Pre-print of article DOI: 10.1109/WACV.2012.6163037

Mutual Occlusion between Real and Virtual Elements in Augmented Reality based on Fiducial Markers

Silvio R. R. Sanches[1], Daniel M. Tokunaga[1], Valdinei F. Silva[1], Antonio C. Sementille[2], Romero Tori[1]
[1]Universidade de São Paulo, São Paulo, SP, Brasil
[2]Universidade Estadual Paulista, Bauru, SP, Brasil
silviorrs, dmtokunaga, valdinei.freire{@usp.br}, semente@fc.unesp.br, tori@acm.org

Abstract

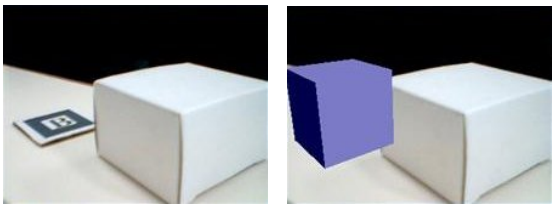
Augmented Reality (AR) systems which use optical tracking with fiducial marker for registration have had an important role in popularizing this technology, since only a personal computer with a conventional webcam is required. However, in most these applications, the virtual elements are shown only in the foreground a real element does not occlude a virtual one. The method presented enables AR environments based on fiducial markers to support mutual occlusion between a real element and many virtual ones, according to the elements position (depth) in the environment.

1. Introduction

Fiducial markers have been an efficient solution to solve the registration problem in AR environments. The three-dimensional coordinates of these markers are obtained by optical tracking and each virtual object is overlaid on a marker to align it with the real environment.

Methods which use fiducial marker for registration [9, 25, 5, 30, 29, 1] have been used in many areas of AR applications, such as education [3, 16], entertainment [21, 10] and industry [22, 29]. Although there are approaches that use specialized hardware [21, 22], applications based on fiducial markers are known for allowing an AR experience with only a conventional camera and a personal computer [24, 17].

However, in most of these systems, there is no mutual occlusion support – a real element in the foreground does not occlude a virtual one in the background. The virtual elements are always generated as foreground because there is no information about the depth of the real elements in the scene. This limitation can produce incoherent scenes which – in terms of cognitive psychology – may confuse the users [26]. As shown in Fig. 1, this usually occurs when a marker is visible[1] (not occluded) to the camera and its associated virtual object is larger than its area.



(a) (b)

Figure 1. An AR environment based on fiducial markers. (a) A visible marker is more distant from the camera than a real element. (b) The virtual object is generated in front of the real element. There is no mutual occlusion between the real and virtual element of the scene.

According to [11, 34, 2] the mutual occlusion between real and virtual elements in AR environments enhances the user's feeling that the virtual objects truly exists in the real world, which makes it a strongly desired attribute for certain applications.

Sophisticated AR systems usually acquire 3D geometric information from the real environment to enable mutual occlusion characteristic [11, 34, 12]. This requires specific hardware based on laser emission [31, 12], HMDs usually based on binocular video for stereo [32, 6, 34] or a moving camera [20] for depth estimation. However, when a single static webcam and a personal computer are application requirements, other strategies must be adopted. In [15], a method is presented which tracks the 3D silhouette of an object using only one camera to provide an occlusion shape, but the solution requires user interaction, which makes it not applicable in real-time.

In order to make a real element appear in front of a virtual one in an AR environment a real element must be ex-

[1]When a marker is not fully visible to the camera, it cannot be detected by optical tracking, and its associated virtual object is not rendered. However, there are methods with strong marker occlusion resilience able to estimate pose and position from a not fully visible marker [1].

tracted from the video frame for later insertion in front of the virtual one. In AR applications this process must be sufficiently computationally efficient to attain live streaming speed. Many real-time segmentation techniques can be found in the literature.

Traditional methods for this purpose presume that the video frame has been captured in a controlled environment, with a single color as background (usually blue or green) and with the environment lights configured to keep that color uniform [28, 19, 27, 7]. Briefly, these methods isolate the element of interest by removing the known background color which produces segmentation with low error.

