

Chapter 4

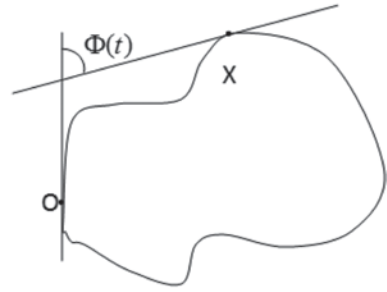
Feature Extraction

An image is worth 1,000 words. Yet, a machine to describe a picture or a color image is not trivial. Of course, some measurements can easily be estimated such as different colors, their intensities, size and dimensions of certain objects if the object can be specified. Yet, the most difficult aspect is to make the decisions as to what constitute an object. In a scene consisting of hand gesture or gestures and a cluttered background, difficulty lies in interpreting these items. Perhaps, the hand gesture recognition offers some help compared to other problems as skin detection can be used to define a hand as was discussed under Pre-processing in Chap. 3. Yet, even when a hand is detected and isolated, what configuration the hand shows is again a difficult question to address.

Feature extraction attempts to extract certain measureable inputs that can be used to classify a section of a signal. If the isolated section of an image contains what the humans interpret as a hand sign with a ‘thumbs up’ gesture, then it is important to extract information that would make this gesture unique compared to other possible gestures. The success of any classification relies on the ability to develop unique and robust features. As would be detailed in Chap. 6 on Sign Languages, even the same user would not be able to precisely perform the same gesture again. That is to say any gesture has certain variability and the certain degree of uniqueness among other gestures. Humans have evolved in a more subtle way to remember and understand this variability and uniqueness. To develop machine capabilities to interpret this information from an image is not trivial. Therefore a robust feature or set of features should uniquely describe the gesture in order to achieve reliable recognition. In other words, different gestures should result in different good discriminable features. Furthermore, shift and rotation invariant features lead to a better recognition of hand gestures even if the hand gesture is captured from a different angle.

This chapter contains few sections on different approaches to extract features that would make successful classification avoiding false positives. It would contain orientation histogram based feature extraction, the highly successful moment Invariant feature extraction; Principal Component Analysis based feature extraction, other feature extraction methodologies based on color and few other feature extraction strategies that results in successful gesture classification.

Fig. 4.1 Description of a point X with respect to origin O using Fourier descriptor



Before the discussion starts on successful features for better classification, it would be versatile to describe the attributes of a good feature. In the context of hand gesture recognition, good features are:

1. Compact set of data representing a unique gesture
2. Sufficient separation of feature clusters. Variety of distance measures such as Euclidean, Mahalanobis, etc. can be used to measure the distance between one gesture cluster and the other gesture clusters. The inter-cluster difference should be sufficient so that statistical variation of same gesture by different users at different times should not confuse the gesture classification.
3. The features should cluster well for different users with different hand sizes and different skin colors and gesture orientations (the features should be invariant)
4. The features obtainable in realtime

4.1 Fourier Descriptors (FD)

Fourier descriptors have been the first features used to describe shapes in image processing and computer vision [1–7]. They have been used for fingerprint recognition as way back as in 1972 due to its simplicity in describing contours which are invariant to scale, shift and rotation [2]. Due to these attributes, they are equally suitable for describing hand gestures.

Figure 4.1 outlines a closed contour that can be described effectively by Fourier Descriptor. To describe point X on the curve as shown above using the arc length s from the origin O , a relationship has to be established using the angle that is formed when two tangents from O and X meet as shown above. Then this point is uniquely described by the angular variation $\Phi(t)$ such that:

$$\Phi(t) = \Phi(t) - t \text{ where } t = 2\pi s/L.$$

In order to introduce the property of scale invariance, the length of the arc is normalized such that entire contour spans an angle of 2π . This function is real, continuous, and periodical with a period 2π and hence can be described by a Fourier series:

$$\Phi(t) = \sum_{k=0}^{\infty} a_k \exp(-jkt).$$

The set of modules of the coefficients a_k is called Fourier descriptors which can be used to describe various shapes such as leaves, finger prints and human hand postures.

Researchers have used extensions to basic Fourier descriptors to analyse shapes with increasing complexity. Lin and Hwang [8] showed that an alternative representation of the Fourier series is possible using elliptic Fourier features. In their approach, a shape was interpreted as a specific composition of feature ellipses having fixed axis lengths and fixed relative positions and orientations. It was shown that a shape can be represented by a set of ellipses which were rotation and translation invariant. Each ellipse also contained invariant major and minor axis lengths and each pair of ellipses had a specific position and orientation. Lin and Jungthirapanich [9] further developed the 2D elliptic Fourier descriptor to a 3D descriptor. Harding and Ellis developed the concept further with to show that using the FD on a set of trajectory data, it would be possible to recognize a range of pointing gestures that is invariant to natural variations due to the single individual or a ‘normal’ population. The 2D spatial data of a sequence of hand centroids was obtained using a single camera, but had the potential to be extended to 3D spatial data.

4.1.1 Elliptic Fourier Descriptors

As shown in Fig. 4.1, a point on a contour can be described by a coordinate pair which can be represented by a complex number $z(k) = x(k) + jy(k)$, so that the discrete Fourier transform of $z(k)$ is [10]:

$$a(u) = \frac{1}{N} \sum_{k=0}^{N-1} z(k) e^{\frac{-j2\pi uk}{N}} \quad (1)$$

For $u=0, 1, \dots, N-1$.

