ACCV2002: The 5th Asian Conference on Computer Vision, 23–25 January 2002, Melbourne, Australia.

Side Statistics and Maximum Discriminant Analysis for Real-Time Tracking

Zhengyou Zhang, Ying Wu[*], Zicheng Liu
Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA
Email: zhang@microsoft.com

Abstract

We propose a new technique for tracking based on side statistics and maximum discriminant analysis. The object to be tracked is modeled by a set of sample points on the boundary together with image statistics inside the object. Tracking is conducted by maximizing the discrimination between the object and the background, based on an adapted Kullback-Leibler divergence without knowing the statistics of the background. Since no knowledge of the background is required, our technique is particularly useful in dynamic environments where the background can change substantially during the performance of a visual task, or when a system needs to be deployed in different environments. Because we use both the side statistics and the boundary information, our method is more robust than the traditional approaches that use either just boundaries or just regions. As will be shown experimentally, our technique can deal with complex environment, changing background, and partial occlusion, and it is real-time and accurate. We have used the technique to track a panel such as a piece of paper in an application system called VISUAL PANEL *which serves as a wireless mobile input device to a computer.*

1. Introduction

Tracking moving objects has been a central research area in computer vision, and has a wide range of applications such as video surveillance, 3D reconstruction, video compression, augmented reality, and human computer interaction. Efficiency and robustness of tracking are critical in most of these applications because only a small percentage of computer resources can be allocated for tracking and the environment is usually complex and dynamic. Accuracy is another important consideration because a successful vision application relies on an accurate input provided by tracking, e.g., the pose estimation of an object. This paper addresses these issues and describes a new tracking technique based on side statistics and maximum discriminant analysis. We have used the technique in an application system called VISUAL PANEL which serves as a wireless mobile input device to a computer by tracking a panel such as a piece of paper and a pointer such as a finger [16].

Many tracking techniques have been proposed in computer vision, and many are adaptation or reformulation of those developed in radar target tracking [1, 6]. In radar target tracking, a target can be relatively easily detected through thresholding because a target might be a bright point on a dark background. In computer vision, an image is cluttered, and most of the data is irrelevant to the object being tracked. The problem of matching ambiguity and data missing is therefore more severe in computer vision. Several data association versions of the Kalman filter have been used, including multiple hypotheses [15] and joint probability data association filter [11]. Another category of tracking techniques based on particle filtering have been studied extensively in the past few years [7, 8, 12]. The idea is to generate a lot of hypotheses according to estimated non-parametric distribution. This approach is usually more robust but less accurate than the Kalman-filter-based approach.

Tracking techniques also differ, depending on the representation used to describe the objects to be tracked. The first category is based on region information such as statistic distribution of color and/or texture [2, 4]. The second category is based on boundary information such as snakes or active contours [13, 3, 10]. Contour-based techniques usually provide more accurate tracking results, but are easily trapped by strong edges in the background, and are thus less robust than region-based tracking.

In this paper, we propose a new tracking technique which takes advantage of both boundary- and region-based approach. We represent an object to be tracked by its boundary together with statistics within a narrow band inside the boundary (*side statistics* for short), and we track the boundary based on a *maximum discriminant analysis* principle in order to differentiate the object being tracked from the background. Discriminative methods are widely used in classification [5, 14] and typically yield more robust results. They,

[*] Address: Northwestern University, Evanston, IL 60208, USA

however, get little attention in tracking. In many applications, we can learn (either online or offline) the characteristics of the object to be tracked, but the background can change substantially in a dynamic environment, and can be very different from one setup to another. Our method is particularly useful for this class of applications since we use the statistics of the object of interest but not that of the background. Because we use both the side statistics and the boundary information, our method is more robust than the traditional approaches that use either just boundaries or just regions. As will be shown experimentally, our technique can deal with complex environment, changing background, and partial occlusion, and it is real-time and accurate.

The paper is organized as follows. Section 2 describes our approach to representing objects with side statistics and to tracking using maximum discriminant analysis. Section 3 presents the application of the proposed approach to tracking a quadrangle.

2. Discriminative Tracking With Side Statistics

In this section, we describe how to represent objects with side statistics and how to track them using maximum discriminant analysis.