Recent approaches developed by [13, 4, 33], in turn, make it possible to extract elements not only in real-time, but also from natural images (without a single color background). Although these methods are error-prone, videochats and videoconferencing systems, which replace the original background of a frame before sending it to remote users [4, 33], have become potential applications.

In this research, mutual occlusion was achieved by combining a real-time segmentation method – which extract a real element – with OpenGL framebuffer operations – which recover the pixels belonging to virtual objects immediately after rasterization. The latter information was used to reduce the area to be segmented in order to decrease pixels misclassification in the segmentation process for natural environment with natural background.

2. Methodology for Real Element Extraction

As discussed in section 1, the locations of the real elements are not known for most AR applications based on fiducial markers. Despite the fact that these coordinates can be obtained by fixing a marker in the real element (element of interest for the application) it is still necessary to extract these elements from their original background.

The probabilistic model proposed in [4] was used for foreground extraction in our solution. This model uses an energy minimization framework, in which a frame is represented as an array $z = (z_1, z_2, \cdots, z_n, \cdots, z_N)$ of pixels in the YUV color space, indexed by the single pixel n. A frame at time t is denoted z^t.

Temporal derivatives are denoted $\dot{z} = (\dot{z}_1, \dot{z}_2, \cdots, \dot{z}_n, \cdots, \dot{z}_N)$ and computed as $\dot{z}_n^t = |G(z_n^t) - G(z_n^{t-1})|$ each time t with a Gaussian kernel $G(.)$ at a scale of σ_t pixels. Spatial gradients are denoted $g = (g_1, g_2, \cdots, g_n, \cdots, g_N)$ where $g_n = |\nabla z_n|$ are computed by convolving the images with first-order derivative of Gaussian kernels with standard deviation σ_s (we use $\sigma_s = \sigma_t = 0.8$, as in [4]) .

Spatio-temporal derivatives are computed only on the Y channel. Motion observations are denoted $m = (g, \dot{z})$ and the segmentation task is to infer a binary label $\alpha^t \in \{F, B\}$ with F and B denoting foreground and background, respectively.

The model is a Conditional Random Field (CRF) [14] which models the conditional probability

$$p(\alpha^1, \ldots, \alpha^t | z^1, \ldots, z^t, m^1, \ldots, m^t) \propto \exp - \left\{ \sum_{t'=1}^{t} E^{t'} \right\} \quad (1)$$

where

$$E^t = E(\alpha^t, \alpha^{t-1}, \alpha^{t-2}, z^t, m^t). \quad (2)$$

The energy E^t associated with time t is a sum of terms in which likelihood and prior are not entirely separated. The energy decomposes as a sum of four terms:

$$\begin{aligned} E(\alpha^t, \alpha^{t-1}, \alpha^{t-2}, z^t, m^t) = \quad (3) \\ \eta V^T(\alpha^t, \alpha^{t-1}, \alpha^{t-2}) + \gamma V^S(\alpha^t, z^t) \\ + \rho U^C(\alpha^t, z) + \phi U^M(\alpha^t, \alpha^{t-1}, m^t), \end{aligned}$$

in which the first two terms are "prior-like" and the second two are observation likelihoods. η, γ, ρ and ϕ are normalizing parameters.

The temporal prior term $V^T(\cdot)$ imposes a tendency to the temporal continuity of segmentation labels. Second-Order Markov chain is used in the energy minimization framework to incorporate the intuition that a pixel that was in the background at time $t-2$ and in the foreground at time $t-1$ is far more likely to remain in the foreground at time t than to go back to the background. The temporal transition priors are learned from labeled data [18].

Spatial prior term $V^S(\cdot)$ is an Ising term, imposing a tendency to the spatial continuity of labels. This term is inhibited by high contrast.

Color likelihood term $U^C(\cdot)$ evaluates the evidence for pixel labels using the color distributions in the foreground and in the background. Likelihoods are modeled as histograms in the YUV color space.