The obtained $a(u)$ coefficients describe the contour. In order to attain translation invariance of this feature, the DC component of the Fourier series given by $a(0)$ removed from the sequence and the rest of the components are scaled by $a(1)$ so that the feature incorporates scale invariance [10]. The origin of the sequence is encoded into the phase of $a(u)$. The consequence of origin selection is illustrated in Fig. 4.2 as it would change the orientation of the contour. An ellipse can be modeled as a positive and negative sequence of differing amplitudes. If the phase shift affecting both sequences is θ the orientation angle, then the sequence is:

$$A_{pos} e^{+j(\varphi+\theta)} \text{ and } A_{neg} e^{-j(\varphi+\theta)}$$

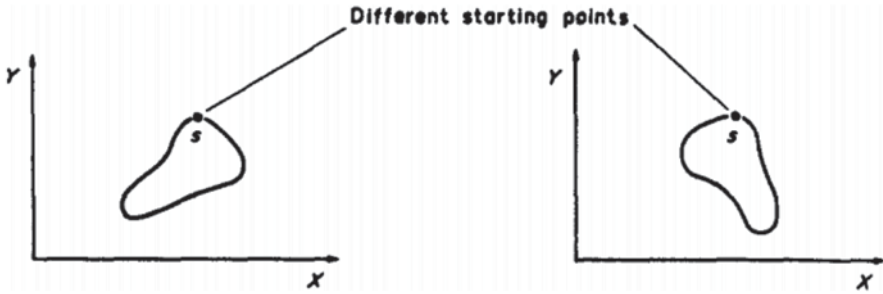


Fig. 4.2 Different starting points due to different orientations. (Courtesy of [8])

The orientation angle, θ can be found by taking the average of the positive and negative sequence phase. The direction and shape of the ellipse depends upon the magnitude of A_{pos} or A_{neg} . The relative size of A_{pos} and A_{neg} affects direction of revolution of the ellipses.

The representation of closed contours based on elliptical basis functions is described in detail by Lin and Hwang [8]. They mathematically demonstrated that a closed contour can be described by its Fourier descriptor feature matrices. A shape can be viewed geometrically as the locus generated by properly moving the feature ellipses.

Harding and Ellis [10], developed hand tracking method based on the work of Lin and Hwang. As shown in Eq. 1, the complex frequency domain data generated by the Fourier descriptor technique is generated by a discrete Fourier Transform algorithm. The number of harmonics generated was equal to the number of samples, N . The sample lengths were all normalized to the same length (64) by a multirate process. They used sixty four samples to encode a typical gesture that was completed within two seconds, at a sample rate of 30 frames per second, and additionally aided the speed of FFT implementation. Figure 4.3 shows ‘elliptic corkscrews’ and the overall trajectory for a gesture- *To Left Should and Return*.

Conseil et al. [11] developed a Fourier descriptor based method to represent hand gestures in an attempt to compare the performance accuracy of Fourier descriptor to Hu Moment based (this is discussed in Sect. 4.3.1) approaches. They used Triesch hand posture database and defined their own gesture vocabulary, with 11 gestures, and performed the acquisition of a large number of images, with 18 persons, and approximately 1,000 images per gesture per person [12].

They claimed that the tests were performed on a more realistic database, with various hand configurations realized by non-expert users. The learning was done with manually selected images of an expert user, with nearly 500 images per gesture. In the tests, they used 6 Fourier descriptors and initially validated the learning stage by running classification on the learning images, and obtained recognition rates of 98.11% for Hu moments and 99.96% for Fourier descriptors. Then images of the other users were classified using this learning data, with approximately 1,000 images per gesture for each user. They obtained a total of 86.22% for Fourier

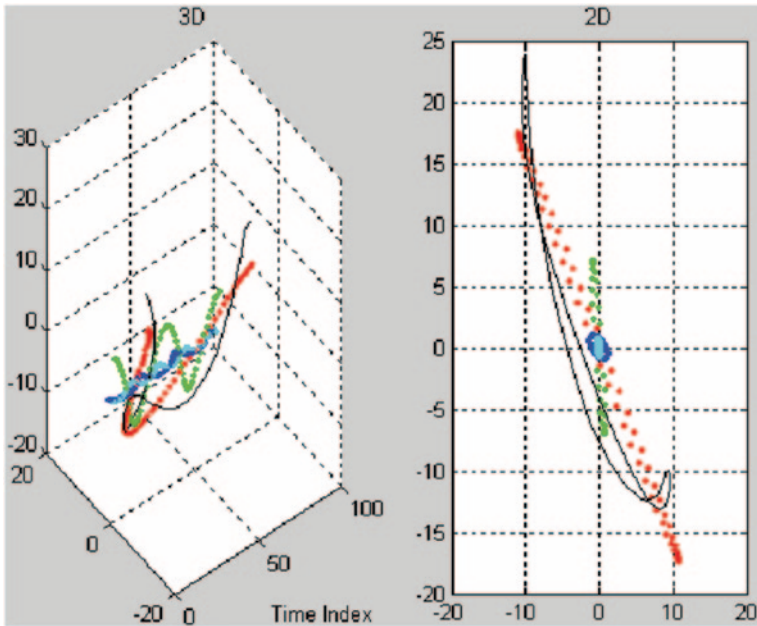


Fig. 4.3 *Left: 3D, Right: 2D*, view of the first 4 ‘elliptic corkscrews [8]’, ‘.’ and overall trajectory (‘-’) of gesture ‘To left shoulder and return’. (Courtesy of [10])

descriptors versus 71.08% for Hu moments. They also observed that FD outperformed Hu moments in terms of discrimination between visually close gestures. Figure 4.4 shows that the low frequency coefficients contain information on the general form of the shape and the high frequency coefficients contain information on the finer details of the shape.