Tracking can be formulated as an estimation problem. Let I be the current image, and $\mathbf{x}$ be the object to be tracked. Bayes' theorem says that the *a posteriori* probability of observing $\mathbf{x}$ given I is

$$p(\mathbf{x}|I) = \frac{p(I|\mathbf{x})p(\mathbf{x})}{p(I)} = \frac{p(I|\mathbf{x})p(\mathbf{x})}{p(I|\mathbf{x})p(\mathbf{x}) + p(I|\bar{\mathbf{x}})p(\bar{\mathbf{x}})} , \quad (1)$$

where $p(I|\mathbf{x})$ is the likelihood function (i.e., a measure of how likely image I is observed given what we know about object $\mathbf{x}$), $p(\mathbf{x})$ is the prior model of $\mathbf{x}$ (i.e., our prior knowledge of the object being tracked), and $\bar{\mathbf{x}}$ denotes everything which does not belong to the object being tracked, i.e., the background. At time t, we observe a sequence of images I_t, I_{t-1}, ..., and the tracking task is then to estimate $\mathbf{x}_t$ that maximizes $p(\mathbf{x}_t|I_t, I_{t-1}, \ldots)$. Applying Bayes' theorem yields $p(\mathbf{x}_t|I_t, I_{t-1}, \ldots)$

$$= \frac{p(I_t|\mathbf{x}_t)p(\mathbf{x}_t|I_{t-1}, \ldots)}{p(I_t|\mathbf{x}_t)p(\mathbf{x}_t|I_{t-1}, \ldots) + p(I_t|\bar{\mathbf{x}}_t)p(\bar{\mathbf{x}}_t|I_{t-1}, \ldots)} . \quad (2)$$

Here, $p(\mathbf{x}_t|I_{t-1}, \ldots)$ is a prediction of $\mathbf{x}$, i.e., prior knowledge about $\mathbf{x}_t$, based on previous estimate of the object and knowledge of the object's dynamics. More precisely, we have

$$p(\mathbf{x}_t|I_{t-1}, \ldots) = \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|I_{t-1}, \ldots) , \quad (3)$$

where $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ describes the probability of object state transition (i.e., knowledge of the object's dynamics). By now, it is clear how an object can be tracked at t if it is tracked until $t-1$ and if we know how the object will likely evolve and if we also know the background. This is the generative approach or graphical modeling to tracking. Although theoretically very interesting and complete, it requires to model the background almost in the same way as for the object of interest, which is difficult, if not possible, because of the complexity and dynamic nature of the environment. Even if possible, a system built in one environment will need to be retrained when deployed in a different environment. Furthermore, the background is usually irrelevant, making its modeling waste of resources. However, if background is ignored, then tracking might not be very stable. For example, the region-based approach may not work for an object with homogeneous regions, while the boundary-based approach may be stuck at a strong edge belonging to the background.

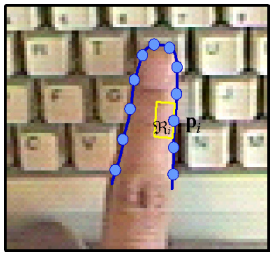


Figure 1. *Boundary representation and side statistics.*

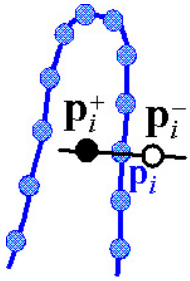


Figure 2. *Inner and outer pixels along a boundary normal line.*

2.1. Representation: Side Statistics

There are many possible ways to represent an object: templates which can only handle in-image translation; parametric templates which can handle global deformation described by a parametric model; region statistics which can handle significant deformation but is usually not detailed enough; boundary which is accurate but may have difficult to distinguish edges of the object of interest from those belonging to the background.

We describe an object by a set of points on the boundary together with statistics within a narrow band inside the boundary. Refer to Fig. 1. The set of points $\{\mathbf{p}_i|i = 1, \ldots, N\}$ can be the polygonal representation of the boundary, or can be the sample points on a parametric curve such as a cubic spline. In the latter case, the number of degrees of freedom is much smaller than $2N$, usually making the tracking more stable. Furthermore, for each sample point $\mathbf{p}_i$, we compute image statistics of a rectangle, denoted by $\mathcal{R}_i$, aligned with the tangent vector at $\mathbf{p}_i$ and located inside the boundary, as shown as a yellow rectangle

in Fig. 1. For image statistics, we simply model the color variation within $\mathcal{R}_i$ as a Gaussian distribution and compute the mean $\mathbf{c}_i$ and covariance matrix $\mathbf{C}_i$ in the RGB space. In summary, an object of interest $\mathbf{x}$ is represented by

$$\mathbf{x} = \{(\mathbf{p}_i, \mathbf{c}_i, \mathbf{C}_i) | i = 1, \ldots, N\} \ . \tag{4}$$