The motion likelihood term $U^M(\cdot)$ uses spatial and temporal (computed from frames $t-1$ and t) derivatives $m = (g, \dot{z})$ to capture the characteristics of the features under foreground and background conditions. The motion likelihood is learned from some labeled ground-truth data [18] and then stored as 2D histograms to be used in likelihood evaluation.

3. Composition of the Augmented Reality Environment

In order to build an Augmented Reality environment we presumed that there is one real element in foreground and many virtual objects in the scene. One marker is fixed in the real element to obtain it position and the others are overlaid

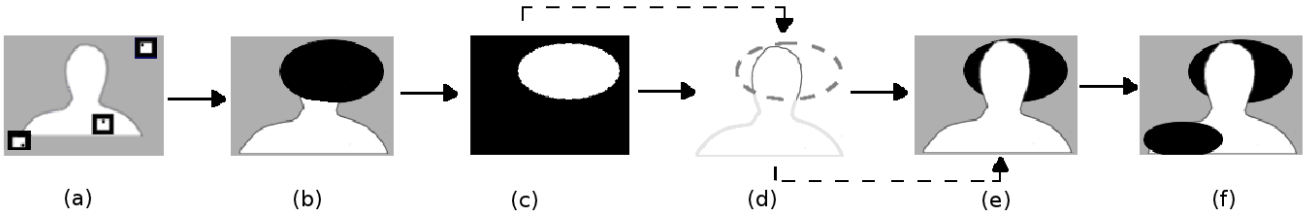


Figure 2. Scene composition process. (a) Original image with fiducial markers. (b) Virtual object in foreground generation. (c) Mask containing the pixels belonging to the virtual object in the foreground. (d) Foreground layer extracted using the mask. (e) Foreground layer overlaid the virtual object in the foreground. (f) Virtual object in the foreground overlaying the foreground layer. Mutual occlusion between real and virtual layers is achieved.

by virtual objects, as in traditional fiducial-marker-based applications (Fig. 2(a)).

The current frame sent by a camera is analyzed by the system to estimate pose and position of the fiducial markers. Next, the virtual objects in which their associated markers are more distant from the observer than the marker fixed on the real element are rendered. This incomplete version of the scene is sent to OpenGL framebuffer to generate a rasterized image which contains the virtual objects (Fig. 2(b)). After that, this rasterized image is recovered from the framebuffer to generate a mask to help the segmentation process (Fig. 2(c)).

The main challenge to solve the mutual occlusion problem in AR applications is the real element extraction. Methods for real-time image segmentation in natural environments are known to be error-prone, which can make them unsuitable for AR applications.

Our method, which uses the algorithm described in section 2 to generate a foreground layer (Fig. 2(d)), does not show all the pixels of the foreground layer in order to avoid misclassified pixels exhibition. Only the area of the foreground layer that overlies the virtual one needs to be shown. In other words, the area to be segmented (or shown to the user) is limited by a mask that contains the pixels belonging to virtual objects in the background (Fig. 3). Thus, the foreground layer is overlaid on the current image in order to show the real element as foreground (Fig. 2(e)). Finally, if there are virtual objects closer to the observer than the real element, these objects are generated as foreground (Fig. 2(f)).

4. Experimental Results

Our new method for scene composition is validated in this section. First, the qualitative results of our implementations of the segmentation method are shown. These tests verify the robustness of this algorithm when their parameters are adjusted for best performance in AR applications based on our method. The scene composition with mutual occlusion is validated by several video sequences which show AR environments that support mutual occlusion. The

Figure 3. Region of the frame where the foreground layer is visible. (a) A frame from the SEQ1 with the virtual objects. (b) The mask recovered from the framebuffer which contains the pixels belonging to the virtual objects in foreground. (c) Reduced area in which the segmentation must be applied.

reduction of the area to be segmented in a frame in order to reduce pixels misclassification is demonstrated by using a ground-truth.

