Which Shape Representation Is the Best for Real-Time Hand Interface System?

Serkan Genç[1] and Volkan Atalay[2]

[1] Computer Technology and Information Systems, Bilkent University, Ankara, Turkey
`sgenc@bilkent.edu.tr`
[2] Department of Computer Engineering, METU, Ankara, Turkey
`volkan@ceng.metu.edu.tr`

Abstract. Hand is a very convenient interface for immersive human-computer interaction. Users can give commands to a computer by hand signs (hand postures, hand shapes) or hand movements (hand gestures). Such a hand interface system can be realized by using cameras as input devices, and software for analyzing the images. In this hand interface system, commands are recognized by analyzing the hand shapes and its trajectories in the images. Therefore, success of the recognition of hand shape is vital and depends on the discriminative power of the hand shape representation. There are many shape representation techniques in the literature. However, none of them are working properly for all shapes. While a representation leads to a good result for a set of shapes, it may fail in another one. Therefore, our aim is to find the most appropriate shape representation technique for hand shapes to be used in hand interfaces. Our candidate representations are Fourier Descriptors, Hu Moment Invariant, Shape Descriptors and Orientation Histogram. Based on widely-used hand shapes for an interface, we compared the representations in terms of their discriminative power and speed.

Keywords: Shape representation, hand recognition, hand interface.

1 Introduction

Hands play very important role in inter-human communication and we use our hands for pointing, giving commands and expressing our feelings. Therefore, it is reasonable to mimic this interaction in human-computer interaction. In this way, we can make computer usage natural and easier. Although several electro-mechanical and magnetic sensing devices such as gloves are now available to use with hands in human computer interaction, they are expensive and uncomfortable to wear for long times, and require considerable setup process. Due to these disadvantages, vision based systems are proposed to provide immersive human computer interaction. Vision systems are basically composed of one or more cameras as input devices, and processing capabilities for captured images. Such a system is so natural that a user may not be aware of interacting with a computer system. However, there is no unique vision based hand interface system that can be used in all types of applications. There are several reasons for this. First, there is no computer vision algorithm which reconstructs a hand from an image. This is because a hand has a very complex model with 27 degrees of

freedom (DOF) [1]. Modeling the kinematic structure and dynamics are still open problems, and need further research [2]. Second, even if there was an algorithm which finds all 27 parameters of a hand, it would be very complex, and it may not be appropriate for real-time applications. Third, it is unnecessary to use complex algorithms for a simple hand interface, since it consumes considerable or even all available computing power of the system. Appearance-based techniques that analyze the image without using any 3-dimensional hand model work faster than 3-dimensional model based techniques and they are more appropriate for real time hand interface applications [2],[3].

This study mainly focuses on appearance-based methods for static hand posture systems. However, the shape representations presented in this study can be incorporated as feature vectors to standard spatio-temporal pattern matching methods such as Hidden Markov Models (HMM) [18] or Dynamic Time Warping (DTW) [22] to recognize dynamic hand movements or hand gestures.

Our initial motivation was to develop an application which was controlled by hand. In this application, the setting was composed of a camera located on top of the desk, and the user gave commands by hand. Although capturing from above limits possible hand shapes, this is very frequent setup for hand interface systems. For example, the system described by Quek *et al.* controls the behavior of a mouse by a hand on keyboard [4]. ITouch uses hand gestures appearing on the monitor similar to touch screen monitors [5]. Licsar and Sziranyi present another example that enables a user to manage presentation slides by hands [6]. Freeman *et al.* let the users play games by their hands [7]. Nevertheless, there is no study comparing techniques employed in such a setup. The aim of our paper is to assess various representations for hand shape recognition system having a setup where a camera is located above the desk and is looking downward to acquire the upper surface of the hand.