Since we attach side statistics to each sample point on the boundary, we obtain quite accurate description of the object of interest, and the accuracy increases by using a larger number of sample points, at the expense of more computational cost. Furthermore, side statistics is computed from a window, allowing us to deal with object deformation in image due to camera perspective projection when the object and/or the camera move. In our implementation, the sample points are separated by 5 to 15 pixels, and the window size for side statistics is 5×10 pixels (5 pixels in the normal direction and 10 pixels along the tangent direction). To deal with the camera noise and illumination variation, $\mathbf{C}_i$ is increased by adding diag $(15^2, 15^2, 15^2)$.

2.2. Maximum Discriminant Analysis

Since we do not have a model of the background, instead of maximizing the *a posteriori* probability (2), we propose to track a moving object by estimating the boundary $\{\mathbf{p}_i | i = 1, \ldots, N\}$ which maximizes the discriminance between the two sides of the boundary while maximizing the object likelihood based on learned side statistics. A natural choice of the merit function is the Kullback-Leibler divergence [5].

The Kullback-Leibler divergence (or Kullback divergence/distance) is the information-theoretic divergence between two probability distributions $p(\mathbf{x})$ and $q(\mathbf{x})$, given by

$$D_{KL}(p(\mathbf{x}), q(\mathbf{x})) = \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} \ . \tag{5}$$

This is a non-negative measure of the discrimination power between two probability distributions that equals zero only when they are identical.

Now, we are adapting the Kullback-Leibler divergence to serve tracking. Considering the normal through a sample point $\mathbf{p}_i$ on the boundary (see Fig. 2). We denote the immediate neighboring pixel within the object region by $\mathbf{p}_i^+$, and the immediate neighboring pixel in the background area by $\mathbf{p}_i^-$. Collectively, we have

$$\mathbf{x}^+ = \{(\mathbf{p}_i^+, \mathbf{c}_i, \mathbf{C}_i) | i = 1, \ldots, N\} \tag{6}$$
$$\mathbf{x}^- = \{(\mathbf{p}_i^-, \mathbf{c}_i, \mathbf{C}_i) | i = 1, \ldots, N\} \ . \tag{7}$$

Following previous discussions, we can readily define the merit function for tracking as $D(\mathbf{x}_t | I_t, I_{t-1}, \ldots)$

$$= \int p(I_t | \mathbf{x}_t^+) p(\mathbf{x}_t | I_{t-1}, \ldots) \log \frac{p(I_t | \mathbf{x}_t^+)}{q(I_t | \mathbf{x}_t^-)} d\mathbf{x}_t \tag{8}$$

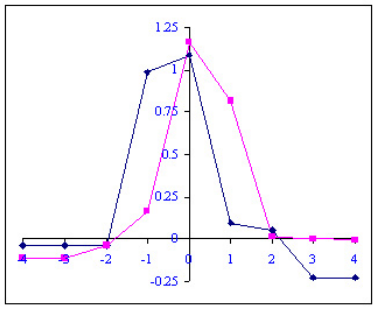


Figure 3. *Kullback-Leibler distances along the normal line across the boundary for two sample points arbitrarily chosen.*

where $p(\mathbf{x}_t | I_{t-1}, \ldots)$ is given by (3). If we consider individual sample points, then we have $D(\mathbf{x}_t | I_t, I_{t-1}, \ldots)$

$$= \sum_{i=1}^{N} p(I_t | \mathbf{p}_{it}^+, \mathbf{c}_{it}, \mathbf{C}_{it}) p(\mathbf{x}_t | I_{t-1}, \ldots) \log \frac{p(I_t | \mathbf{p}_{it}^+, \mathbf{c}_{it}, \mathbf{C}_{it})}{q(I_t | \mathbf{p}_{it}^-, \mathbf{c}_{it}, \mathbf{C}_{it})} \ . \tag{9}$$

