Augmented reality on cloth with realistic illumination

Bradley, D.; Roth, Gerhard; Bose, J.

Publisher's version / Version de l'éditeur:
https://doi.org/10.1007/s00138-007-0108-9
Machine Vision and Applications Journal, 20, 2, pp. 85-92, 2007-11-23

National Research Council Canada
Institute for Information Technology

Conseil national de recherches Canada
Institut de technologie de l'information

NRC-CNRC

*Augmented Reality on Cloth with Realistic Illumination **

Bradley, D., Roth, G., Bose, J.
November 2007

* published in the Machine Vision and Applications Journal. 2007. NRC 49871.

Canada

Machine Vision and Applications manuscript No.
(will be inserted by the editor)

Derek Bradley · Gerhard Roth · Prosenjit Bose

Augmented Reality on Cloth with Realistic Illumination

Received: September 2005 / Accepted: November 2006

Abstract Augmented reality (AR) is the concept of inserting virtual objects into real scenes. Often, augmentations are aligned with rigid planar objects in the scene. However, a more difficult task is to align non-rigid augmentations with flexible objects like cloth. To address this problem, we present a method to perform real-time flexible augmentations on cloth. Our method involves sparse cloth-tracking in video images using a new vision-based marker system with temporal coherence. We include an image-based method to automatically acquire real world illumination and shadows from the input frame. Non-rigid augmentations are achieved by rendering a textured 2D mesh aligned with the cloth surface, and combining the illumination result. The ability to perform realistic augmentations on cloth leads to applications in fashion, advertising, home decor and entertainment. We demonstrate our cloth augmentations with an application to interactively design T-shirts by demonstrating different virtual logos on a physical shirt in real-time.

Keywords Augmented reality · Non-rigid object tracking · Common illumination · Marker systems

1 Introduction

In augmented reality systems, virtual objects are inserted into a real scene by superimposing them onto video images or a head-mounted display (HMD), usually in real-time. This process requires the calculation of the camera pose in order to precisely align the virtual objects with the real objects in the scene. In many AR applications, augmentations are aligned with a 2D plane. This plane can be defined by a rigid pattern on a real object [1,4] or, in the case of markerless AR, planes aligned with rigid real objects in the scene [8,29,14]. We present a method to perform real-time 2D augmentations on a non-rigid object, such as a flexible piece of cloth, using a single camera. Our technique involves increased realism by applying real-world illumination and shadows to the augmentation.

In order to perform augmentations on a flexible piece of cloth, a mesh representation of the cloth is acquired from the video stream. Cloth tracking is a difficult problem due to its flexibility and ability to self-occlude. Typically, complex physical simulations are used to achieve correct folds and wrinkles in virtual cloth [7]. Techniques to track cloth for the purpose of 3D shape reconstruction do exist [25,27,28], however these methods focus on dense reconstruction and therefore require too much processing to operate in real-time. Sparse reconstruction of cloth geometry from real-time video has been performed [17,18], although not for the purpose of augmented reality. In our method we acquire a sparse representation of the cloth by placing markers on its surface that can be identified in video images. Using this representation we render a 2D augmentation that appears as an image on the cloth. The quality of the augmentation is enhanced by applying real illumination and shadows. The addition of realistic lighting to AR scenes increases perception of spatial relations between real and virtual objects [21]. There are a number of techniques to insert virtual objects into a real scene using the real-world illumination. Based on pioneering work by Fournier et al. [13], Drettakis et al. present the first system for rendering moving virtual objects in a static scene with correct illumination using radiosity [11]. Debevec uses a light probe to gather the light model in a scene and then render virtual objects with global illumination and com-

D. Bradley
Department of Computer Science
Carleton University, Canada
derek@derekbradley.ca
(now at the University of British Columbia)

G. Roth
Institute for Information Technology
National Research Council of Canada
Gerhard.Roth@nrc-cnrc.gc.ca

P. Bose
Department of Computer Science
Carleton University, Canada
jit@scs.carleton.ca

posite them with a photograph [10]. Stauder estimates real point light sources in video by examining changes between neighbouring frames [30]. Gibson et al. use an omni-directional image to capture the real-world illumination and render virtual objects using sphere mapping [16] and hierarchical shaft data structures [15]. Haller et al. achieve real-time shadow casting from virtual to real objects and vice versa by modifying a shadow volume algorithm [19]. Our system includes a simple image-based method to acquire the illumination and shadows from the input frame and apply them to the 2D augmentation in real-time. This illumination technique allows for virtual and real dynamic, non-rigid objects in the scene.

