View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Leeds Beckett Repository

Visual Servoing-based Augmented Reality

V. Sundareswaran and R. Behringer

Rockwell Science Center

1049 Camino Dos Rios

Thousand Oaks, CA 92360.

email: vsundar@rsc.rockwell.com, rbehringer@rsc.rockwell.com

1 Introduction

The notion of Augmented Reality (AR) is to mix computer-generated, synthetic elements (3D/2D graphics, 3D audio) with real world in such a way that the synthetic elements appear to be part of the real world. There are various techniques to accomplish this, including magnetic tracking of position and orientation, and video-based tracking. This paper focuses on the video-based AR in which a camera is used to image the real world, and the image is processed to determine where the computer-generated elements are to be displayed. Video-based AR is particularly popular among AR researchers because of the accuracy that can be achieved in video image processing.

The central problem in video-based Augmented Reality (AR) is the registration, or alignment of 3D graphical information with the image of a real scene. There are two ways in which an AR display could be presented to the user: a video monitor where the graphical elements are overlaid on the image seen by the camera, or a see-thru display in which only the graphics is displayed. In the case of a video monitor display, the graphical information should be positioned appropriately relative to the image of the object as seen by the camera. In the case of a see-thru display, the graphical information should be positioned on the display in such a way as to appear registered with the object being viewed through the display. In either case, if the position and orientation of the camera relative to the scene are known, the graphical rendering can be done correctly. In the

case of the see-thru display, the transformation from the camera to the viewer's eye should also be taken into account while rendering. Research in AR has focused on the registration problem during the past several years.

2 Background

There are three major approaches to solve the registration problem. These are: object pose estimation methods, observer pose estimation methods, and camera motion estimation methods. In theory, the first two approaches should map to the same general type of solution to determine the relative pose between the object and the observer. However, specific object/observer pose estimation methods from computer vision have been adapted for AR applications.

Object pose estimation methods determine the position and orientation of the object, for e.g., a plane containing landmarks such as LED patterns (MIT) or colored dots (Neumann and Cho, 1996). These are based on pose determination schemes such as Fischler and Bolles (1981). The aforementioned method involves solving a quadratic polynomial. Typically, the landmarks are located using image processing, and the pose is determined in each frame. The pose is then used to render the 3D model. Geometric invariants around the landmarks (Uenohara and Kanade, 1995) and affine coordinate system-based approches (Kutulakos and Vallino, 1996) are also used to determine object pose.

Magnetic tracking can readily provide the position and orientation of the observer (e.g., Webster et al., 1996, State et al., 1996). The major limitations of magnetic tracking are its short range (typically 8 ft radius) and sensitivity to metallic objects in the vicinity. Video-based observer pose estimation methods attempt to compute the position and orientation of the camera from the position of landmarks in the images. In a surgical application, Grimson et al. (1996) used a data-model minimization, a least squares minimization of distance between the image data and 3D model data obtained a priori by scanning with a laser rangefinder. Using the "Hung-Yeh-Harwood pose estimation method," Hoff et. al (1996) at Colorado School of Mines developed an observer

pose estimation from concentric circle markers. The work at UNC (State et al., 1996) integrated magnetic tracking and video-based observer pose estimation to demonstrate a robust system for AR.

Our approach falls under the third category of approaches: motion estimation. The general problem of reliable 3D motion estimation from image features is largely an unsolved problem in computer vision. However, by restricting to the subproblem of easily identifiable landmarks, the motion estimation problem can be solved. Koller et al. (1997) used a linear acceleration model for the camera motion to determine the motion of the camera to determine where the graphical elements are to be displayed. Our approach is based on the mathematical formalism of visual servoing, explained in the next section.

3 3D tracking

In this section, we describe the algorithm for tracking an object in three dimensions based on 2D coordinates of object features, and known 3D model of the object. The algorithm is based on principles from visual servoing.

3.1 Visual Servoing

Visual servoing is controlling a system - typically, a robot end-effector - based on processing visual information. It is a well-developed theory for robotic vision (Espiau et al., 1992, Feddema and Mitchell 1989, Hashimoto 1993, Papanikolopoulos et. al 1993, Sundareswaran et al., 1996, Weiss et al., 1987). Visual servoing is carried out in a closed-loop fashion, as shown in Fig. 1. We would like the set of system states $\mathbf{s}$ to attain certain target values $\mathbf{s}_r$. The current values of the states $\mathbf{s}$ are measured by a camera looking at the scene. We will illustrate the measurements by an example: we want to move the camera in such a way as to position the camera exactly 12 inches from a planar configuration of four dots with known geometry. From this known geometry, we can specify