The likelihood for the inner pixel is given by

$$p(I_t | \mathbf{p}_{it}^+, \mathbf{c}_{it}, \mathbf{C}_{it}) = \exp\left(-d_M(\mathbf{c}(\mathbf{p}_{it}^+); \mathbf{c}_{it}, \mathbf{C}_{it})\right) \ , \tag{10}$$

where $\mathbf{c}(\mathbf{p}_{it}^+)$ is the color vector of the pixel at $\mathbf{p}_{it}^+$, and $d_M(\mathbf{g}; \mathbf{c}, \mathbf{C})$ is the Mahalanobis distance between two vectors $\mathbf{g}$ and $\mathbf{c}$, i.e.,

$$d_M(\mathbf{g}; \mathbf{c}, \mathbf{C}) = (\mathbf{g} - \mathbf{c})^T \mathbf{C}^{-1} (\mathbf{g} - \mathbf{c}) \ . \tag{11}$$

The likelihood for the outer pixel is given by

$$q(I_t | \mathbf{p}_{it}^-, \mathbf{c}_{it}, \mathbf{C}_{it}) = \begin{cases} p_{it}^- & \text{if } p_{it}^- > \lambda, \\ \lambda & \text{otherwise.} \end{cases} \tag{12}$$

where $p_{it}^- = \exp\left(-d_M(\mathbf{c}(\mathbf{p}_{it}^-); \mathbf{c}_{it}, \mathbf{C}_{it})\right)$. Here, λ is introduced to deal with the numerical instability due to the division operation in (9) when $q(I_t | \mathbf{p}_{it}^-, \mathbf{c}_{it}, \mathbf{C}_{it})$ is very small. When $\exp(-d_M(\mathbf{c}(\mathbf{p}_{it}^-); \mathbf{c}_{it}, \mathbf{C}_{it})) < \lambda$, it is almost sure that pixel $\mathbf{p}_{it}^-$ belongs to the background, and there is no need to further differentiate it from other background pixels. In our implementation, we set $\lambda = 0.3$.

From (9), we see that tracking a moving object is to find the most discriminant pixel for each sample point, and this can be done by searching along the normal line going through the predicted position for the pixel which maximizes the adapted Kullback-Leibler divergence. Notice that the resulting $\mathbf{p}_{it}$ is not necessarily the same physical point on the object as $\mathbf{p}_{i,t-1}$. The tracking works as long as $\mathbf{p}_{it}$ stays in the same image rectangle $\mathcal{R}_i$ as $\mathbf{p}_{i,t-1}$ because of using the side statistics which characterizes $\mathcal{R}_i$. As an example, Figure 3 shows the Kullback-Leibler divergence on pixels along the normal line across the boundary for two

sample points arbitrarily chosen. The horizontal axis indicates the pixel position; 0 indicates the expected boundary pixel, a positive value indicates an inside pixel, and a negative value indicates a background pixel. The vertical axis is the Kullback-Leibler divergence. We observe clearly that the Kullback-Leibler divergence maximizes at the expected boundary position.

The side statistics $(\mathbf{c}_i, \mathbf{C}_i)$ can also be updated using the side statistics of the tracked sample points. In order to deal with occlusion and disappearance, we only update side statistics if the color in the current side rectangle $\mathcal{R}_i$ is not very different from the predicted mean color $\mathbf{c}_i$.

3. Example: Tracking a Quadrangle

In this section, we apply the proposed technique to tracking a quadrangle. This is used in a system called *Visual Panel* to serve a wireless input device of a computer system. The reader is referred to [16] for details of that system.

3.1. Automatic Detection

We have developed a simple technique based on Hough transform to automatically detect a quadrangle in an image [9]. Take the image shown in Fig. 4a as an example. A Sobel edge operator is first applied, and the resulting edges are shown in Fig. 4b. We then build a 2D Hough space for lines. A line is represented by (ρ, θ), and a point (u, v) on the line satisfies $\cos(\theta)u + \sin(\theta)v - \rho = 0$. An edge point with orientation is mapped into the (ρ, θ) space. In our implementation, θ is divided into 90 intervals from -90° to 90°, and ρ is divided into 100 intervals from range from $-d$ to d, where d is the half of the image diagonal. The Hough space for the edges in Fig. 4b is shown in Fig. 4c.