One of the earliest works of hand gesture recognition using gesture feature extraction was attempted by Utsumi et al. in 1995 [18]. They proposed very simple feature extraction method that relied on centre of gravity of the hand and the finger locations based on Fourier descriptors. However, they used multiple cameras and tracked 3D position, posture, and shapes of human hands from multiple viewpoint images. This reduced self-occlusion and hand-hand occlusion by employing multiple-viewpoint and viewpoint selection mechanism. Each hand position was tracked with a Kalman filter and the motion vectors were updated with image features in selected images that did not include hand-hand occlusion. In their approach, 3D hand postures were estimated with a small number of reliable image features using COG and fingertip positions. These features were extracted based on distance transformation, and were found to be robust against changes in hand shape and self-occlusion. Finally, a “best view” image was selected for each hand for shape recognition. The shape recognition process was based on Fourier descriptors. The outline of their approach is depicted in Fig. 4.5.

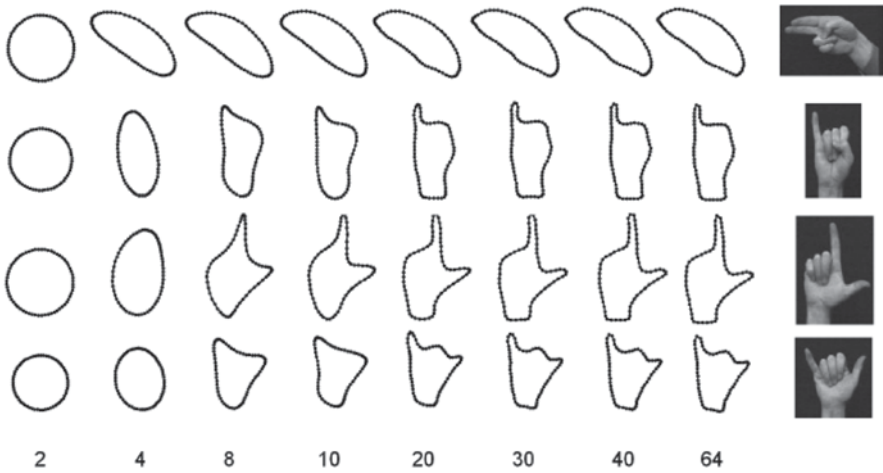


Fig. 4.4 Example of reconstruction with FD, as a function of the cut-off frequency, with an initial contour sampled at 64 points. (Courtesy of [12])

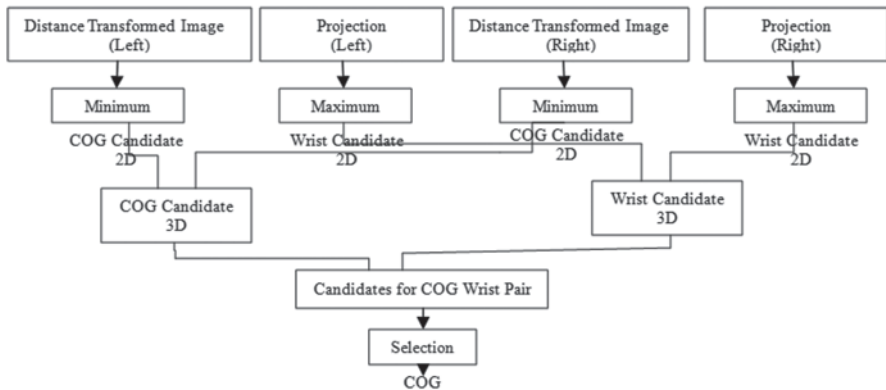


Fig. 4.5 COG Detection. (Courtesy of [13])

4.1.2 Modified Fourier Descriptors

Licsár and Szirányi applied a boundary-based Fourier descriptors for feature extraction based on widely used for shape description method used for content-based image retrieval systems [14, 15]. The extracted features were classified using neural networks classification algorithms [16, 17] resulting in about 91 % recognition rate for 6 gestures. In their method, the gesture contours were classified by the nearest neighbor rule and the distance metric based on the Modified Fourier Descriptors (MFD) [15]. This metric is invariant to the rotation, translation, reflection and scaling of shapes. The strategy requires that the examined shape should be defined by

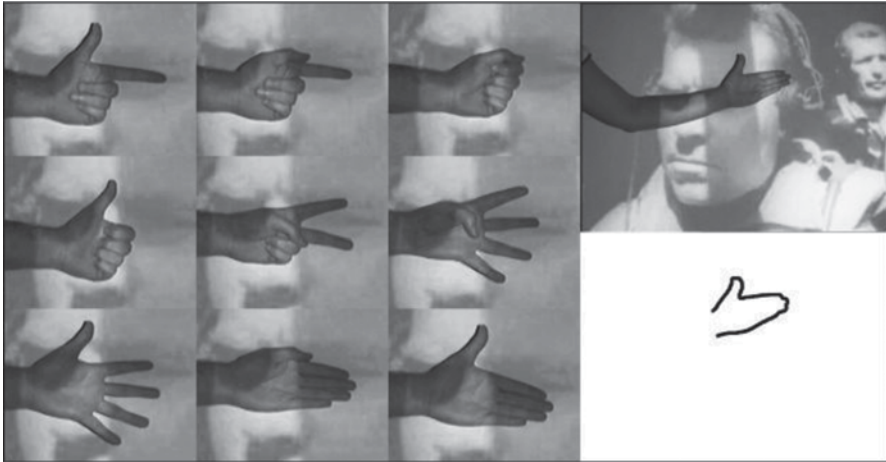


Fig. 4.6 Gesture vocabulary and segmentation result. (Courtesy of [14])