Existing examples of non-rigid augmented reality are limited, as this is a new area of research. In our previous work [5], we present a system based on ARToolKit markers that includes a method to approximate the illumination and shadows. However, tracking failures result in large holes in the augmentation, and the approximate illumination calculation does not capture all shadows correctly. In our new system, these drawbacks have been overcome. Pilet et al. [24] have developed a system for real-time non-rigid surface detection. Their method uses 2D feature point matching between a reference image and the input frame in order to detect the surface of a flexible object. They use augmented reality to demonstrate the results of their system, however in this work AR is not their main focus. Recently, parallel research has yielded several techniques similar to our own. Pilet et al. extend their previous work to include shadows when augmenting deformable objects by comparing individual pixel intensities in each frame to the reference image [23]. White and Forsyth retexture non-rigid objects in video by acquiring texture coordinates at a coarse scale using correspondence reasoning, and a dense irradiance estimate at a fine scale using color reasoning [31]. However, this technique does not achieve real-time augmentations. Scholz and Magnor also retexture videos of deformable objects in an offline manner [26]. They use an array of color markers to locate the object and encode texture coordinates, radial basis functions for time-coherent texture interpolation, and surface reconstruction to determine shading maps. Ehara and Saito overlay a texture onto a T-shirt by learning the relationship between the deformation of an object and its silhouette in the image [12]. A database is constructed from a number of training images, and then a search is performed using the silhouette in each frame to get an estimate of the deformation and overlay a virtual T-shirt.

The remainder of this paper is organized as follows. In Sect. 2 we present our system in detail. Sect. 3 describes applications of our work, and Sect. 4 presents our conclusions.

2 Non-Rigid Augmented Reality

Our method to perform non-rigid augmented reality on cloth is divided into three sections: locating the cloth, acquiring the illumination, and rendering augmentations. Each phase is described in detail. However, we first present an overview of the system.

2.1 System Overview

Our system operates on video images from a live camera. We locate the cloth in the scene using a set of trackable markers that are placed on its surface. We use the marker locations to build a 2D mesh representation of the cloth in image-space, and we render the mesh with a texture on top of the video image in each frame. In addition, we acquire real world illumination from the input image and generate a shadow texture that we blend with the augmentation when rendering. The output is a mixed-reality video frame that contains an illumination-correct 2D augmentation on the surface of the flexible cloth. Fig. 1 illustrates the operation of our system for each video frame.

2.2 Locating the Cloth

In order to locate the cloth we visually track a number of circular targets, generated by a photogrammetry application called Photomodeler Pro[1]. Photogrammetry is the process of recovering 3D information from a scene captured by a set of 2D images. One way this is accomplished is by placing 2D markers throughout the scene and then locating them in the images offline. This is similar to our problem for cloth tracking except that we operate on live video images, so we develop an online algorithm to track the targets. The markers consist of an inner black circle surrounded by a white ring in the middle and a sequence of white and black ring segments on the outer border (see Fig. 2). The outer ring segments can be thought of as a 10-bit digital barcode wrapped around into a ring. This ring is used to uniquely identify each marker, independent of scale or orientation.

We arrange a set of coded-ring markers in a grid with known topology on the surface of the cloth (as illustrated in Fig. 3). The density of the marker grid affects the resolution of the augmentation mesh. Markers are placed as densely as possibly while ensuring that outer ring segments remain spatially closer to their respective marker centers than to any component of another marker. Leaving more space between markers allows for more flexible interaction at the cost of augmentation precision. In practice we found that a spacing of approximately one marker diameter in each direction yields sufficient resolution and flexibility for realistic augmentations. The

[1] www.photomodeler.com

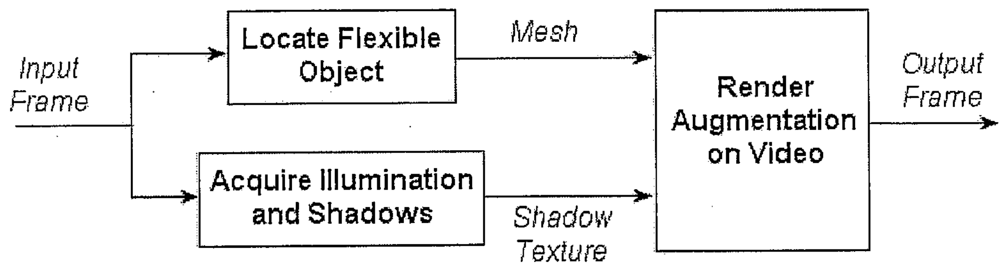


Fig. 1 Overview of augmented reality cloth system.