Usually, a hand interface system is composed of several stages: image acquisition, segmentation, representation, and recognition. Among those stages, segmentation is the main bottleneck in developing general usage HCI applications. Although there are many algorithms attempting to solve segmentation such as skin color modeling [24], Gauss Mixture Model for Background Subtraction [23] and Neural Network methods, all of them impose constraints on working environments such as illumination condition, stationary camera, static background, uniform background, etc. When the restriction is slightly violated, a clean segmentation is not possible, and the subsequent stages fail. Remedy to this problem is to use more complex representations or algorithms to compensate the deficiency of segmentation. However, there is also a limit on compensation. As a result, for the time being, even using the state-of-the-art segmentation algorithms for color images, a robust HCI application is not possible. However, there is a good news recently on segmentation with a new hardware technology, which is called Time-of-Flight (ToF) depth camera [19]. It captures depth information for each pixel in the scene and the basic principle is to measure the distances for each pixel using the round-trip delay of light, which is similar to radar systems. This camera is not affected by illumination changes at all. With this technology, clean segmentation for HCI applications is possible using depth keying technique [20]. Microsoft's Natal Project uses this technology to solve segmentation problem and enable users to interact with their body to control games [21]. Therefore, shape representations from clean segmentation can be used with this technology, and we used clean segmented

hand objects in our study. We believe that future HCI applications will be using ToF camera to solve segmentation problem.

The next stage after segmentation is representation, hand pixels are transformed into a meaningful representation which is useful at recognition stage. Representation is very important for recognition since unsuccessful representation gives unsatisfactory results even with state-of-the-art classifiers. On the other hand, good representation always results in an acceptable result with an average classifier [8].

This paper compares four representation techniques which can be used in shape recognition systems. In the selection of these representations, the following criterions are used: discriminative power, speed and invariance to scale, translation and rotation. Selected representations are Fourier descriptors, Hu moment invariants, shape descriptors, and orientation histogram. In Section 2, these selected representations are explained in detail. To assess the representations, bootstrapping is used to measure the quality of representations while decision tree is used as the classifier. Section 3 gives all the details concerning tests. In Section 4, we comment on the results in terms of discriminative power and real-time issues. Finally, we conclude the paper.

2   Shape Representations

Recognizing commands given by hand depends on the success of shape recognition, and thus, it is closely related to shape representation. Therefore, it is vital to select the appropriate shape representation for hand interfaces. Unfortunately, there is no unique representation that works for all sets of shapes. This is the motivation that leads us to compare and assess popular hand shape representations. Techniques for shape representation can be mainly categorized as contour-based and region based [9]. Contour based techniques use the boundary of the shape while region based techniques employ all the pixels belonging to a shape. Each category is divided into two subcategories: structural and global. Structural methods describe the shape as a combination of segments called primitives in a structural way such as a tree or graph. However, global methods consider the shape as a whole. Although there are many shape representation techniques in these categories, only some of them are eligible to be used in hand interfaces. We took into account certain criterions for selection. The first criterion is the computational complexity of finding the similarity of two shapes, i.e., matching. Contour based and region based structural methods such as polygon approximation, curve decomposition, convex hull decomposition, medial axis require graph matching algorithm for similarity, thus they are computationally complex [9]. Zhang *et al.* show that Fourier Descriptor (FD) which is a contour based global method performs better than Curvature Scale Space (CSS) which is also a contour based global method, in terms of matching and calculating representations [10]. Another contour-based global method, Wavelet Descriptor requires shift matching for similarity, which is costly compared to FD. Therefore, we have selected FD as a candidate. Freeman *et al.* use Orientation Histogram which is a region-based global method, for several applications controlled by hand [7],[11]. Since there is no comparison of Orientation Histogram with others, and authors promote it in terms of both speed and recognition performance, we have also chosen it. Peura *et al.* claim that practical application does not need too sophisticated methods, and they use the combination of simple shape descriptors for shape recognition [12]. Since Shape Descriptors are semantically simple,

fast and powerful, we have chosen Shape Descriptors (SD) as well. Each Shape Descriptor is either a contour or a region based global method. Flusser asserts that moment-invariants such as Hu Moment Invariants are important [13] since they are fast to compute, easy to implement and invariant to rotation, scale and translation. Therefore, Hu Moment Invariants are also selected for hand interface.

In conclusion, we have opted for four shape representation techniques; Shape Descriptors, Fourier Descriptors, Hu Moment Invariants and Orientation Histogram to assess their discrimination power and speed on a hand shape data set. In the rest of this section, we describe each selected method.