a feature vector, which is periodic, to expand it into Fourier series. The approach generated a feature sequence between the two wrist points, as shown in Fig. 4.6, along the shape boundary leading to more unambiguous features. This is due to the fact that the shape contours of the palm when showing only the index or the thumb finger is very similar to each other, while the contour between wrist points are distinctively different. The defined boundary sequence was constructed as a complex sequence of the x and y coordinates of the boundary points. These boundary points were then used to calculate the discrete Fourier transform (DFT) of this complex sequence. They further used the magnitude values of the DFT coefficients to retain invariance to rotation and extended the MFD method to obtain symmetric distance computations. They reported that when the trainer and the user were the same, recognition rates were above 97% while different users resulted in an accuracy around 86%.

4.2 Contour Description using 1D Sequence

Fourier descriptors always had a strong appeal as an excellent descriptor of the shape boundary or contour with invariance for translation, scale, rotation and reflection or mirror image offered by MFD techniques. One of the drawbacks in the Fourier descriptor is that the non-smooth contours result in very poor description of the shape resulting in classification error. Even though many researchers willingly state this in their research, this is indeed the reason why many others deviated from the very promising Fourier descriptors. Malima et al. in 2006 reported a new development inspired by Fourier descriptors to recognize hand gestures [18]. Their approach had limited focus and was not intended to develop a highly accurate

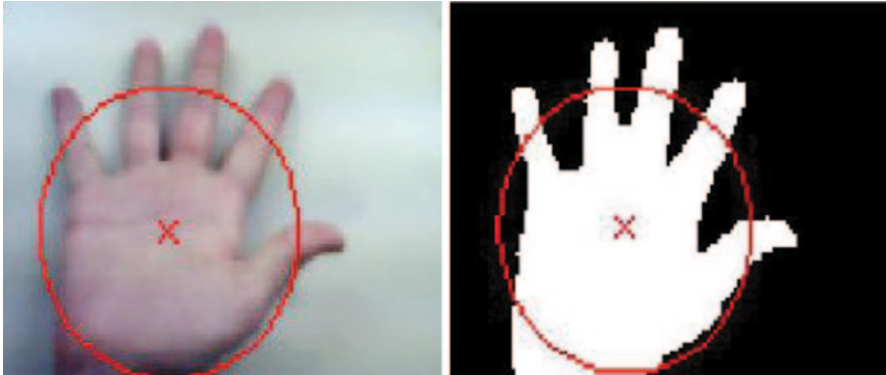


Fig. 4.7 Original image (*left*) with circle overlapped and the skin segmented binary image (*right*) with the circle with COG as the center. (Courtesy of [18])

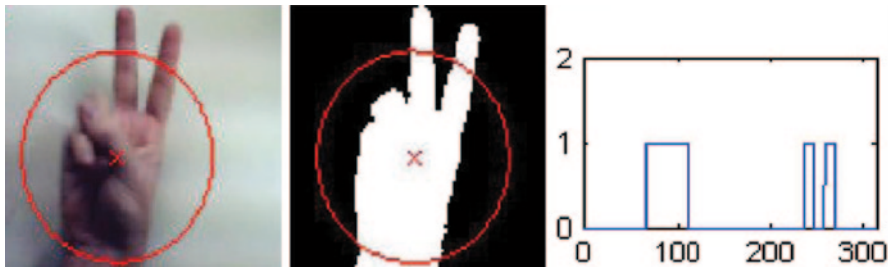


Fig. 4.8 Circle overlapping the hand (*left*), binary image (*middle*) and the zero-to-one transitions [18]

system as their gesture recognition was used to control a robot arm. Nevertheless, the approach had many positive developments.

As shown in Fig. 4.7, the initial step in extracting features was to select the region of importance. This is achieved by drawing a circle whose radius is 0.7 of the farthest distance from the Centre of Gravity (COG). Such a circle is likely to intersect all the fingers active in a particular gesture as demonstrated in Fig. 4.7. Once the skin segmentation is performed and the image is binarized, the 1D signal or the feature vector that describes the gesture is obtained by tracking the circle constructed in the previous step. As conceivable, the uninterrupted ‘white’ portions of this signal correspond to the fingers or the wrist. The total transitions of zeros to one can be counted to indicate the signal. By subtracting one from this number removes the transition due to the wrist. Estimating the number of fingers leads to the recognition of the gesture. This process is shown in Fig. 4.8.

This algorithm simply counts the number of active fingers without any regard to which particular fingers are active. Different combination of active fingers may result in the same configuration. A user may potentially use any finger combination for ‘on’ or ‘off’ state to activate robotic commands which limits its use as a solid



Fig. 4.9 Set of gestures used by Hasan and Misra [19]



Fig. 4.10 Edges of the gestures. (Courtesy of [19])



Fig. 4.11 Normalization operation and features calculation via dividing the gesture edge map with remapping. (Courtesy of [19])

hand gesture recognition approach. This algorithm is scale invariant as any size of hand or image of a hand will result in the same 1D signal. It is also rotation invariant, since the orientation of the hand does not hinder the algorithm from recognizing the gesture. In addition, the position of hand is also not an issue leading to translation invariance.