4.1. Segmentation Tests

In order to test the segmentation method for natural environment, a more complex setup was required for its initialization and parameter adjustment. A ground truth [18] was used, the videos were labeled from 5-to-5 or 10-to-10 frames to obtain the temporal prior and the motion likelihood as well as to obtain the optimal parameters used in CRF ($\eta = 0.0018$, $\gamma = 1$, $\rho = 0.0338$, $\phi = 0.0413$) shown in Eq. 3. Fig. 4 show the qualitative results of the natural segmentation method.

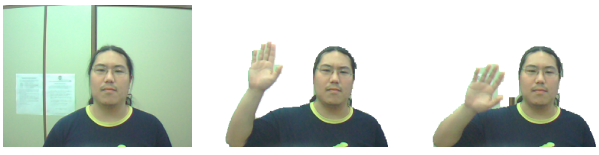
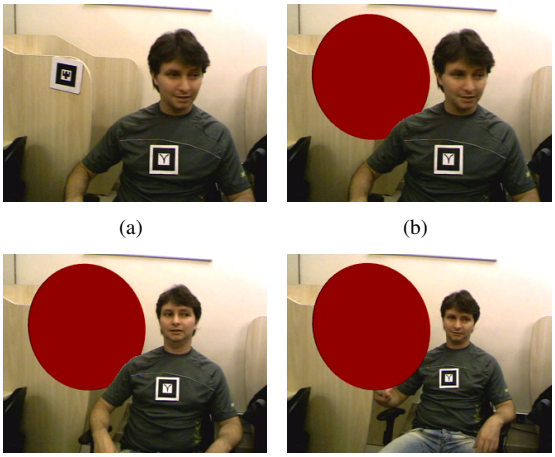


Figure 4. Segmentation. (a) Original image. (b,c) The original background was replaced with a constant white color by the algorithm shown in Eq 3.

4.2. Scene Composition Experiments

In order to show the results obtained, an environment with two markers was built: (1) a marker fixed on the real

element and (2) a marker fixed on an object in the environment. In this video sequence, the real element moves from the background to the foreground. The correct occlusion can be visualized in Fig 5.



(a) (b) (c) (d)

Figure 5. Scene composition. (a) A frame from SEQ2 sequence in which an original environment contains two fiducial makers. A virtual object was rendered over the marker fixed in the environment. The actor in the foreground (b,c) moves to the background (d). Note that the mutual occlusion was achieved.

The main problem in applications which use methods for segmentation in natural environments is the pixels misclassification in the segmentation task. Changes in lighting, distracting events such as element moving in the background and camera shake are situations that may occur while the video is being captured, which may generate segmentation errors [4, 33].

In order to avoid a large number of errors, the total of the pixels analyzed in the segmentation task was decreased, as shown in section 3. 30 frames (320x240 pixels) of the three sequences which simulate a typical fiducial-maker-based AR environment were analyzed to show that the segmentation method discussed in section 2 is suitable for our approach. Fig 7 shows that few segmentation errors were displayed to the user. Lowering the number of pixels clearly lowers the number of misclassified pixels.

The real element remains between the virtual ones in all the frames in SEQ1, and the segmentation area is limited by the virtual object in the background. In turn, the real element moves from the background to the foreground three times in SEQ2, and the virtual one moves from the foreground to the background in SEQ3. In this last sequence, there were no visible markers in the initial frames. Fig. 6 compares the total frames analyzed in SEQ1, SEQ2 and SEQ3 sequences.

Next, such test sequences were labeled (5-to-5 frames) to quantify the segmentation errors in order to demonstrate

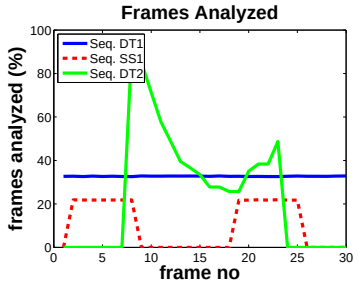
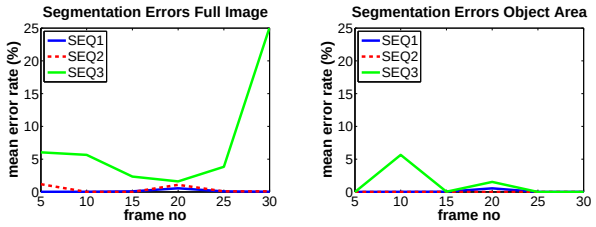


Figure 6. Percentage of frames analyzed in SEQ1, SEQ2 and SEQ3 frame sequences.

the applicability of the natural environment segmentation method in our approach. Most test sequences contain near stationary foreground since the segmentation method precision is smaller in this situation (the segmentation algorithm is based on pixel movement).