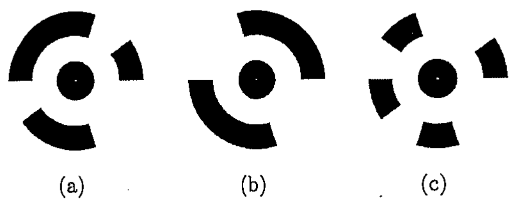


Fig. 2 Coded ring markers. Outer ring is a 10-bit binary code wrapped around a black circle and inner white ring. The codes are: a) 1010011011; b) 1110011100; c) 0010010101.

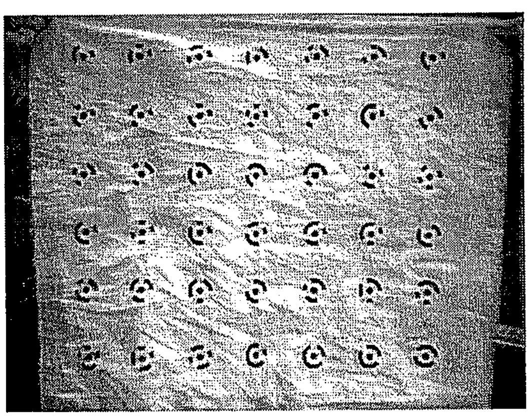


Fig. 3 Grid of circular markers on the cloth.

optimal scale of the markers depends on the distance of the cloth from the camera. We found that a projection of approximately 30 pixels in diameter is sufficient for robust tracking.

The first step in tracking is to adaptively threshold the input frame to create a binary image [6]. Contours of the connected regions in the binary image are found and a contour size filter is applied to remove unwanted noise. At this point we must determine which contours belong to the same marker, and then which specific contour represents the center of the marker. Simply processing the contours individually is not sufficient, so we perform a clustering step to group together contours belonging to the same marker. The clustering algorithm proceeds by calculating the pixel center of each contour to produce a set of 2D points on the image plane. Then the Delaunay triangulation of the points is computed. The Delaunay triangulation is used since it effectively captures the spatial relations within a point set at an affordable computational cost [22]. More specifically, points that are spatially close to each other will be connected together in a small sub-graph, since long and thin triangles are automatically avoided where possible. We attempt to extract the clusters from the triangulation by using an approach similar to Kruskal's algorithm for computing the Minimum Spanning Tree of a graph [9]. Initially, each point is considered to be a single cluster. The edges of the triangulation are then processed from shortest to longest. When an edge is processed, the two adjacent clusters are merged using an efficient Union-Find data structure [9]. The edges are processed until the number of clusters is at most half of the number of original points, and the length of the current edge is more than 2% longer than the previously processed edge. These parameters are chosen empirically based on the density of the marker grid. At this point we conclude that each cluster represents one coded ring target. We then select the elliptically-shaped contour closest to the center of the cluster and mark it as the center circle of the target. Fig. 4 illustrates these steps to find the target centers in a magnified region of an input image.

Once a set of possible target centers are found we attempt to uniquely identify each target using the coded outer ring. To do this, we scale the elliptical projection of the target center to obtain an ellipse that represents the outer border, and then we compute a homography from that ellipse to a circle in order to remove the perspective distortion. A homography is a 3 x 3 matrix that maps points from one plane to another [20]. The un-warped target is then radially sampled at multiple locations to determine the binary representation of the outer ring. We sample at three different radius values and five orientations per radius, for a total of fifteen 10-bit samples (or