Fourier descriptor-based methods predominantly use edge contours as the source of features. Hasan and Misra proposed an approach where the edge map of gestures were remapped to 25×25 blocks with each block comprising the output of the edge map due to 5×5 pixels. The edge detection is achieved by convolving the binary image with a Laplacian Mask. Figure 4.9 shows the set of hand gestures they were using with skin segmented edge maps shown in Fig. 4.10. These edge maps were then normalized as shown in Fig. 4.11 and mapped to a 25×25 block feature map representation as shown on Fig. 4.11 (right most). This represents a hand gesture feature vector of size 625 (25×25) and the pixel value of the 25×25 block is determined by the following calculation:

$$B = \sum \text{pixel value in each } 5 \times 5 \text{ block.}$$

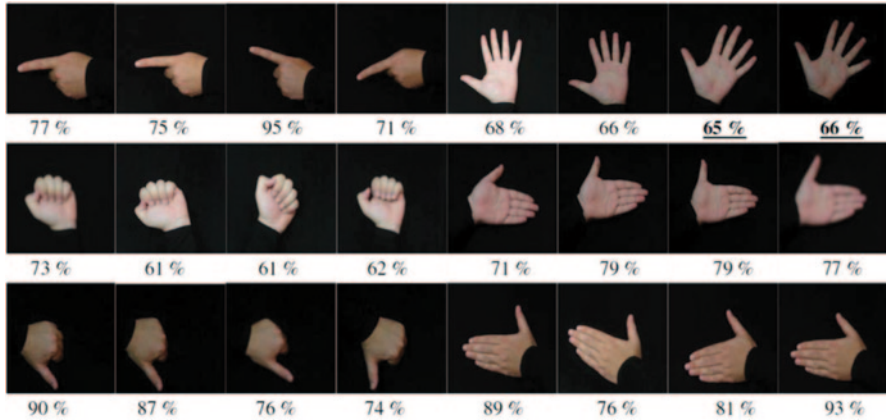


Fig. 4.12 Recognition accuracy for each gesture. (Courtesy of [19])

Their primary objective was to establish a system which could identify specific human gestures and utilize these gestures to control machines in a natural way. They used HSV (Hue, Saturation and Value) color model for segmentation and identified a feature vector of 25×25 after remapping of edges as discussed earlier. Their experiment showed that more than 65% of these features were zero values which leads to minimum storage requirements and the recognition rate achieved surpassed 91% using 36 training gestures and 24 different testing gestures. Their classification results are shown in Fig. 4.12 different gestures.

Li [20] attempted feature extraction techniques that were similar to Hasan and Misra, to classify hand gestures for robot control. It was called Fuzzy C-Means clustering technique however, the feature extraction stage remain very basic. In this approach, the segmented hand shape was converted into a feature vector. Their system used the approach designed by Wachs and Kartoun [21]. In this approach, a feature vector of the image with 13 parameters was created where the first feature is the aspect ratio of the hand's bounding box. The other 12 features were the values representing a coarse discretization of the image, where each grid cell is the mean gray level in the 3 by 4 block partition of the input image. The mean values of each cell represented the average brightness of those pixels in the image. Figure 4.13 illustrates typical user gestures, their binary representation after skin segmentation and the block mean gray scale values and the resultant feature vector on the third bottom row.

Initial research carried out prior to year 2000 focussed on less image processing tasks compared to what is attempted today. The major reasons behind this were the amount of computing power available on ordinary desktop computers to the resolution and the accuracy of cameras and the maturity of the developmental tools and programming languages available at that time. Obtaining the hand outline for human computer interaction was first proposed by Segan and Kumar [22]. In their effort, the outline of the hand is extracted using an edge tracking algorithm. The