We then examine the strong peaks in the Hough space whether four of them form a reasonable quadrangle. By "reasonable", we mean:

- the neighboring sides should differ at least by 20° in orientation;
- the opposite sides are close to be parallel (the orientation difference is less than 20°);
- the opposite sides are not close to each other (at least 40 pixels of difference in ρ); and
- there are indeed a large number of edges on the quadrangle.

The last test is necessary because a point in the Hough space corresponds to an infinite line, and a quadrangle formed by 4 lines may not correspond to any physical quadrangle in an image. The quadrangle detected in Fig. 4a is shown with red lines on the image. Our current implementation of quadrangle detection achieves 22 frames per second for image resolution $320x240$ on a PC III 1G Hz.

(a) (b)

Figure 5. *Comparison between (a) tracking with edges and (b) tracking with side statistics.*

3.2. Tracking

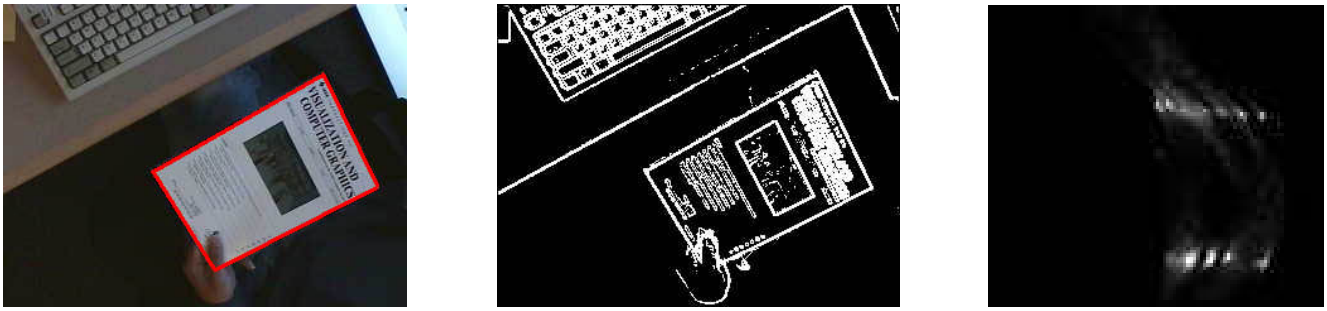
A quadrangle is represented by its four corners. Each side is represented by 15 points, and the side statistics is computed for each point when the quadrangle is detected. After tracking, a line is fitted to the sample points for each side of the quadrangle, and the intersections of the neighboring lines give the updated corners of the quadrangle. Because the fitting is based on a robust technique (least-median squares), partial occlusion or disappearance is allowed. Furthermore, the corners of the quadrangle are tracked with subpixel precision because they are computed by intersecting fitted lines. With our proposed discriminative tracking technique, we almost achieve real-time tracking (more than 29.5 frames per second), while our previous edge-based tracking technique achieves 26 frames per second.

Figure 5 shows a comparison of the two techniques. When I was moving a book from top to down, one side of the quadrangle was stuck at the desk border with the edge-based technique, as shown in Fig. 5a, while the new technique based on side statistics tracked the book very well, as shown in Fig. 5b.

Figures 6 and 7 show another tracking sequence with different background. Figure 6 shows the automatic detection result, while Figure 7 shows a few sample results of the tracking under various situations. Note that this sequence is quite difficult since the background contains books of similar color and there are a large number of edges. Note also that we have not used any background subtraction or frame difference technique to reduce the background clutter. As can be observed, our technique tracks very well under perspective distortion, illumination change, partial disappearance, size change, and partial occlusion.

4. Conclusion

We have described a new technique for tracking based on side statistics and maximum discriminant analysis. The object to be tracked is modeled by a set of sample points on the boundary together with image statistics inside the object. Tracking is conducted by maximizing the discrimination



(a) (b) (c)

Figure 4. *Automatic quadrangle detection. (a) Original image with detected quadrangle overlaid as green lines; (b) Edges image obtained with Sobel detector; (c) Hough space.*

between the object and the background, based on an adapted Kullback-Leibler divergence without knowing the statistics of the background. Since no knowledge of the background is required, our technique is particularly useful in dynamic environments where the background can change substantially during the performance of a visual task, or when a system needs to be deployed in different environments. Because we use both the side statistics and the boundary information, our method is more robust than the traditional approaches that use either just boundaries or just regions. As have been shown experimentally, our technique can deal with complex environment, changing background, and partial occlusion, and it is real-time and accurate.