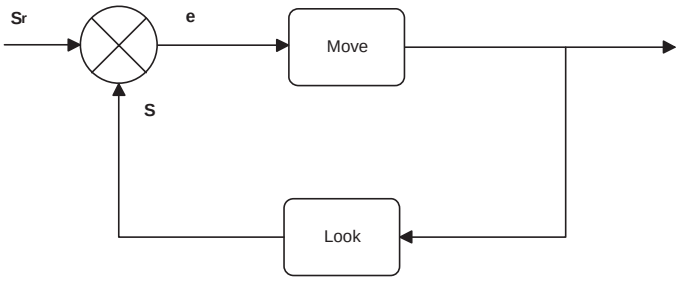

Figure 1: Schematic of the visual servoing approach.

the target configuration $\mathbf{s}_r$ of the four dots as will be seen by the camera in the final position of the camera. By processing the current image captured by the camera, we can also determine the current configuration $\mathbf{s}$ of the four dots.

The system uses the error (difference between the target values and current values) to determine the motion parameters T and Ω to move the camera in order to reduce the error. We adopt the standard coordinate systems shown in Figure 2 (the spherical coordinates could be adopted as readily). The translational velocity T has components U, V, and W. The components of the rotational velocity Ω are A, B and C.

To do this, we need to know the analytical relationship between the motion parameters and the state $\mathbf{s}$. Usually, the forward relationship, namely the change in $\mathbf{s}$ due to parameters T and Ω is known. The goal is to minimize $\|\mathbf{s} - \mathbf{s}_r\|$. Let us define the error function

$$\mathbf{e} = \mathbf{s} - \mathbf{s}_r \tag{1}$$

The change in the error is given by

$$\dot{\mathbf{e}} = \dot{S}_r.$$

We would like the error function to decay exponentially:

$$\dot{\mathbf{e}} = -\lambda \cdot \mathbf{e},$$

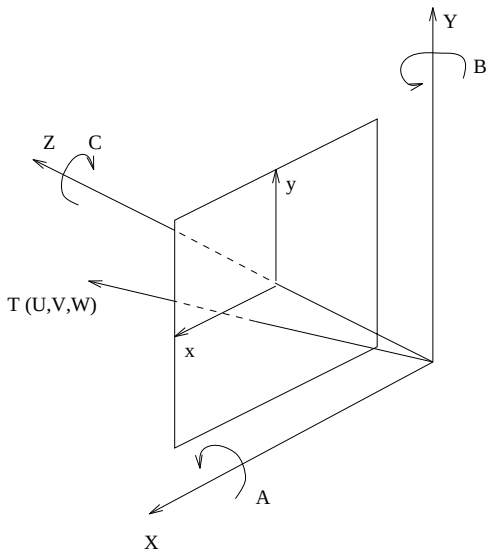

Figure 2: Coordinate systems and motion parameters.

where λ, the constant in the exponential, controls the decay rate (i.e., speed of convergence). Therefore $\dot{\mathbf{s}} = -\lambda \cdot (\mathbf{s} - \mathbf{s}_r)$ From standard optic flow equations (see for e.g., Horn, 1987), we know that we can write the 2D displacement of an image feature at (x_p, y_p) as:

$$\begin{aligned} \dot{x_p} &= \frac{1}{Z(x_p,y_p)} \left[-U + x_p W\right] + A x_p y_p - B\left[1 + x_p^2\right] + C y_p, \\ \dot{y_p} &= \frac{1}{Z(x_p,y_p)} \left[-V + y_p W\right] + A\left[1 + y_p^2\right] - B x_p y_p - C x_p. \end{aligned} \tag{2}$$

We assume that the images are planar, obtained by the pin-hole perspective approximation with a focal length of unity (see Figure 2). This relationship between change in 2D projection of a point and the motion parameters is of the form

$$\dot{\mathbf{s}} = L \begin{pmatrix} T \\ \Omega \end{pmatrix}, \tag{3}$$

where L is the "interaction matrix" consisting of 2D coordinates (x_p, y_p) and the depth Z of the 3D point projected at (x_p, y_p), T is the translation vector and Ω is the rotation vector. We would like

to determine T and Ω. Assuming that the motion of features $\mathbf{s}$ is due to the motion T and Ω, we obtain:

$$L \begin{pmatrix} T \\ \Omega \end{pmatrix} = -\lambda \mathbf{e}. \tag{4}$$

Inverting Eqn. 4, we get the control law

$$\begin{pmatrix} T \\ \Omega \end{pmatrix} = -\lambda L^{+} \mathbf{e}, \tag{5}$$

where L^{+} is the pseudo-inverse of L.

This allows us to compute the motion of the camera required to minimize the error $\mathbf{e}$. When performed in closed-loop, the value $\mathbf{s}$ will reach $\mathbf{s}_r$ when error $\mathbf{e}$ is reduced to zero.