2.1 Shape Descriptors

Shape Descriptor is a quantity which describes a property of a shape. Area, perimeter, compactness, rectangularity are examples of shape descriptors. Although a single descriptor may not be powerful enough for discrimination, a set of them can be used for shape representation [12]. In this study, five shape descriptors; compactness, ratio of principal axes, elliptical ratio, convexity and rectangularity are chosen because they are invariant to scale, translation and rotation, and easy to compute. Also they are reported as successful descriptors in [9],[12].

2.1.1 Compactness
A common compactness measure, called the circularity ratio, is the ratio of the area of the shape to the area of a circle (the most compact shape) having the same perimeter. Assuming P is the perimeter and A is the area of a hand shape, circularity ratio is defined as follows.

$$\Psi = \frac{4\,\pi\,A}{P^2}$$

For a circle, circularity ratio is 1, for a square, it is $\frac{\pi}{4}$, and for an infinite long and narrow shape, it is 0.

2.1.2 Ratio of Principal Axes
Principal axes of a 2-dimensional object are two axes that cross each other orthogonally in the centroid of the object and the cross-correlation of boundary points on the object in this coordinate system is zero [12]. Ratio of principal axes, ρ provides the information about the elongation of a shape. For a hand shape boundary B which is an ordered list of boundary points, ρ can be determined by calculating covariance matrix $\sum$, of a boundary B, and then finding the ratio of $\sum$'s eigenvalues; λ_1 and λ_2. Eigenvectors e_1, e_2 of $\sum$ are orthogonal and cross-correlation of points in B with e_1 and e_2 is zero since $\sum$ is a diagonal matrix. The values of λ equal to the length of the principal axes. However, to find the ratio of λ_1 and λ_2 or principal axes, there is no need to explicitly compute eigenvalues, and ρ can be calculated as follows [12]:

$$\sum = \begin{vmatrix} \sum_{xx} & \sum_{xy} \\ \sum_{yx} & \sum_{yy} \end{vmatrix}, \text{ where } \sum_{ij} \text{ represents covariance of } i \text{ and } j.$$

$$\rho = \frac{\Sigma_{yy} + \Sigma_{xx} - \sqrt{\left(\Sigma_{yy} + \Sigma_{xx}\right)^2 - 4\left(\Sigma_{yy}\Sigma_{xx} - \Sigma_{xy}^2\right)}}{\Sigma_{yy} + \Sigma_{xx} + \sqrt{\left(\Sigma_{yy} + \Sigma_{xx}\right)^2 - 4\left(\Sigma_{yy}\Sigma_{xx} - \Sigma_{xy}^2\right)}}$$

2.1.3 Elliptical Ratio

Elliptical Ratio, ε is the ratio of minor axis, b to major axis, a of an ellipse which is fitted to boundary points.

$$\varepsilon = \frac{b}{a}$$

Ellipse fitting is performed using a least-square fitness function. In the implementation, ellipse fitting algorithm proposed by Fitzgibbon *et al.* [14] and provided by OpenCV is used [17].

2.1.4 Convexity

Convexity is the ratio of perimeter of the convex-hull, $\pi_{convexHull}$ to the perimeter of the shape boundary, π_{Shape}, where convex hull is the minimum convex polygon covering the shape.

$$\varsigma = \frac{\pi_{convexHull}}{\pi_{Shape}}$$

2.1.5 Rectangularity

Rectangularity measures the similarity of a shape to a rectangle. This can be calculated by the ratio of the area of the hand shape, $A_{handShape}$ to the minimum bounding box of hand shape, A_{MBB}. Minimum bounding box (MBB) is the smallest rectangle covering the shape.

$$\Gamma = \frac{A_{handShape}}{A_{MBB}}$$

For a rectangle, rectangularity is 1; for a circle, it is $\frac{\pi}{4}$.

2.2 Fourier Descriptors

Fourier Descriptors (FDs) represent the spectral properties of a shape boundary. Low frequency components of FDs correspond to overall shape properties; while high frequency components describe the fine details of the shape.

