e., both the real object and virtual object are planar, either a homograhy or a 3D pose can be used. For the general case, where the goal model is not necessarily planar, we use 3D pose parameterization.

For projector-based AR, the combined image is a function of light and surface reflectance. Assuming no environmental light and that real objects are planar and diffuse, then the observed combined image $\hat{C}$ can be approximated as a multiplicative modulation of the projected light $\hat{V}$, called the *virtual image*, the surface reflectance $\hat{R}$, called the *real image*, and the cosine of the angle θ between the surface normal and projector light:

$$\hat{C}(\mathbf{u}) = \hat{V}(W(\mathbf{u};\mathbf{p})) \cdot \hat{R}(\mathbf{u}) \cdot \cos\theta \tag{2}$$

where the virtual image $\hat{V}$ is warped onto the coordinate frame of the real image $\hat{R}$. The coordinate frames of $\hat{C}$ and $\hat{R}$ are the same. The projector-camera system is assumed to be geometrically calibrated.

Plugging Equation (2) into Equation (1) and using the L2 norm as our error measure, we obtain

$$\sum_{\mathbf{u}} \|\hat{V}(W(\mathbf{u};\mathbf{p})) \cdot \hat{R}(\mathbf{u}) \cdot \cos\theta - \hat{G}(W(\mathbf{u};\mathbf{p}))\|^2 \tag{3}$$

Note that both $\hat{V}$ and $\hat{G}$ are warped using the same warping parameters since they essentially have the same behavior. Thus a nonlinear optimization problem is formulated.

To simplify the first term in Equation (3), we apply a logarithmic transformation to linearize it, assuming uniform illumination:

$$\sum_{\mathbf{u}} \|V(W(\mathbf{u};\mathbf{p})) + R(\mathbf{u}) + \log\cos\theta - G(W(\mathbf{u};\mathbf{p}))\|^2 \tag{4}$$

where V, R , and G are referred as the *log virtual image*, *log real image* and *log goal image*, respectively. Likewise, the combined image $\hat{C}$ also has a log form C:

$$C(\mathbf{u}) = V(W(\mathbf{u};\mathbf{p})) + R(\mathbf{u}) + \log\cos\theta \tag{5}$$

Algorithm 1: Real-Virtual Model-based Registration

```
Pre-compute Gradients $\nabla V$ and $\nabla G$ of images V and G
repeat
    Transform $G$ with $W(\mathbf{u};\mathbf{p})$ to compute $G(W(\mathbf{u};\mathbf{p}))$
    Compute the error image $D(\mathbf{u}) = C(\mathbf{u}) - G(W(\mathbf{u};\mathbf{p}))$
    Warp the gradient $\nabla V$ and $\nabla G$ with $W(\mathbf{u};\mathbf{p})$
    Evaluate the Jacobian $\frac{\partial W}{\partial \mathbf{p}}$ at $(\mathbf{u};\mathbf{p})$
    Compute the steepest descent image $(\nabla V - \nabla G)\frac{\partial W}{\partial \mathbf{p}}$
    Compute the Hessian matrix using Equation (7)
    Compute $\sum_{\mathbf{u}} \left[(\nabla V - \nabla G)\frac{\partial W}{\partial \mathbf{p}}\right]^T D(\mathbf{u})$
    Compute $\Delta\mathbf{p}$ using Equation (6)
    Update the parameters: $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$
until $\|\Delta\mathbf{p}\| \leq \varepsilon$
```

Equation (4) can be effectively solved using conventional gradient descent techniques. In our implementation, we use the Gauss-Newton method, and apply an additive rule to update the motion parameters. The solution of Equation (4) is:

$$\Delta\mathbf{p} = -H^{-1} \sum_{\mathbf{u}} \left[(\nabla V - \nabla G)\frac{\partial W}{\partial \mathbf{p}}\right]^T D(\mathbf{u}) \tag{6}$$

where $\Delta\mathbf{p}$ is the incremental motion vector, ∇V and ∇G are gradients of V and G, $D(u)$ denotes the error image, i.e., $D(\mathbf{u}) = C(\mathbf{u}) - G(W(\mathbf{u};\mathbf{p}))$, and H is the *Hessian* matrix:

$$H = \sum_{u} \left[(\nabla V - \nabla G)\frac{\partial W}{\partial \mathbf{p}}\right]^T \left[(\nabla V - \nabla G)\frac{\partial W}{\partial \mathbf{p}}\right] \tag{7}$$

