Object Tracking, Hand Pose Estimation and Direct Manipulation Interfaces

Peter Juritz

Abstract

I this article we review the research and techniques developed in three areas: object tracking, hand pose estimation and direct manipulation and tangible interfaces.
An introduction to object tracking is given and the role that object representation and image features play in object tracking is explained. This is followed by a description of the popular techniques in background subtraction and silhouette tracking.
We give a brief and wide review of popular techniques in hand pose estimation. More specifically we concentrate on model based pose estimation and single frame pose estimation. The differences and benefits of these topics are discussed.
Finally a brief description is given of tangible based interfaces, difficulties in their implementation and their benefits.

Keywords: hand pose estimation, object tracking, direct manipulation, terrain

1 Object Tracking

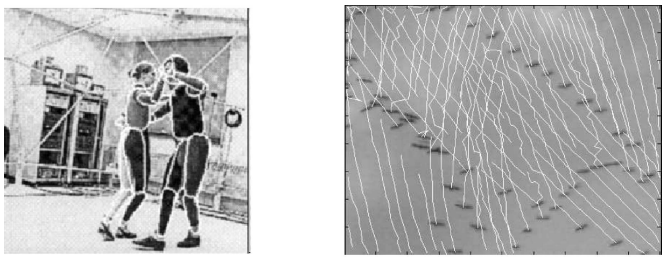
Object tracking is the task of detecting the position of an object in a scene consisting of single image or sequence of frames[Yilmaz et al. 2006]. There are many techniques which can be applied to solve this problem, however, each techniques is often best applied to a specific domain. Yilmaz et. al. suggest that the techniques can be categorised based on the following factors: the object representation used, the image features used and the method of modelling motion, appearance and shape[Yilmaz et al. 2006].

1.1 Difficulties

There are many difficulties involved in object tracking. Yilmaz et al identify the following common obstacles to object tracking[Yilmaz et al. 2006].

- loss of information caused by projection of the 3D world on a 2D image
- noise in images
- complex shapes and motion
- nonrigid or articulated nature of objects
- partial and full object occlusions
- scene illumination changes
- real-time processing requirements

Many of these problems are addressed by constraining the search space and parameters during evaluation. For example, assumption will be made about an objects position in frame T from its detected position in frame $T-1$. Constraints can also be applied to features of an object with articulated geometry, more on this will be discussed in later sections.



(a) (b)

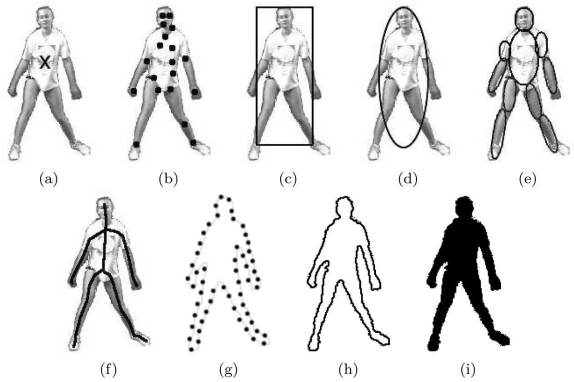
Figure 1: (a) Human tracking segmentation from Gavrila and Davis [Gavrila and Davis 1996](b) Bird tracking from Shafique and Shah[Shafique and Shah 2005].

1.2 Object representation and Image Features

In order to track an object through a sequence of frames it is first necessary to detect the object. This process depends on two factors. Tracking algorithms must use the most applicable object representation and from this choose the best image features to identify the object by.

1.3 Object Representation

Object representation refers to how the target object is modelled in the scene. This representation can range from a single point to a complex composite of shapes as illustrated in figure X. Object representation must be chosen to best match the target object and required information. Figure 1 illustrates how the requirement of the application affects the object representation. Shafique and Shah use a point representation to track the position of birds while Gavrilla and Davis use a composite of edge boundaries to track and analyse movements of people [Shafique and Shah 2005; Gavrila and Davis 1996].
Figure 2 shows different object representations derived for a single image.



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

(f) (g) (h) (i)

Figure 2: Object representations. (a) Centroid, (b) multiple points, (c) rectangular patch, (d) elliptical patch, (e) part-based multiple patches, (f) object skeleton, (g) complete object contour, (h)

control points on object contour, (i) object silhouette. Taken from Yilmaz et al.