FDs are calculated using Fourier Transform of shape boundary points, (x_k,y_k), $k=0,..., N-1$ where N is the number of points in the boundary. Boundary can be represented by an ordered list of complex coordinates called complex coordinate signature, as $p_k=x_k+i\ y_k$, $k=0,...,N-1$ or a boundary can be represented by an ordered list of distances, r_k, of each boundary point (x_k,y_k) to centroid of the shape (x_c,y_c) called centroid distance signature.

Zhang and Lu compared the effect of four 1-dimensional boundary signatures for FDs; these signatures are complex coordinates, centroid distances, curvature signature and cumulative angular function [15]. The authors concluded that FDs derived from

centroid distance signature is significantly better than the others. Therefore, we use centroid signature of the boundary. To calculate FDs of a boundary, the following steps are pursued.

1. Calculate the centroid of the hand shape boundary

$$x_c = \frac{1}{N}\sum_{k=0}^{N-1} x_k \ \ and \ y_c = \frac{1}{N}\sum_{k=0}^{N-1} y_k$$

2. Convert each boundary point (x_k, y_k) to centroid distance r_k,

$$r_k = [(x_k - x_c)^2 + (y_k - y_c)^2], k = 0, \dots, N-1$$

3. Use Fourier Transform to obtain FDs.

$$FD_f = \frac{1}{N}\sum_{k=0}^{N-1} r_k \cdot e^{-j2\pi fk/N}$$

FD_f, $f=0,\dots,N-1$ are Fourier coefficients.

4. Calculated FDs are translation invariant since centroid distance is relative to the centroid. In Fourier Transform, rotation in spatial domain means phase-shift in frequency domain so using magnitude values of coefficients make FDs rotationally invariant. Scale invariance is achieved by dividing FDs by FD_0. Since each r_k is real valued, first half of FDs are the same with second half. Therefore, half of the FDs are enough to represent shape. As a result, a hand shape boundary is represented as follows [15]:

$$FD = \left[\frac{|FD_1|}{|FD_0|}, \frac{|FD_2|}{|FD_0|}, \dots, \frac{|FD_{N/2}|}{|FD_0|} \right]$$

2.3 Hu Moment Invariants, Φ

Hu derived 7 moments which are invariant to translation, rotation and scaling [13],[16]. This is why Hu moments are so popular and many applications use them in shape recognition systems. Each moment shows a statistical property of the shape. Hu Moment Invariants can be calculated as follows. Note that μ shows the 2nd and 3rd order central and normalized moments.

$\phi_1 = \mu_{20} + \mu_{02}$

$\phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2$

$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2$

$\phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2$

$\phi_5 = (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})((\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2) + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})(3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2)$

$\phi_6 = (\mu_{20} - \mu_{02})((\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2) + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03})$

$\phi_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})((\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2) - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})(3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2)$

The main problem of Hu moments in classification are the large numerical variances in the values of moment invariants. Therefore, the use of Euclidian distance to

compute similarity is not possible. In our implementation, decision tree is used as the classifier.

2.4   Orientation Histogram

Orientation Histogram (OH) is a histogram of local orientations of pixels in the image [11]. Freeman *et al.* applied the idea of OH to create fast and simple hand interfaces [7]. The basic idea of OH is that hand pixels may vary in illumination, and pixel-by-pixel difference leads to huge error in total. Instead of using pixels themselves for comparison, their orientations are used to overcome the illumination problem. To make it translation invariant, orientations are collected in a histogram with 36 bins where each bin represents 10 degree. Scale and rotation invariance are not pointed in [11]. However, our implementation normalized the magnitude of the histogram to overcome scaling problem, and updated by shifting all orientations relative to peak one to make it rotationally invariant. Instead of using the whole image, its dimension is reduced to about 100 by 80 pixels for faster computation [11]. The problems of the method are also reported as two similar shapes can produce very different histograms, and hand shape must not be a small part of the image. Thinking each normalized histogram as a vector in 36 dimensional space, we classified them using a decision tree algorithm similar to other three methods.