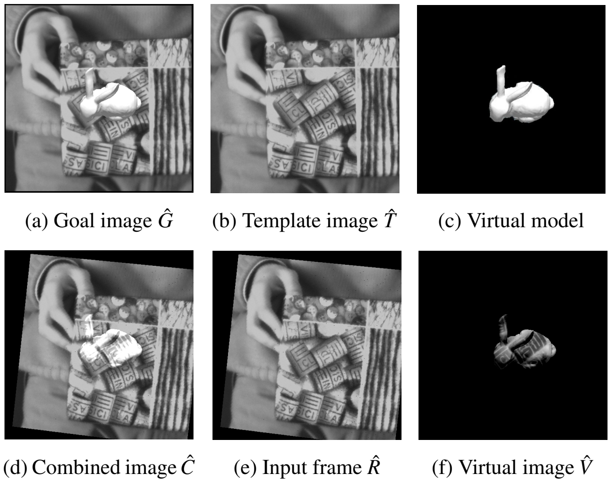
A summary of the algorithm is shown in Algorithm 1. Note that in the solution Equation (6), it is not necessary to compute the angle between the surface normal and the projector light for computing the combined image in Equation (5)). The reason is that in projector-based AR, the combined images are "computed" (combined) optically. To get the combined image, we project the virtual image onto the scene then capture the resulting appearance using a camera. Hence, in Algorithm 1 $C(\mathbf{u})$ is implicitly "computed" by capturing the scene and then doing a logarithmic transformation.

3.2 Extension to Other AR Paradigms

To extend the proposed approach to other AR paradigms in which the virtual and real do not coexist in the same space, the combined real and virtual image needs to be generated via simulation rather than being combined optically and captured with a camera, as in projector-based AR. A simple way to do this is to consider the relationship as addition, i.e., $\hat{C} = \hat{V} + \hat{R}$. Then to compute the virtual image, we can simply subtract the template object image $\hat{T}$ from goal image $\hat{G}$, i.e., $\hat{V} = \hat{G} - \hat{T}$. An illustration of the various images and computations is shown in Figure 2. With this simplified relationship between the real and virtual images, the algorithm can be used without change for different AR paradigms.

4 Evaluation

We performed both qualitative and quantitative experiments to evaluate the proposed approach. All of these experiments were focused on tracking and augmenting planar objects with 2D or 3D virtual



(a) Goal image $\hat{G}$ (b) Template image $\hat{T}$ (c) Virtual model

(d) Combined image $\hat{C}$ (e) Input frame $\hat{R}$ (f) Virtual image $\hat{V}$

Figure 2: Illustration of combined image formulation. The goal image $\hat{G}$ in (a) is the 2D appearance of the goal model, which is comprised of the template image $\hat{T}$ (the real object) in (b) and the virtual object in (c). The virtual image $\hat{V}$ in (f) is computed as the subtraction of $\hat{G}$ and $\hat{T}$. The combined image $\hat{C}$ in (d) is the addition of the input frame $\hat{R}$ in (e) and $\hat{V}$. In the registration process, $\hat{G}$ is iteratively acquired by transforming the goal model using the current pose estimate, until it matches $\hat{C}$.

content. The method can be readily extended to track and augment non-planar objects if their 3D structure is known. For motion parameterization, we tried both a 2D homography and a 3D pose.

The software for the experiments is implemented using OpenCV for image processing and OpenGL for rendering. The multithreading API OpenMP is also used to parallelize and speed up the algorithm. Our test hardware consisted of a Flea-HICOL (1024x768) camera and an InFoucus 1503D (1280x800) projector, both connected to a computer with an Intel Xeon 2.27 GHz CPU.

4.1 Qualitative Experiments

To show the feasibility of our approach under various AR paradigms, we did three qualitative experiments for each mentioned AR paradigm.

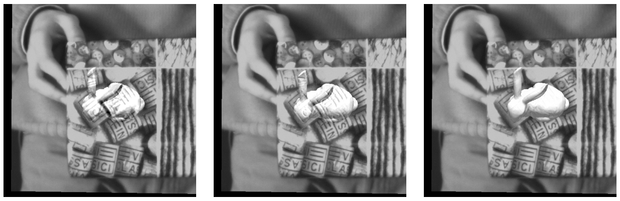
4.1.1 Experiment 1: Projector-Based AR

Here we show a projector-based application similar to [2], where parts of the expected imagery are printed on the board while the others are projected. The projector-camera system was geometrically calibrated using [1] without color calibration. We chose to optimize for a 2D homography and then extract the 3D pose from it [11]. We achieved 10 fps with the current implementation. The algorithm successfully converged for the test sequence, which contains large inter-frame motion and noise. Results are shown in Figure 1.