1.4 Image features

Image features are components of an image or sequence of frames which gives rise to its structure. Common image features are colour, edges, optical flow, and texture.

1.4.1 Colour

Colour is the most widely used image feature used in object tracking although it is sensitive to illumination.[Yilmaz et al. 2006]. Colour data can be represented in a different number of ways. Most commonly the RGB colour space is used however it does not handle illumination changes well[Yilmaz et al. 2006]. Human perceptual differences in colour are known to differ from changes in RGB and hence RGB is not perceptually uniform[Paschos 2001].The $L*u*v*$, $L*a*b*$ and HSV (Hue, Saturation, Value) colour spaces have been shown to handle illumination more effectively although they are sensitive to noise[Song et al. 1996].

1.4.2 Edges

Edges are sections of images where there is a large transition in image intensity[Yilmaz et al. 2006]. An important feature of edges is that their detection is robust to changes in illumination. Edges are often used when the boundary of an object is being detected[Yilmaz et al. 2006]. The most commonly used edge detection algorithm is the Canny edge detector[Canny 1986].

1.4.3 Texture

Texture describes the intensity variation of a surface and represents features such as smoothness or roughness[Yilmaz et al. 2006]. Image texture has also been shown to be less sensitive to illumination changes than colour[Yilmaz et al. 2006]. The most common texture descriptors are: Gray-Level Cooccurence Matrices [Dinstein et al. 1977], Law's texture meshes[Laws 1980], wavelets[Mallat 1989] and steerable pyramids[Greenspan et al. 1994].

1.5 Background Subtraction

An object can be tracked and detected by modelling the background of the scene and then finding deviations from this in each coming frame. This technique is called background subtraction. Many different methods have been used to achieve this. Difficulties include: changes in illumination, shadows, camera jitter and non-static backgrounds. We will briefly explain some of the most popular methods used for background subtraction.

To handle temporal changes in illumination Wren et. al. modelled the colour at each pixel $I(x, y)$ by a Gaussian function[Wren et al. 1997]. As new pixel data comes in over time the mean and variance of this Gaussian is updated. Incoming pixels are then tested against these Gaussian and marked as either background or foreground pixels. However, Gao et al showed that a single Gaussian does not perform well for outdoor scenes since it can not compensate for repetitive object motion, shadows and reflectance[Boult et al. 2000]. Stauffer and Grimson addressed this by using multiple Gaussian for each pixel[Stauffer and Grimson 2000]. If a match to one of the Gaussian is not found for an incoming pixel a new Gaussian is created with the mean set to the value of the new pixel.

More complex methods process more information than just colour values for each pixel. Elgammal and Davis have proposed a technique where the background pixels are modelled by a kernel density (Kernel refers to the shape and appearance of an object)[Elgammal et al. 2000]. Their method does not only look at a single pixel to classify an incoming pixel but its neighboring pixels as well. Their method can therefore handle camera jitter and small background movements.

Another approach to background subtraction is to model each pixel or block of pixels as being in a certain state - usually background or foreground. Ritcher et al use Hidden Markov Models to model the state and state transition of blocks of pixels to perform background subtraction[Rittscher et al. 2000].

Oliver et al. have suggested using eigenspace decomposition of a sequence of frames. Incoming images are then projected onto the eigenspace and foreground pixels are found by examining the differences.

Many of these techniques can not handle dynamic backgrounds well. Monnet et al, and Zhong and Sclaroff have proposed methods which handle time varying backgrounds by applying autoregressive moving average processes [Monnet et al. 2003; Zhong and Sclaroff 2003]. These processes can learn how an environment changes over time.

Modern methods use a combination of image features and attempt to compensate for dynamic backgrounds, changes in illumination and temporal effects. These methods are computationally efficient and can be run in real time[Yilmaz et al. 2006]. In practice these methods often lead to small errors which must compensated for in resulting algorithms. An important limitation is that many of these techniques require the camera to be static[Yilmaz et al. 2006]. This has been addressed by rebuilding the background every few frames or by attempting to model the changing projection of the camera. These techniques however are limited themselves as they make assumptions about camera movement and environment geometry[Yilmaz et al. 2006].

1.6 Silhouette Tracking

Silhouette tracking aims to determine an object region based on a computed object region from previous frames[Yilmaz et al. 2006]. This is necessary when the target object can not be represented by simple shapes such as a point or ellipse. Silhouette tracking algorithms are beneficial in that they can handle a wide variety of shapes[Yilmaz et al. 2006].