3   Experimental Results

To evaluate the performance of 4 shape representations, we collected 10160 samples of widely-used 15 hand shapes from 5 different people. There is approximately the same number of samples for each person and hand commands. A sample set of collected 15 hand shapes are shown in Fig. 1. The evaluation is based mainly on discrimination power and speed. Furthermore, we have also investigated the performance with respect to the number of people and samples in the training set.

Fig. 1. Hand commands used in our experiments

We have first divided the samples into two sets: training set and test set. Training set is used to train a decision tree for each representation, and samples in the test set are classified by the corresponding decision tree. Hit ratio, which is the percentage of correctly classified samples in the test set, is used as the measurement for discrimination power of each representation. The division of training and test set is based on two

parameters: number of people in the training set, and the percentage of the samples of each person selected for training.

We first assessed the effect of training set size when the system is trained with samples only from one person. We selected randomly one person among 5 people, and used 10% of samples from each hand command of the selected person in training. All the remaining samples, that is, remaining 90% samples of the same person and all samples from other 4 people, were employed in test set. This test is repeated 5 times and the average of the hit ratios is used as the measurement of discrimination power for each representation. This procedure is repeated with 30% and 50% of samples in training for one person, and Fig. 2.a shows the results graphically. We repeated the above procedure for 2, 3, 4, and 5 people, and only the results for 5 people is depicted in Fig. 2.b. Results for all number of people with all number of training set size can be found in Table 1.

Table 1. Hit ratios for all parameters used in the experiment

Parameters		Hit Ratios (%)			
Num. of people	Training Size (%)	SD	FD	HU	OH
1	10	55.63	65.08	62.1	38.15
	30	64.99	77.5	70.97	45.87
	50	62.2	76.45	70.16	47.24
2	10	75.47	86.09	85.19	49.53
	30	75.54	88.11	86.48	55.82
	50	75.97	87.9	85.6	57.12
3	10	81.05	89.29	89.81	53.23
	30	83.44	92.98	91.8	63.33
	50	82.73	93	91.1	65.48
4	10	83.89	93.56	93.86	58.56
	30	87.05	94.12	94.82	67.38
	50	87.07	94.05	94.27	69.65
5	10	87.29	96.84	95.73	63.45
	30	91.18	98.64	98.13	70.17
	50	92.35	99.61	98.58	75.43

Fig. 2.a and Fig. 2.b show the hit ratios of representations when 10%, 30% and 50% of samples from the people used in training. It is observed that the performances of SD, FD and HU are not influenced considerably by increasing the number of samples from the same person. As a result, SD, FD and HU representations produce low variances for the representations of similar hand shapes from the same person. This is a desired property since a few training samples from a person are adequate to train the system for that person.

(a)   (b)

(c)
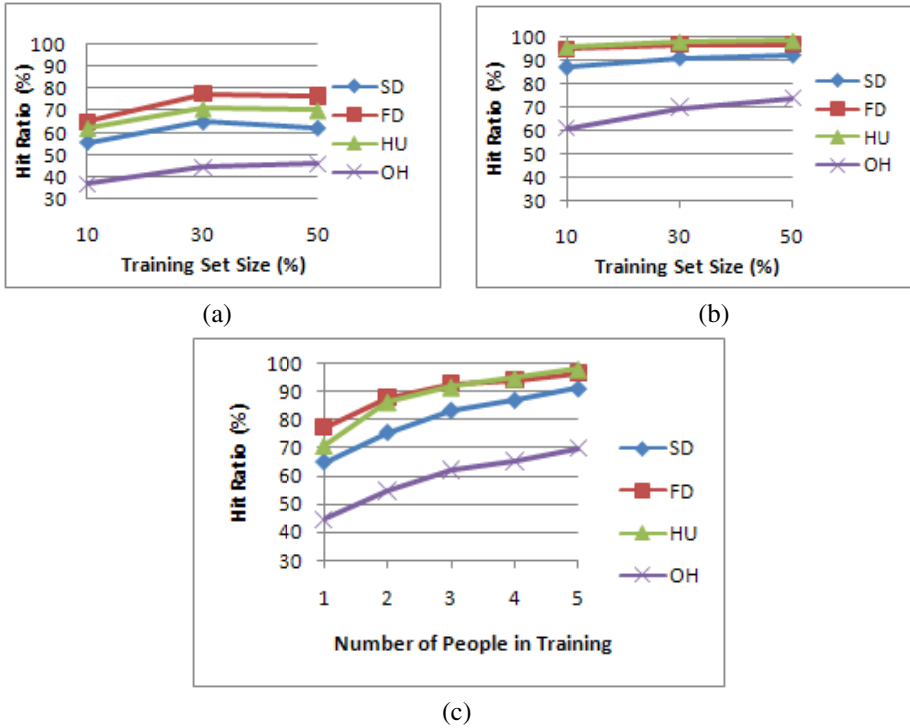
Fig. 2. Hit ratios for (a) randomly selected one person for training, (b) all 5 people used in training, (c) 30% of samples of selected number of people for training