Fig. 7(a) and 7(b) show the percentage errors in the SEQ1, SEQ2 and SEQ3 sequences when the fully and the partial image (virtual objects in the background area) are analyzed, respectively.



(a) (b)

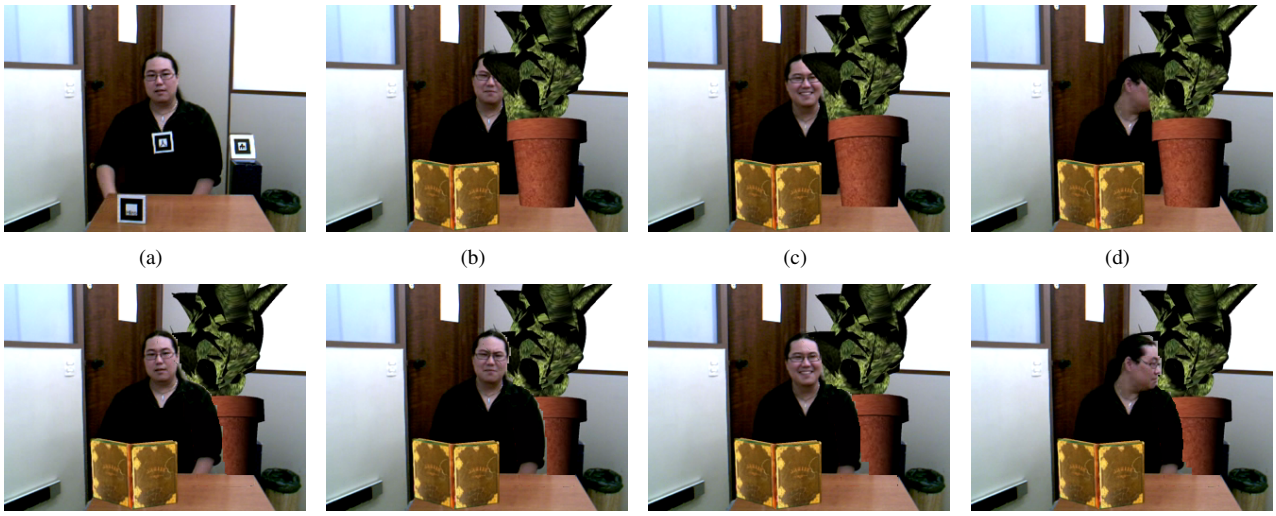
Figure 7. Quantitative Analysis. (a) Total errors when the full pixels are analyzed. (b) Total errors when the segmentation is applied in AR application based on our method (only the virtual objects in the background area). Note that the image was segmented only where the real element occludes a virtual one and there was at least one marker visible in the background.

Although our method allows mutual occlusion in fiducial-maker-based AR applications, some limitations were found in our solution. As the real element is associated with a fiducial marker, the latter is an image layer (2D element). In addiction, the scene must have a single real element moving in the environment.

A final demonstration of our method is shown in Fig. 8. The color likelihood model was manually initialized in the first frame for this demonstration.

5. Conclusion

The mutual occlusion is an important characteristic in AR systems since it allows a coherent visualization of the scene. However, AR systems based in fiducial markers usually generate the virtual objects in the foreground. The



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 8. The final demonstration. An AR environment based on fiducial marker with mutual occlusion. (a) A real environment with three visible markers. (b,c,d) Three frames from SEQ1 which show an AR environment without mutual occlusion. (e,f,g,h) Four frames from SEQ1 which show an AR environment with mutual occlusion. In order to hide the marker fixed in the real element, a coherent virtual object overlays it.

method presented here allows the generation of the AR environments which support mutual occlusion between real and virtual layers by combining a method for real-time foreground extraction and framebuffer operations.