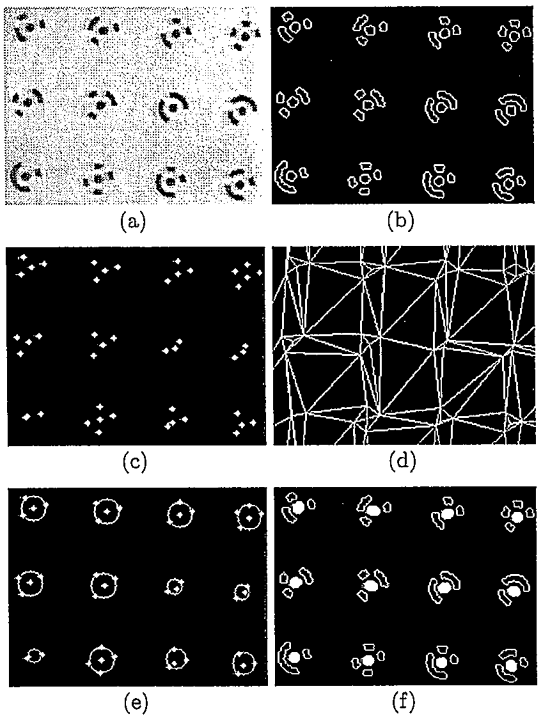


Fig. 4 Finding the circular target centers (magnified region of an input image). a) Input; b) Contours are found; c) Center of each contour computed; d) Delaunay triangulation; e) Result of clustering (each cluster shown with a minimum enclosing circle of the cluster points); f) Target centers are found.

one hundred and fifty pixel locations). The large number of samples provide increased tracking robustness. We use voting to determine if enough samples agree on the identity of the target or if the cluster is simply noise. Fig. 5 illustrates the application of the homography and the fifteen samples for one target.

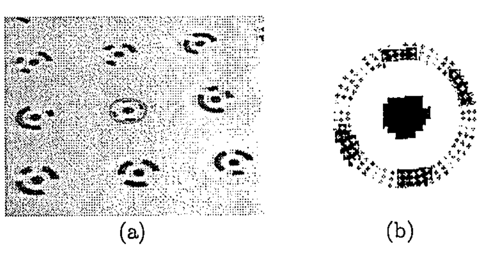


Fig. 5 Removing perspective distortion and sampling to uniquely identify a target. a) Elliptical border of the target in question is highlighted; b) Homography is applied and the now circular target is sampled to produce fifteen 10-bit samples.

The result of our tracking method is a set of pixel locations, indicating the target centers of the uniquely identified targets. The targets map to known locations on the surface of the cloth, so the pixel locations are triangulated to provide a 2D mesh representation. The triangulation is performed dynamically in each frame using the targets that are located (see Fig. 6a). Since we triangulate *only* the targets that are located, then the mesh is automatically interpolated over locations where a target is not found in a frame (for instance, the two missing targets in Fig. 6b highlighted by circles). In this case, triangles are automatically formed using neighbouring targets of the missing one. Although this is an approximation it prevents holes from appearing in the mesh, which is a much more visible and distracting artifact. Interpolation is linear in image space in order to reduce complexity and computational cost. Although a piece-wise linear representation of the cloth is unrealistic for 3D modeling, it is sufficient for 2D surface augmentations and affordable for real-time performance.

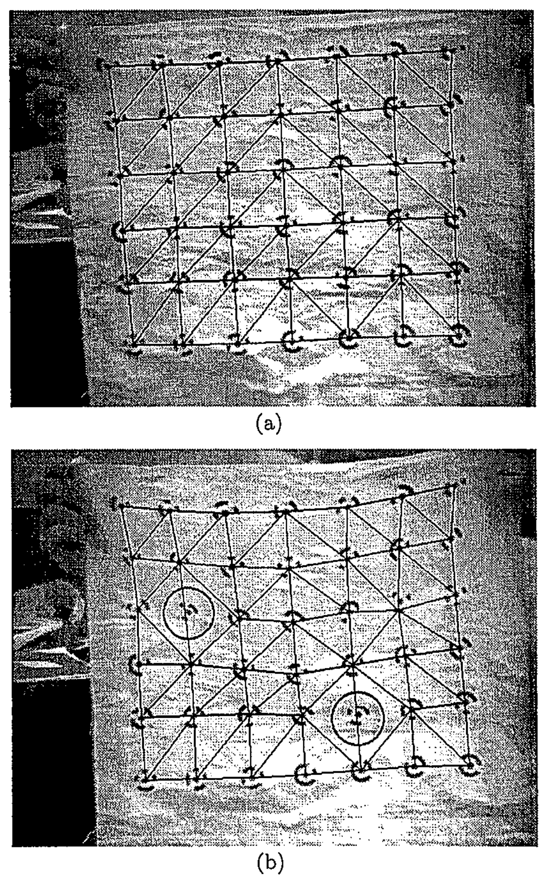


Fig. 6 Dynamic triangulation of the marker locations. a) When all the targets are located; b) When two targets (shown in circles) are not found due to tracking failure.