Fig. 2.c indicates the influence of number of people in training on hit ratio. It is apparently observed that hit ratios of all representations improve drastically when the number of people participating in training increases. Thus, it is reasonable to use many people to train a hand interface system which uses one of four shape representations described in this paper.

Our aim is to find out the best of four representations in terms of discrimination power and speed. According to Fig. 2, FD and HU outperform SD and OH, and the results of FD and HU are very close to each other in terms of hit ratio. FD is slightly better than HU.

Real-time is an indispensable requirement for a hand interface system. Noticeable delay prevents user from immersive usage of the system. Therefore, we analyzed the running time performance of representations. To measure running time performances of each representation, we computed the total elapsed time of each representation for all samples. Fig. 3 shows the results of total elapsed time of 10160 samples for each representation. According to the results, HU is the fastest method among these four representations. Remark that FD is also a relatively fast method. On the average, calculation of a HU or FD representation of a segmented image is less than a millisecond (dividing total application time by total samples according to Fig. 3) with a Pentium IV – 3GHz computer with 1GB RAM of memory.
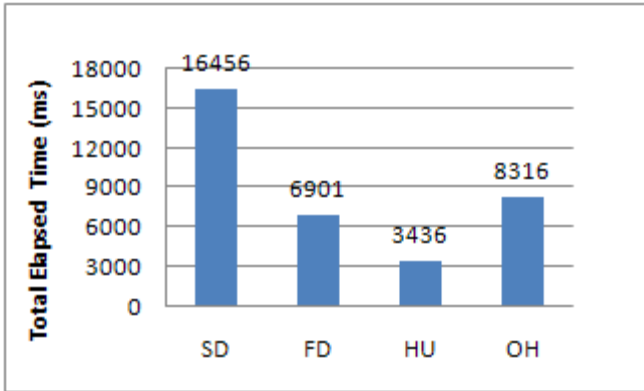
Fig. 3. Running times of 4 shape representations for 10160 samples

In conclusion, both HU and FD provide acceptable results in terms of discrimination power and speed, and they can be used for computer systems which employ static hand shapes as commands.

4   Conclusion

The main component of a hand interface is the part that recognizes the given hand command. Shape is an important property of a hand, and it can be used for representation of a hand. In this paper, four representation techniques; Shape Descriptors (SD), Fourier Descriptors(FD), Hu Moment Invariants (HU) and Orientation Histogram (OH) are selected, and compared in terms of their discrimination power and speed. When forming the test environment, a widely-used hand interface setup is chosen. In this setup, a camera is located above the desk in such a way that its viewing direction is to the top of the desk. In the experiments, there are totally 10160 samples of 15 different hand commands from 5 people. Those samples are divided into training set and test set. Each sample in training set is converted to four representations, and each representation is used to train a decision tree classifier for recognition. Furthermore, running times of calculating the representations are accumulated and used as the measurement of speed for each representation. According to test results, HU and FD outperform SD and OH in terms of both discrimination power and speed. Therefore, HU and FD are reasonable to use in hand shape recognition systems as posture recognizer or as a feature vector to a spatio-temporal pattern recognizers such as HMM.