Yilmaz et al suggest that silhouette tracking algorithms can be divided into two categories: shape matching and contour evolution.

1.7 Shape Matching

Shape matching tracks a silhouette by searching the image for the object in every frame. The search is performed by computing the similarity of a proposed object silhouette with that of the previous frame. The object is assumed to have only translated from the previous frame. This assumption means that from frame to frame an object must exhibit rigid body motion only. To handle this problem the object silhouette representation is updated at each frame which allows for non rigid body motion over time. Shape matching algorithms differ mainly in how they attempt to match the hypothesised silhouette to a candidate silhouette in the new frame. Here we will detail some techniques which have been employed for this purpose.

In order to compute the score of a silhouette match Huttenlotcher et al. use the Hausdorff metric on an edge map representation of the proposed position to measure how well the predicted position of the silhouette matches the input frame[Huttenlocher et al. 1993]. The Hausdorff metric measures the magnitude of differences between

two sets of points.
Another method attempts to match two silhouettes detected in consecutive frames. Each individual silhouette is usually detected through background subtraction. Once these silhouettes have been detected, matching is performed by calculating a distance between the old silhouette and candidate silhouette[Yilmaz et al. 2006].
Kang et al. suggesteusing colour and edge histograms built from circles covering the silhouette[Kang et al. 2004]. These histograms are then matched to the proposed silhouette using three different measures: cross-correlation, Bhattacharya distance and Kullback-Leibler divergence.
Finally, Sato and Agger wal compute flow vectors of pixels within a silhouette and perform Hough transforms on pixel/time blocks to obtain a Temporal Spatio Velocity image for each frame. Their method is less sensitive to appearance variations than other methods[Yilmaz et al. 2006].

1.8 Contour Evolution

As opposed to shape matching methods which attempt to find a silhouette in each frame, contour evolution methods evolve an object contour from frame to frame given an initial silhouette. The object's contour consists of the boundary between the silhouette region and everything else. Contours are often represented by a set of control points along the object boundary.
Yilmaz et al divide contour evolution methods into two categories: tracking using state space models and tracking by energy minimisation.

1.8.1 State Space Models

State space models work by assigning a state to the parameters of the contour (such as shape, motion and control point position).
Terzopoulos and Szeliski assign state according to the dynamics of the control points which are modelled by a spring model[Terzopoulos and Szeliski 1993]. Control point positions for the next frame are computed from this. The new position is then matched and corrected with a trained particle filter.
Chen et al. use Hidden Markov Models on elliptical control points to predict and evolve the contour state.[Chen et al. 2001b].

1.8.2 Energy Minimisation

Energy models ustilise an energy function on the contour and attempt to minimise this according to a set of parameters. Minimisation is usually performed through greedy methods or gradient descent[Yilmaz et al. 2006]. Contour energy is often defined by either temporal gradient or appearance statistics[Yilmaz et al. 2006].

1.9 Silhouette Representation

Silhouette are represented in different manners. They can be represented by binary indicator functions, which take the value one on the silhouette and zero elsewhere. Alternate representations are explicit and implicit surfaces. Explicit surfaces represent the silhouette by a set of control points along the object contour. Implicit surfaces represent the silhouette by a function define on the image.

2 Hand Pose Estimation

Pose estimation refers to the problem of detecting the three dimensional articulation of an object from a single image or sequence of frames. Hand pose estimation attempts to compute this articulation for the human hand.
There are two main approaches to this problem. The first, called model based tracking, attempt to model the kinematics and dynamics of a three dimensional model of the object. These techniques make use of temporal effects or time coherence and use data from previous frames to predict the new hand position.
The second type of method, called single frame pose estimation, attempts to compute hand pose independently in each frame. These techniques ignore temporal coherence. This can be partly justified due to how rapidly hand silhouettes can change during hand movement which makes temporal effects useless[Erol et al. 2007].

2.1 Common Difficulties

Hand pose estimation is, by the nature of the problem, a theoretically and computationally difficult task. There are many approaches to solving the problem with continued research into new techniques. There are alternative methods to pose estimation such as the use of sensor gloves, however here we will focus on vision based solutions.
Erol et al. have identified the following common difficulties encountered in the field.