3.2 Application to AR

In vision-based AR, landmarks or features are chosen on the object to be registered with, and these can be measured on the image taken by the camera, as well as on the 2D rendering of the 3D scene as seen by the "virtual" camera. The goal of the AR process is to maintain the alignment between the image landmarks and the graphical rendering of the landmarks (these need not actually be rendered). In other words, we would like to minimize the error between these two sets of values, yielding the following choice: the target values ($\mathbf{s}_r$) are measured from the image, the current values ($\mathbf{s}$) measured from the graphics, and the control (computed from Eqn. 5)is applied to the virtual camera that renders the 3D scene. The goal of the control process is to minimize the difference between the image location of the landmarks and the rendered position of the landmarks. Evidently, if successful, this control process will achieve the registration of the image and the model. If operated continuously, the control process will keep the model registered with the image.

4 Experimental results

We have implemented the visual servoing-based AR on a PC with 200 MHz Pentium Pro processor, an Imaging Technology framegrabber, and an OpenGL accelerator card. The landmarks are circular concentric ring markers. Each marker has a unique internal structure, and this identifies the marker uniquely in images. The circular shape simplifies the image processing to locate and identify the markers. The overall flow diagram of the system is shown in Figure 3.

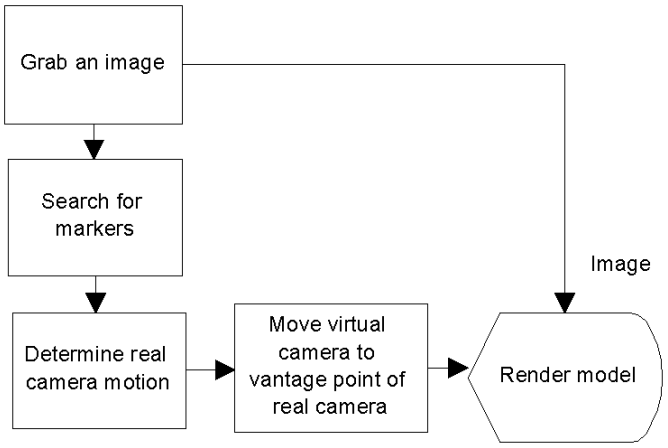

Figure 3: Flow diagram of the system

The image is grabbed by the Imaging Technology framegrabber, and the processing is carried out in the CPU. The graphical rendering is done using World Tool Kit (WTK). The system runs at 8-10 fps. The registration of the 3D wireframe model of a PC with an image of the PC is shown in Figure 4.

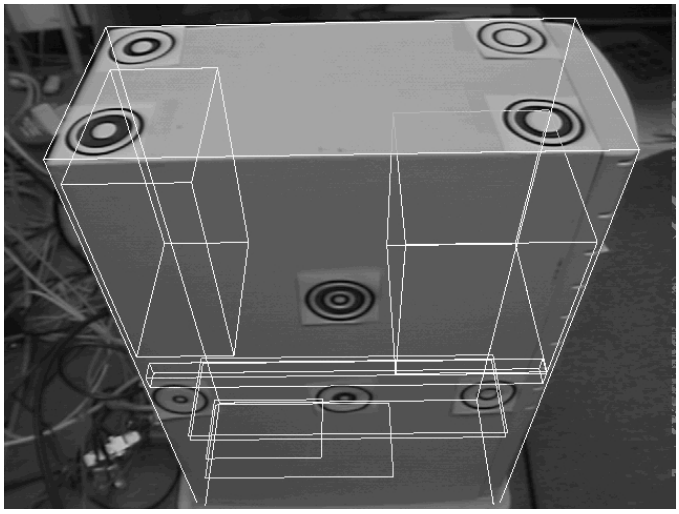

Figure 4: Sample frame from the visual servoing-based AR display. A sequence can be viewed at http://hci.rsc.rockwell.com/3Dreg.shtml.

5 Discussion

Minimizing the distance between image landmarks and their counterparts in the model projection is, by definition, the registration requirement. This minimization is carried out directly by the approach described in this paper. Thus, we believe that the visual registration problem is solved in a direct way by our approach. Also, since the loss of alignment is due to motion, computing the motion to reduce misalignment is a direct solution. The six motion parameters-three translational velocity components, and three rotational velocity components-are computed, and the virtual camera is "moved" with these parameters. Since the computation is carried out in closed loop, the motion of the camera is imitated by the virtual camera rendering the graphics.

The approach is independent of the type of landmark, as long as at least four of them can be detected and identified (i.e., the corresponding model landmarks can be determined). Three landmarks are required to determine the six motion parameters (three landmarks yield two equations each, as in Eqn. 2). However, to resolve the ambiguity of planar orientation, we need at least four

landmarks. When more than four landmarks can be detected, we have an overdetermined system, and the redundancy increases robustness. Non-coplanar landmarks also increase the robustness.