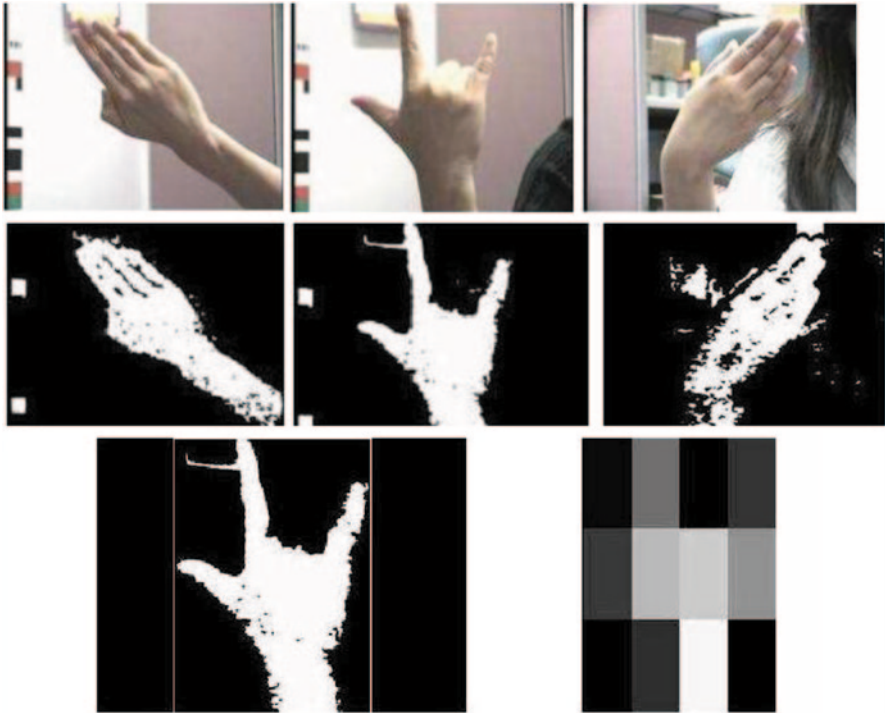


Fig. 4.13 Hand gestures in HSV space (*top row*), their binary representation after skin segmentation (*middle row*) and a gesture and its gray scale block feature vector (*bottom row*). (Courtesy of [20])

system was capable of recognizing both hand postures and gestures which was a remarkable feat at that time. In this approach, the local features were represented by the local extrema of the outline; peaks and valleys. The peaks are found at the finger tips, whereas the valleys are rather found in the regions where two fingers join the palm of the hand. This is shown in Fig. 4.14.

Segan and Kumar restricted their system to identify one of four possible gesture classes: *Point*, *Reach*, *Click*, and *Ground*, shown in Fig. 4.15. *Point* and *Reach* are static gestures, while *Click* is a dynamic gesture that involves quick bending of the index finger. The *Ground* class includes all gestures other than the remaining three, as well as an empty image.

An image that belongs to the *Point* class, was further analyzed to compute the position and orientation of the pointing finger in the image plane, that is a three degrees of freedom (3DOF) pose $(x; y; \theta)$. The classification method consisted of two stages: an initial classification based on analysis of local features, and final classification involving a finite state machine.

Extracting the hand outline of the connected regions was extracted by comparing the input image with a previously acquired background image. After extracting the regions, the boundary of each region was represented as a list of pixel positions in a

Fig. 4.14 Peaks (*circles*) and valleys (*squares*) used in initial gesture classification. (Courtesy of [22])

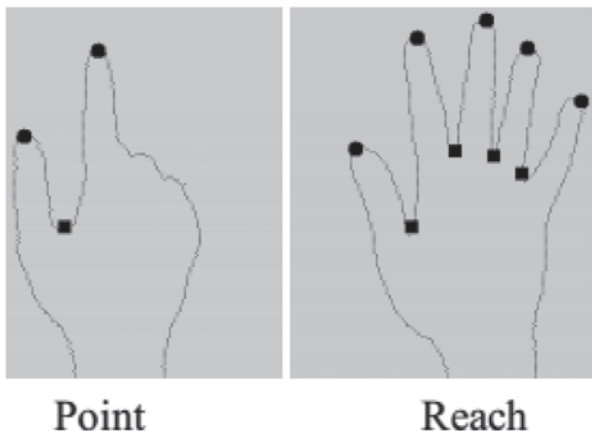
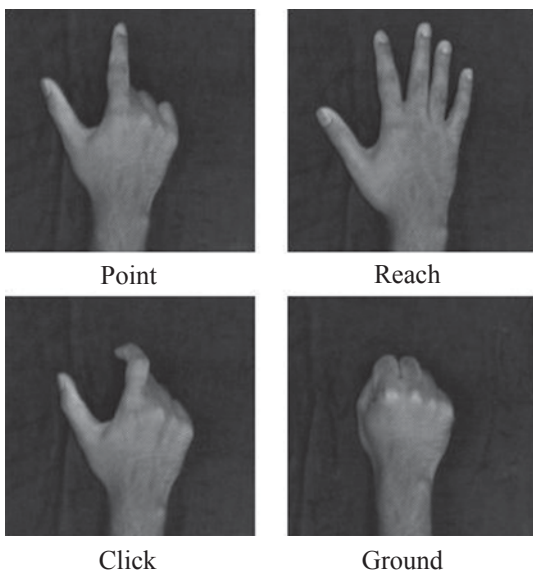


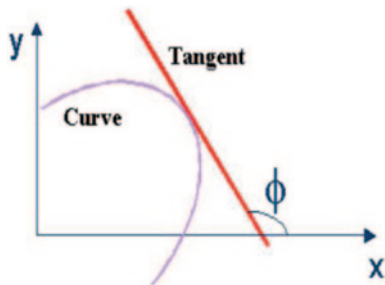
Fig. 4.15 Four possible gesture classes outlined by Segan and Kuma. (Courtesy of [22])



clockwise order. A heuristic screening of the regions based on perimeter length led to the identification of a hand.

The boundary of the region selected as a possible “hand” is further analyzed to extract local features. At each point the k -curvature measure at each point. The k -curvature is the angle $C(i)$ between two vectors $[P(i-k); P(i)]$ and $[P(i); P(i+k)]$, where k is a constant. The points along the boundary where the curvature reached a local extremum, that is the “local features”, were then identified. Some of these local features were labelled “peaks” or “valleys”. Peaks were defined as having a positive curvature above P_{thr} and the ‘Valleys’ were defined as having a negative curvature less than V_{thr} [22].

Fig. 4.16 The curvature of a planar curve. (Courtesy of [23])



One advantage of such features is the quick exclusion of inappropriate gestures using the number of peaks and valleys as indicators. One of the disadvantages was that this simplistic approach limited the available gestures to a minimum of four.