Due to the difference in our cost function formulation compared to [2], we project an image for *each* iteration using incrementally estimated pose parameters. This means that the real-virtual optimization (augmentation with lighting) is affected and directly measured optically in the scene space every iteration, as opposed to being simulated. Another difference is that in our optimization we obtained an analytical solution while [2] evaluated the Jacobians numerically. Moreover, even without color calibration or synchronization between the projector and the camera, our method worked well and was robust in handling the test sequences.

4.1.2 Experiment 2: Video-Based AR

For video-based AR, since the goal model is not planar, we directly optimized for the 3D pose, which is parameterized using the twist representation as in [12]. We tested our approach with two synthetic sequences both of which were accurately tracked and augmented in real time. Figure 3 shows the progression from an initial state with some noise and large registration error to reduced error and finally almost no registration error after nine iterations for a single frame.



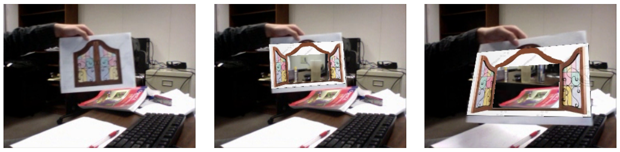
(a) 1st iteration (b) 6th iteration (c) 9th iteration

Figure 3: (a) Initial misregistered appearance (note the bunny region). (b) Decreased registration error. (c) Converged state with almost no error.

4.1.3 Experiment 3: Diminished Reality

Diminished reality removes an object or collection of objects and replaces it with an appropriate background image [13]. It can be considered a real-virtual registration process where objects are tracked and augmented with virtual content that hides them.

We did a simple proof-of-concept diminished reality experiment in which we computed the homography between consecutive frames. For a single static camera view with a known static background, we tracked and "camouflaged" a portion of the planar real object in real time, as shown in Figure 4. The result was achieved using the same process as video-based AR.



(a) Real object (b) Frame 1 (c) Frame 106

Figure 4: (a) shows the real object, which is tracked and replaced with background imagery and also augmented with an "opened" window. Results of two frames are shown in (b) and (c).

4.2 Numerical Experiments

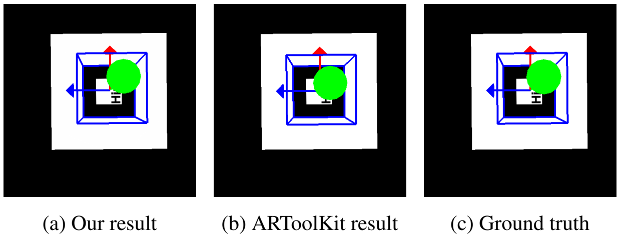
Two quantitative experiments were conducted, the results of which show that our approach outperforms conventional AR systems in terms of registration accuracy. We chose ARToolKit [6] to compare our method with, as it is widely used in current AR systems. Both of the test sequences used were synthetic so that we could also calculate the absolute ground-truth registration data easily and have pefect knowledge and control of the calibration. The parameters we optimized for were the 3D pose. The error metric we used was the mean-absolute-error in image intensity

$$E = \frac{1}{N}\sum_{\mathbf{u}} |A(\mathbf{u}) - B(\mathbf{u})| \quad (8)$$

where $A(\mathbf{u})$ and $B(\mathbf{u})$ represent the ground-truth image and result image respectively, and N denotes the number of pixels in an image. The images were grayscale with intensity values in [0,255].

4.2.1 Experiment 4: Tracker Error

In this experiment, both our approach and ARToolKit were provided with correct calibration parameters, meaning the registration



(a) Our result (b) ARToolKit result (c) Ground truth

Figure 5: Comparison of rendered results of frame 241. Our result is much closer to the ground truth (note the green arrow).

accuracy can be attributed purely to pose estimates from the tracking. The test sequence contains a marker, which is initially almost perpendicular to the viewing camera, undergoing a small amount of movement. Visual registration results are shown in Figure 5.

Figure 6 shows numerical results of registration error for each frame. Our results are more stable and accurate while there is a significant amount of jitter in the ARToolKit result. This is because ARToolKit tends to produce unreliable jittery pose estimates with sequences captured from a frontal direction [9]. Our method enjoys the benefit of measuring and correcting tracker errors by feedback from the real-virtual registration.