In the description of Equation 3, we noted that the computation requires the Z values of the landmarks. Normally, in an image processing situation, this will be a problem since the depth values are not known. But in the AR application, this is obtained from the rendered scene since the geometry of the model and the position of the viewpoint are known. However, the implication of this, and of Equation 2 is that the motion computations are valid only in a region around the current coordinates. If the motion is so fast that the control cannot keep up with the motion, the control will break down. In practice, this did not appear to cause problem for reasonable, hand-held motion of the camera. If the camera were to be head-mounted, additional sensors (e.g., inertial) may be needed to supplement the tracking while the visual control catches up.

In our current implementation, there is no temporal flow of information. A Kalman filtering approach could be used to enhance the performance by predicting the state of the system, as done for example in Azuma 1995, and Koller 1997.

6 Conclusions

In this paper we presented a new approach to solving the registration problem in vision-based AR, by using the visual servoing technique. Our implementation results are encouraging, and the registration algorithm is being filed for patent.

References

[1] Azuma, R. 1995. Predictive Tracking for Augmented Reality. Ph.D. Dissertation, University of North Carolina at Chapel Hill. Computer Science technical report TR#95-007, February 1995.

[2] Espiau, B., Chaumette, F., and Rives, P. 1992. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326.

[3] Feddema, J.T., and Mitchell, O.R. 1989. Vision-guided servoing with feature-based trajectory generation. *IEEE Trans. on Robotics and Automation*, 5(5):691–700.

[4] Fischler, M.A., and Bolles, R.C. 1981. Random Sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381-395.

[5] Grimson, W.E.L., Ettinger, G. J., White, S.J., Lozano-Perez, T., Wells III, W.M., and Kikinis, R. 1996. An automatic registration method for frameless stereotaxy, image guided surgery, and enhanced reality visualization. *IEEE Transactions on Medical Imaging*, 15(2):129-140.

[6] Grosso, E., Tistarelli, M., and Sandini, G. 1992. Active/dynamic stereo for navigation. *Lecture Notes in Computer Science 588: Computer Vision–ECCV '92*, ed. G. Sandini. Santa Margherita: Springer-Verlag, pp. 516–525.

[7] Hashimoto, K. ed. 1993. *Visual Servoing*, volume 7 of World Scientific Series in Robotics and Automated Systems. Singapore: World Scientific.

[8] Hoff, W. A., Lyon, T., and Nguyen, K. 1996. Computer Vision-based registration techniques for augmented reality. *Proc. of Intelligent Rootics and Computer Vision XV*, Vol 2904 in Intelligent Systems and Advanced Manufacturing, SPIE, Boston, Massachusetts, pp.538-548.

[9] Horn, B.K.P. 1987. *Robot Vision*. Cambridge: The MIT Press.

[10] Koller, D., Klinker, G., Rose, E., Breen, D., Whitaker, R., and Tuceryan, M. 1997. Real-time Vision-Based Camera Tracking for Augmented Reality Applications. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology (VRST-97)*, Lausanne, Switzerland, pp. 87-94.

[11] Kutulakos, K.N., and Vallino, J. 1996. Affine object representations for calibration-free Augmented Reality. *Proc. IEEE Virtual Reality Annual Symposium (VRAIS)*.

[12] MIT wearables group. http://wearables.www.media.mit.edu/projects/wearables/augmented-reality.html

[13] Neumann, U., and Cho, Y. 1996. A Self-Tracking Augmented Reality System. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pp. 109-115.

[14] Papanikolopoulos, N., Khosla, P., and Kanade, T. 1993. Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision. *IEEE Trans. on Robotics and Automation*, 9(1):14–35.

[15] State, Andrei, Gentaro Hirota, David T. Chen, William F. Garrett, and Mark A. Livingston. 1996. Superior Augmented-Reality Registration by Integrating Landmark Tracking and Magnetic Tracking. *Proceedings of SIGGRAPH 96*, pp. 429-438.

[16] Sundareswaran, V., Bouthemy, P., and Chaumette, F. 1996. Exploiting Image Motion for Active Vision in a Visual Servoing Framework. *International Journal of Robotics Research*, 15(6):629-645.

[17] Uenohara, M., and Kanade, T. 1995. Vision-based object registration for real-time image overlay. *Proc. 1st Intl. Conf. on Computer Vision, Virtual Reality, and Robotics in Medicine,* Nice, France.

[18] Webster, A., Feiner, S., MacIntyre, B., Massie, W., and Krueger, T. 1996. Augmented Reality in architectural construction, inspection, and renovation. *Computing in Civil Engineering*, pp. 913-919.

[19] Weiss, L.E., Sanderson, A.C., and Neuman, C.P. 1987. Dynamic sensor-based control of robots with visual feedback. *IEEE Trans. on Robotics and Automation*, 3(5):404–417.