The approach presented for this proposal reduces the segmentation area, which is limited by the virtual objects in the background. Quantitative evaluation has confirmed the validity of the proposed method and highlighted advantages with respect to segmentation of the full image. This made natural environments segmentation methods applicable in this context since the number of segmentation errors is lower than in applications such as videoconferencing or videochats, in which all the pixels misclassified in a frame must be shown to the user.

Next, we would like to apply a method for fractional pixels transparency [23] to achieve antialiasing effects on the edges of the real element. A method for subject assessment [8] should be applied to verify if the mutual occlusion achieved based on layers influences the user's perception concerning the depth of the real and virtual elements in the scene. We believe that our approach has potential application in teleconferencing, videochats and educational systems in which an AR environment based on fiducial markers can be used to build a scene.

Methods with strong marker occlusion resilience able to estimate pose and position from a not fully visible marker [1] may be suitable to be used with our approach since the situations in which our method is useful (when there are markers recognized in the background with large objects associated with them) occur more frequently.

References

[1] F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello. Rune-tag: A high accuracy fiducial marker with strong occlusion resilience. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 113–120, june 2011. 49, 53

[2] O. Cakmakci, Y. Ha, and J. P. Rolland. A compact optical see-through head-worn display with occlusion support. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '04, pages 16–25, Washington, DC, USA, 2004. IEEE Computer Society. 49

[3] Y.-C. Chen. A study of comparing the use of augmented reality and physical models in chemistry education. In *VRCIA '06: Proceedings of the ACM international conference on Virtual reality continuum and its applications*, pages 369–372, New York, NY, USA, 2006. ACM. 49

[4] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. Bilayer segmentation of live video. In *CVPR '06: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 53–60, Washington, DC, USA, June 2006. IEEE Computer Society. 50, 52

[5] M. Fiala. Artag, a fiducial marker system using digital techniques. In *CVPR '05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, pages 590–596, Washington, DC, USA, 2005. IEEE Computer Society. 49

[6] A. Fuhrmann, G. Hesina, F. Faure, and M. Gervautz. Occlusion in collaborative augmented environments. *Computers & Graphics*, 23(6):809 – 819, 1999. 49

[7] S. Gibbs, C. Arapis, C. Breiteneder, V. Lalioti, S. Mostafawy, and J. Speier. Virtual studios: an overview. *Multimedia, IEEE*, 5(1):18–35, Jan-Mar 1998. 50

[8] ITU-R. Recommendation ITU-R BT.1788 – methodology for the subjective assessment of video quality in multimedia applications. BT Series Broadcasting service (television) BT.1788, International Telecommunications Union, 2007. 53

[9] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IWAR '99: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality*, pages 85–94, Washington, DC, USA, 1999. IEEE Computer Society. 49

[10] C. Kirner, E. Zorzal, and T. Kirner. Case studies on the development of games using augmented reality. In *Systems, Man and Cybernetics. SMC '06. IEEE International Conference on*, volume 2, pages 1636 –1641, 8-11 2006. 49

[11] K. Kiyokawa, Y. Kurata, and H. Ohno. An optical see-through display for mutual occlusion with a real-time stereovision system. *Computers & Graphics*, 25(5):765 – 779, 2001. 49

[12] A. Kolb, E. Barth, R. Koch, and R. Larsen. Time-of-flight sensors in computer graphics. *Eurographics State of the Art Reports*, pages 119–134, 2009. 49

[13] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother. Bi-layer segmentation of binocular stereo video. In *CVPR '05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 407–414, Washington, DC, USA, June 2005. IEEE Computer Society. 50

[14] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. 50

[15] V. Lepetit and M. O. Berger. A semi-automatic method for resolving occlusion in augmented reality. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, number 2, pages 225 – 230, 2000. 49