References

1. La Viola, J.J.: A Survey of Hand Posture and Gesture Recognition Techniques and Technology. Technical Report CS-99-11, Department of Computer Science, Brown University (1999)
2. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X.: A Review on Vision Based Full DOF Hand Motion Estimation. In: IEEE Workshop on Vision for Human-Computer Interaction (in conjunction with CVPR 2005), San Diego, CA, June 21 (2005)

3. Wu, Y., Huang, T.S.: Hand Modeling, Analysis and Recognition. IEEE Signal Processing Magazine, 51–58 (May 2001)
4. Quek, F.K.H., Mysliwiec, Y., Zhao, M.: FingerMouse: a Freehand Pointing Interface. In: Proc. of Int. Workshop on Automatic Face and Gesture Recognition, pp. 372–377 (1995)
5. Genc, S., Atalay, V.: ITouch: Vision-based Intelligent Touch Screen in a Distributed Environment. In: Int. Conf. on Multimodal Interfaces (ICMI), Doctoral Spotlight (October 2005)
6. Licsar, A., Sziranyi, T.: Dynamic Training of Hand Gesture Recognition System. In: Proc. Intl. Conf. on Pattern Recognition, ICPR 2004, vol. 4, pp. 971–974 (2004)
7. Freeman, W.T., Anderson, D., Beardsley, P., Dodge, C., Kage, H., Kyuma, K., Miyake, Y., Roth, M., Tanaka, K., Weissman, C., Yerazunis, W.: Computer Vision for Interactive Computer Graphics. IEEE Computer Graphics and Applications 18(3), 42–53 (1998)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley-Interscience, Hoboken (2000)
9. Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern Recognition 37(1), 1–19 (2004)
10. Zhang, D., Lu, G.: Content-Based Shape Retrieval Using Different Shape Descriptors: A Comparative Study. In: Proc. of IEEE Conference on Multimedia and Expo., Tokyo, August 2001, pp. 317–320 (2001)
11. Freeman, W.T., Roth, M.: Orientation Histograms for Hand Gesture Recognition. In: Intl. Workshop on Automatic Face and Gesture Recognition (1995)
12. Peura, M., Livarinen, J.: Efficiency of Simple Shape Descriptors. In: Aspects of Visual Form, pp. 443–451. World Scientific, Singapore (1997)
13. Flusser, J.: Moment Invariants in Image Analysis. In: Proceedings of World Academy of Science, Engineering and Technology, February 2006, vol. 11 (2006) ISSN 1307-6884
14. Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct Least Square Fitting of Ellipses. Pattern Analysis and Machine Intelligence 21(5) (May 1999)
15. Zhang, D., Lu, G.: A Comparative Study on Shape Retrieval Using Fourier Descriptors with Different Shape Signatures. In: Proc. of International Conference on Intelligent Multimedia and Distance Education, Fargo, ND, USA, pp. 1–9 (2001)
16. Hu, M.K.: Visual Pattern Recognition by Moment Invariants. IRE Transactions on Information Theory IT-8, 179–187 (1962)
17. Gary, K.A.: Learning OpenCV. O'Reilly, Sebastopol (2008) (first print)
18. Chen, F.S., Fu, C.M., Huang, C.L.: Hand Gesture Recognition Using a Real-time Tracking Method and Hidden Markov Models. Image and Vision Computing 21, 745–758 (2003)
19. Kolb, A., Barth, E., Koch, R.: ToF-sensors: New dimensions for realism and interactivity. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Workshop on ToF Camera based Computer Vision (TOF-CV), pp. 1–6 (2008), doi:10.1109/CVPRW
20. Gvili, R., Kaplan, A., Ofek, E., Yahav, G.: Depth Keying, http://www.3dvsystems.com/technology/DepthKey.pdf
21. Microsoft Natal Project: http://www.xbox.com/en-US/live/projectnatal
22. Nakagawa, S., Nakanishi, H.: Speaker-Independent English Consonant and Japanese word recognition by a Stochastic Dynamic Time Warping Method. Journal of Institution of Electronics and Telecommunication Engineers 34(1), 87–95 (1988)
23. Stauffer, C., Grimson, W.E.L.: Adaptive Background Mixture Models for Real-Time Tracking. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), pp. 246–252 (1999)
24. Jones, M.J., Rehg, J.M.: Statistical Color Models with Application to Skin Detection. Int. J. of Computer Vision 46(1), 81–96 (2002)