Tracking robustness is further increased by using temporal coherence. Image noise and cloth flexibility can lead to false-negatives. In practice, we find that this type of tracking failure occurs for a given marker in only a small number of consecutive frames, and so temporal coherence can be used to avoid it. When processing a frame, if one of the markers is not found, but yet it was found in at least one of the previous n frames then we compare its last known location with the set of possible markers that were not successfully identified in the current frame. If one of the possible marker locations is within d pixels of the last known location, then we assume this marker is the missing one. In practice we found that a value of 5 for n and 30 for d provide suitable results for video of size 640 x 480 pixels and cloth at a distance of approximately 1 meter from the camera. Using temporal coherence helps to reduce tracking failure, particularly due to rapid changes in illumination and surface deformations.

2.3 Acquiring the Illumination

The flexibility of cloth produces many small ripples and wrinkles, which lead to self-shadows on its surface. Additionally, most cloth exhibits diffuse reflection, and ripples and wrinkles result in spatial variations in illumination. By replicating the diffuse illumination and the shadows in the augmentation, we achieve a realistic augmented experience. These illumination effects are acquired directly from the input image. We use the fact that the cloth is light in color, and that the *observed* reflectance and shadows are the exact effects that we wish to apply to the augmentation. Our technique is to dynamically generate an illumination texture that we blend with the augmentation in each frame. We observe that this illumination texture is exactly the grayscale image of the input without the coded ring targets. We use image inpainting to remove the markers from the intensity image and replace the black areas with a blended average of the local neighbourhood. Image inpainting is an ancient technique to make undetectable modifications to images. Automatic digital inpainting methods have been developed to fill in a user-defined mask area using information from surrounding pixels [3,2]. However, these inpainting techniques typically require many iterations and are executed offline for a single image. We therefore implement a simple and fast algorithm which is less accurate but can be used on video frames as they are processed in real-time. Using a mask defined by the contours of the coded rings, the grayscale image is inpainted by shifting a 3 x 3 window around each contour in a decreasing spiral and averaging non-zero intensity values. This essentially provides a uniform blend of the local neighbourhood in the unknown region. The simplicity of the algorithm can sometimes result in illumination artifacts. However, since individual marker contours are generally small and most of the source image consists of low frequency variations, the artifacts are less noticeable. Furthermore, the interactive frame-rate is of more importance than per-pixel correct illumination. Fig. 7 demonstrates the results of our inpainting method. The inpainted image is used as the illumination texture when rendering.

2.4 Rendering Augmentations

The cloth augmentation is rendered in two passes. In the first pass we start with the input image (with the targets removed by inpainting), and we render the textured triangular mesh created from the target locations. In the second pass we re-draw the mesh, modulating the shadow texture using OpenGL blending. The projected screen coordinates for each mesh vertex are used as the texture coordinates for the shadow texture. This technique produces an illumination-correct textured mesh. As we mentioned, the flexibility of cloth produces many ripples and wrinkles during interaction, and these effects generate spatial illumination variations on the cloth surface. Our method captures these variations and renders realistic cloth augmentations. The results of rippling and wrinkling cloth are shown in Fig. 8.

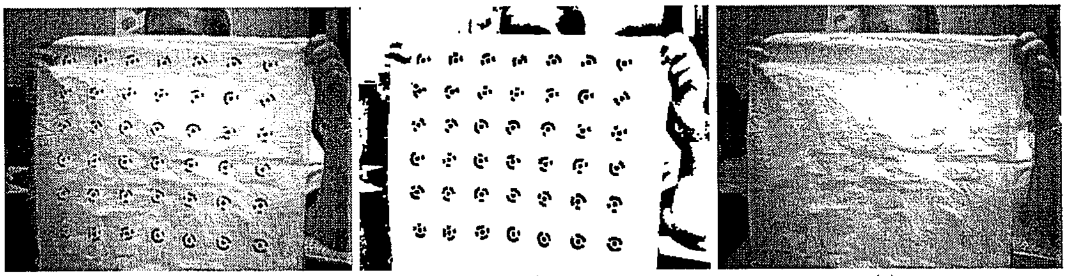
In order to demonstrate the full benefit of using correct illumination for cloth augmentations, consider Fig. 9 that shows an augmentation with constant illumination (Fig. 9a) and the same augmentation with real illumination applied (Fig. 9b). This clearly indicates that applying the illumination and shadows increases the realism of the augmentation.

We further highlight our method to establish common illumination in Fig. 10, where the shadow of a hand is rendered correctly, along with the more difficult illumination condition created by using a flashlight.