References

[1] Y. Bar-Shalom and T. Fortmann. *Tracking and Data Association*. Academic, New York, 1988.

[2] B. Bascle and R. Deriche. Region tracking through image sequences. In *Proceedings of the 5th International Conference on Computer Vision*, pages 302–307, Boston, MA, June 1995. IEEE Computer Society Press.

[3] A. Blake, M. Isard, and D. Reynard. Learning to track the visual motion of contours. *Artificial Intelligence*, (78):101–133, 1995.

[4] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 142–149, South Carolina, June 2000. IEEE Computer Society.

[5] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley, New York, 2001.

[6] N. Gordon, D. Salmond, and A. Smith. A novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. Radar, Sonar and Navigation*, 140(2):107–113, 1996.

[7] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *The International Journal of Computer Vision*, 29(1):5–28, 1998.

[8] M. Isard and A. Blake. ICondensation: Unifying low-level and high-level tracking in a stochastic framework. In H. Burkhardt and B. Neumann, editors, *Proceedings of the 5th European Conference on Computer Vision*, pages 893–908, Freiburg, Germany, June 1998.

[9] R. Jain, R. Kasturi, and B. Schunck. *Machine Vision*. McGraw-Hill, New York, 1995.

[10] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:266–280, Mar. 2000.

[11] C. Rasmussen and G. Hager. Probabilistic data association methods for tracking complex visual objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):560–576, June 2001.

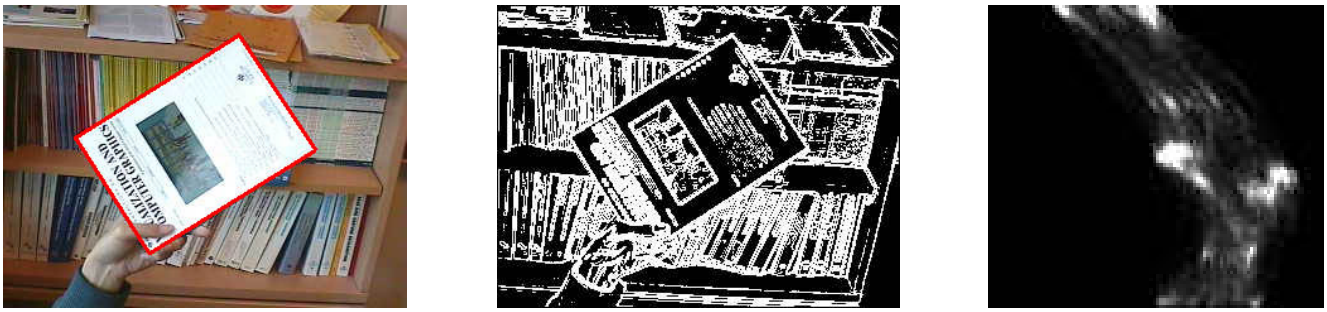
[12] H. Sidenbladh, M. Black, and D. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In D. Vernon, editor, *Proceedings of the 6th European Conference on Computer Vision*, volume II, pages 702–718, Dublin, Ireland, June 2000.

[13] D. Terzopoulos and R. Szeliski. Tracking with kalman snakes. In A. Blake and A. Yuille, editors, *Active Vision*, pages 3–20. MIT Press, 1992.

[14] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[15] Z. Zhang and O. Faugeras. Three-Dimensional Motion Computation and Object Segmentation in a Long Sequence of Stereo Frames. *The International Journal of Computer Vision*, 7(3):211–241, 1992.

[16] Z. Zhang, Y. Wu, Y. Shan, and S. Shafer. Visual Panel: Virtual mouse, keyboard and 3d controller with an ordinary piece of paper. In *Proceedings of ACM Workshop on Perceptive User Interfaces (PUI)*, Orlando, Florida, Nov. 2001.



(a) (b) (c)

Figure 6. *Another example of automatic quadrangle detection. (a) Original image with detected quadrangle overlaid as red lines; (b) Edges image obtained with Sobel detector; (c) Hough space.*

(a) (b) (c)

(d) (e) (f)

Figure 7. *Sample results of a tracking sequence under various situations.*