4.2.1 Contour Description using Curvature Scale Space Features

In a race to develop ideal features that would separate hand gestures apart and brings each gesture by different users closer, Chang et al. presented a novel feature extraction approach based on Curvature Scale Space (CSS) for translation, scale, and rotation invariant recognition of hand postures [23]. Initially, the CSS images were used to represent the shapes of boundary contours of hand postures followed by extraction of multiple sets of CSS features to overcome the problem of deep concavities in contours of hand postures [23]. These CSS images can then be classified using techniques such as nearest neighbour classification to establish matchings between multiple sets of input CSS features and the stored CSS features for hand postures. Chang et al. produced results to show the proposed approach was able to extract multiple sets of CSS features from input images with good recognition accuracy.

Mokhtarian and Mackworth [24, 25] first proposed the object contour-based shape descriptor based on the CSS image of the contour [23]. The CSS descriptor provides translation, scale and rotation invariant features of curves.

The curvature κ of a planar curve is defined as the derivative of the tangent angle ϕ with respect to the arc length s , as shown in Fig. 4.16 [23]. The curvature κ is written as follows [23–25]:

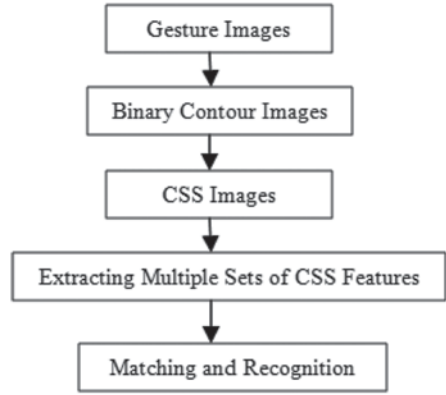
$$\kappa = \frac{d\phi}{ds}$$

and Letting $T = \{x(u), y(u) | u \in [0, 1]\}$ where T is the planar curve and u is the normalized arc length parameter.

Curvature κ can be expressed in terms of u and σ , standard deviation as

$$\kappa(u, \sigma) = \frac{X_u(u, \sigma)Y_{uu}(u, \sigma) - X_{uu}(u, \sigma)Y_u(u, \sigma)}{(X_u(u, \sigma)^2 + Y_u(u, \sigma)^2)^{3/2}}$$

Fig. 4.17 Curvature scale space feature extraction and gesture matching. (Courtesy of [23])



Where

$$\begin{aligned}
 X_u(u, \sigma) &= x(u) * g_u(u, \sigma), \\
 X_{uu}(u, \sigma) &= x(u) * g_{uu}(u, \sigma), \\
 Y_u(u, \sigma) &= y(u) * g_u(u, \sigma),
 \end{aligned}$$

And

$$Y_{uu}(u, \sigma) = y(u) * g_{uu}(u, \sigma), \text{ where } * \text{ denotes convolution,}$$

$$g_u(u, \sigma) = \frac{\partial}{\partial u} g(u, \sigma)$$

and

$$g_{uu}(u, \sigma) = \frac{\partial^2}{\partial u^2} g(u, \sigma).$$

The function defined implicitly by $\kappa(u, \sigma) = 0$ is the CSS image of T [23–25].

In Chang et al.'s approach as outlined in Fig. 4.17, when an image is captured with a potential hand gesture, its contours are extracted using edge detection. It is important to have a continuous contour for the next steps to be successful. Then the contour is successively low-pass filtered with a kernel. For 201, 534, 640, 724 and 731 iterations, the curvature of the curves determine the CSS image. This process is illustrated in Fig. 4.18. With each passing of low-pass filter, the contour smoothens as expected reducing the curvature in many regions.

A good set of features would be expected to be stable when a unique hand gesture is made. Unfortunately, CSS is somewhat unstable with subtle gesture changes as seen in Fig. 4.19. Figure 4.19a and 4.19c denote the same hand posture 4.19b and 4.19d show the respective CSS images of Fig. 4.19a and 4.19c. The locations of the largest peaks which are related to finger directions are unstable in the CSS images.