[16] J. Martín-Gutiérrez, J. L. Saorín, M. Contero, M. A. niz, D. C. Pérez-López, and M. Ortega. Design and validation of an augmented book for spatial abilities development in engineering students. *Computers & Graphics*, 34(1):77 – 91, 2010. 49

[17] P. S. Medicherla, G. Chang, and P. Morreale. Visualization for increased understanding and learning using augmented reality. In *MIR '10: Proceedings of the international conference on Multimedia information retrieval*, pages 441–444, New York, NY, USA, 2010. ACM. 49

[18] Microsoft Corporation. Microsoft research – free research data, 2010. Accessed at May. 2010. 50, 51

[19] Y. Mishima. Soft edge chroma-key generation based upon hexoctahedral color space. U.S. Patent 5,355,174, Out. 1994. 11-10-1994. 50

[20] R. A. Newcombe and A. J. Davison. Live dense reconstruction with a single moving camera. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1498 – 1505, 2010. 49

[21] W. Piekarski and B. Thomas. Arquake: the outdoor augmented reality gaming system. volume 45, pages 36–38, New York, NY, USA, 2002. ACM. 49

[22] H. Regenbrecht, G. Baratoff, and W. Wilke. Augmented reality projects in the automotive and aerospace industries. *Computer Graphics and Applications, IEEE*, 25(6):48 – 56, nov.-dec. 2005. 49

[23] C. Rother, V. Kolmogorov, and A. Blake. "Grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004. 53

[24] S. R. R. Sanches, A. C. Sementille, I. A. Rodello, and J. R. F. Brega. The generation of scenes in mixed reality environments using the chromakey technique. In *ICAT '07: Proceedings of the 17th International Conference on Artificial Reality and Telexistence*, pages 296–297, Washington, DC, USA, 2007. IEEE Computer Society. 49

[25] D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavári, L. M. Encarnaçäo, M. Gervautz, and W. Purgathofer. The studierstube augmented reality project. *Presence: Teleoper. Virtual Environ.*, 11(1):33–54, 2002. 49

[26] A. B. Sekuler and S. E. Palmer. Perception of partly occluded objects: a microgenetic analysis. *Journal of Experimental Psychology: General*, 121(1):95–111, 1992. 49

[27] F. van den Bergh and V. Lalioti. Software chroma keying in an immersive virtual environment. *South African Computer Journal*, 24:155–162, Nov 1999. 50

[28] P. Vlahos. Comprehensive electronic compositing system. U.S. Patent 4,100,569, Jul. 1978. 50

[29] D. Wagner, T. Langlotz, and D. Schmalstieg. Robust and unobtrusive marker tracking on mobile phones. In *Mixed and Augmented Reality. ISMAR 2008. 7th IEEE/ACM International Symposium on*, pages 121 –124, 15-18 2008. 49

[30] D. Wagner and D. Schmalstieg. Artoolkitplus for pose tracking on mobile devices. In *Proceedings of 12th Computer Vision Winter Workshop (CVWW'07)*, pages 139–146, 2007. 49

[31] J. Wither, C. Coffin, J. Ventura, and T. Hollerer. Fast annotation and modeling with a single-point laser range finder. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '08, pages 65–68, Washington, DC, USA, 2008. IEEE Computer Society. 49

[32] M. M. Wloka and B. G. Anderson. Resolving occlusion in augmented reality. In *Proceedings of the 1995 symposium on Interactive 3D graphics*, I3D '95, pages 5–12, New York, NY, USA, 1995. ACM. 49

[33] P. Yin, A. Criminisi, J. Winn, and I. Essa. Bilayer segmentation of webcam videos using tree-based classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(1):30–42, 2011. 50, 52

[34] Y. Zhou, J.-T. Ma, Q. Hao, H. Wang, and X.-P. Liu. A novel optical see-through head-mounted display with occlusion and intensity matching support. In *Proceedings of the 2nd international conference on Technologies for e-learning and digital entertainment*, Edutainment'07, pages 56–62, Berlin, Heidelberg, 2007. Springer-Verlag. 49