3 Applications

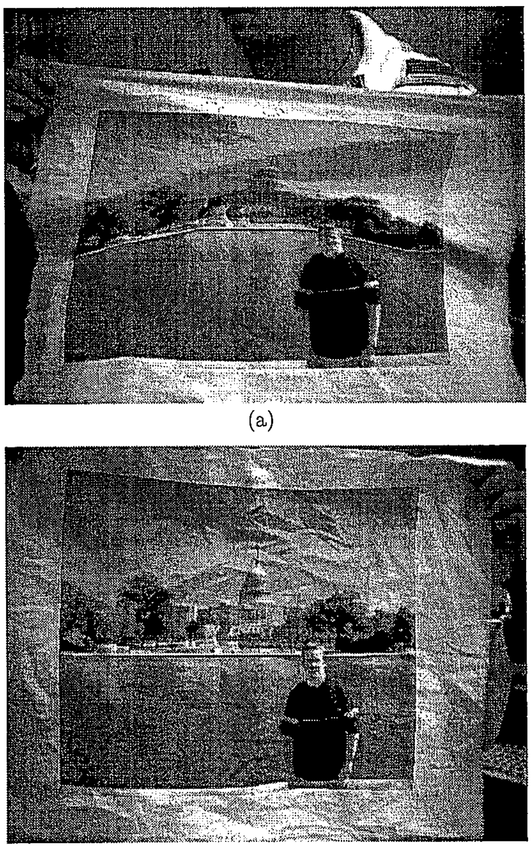
The ability to render augmentations on cloth leads to applications in the fashion industry. For example, users could wear clothing with markers and then virtually "try on" different outfits. We implement a related application for T-shirt design. In our application, users wear a marked T-shirt and select different logos for the front of the shirt. The selected logo is augmented on the shirt so that the user can interactively see what the final T-shirt would look like. Fig. 11 shows a screenshot of our augmented T-shirt application.

Related applications would be to augment logos or emblems on sheets or flags, add virtual advertising to professional sport uniforms in televised broadcasts, or interactively select curtain patterns. In addition, targets could be placed directly on a human body and medical augmentations could be rendered right onto the skin.



(a) (b) (c)

Fig. 7 Simple image inpainting. a) Source image; b) Mask image; c) Result of inpainting algorithm.



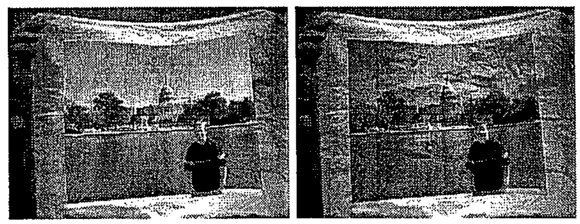
(a)

(b)

Fig. 8 Augmentation with correct illumination on cloth. a) Rippling cloth; b) Wrinkled cloth.

4 Conclusion

We present a method to perform real-time 2D augmented reality on a non-rigid object, such as cloth. We are able to sparsely track the cloth using a new circular marker system and apply correct illumination and shadows. Our system can generate realistic augmentations on the flexible cloth while undergoing rippling and wrinkling, even in difficult illumination conditions. Our method operates at an average of 10 frames per second on a Pentium 4, 3.4Ghz machine using a color Point Grey Dragonfly camera at a resolution of 640 x 480 pixels.

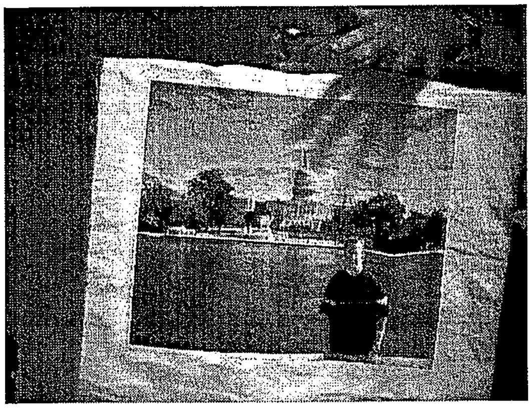


(a) (b)

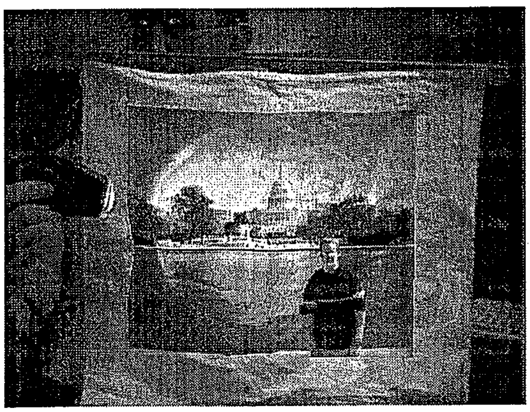
Fig. 9 Benefit of using correct illumination with cloth augmentations. a) Using constant illumination; b) Using correct, real illumination.