- High Dimensional Problem: Due to hand articulation the problem has many degrees of freedom. Most studies use a hand model that has over 20 degrees of freedom[Erol et al. 2007].
- Occlusions: Due to the concave nature of the hand, occlusions are regular. This presents a significant challenge to pose estimation algorithms as many parameters become unbounded during occlusions.
- Processing speed: Most of the uses for hand pose estimation require that the hand pose is computed in real time.
- Uncontrolled environments: To be a widespread and useful tool in HCI, hand pose estimation algorithms will need to cope with a wide variety of different environments.
- Rapid hand motion: During movement the hand can reach velocities of 5m/s and rotation speeds of 300 degrees/s [Erol et al. 2007]. This presents a problem due to hardware restrictions (refresh rates of sensors) and tracking algorithms. This can also make it impossible to use temporal coherence.

2.2 Model based tracking

Model based tracking algorithms attempt to calculate hand pose by use of a constantly evolving model of the current hand articulation. Various constraints and types of models are used in the field. Figure 3 shows the different complexities of models used for pose estimation.

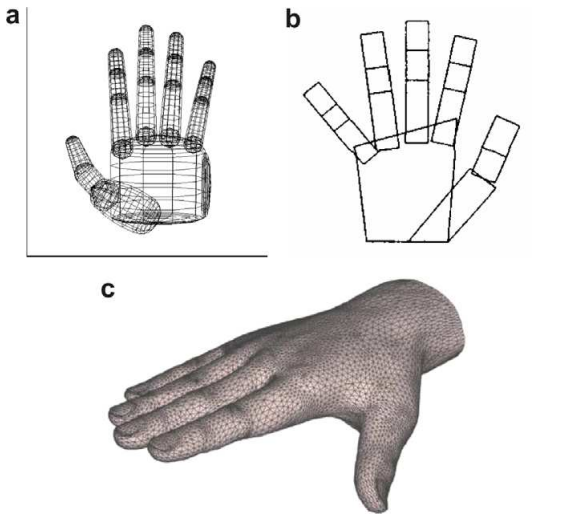


Figure 3: Hand models with varying complexity: (a) Quadrics based model from Stenger et al.(b) Cardboard model from Wu et al. (c) realistic model from Bray et al.

Model based tracking can be further divided into two categories: single and multiple hypothesis tracking. Single hypothesis tracking maintains a single hypothesis about hand articulation throughout the tracking. While this method is more simple it is susceptible to imperfections in the error function, background clutter, self-occlusion and complex motion[Erol et al. 2007].
Multiple Hypothesis methods keep multiple pose estimates at each frame. If the best estimate fails the system will continue tracking with another estimate[Erol et al. 2007].

2.2.1 Single Hypothesis

There are various approaches within single hypotheses pose estimation. Here we briefly discuss some of these methods.

- Optimisation methods: These methods make use of standard optimisation techniques. The optimisation criteria or fitness function is usually based on an error metric. Visual deviations from hypothesised pose and observed pose help guide the algorithm to a solution. Optimisation techniques which have been used include the Gauss-Newton method[Rehg and Kanade 1993], Genetic Algorithms and Simulated Annealing[Nirei et al. 1996], Stochastic gradient descent [Bray et al. 2004] and divide and conquer algorithms[Wu and Huang 1999].

- Physical Force Models: These methods compute a pose and then use the match error to derive some physical forces. These physical forces are then applied to the hand model and dynamics simulations produce a new hand articulation. These techniques have been used in various different implementations [Lu et al. 2003; Delamarre and Faugeras 2001].

2.2.2 Multiple Hypothesis

Multiple hypothesis methods have been shown to have better resilience to error than single hypothesis methods[Erol et al. 2007]. Here we briefly discuss the different approaches to multiple hypothesis post estimation.
Techniques include particle filters, tree based filters (where a tree of hand pose images is stored and traversed), Bayesian networks, template database searching (multiple hypotheses are stored and their neighborhoods are searched at each frame).

2.3 Single frame pose estimation

Single frame pose estimation is a more difficult task than Model based tracking since no previous information can be used. However, this method is attractive since one does not need complex data structures and algorithms to store previous hand pose estimates[Erol et al. 2007]. We will briefly describe some of the single frame pose estimation methods.

- Object detection: Large datasets of artificially generated hand silhouettes are created and stored in a tree data structure which corresponds to “clusters of similar hand poses” [Erol et al. 2007].