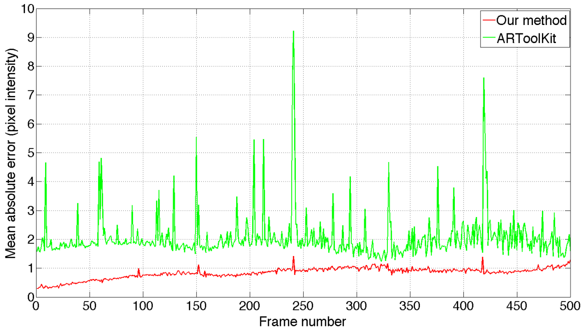


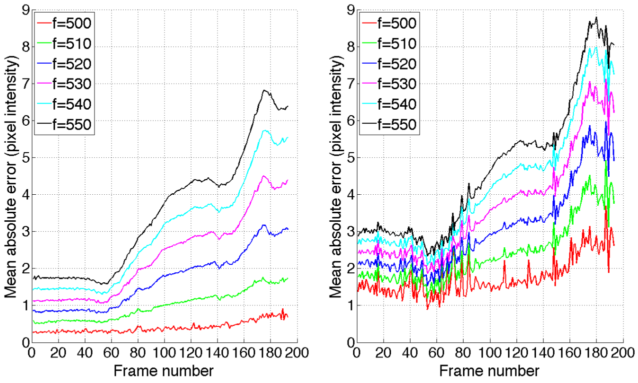
Figure 6: Numerical comparison from experiment 4. Our results (red curve) are more accurate and stable than ARToolKit (green curve) in terms of mean-absolute-error in pixel intensity [0, 255].

4.2.2 Experiment 5: Calibration Error

In this experiment, we tested the same two approaches with inaccurate calibration data, to simulate another common source of misregistration in AR systems. Specifically, the focal length parameter of the camera calibration data is increasingly degraded. Figure 7 shows the registration error for different focal lengths, where focal length f = 500mm is the correct value. For both of the approaches, the registration error increases with the error in focal length. However, our approach still outperformed ARToolKit for all calibration focal lengths.

5 Conclusions and Future Work

In this paper, we have presented a new approach for closed-loop registration in AR, which is simple, effective and general for various AR paradigms. This approach combines tracking and augmentation for the purpose of registration in a more compact closed-loop framework without using an extra step for correction. It can also be easily combined with conventional open-loop trackers to cope with significantly larger inter-frame motion. For future work, we plan to fully correct registration errors due to calibration error and handle complex situations such as lighting and partial occlusions.



(a) Our result (b) ARToolKit result

Figure 7: Comparison of registration accuracy with different amounts of error in focal length calibration. Our result (a) is better and contains less registration error than ARToolKit (b). The error metric is mean-absolute-error in pixel intensity [0,255].

Acknowledgements

This work was supported by U.S. Office of Naval Research grants N00014-09-1-0813 and N00014-12-1-0052, both titled "3D Display and Capture of Humans for Live-Virtual Training," (Clarke Lethin), and in part by the Singapore Interactive Digital Media program office sponsored BeingThere Centre (Michael Yap).

References

[1] S. Audet and M. Okutomi. A user-friendly method to geometrically calibrate projector-camera systems. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition - Workshops (Procams 2009)*, pages 47–54, 2009.

[2] S. Audet, M. Okutomi, and M. Tanaka. Direct image alignment of projector-camera systems with planar surfaces. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 303 –310, 2010.

[3] M. Bajura and U. Neumann. Dynamic registration correction in video-based augmented reality systems. *IEEE Computer Graphics and Applications*, 15(5):52–60, Sept. 1995.

[4] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *Int'l J. Computer Vision*, 56(3):221–255, Feb. 2004.

[5] R. L. Holloway. Registration error analysis for augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4):413–432, 1997.

[6] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proc. of IEEE/ACM Int'l Workshop on Augmented Reality*, pages 85–94, 1999.

[7] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. Int'l Joint Conf. on Artificial Intelligence*, pages 674–679, 1981.

[8] M. Meilland, A. Comport, and P. Rives. Real-time dense visual tracking under large lighting variations. In *Proc. of the British Machine Vision Conference*, pages 45.1–45.11, 2011.

[9] A. Mohan, G. Woo, S. Hiura, Q. Smithwick, and R. Raskar. Bokode: Imperceptible visual tags for camera based interaction from a distance. *ACM Trans. on Graphics*, 28(3):98:1–98:8, July 2009.

[10] R. Newcombe, S. Lovegrove, and A. Davison. DTAM: Dense tracking and mapping in real-time. In *Proc. IEEE Int'l Conf. on Computer Vision*, pages 2320 –2327, nov. 2011.

[11] B. Triggs. Autocalibration from planar scenes. In *Proc. European Conf. on Computer Vision*, pages 89–105, 1998.

[12] J. Xiao, T. Kanade, and J. F. Cohn. Robust full-motion recovery of head by dynamic templates and re-registration techniques. In *Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 163–169, 2002.

[13] S. Zokai, J. Esteve, Y. Genc, and N. Navab. Multiview paraperspective projection model for diminished reality. In *Proc. IEEE/ACM Int'l Symposium on Mixed and Augmented Reality*, pages 217–226, 2003.