Our method does include a number of limitations. We assume that the cloth is diffuse and very light in color. The augmentations on the cloth are also rendered with only diffuse illumination. Although one could argue that logos on cloth often exhibit a significant specular component, simplifying the illumination is acceptable for the purpose of visualizing the logo in real-time. Also, self-occlusion will sometimes result in a tracking failure. This is mainly due to the clustering algorithm, as folds in the cloth can place two markers very close together and they will be clustered incorrectly. The dynamic triangulation step for building the augmentation will linearly interpolate over non-visible regions due to folds. However, this is only an approximation of the correct augmentation, so artifacts are noticeable. Real-time detection of cloth folds is a difficult task, and is outside the scope of this work. Finally, due to the complexity of cloth tracking, our system is limited in the range of motion allowed. Extreme deformations and occlusions by other objects typically result in a tracking failure or artifacts in the augmentation.

Acknowledgements This work was supported by the Natural Sciences and Engineering Research Council of Canada.



(a)



(b)

Fig. 10 Augmentation with correct shadows. a) Shadow of a hand showing each finger individually; b) Flashlight illumination (shown with increased brightness and contrast by 15% for better visualization).

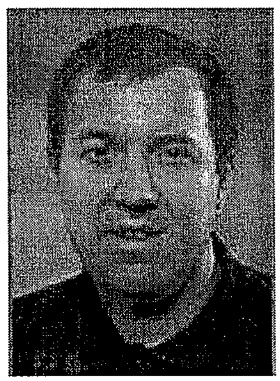
Fig. 11 Augmented T-shirt application to interactively design T-shirt logos. Users add images to the list on the right and then select the current logo for the live augmentation on the left.

References

1. ARToolKit: http://www.hitl.washington.edu/artoolkit
2. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Transactions on Image Processing 10(8), 1200–1211 (2001)
3. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: K. Akeley (ed.) Siggraph 2000, Computer Graphics Proceedings, pp. 417–424. ACM Press / ACM SIGGRAPH / Addison Wesley Longman (2000). URL citeseer.ist.psu.edu/bertalmio00image.html
4. Billinghurst, M., Kato, H., Poupyrev, I.: The magicbook - moving seamlessly between reality and virtuality. IEEE Comput. Graph. Appl. 21(3), 6–8 (2001)
5. Bradley, D., Roth, G.: Augmenting non-rigid objects with realistic lighting. Tech. Rep. NRC 47398, National Research Council of Canada (2004)
6. Bradley, D., Roth, G.: Adaptive thresholding using the integral image. Journal of Graphics Tools (Submitted 2005, accepted for publication)
7. Bridson, R., Marino, S., Fedkiw, R.: Simulation of clothing with folds and wrinkles. In: SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation, pp. 28–36. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2003)
8. Comport, A.I., Marchand, E., Chaumette, F.: A real-time tracker for markerless augmented reality. In: IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2003), pp. 36–45. Tokyo, Japan (2003)
9. Cormen, T.H., Stein, C., Rivest, R.L., Leiserson, C.E.: Introduction to Algorithms. McGraw-Hill Higher Education (2001)
10. Debevec, P.: Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques, pp. 189–198. ACM Press (1998). DOI http://doi.acm.org/10.1145/280814.280864
11. Drettakis, G., Robert, L., Bougnoux, S.: Interactive common illumination for computer augmented reality. In: Proceedings of the Eurographics Workshop on Rendering Techniques '97, pp. 45–56. Springer-Verlag (1997)
12. Ehara, J., Saito, H.: Texture overlay for virtual clothing based on pca of silhouettes. In: IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2006) (2006)
13. Fournier, A., Gunawan, A.S., Romanzin, C.: Common illumination between real and computer generated scenes. In: Proceedings of Graphics Interface '93, pp. 254–262. Toronto, ON, Canada (1993)
14. Genc, Y., Riedel, S., Souvannavong, F., Akinlar, C., Navab, N.: Marker-less tracking for ar: A learning-based approach. In: IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2002), pp. 295–304 (2002). Darmstadt, Germany
15. Gibson, S., Cook, J., Howard, T., Hubbold, R.: Rapid shadow generation in real-world lighting environments. In: Proceedings of the 14th Eurographics workshop on Rendering, pp. 219–229. Eurographics Association (2003)