- Image databases: These techniques aim to develop algorithms to rapidly retrieve similar hand pose images from a database given an input image. If a match is found the corresponding pose is known.

- Direct mapping: These techniques train machine learning systems to directly map a pose silhouette from 2D to 3D.

- Inverse kinematics: Inverse kinematics aims to calculate joint angles given finger tip, joint and palm positions. Closed form solutions for joint angles have been developed for this purpose, although they operate under a constrained model[C.S. Chua 2000]. One restriction of these methods is that they often require markers on the fingers and palm[Erol et al. 2007].

3 Direct Manipulation and Tangible Interfaces

Direct manipulation systems allow users to interact with an interface in a natural manner. There has been increasing interest in tangible interfaces over the past few years. Many of these interfaces are designed as table interfaces.

3.1 Table Interfaces

Table interfaces are tangible interfaces where users can interact with on screen elements through direct manipulation. These systems usually consist of projector camera systems. An image is projected onto a surface. This image is also in view of a camera. The users are usually given devices which they can use to interact with the system. Examples of these include fiducial markers [Kaltenbrunner and Bencina 2007] and light projecting pens[Piazza and Fjeld 2007].

3.2 Difficulties

There are various difficulties involved in implementing a table based interface.

- Changes in illumination: Real environments often exhibit large changes in illumination. These include factors such as time of day and shadows created by users.

- Occlusion through interaction: While interacting with a table interface the users often occlude either the projector or camera system. To address this some system architectures use a translucent table top and place the camera below the surface.

- Projection distortion: The projector does not always face the table surface orthogonally. Furthermore some systems allow the users to tilt the projection surface[Song et al.].

- Dynamic or complex backgrounds: Changes in the background and background clutter can present significant problems to computer vision algorithms employed.

- Surface orientation: Some systems allow the table surface to be moved. The result of this is that the camera needs to correctly identify the position and orientation of this surface. Markers and fiducial are often used for surface detection and orientation.

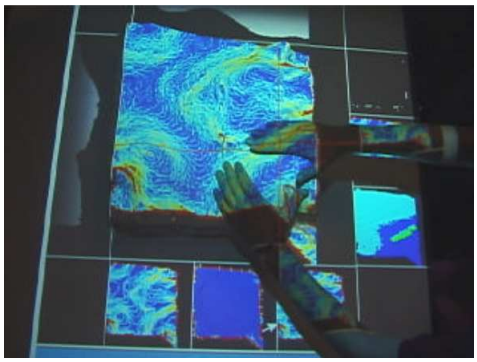


Figure 4: The landscape table interface of Piper et al. [Piper et al. 2002]

3.3 Conclusion

Tangible interfaces allow for novel methods of human computer interaction. Piper et al. have implemented a tangible interface for landscape analysis[Piper et al. 2002]. Their system makes use of a projector and laser scanner which scans a table surface as can be seen in figure 4. The landscape is represented by a square of white Plasticine onto which a colour map is projected. Users are able to directly sculpt the landscape after which their changes are detected by the laser scanner.
Clearly tangible interfaces offer promising results for natural interaction. There are various approaches to implementing tangible interfaces, each having its own weaknesses and merits.

References

BOULT, X. G., GAO, X., AND RAMESH, V. 2000. Error analysis of background adaption. 503–510.

BRAY, M., KOLLER-MEIER, E., MÜLLER, P., GOOL, L. V., AND SCHRAUDOLPH, N. N. 2004. http://nic.schraudolph.org/pubs/BraKolMueVanetal04.pdf 3D Hand Tracking by Rapid Stochastic Gradient Descent Using a Skinning Model. In *First European Conference on Visual Media Production (CVMP)*, 59–68.

CANNY, J. 1986. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell. 8*, 6, 679–698.

CHEN, Y., RUI, Y., AND HUANG, T. 2001. Jpdaf based hmm for real-time contour tracking. I:543–550.

CHEN, Y., RUI, Y., AND HUANG, T. 2001. Jpdaf based hmm for real-time contour tracking. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, I–543 – I–550 vol.1.

C.S. CHUA, H.Y. GUAN, Y. H. 2000. Model-based finger posture estimation. In *Fourth Asian Conference on Computer Vision*, 43–48.

DELAMARRE, Q., AND FAUGERAS, O. 2001. 3d articulated models and multiview tracking with physical forces. *Comput. Vis. Image Underst. 81*, 3, 328–357.