16. Gibson, S., Murta, A.: Interactive rendering with real-world illumination. In: Proceedings of the Eurographics Workshop on Rendering Techniques 2000, pp. 365–376. Springer-Verlag (2000)
17. Guskov, I.: Efficient tracking of regular patterns on non-rigid geometry. In: Proceedings of ICPR, pp. 1057–1060 (2002)
18. Guskov, I., Klibanov, S., Bryant, B.: Trackable surfaces. In: ACM/EG Symposium on Computer Animation (2003)
19. Haller, M., Drab, S., Hartmann, W.: A real-time shadow approach for an augmented reality application using shadow volumes. In: ACM Symposium on Virtual Reality Software and Technology (VRST 2003), pp. 56–65 (2003)
20. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521623049 (2000)
21. Madsen, C.B., Sørensen, M.K.D., Vittrup, M.: The importance of shadows in augmented reality. In: Proceedings: 6th Annual International Workshop on Presence, Aalborg, Denmark (2003)
22. Okabe, A., Boots, B., Sugihara, K., Chiu, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, 2nd Edition. John Wiley and Sons (2000)
23. Pilet, J., Lepetit, V., Fua, P.: Augmenting deformable objects in real-time. In: ISMAR '05: Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 134–137. IEEE Computer Society (2005)
24. Pilet, J., Lepetit, V., Fua, P.: Real-time non-rigid surface detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005), pp. 822–828 (2005)
25. Pritchard, D., Heidrich, W.: Cloth motion capture. Computer Graphics Forum (Eurographics 2003) **22**(3), 263–271 (2003)
26. Scholz, V., Magnor, M.: Texture replacement of garments in monocular video sequences. In: W. Heidrich, T. Akenine-Moeller (eds.) Rendering Techniques 2006, Eurographics Symposium on Rendering, pp. 305–312. Eurographics (EG), ACM SIGGRAPH, Eurographics, Nicosia, Cyprus (2006)
27. Scholz, V., Magnor, M.A.: Cloth motion from optical flow. In: B. Girod, M. Magnor, H.P. Seidel (eds.) Proc. Vision, Modeling and Visualization 2004, pp. 117–124 (2004)
28. Scholz, V., Stich, T., Keckeisen, M., Wacker, M., Magnor, M.: Garment motion capture using color-coded patterns. Computer Graphics Forum (Proc. Eurographics EG'05) **24**(3), 439–448 (2005)
29. Simon, G., Fitzgibbon, A.W., Zisserman, A.: Markerless tracking using planar structures in the scene. In: Proc. International Symposium on Augmented Reality, pp. 120–128 (2000). URL http://www.robots.ox.ac.uk/ vgg
30. Stauder, J.: Augmented reality with automatic illumination control incorporating ellipsoidal models. IEEE Transactions on Multimedia **1**(2), 136–143 (1999). URL citeseer.ist.psu.edu/stauder99augmented.html
31. White, R., Forsyth, D.: Retexturing single views using texture and shading. In: European Conference on Computer Vision, vol. LNCS 3954, pp. 70–81. Springer (2006)

D. Bradley received a Bachelor of Computer Science degree with Highest Honours and the Senate Medal at Carleton University (Canada) in 2003, and a Master of Computer Science degree with the Senate Medal at Carleton University in 2005. He is currently a PhD student in the Department of Computer Science at the University of British Columbia (Canada) in the Imager Laboratory for Graphics, Visualizaton and HCI.



G. Roth received a B.Math from the University of Waterloo in 1976, a Master's of computer science from Carleton University in 1984, and a Ph.D. in electrical engineering from McGill in 1993. He is currently the Group Leader of the Computational Video Group at the Institute for Information Technology at the National Research Council of Canada, working in the fields of perceptual audio/video user interfaces, augmented reality, and building models from sensor data.

P. Bose received his B. Math (1990) and M. Math (1991) in Computer Science and Combinatorics from the University of Waterloo, Canada. He completed his Ph.D. (1994) in Computer Science at McGill University, Canada, where he received the D.W. Ambridge Award. Currently, he is an associate professor at the School of Computer Science in Carleton University, Canada.