DINSTEIN, I., SHANMUGAM, K., AND HARALICK, R. 1977. Textural features for image classification. In *CMetImAly77*, 141–152.

ELGAMMAL, A. M., HARWOOD, D., AND DAVIS, L. S. 2000. Non-parametric model for background subtraction. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II*, Springer-Verlag, London, UK, 751–767.

EROL, A., BEBIS, G., NICOLESCU, M., BOYLE, R. D., AND TWOMBLY, X. 2007. Vision-based hand pose estimation: A review. *Comput. Vis. Image Underst. 108*, 1-2, 52–73.

GAVRILA, D. M., AND DAVIS, L. S. 1996. 3-d model-based tracking of humans in action: a multi-view approach. In *CVPR '96: Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, IEEE Computer Society, Washington, DC, USA, 73.

GREENSPAN, H., BELONGIE, S., PERONA, P., GOODMAN, R., RAKSHIT, S., AND ANDERSON, C. 1994. Overcomplete steerable pyramid filters and rotation invariance. 222–228.

HUTTENLOCHER, D., NOH, J., AND RUCKLIDGE, W. 1993. Tracking non-rigid objects in complex scenes. In *Computer Vision, 1993. Proceedings., Fourth International Conference on*, 93 –101.

KALTENBRUNNER, M., AND BENCINA, R. 2007. reactivision: a computer-vision framework for table-based tangible interaction. In *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, ACM, New York, NY, USA, 69–74.

KANG, J., COHEN, I., AND MEDIONI, G. 2004. Object reacquisition using invariant appearance model. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4*, IEEE Computer Society, Washington, DC, USA, 759–762.

LAWS, K. 1980. *Textured image segmentation*. PhD thesis, University of Southern California.

LU, S., METAXAS, D., AND SAMARAS, D. 2003. Using multiple cues for hand tracking and model refinement. In *In International Conference on Computer Vision and Pattern Recognition*, 443–450.

MALLAT, S. G. 1989. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell. 11*, 7, 674–693.

MONNET, A., MITTAL, A., PARAGIOS, N., AND RAMESH, V. 2003. Background modeling and subtraction of dynamic scenes. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, 1305.

NIREI, K., SAITO, H., MOCHIMARU, M., AND OZAWA, S., 1996. Human hand tracking from binocular image sequences.

PASCHOS, G. 2001. Perceptually uniform color spaces for color texture analysis: An empirical evaluation. 932–937.

PIAZZA, T., AND FJELD, M. 2007. Ortholumen: Using light for direct tabletop input. *Horizontal Interactive Human-Computer Systems, International Workshop on 0*, 193–196.

PIPER, B., RATTI, C., AND ISHII, H. 2002. Illuminating clay: a 3-d tangible interface for landscape analysis. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, New York, NY, USA, 355–362.

REHG, J. M., AND KANADE, T. 1993. Digiteyes: Vision-based human hand tracking. Tech. rep., Pittsburgh, PA, USA.

RITTSCHER, J., KATO, J., JOGA, S., AND BLAKE, A. 2000. A probabilistic background model for tracking. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II*, Springer-Verlag, London, UK, 336–350.

SHAFIQUE, K., AND SHAH, M. 2005. A noniterative greedy algorithm for multiframe point correspondence. *IEEE Trans. Pattern Anal. Mach. Intell. 27*, 1, 51–65.

SONG, P., WINKLER, S., AND TEDJOKUSUMO, J. A tangible game interface using projector-camera systems.

SONG, K., KITTLER, J., AND PETROU, M. 1996. Defect detection in random color textures. 667–683.

STAUFFER, C., AND GRIMSON, W. E. L. 2000. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell. 22*, 8, 747–757.

TERZOPOULOS, D., AND SZELISKI, R. 1993. Tracking with kalman snakes. 3–20.

WREN, C. R., AZARBAYEJANI, A., DARRELL, T., AND PENTLAND, A. P. 1997. Pfinder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Mach. Intell. 19*, 7, 780–785.

WU, Y., AND HUANG, T. 1999. Capturing articulated human hand motion: A divide-and-conquer approach. 606–611.

YILMAZ, A., JAVED, O., AND SHAH, M. 2006. Object tracking: A survey. *ACM Comput. Surv. 38*, 4, 13.

ZHONG, J., AND SCLAROFF, S. 2003. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, 44.