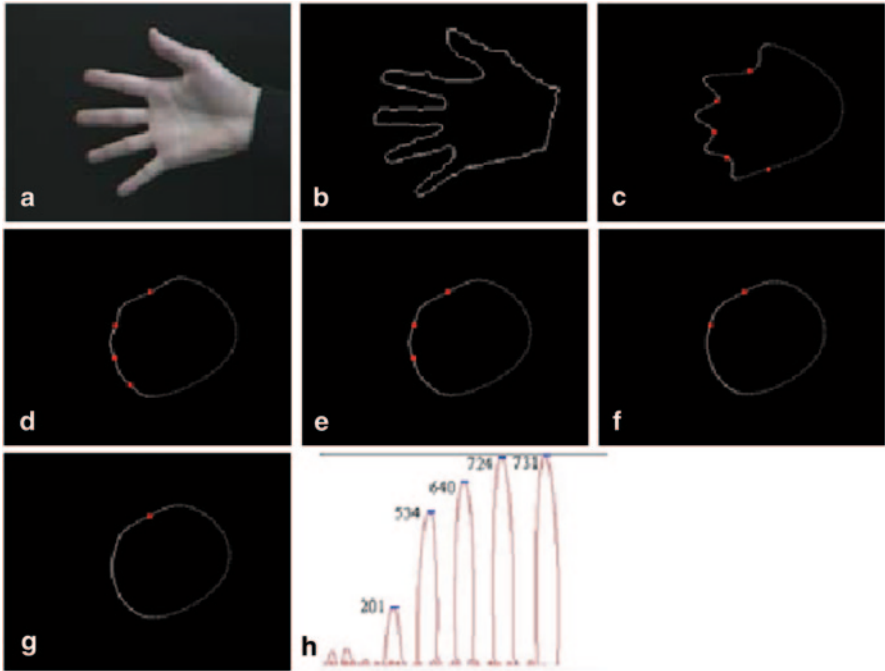


Fig. 4.18 **a** shows the input hand posture. **b** is the contour of the hand posture. **c** to **g** show the resulting contours of the hand pose contour iteratively low-pass filtered by performing a convolution with the (0.25, 0.5, 0.25) kernel for 201, 534, 640, 724 and 731 iterations, respectively. **h** shows the resulting CSS image. (Courtesy of [23])

As shown in Fig. 4.19, the locations of the maximal peaks in the CSS image approximately correspond to the deep concavities in original hand posture contour corresponding to five fingers [23]. Chang et al. extracted multiple sets of CSS features in order to overcome the above instability. They improved their recognition ability by confining their hand posture library to 6 as shown in Fig. 4.20. They reported a recognition rate of 98.3%.

4.3 Features from Karhunen Loeve (K-L) Transform

K-L Transform is well-known for its ability compact data. It is known as the ideal transform for data compression. This ability is very useful in shape description as the shape can be described with minimum number of coefficients opposed to other approaches. The K-L transformation is also known as the principal component transformation, the eigenvector transformation or the Hotelling transformation. The advantages are that it eliminates the correlated data, reduces dimension keeping average square error minimum and provides good clustering characteristics. It establishes a new co-ordinate system whose origin will be at the centre of the object

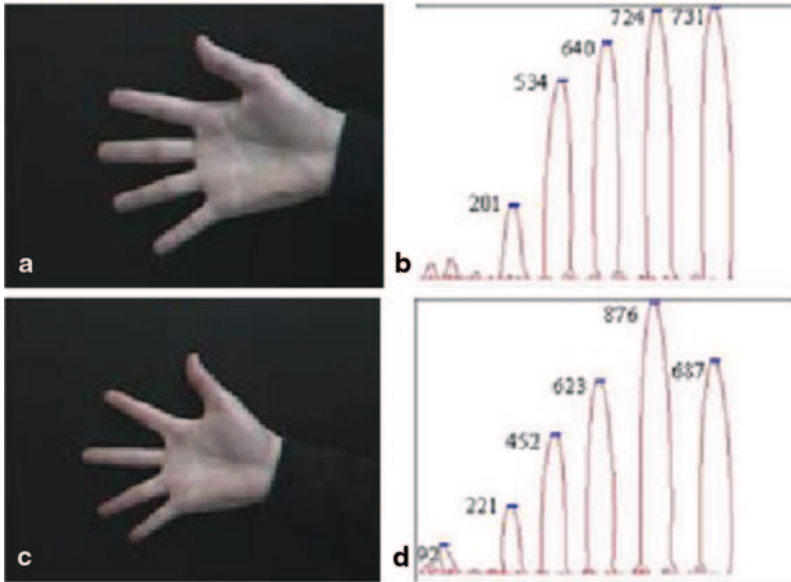
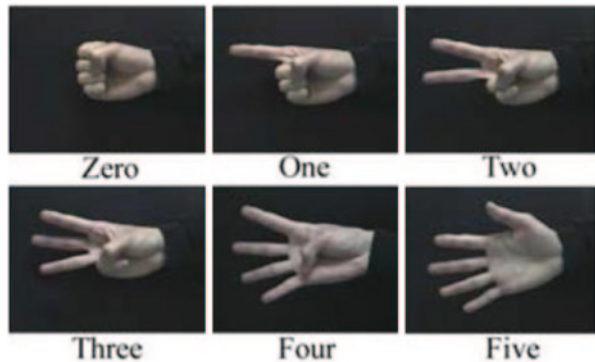


Fig. 4.19 **a** and **c** are the same hand postures. **b** and **d** are the CSS images of **a** and **c**, respectively and shows that the locations of the largest peaks are unstable in the CSS images. (Courtesy of [23])

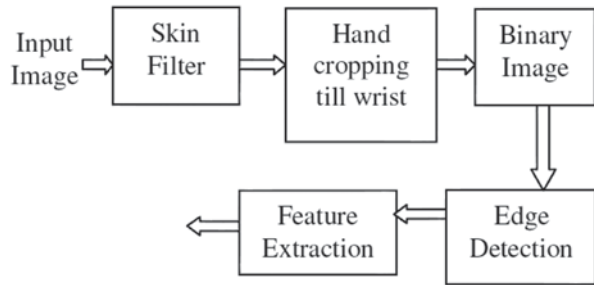
Fig. 4.20 Hand posture library used by Chang et al. (Courtesy of [23])



and the axis of the new co-ordinate system will be parallel to the directions of the Eigen vectors. It is often used to remove random noise.

Singha and Das recently proposed a technique for hand gesture recognition based on K-L transform [26]. Their system composition is shown in Fig. 4.21 for feature extraction. When they extracted binary hand image after skin segmentation and successive cropping, Canny edge detection was used for edge extraction which is then used for K-L feature extraction. K-L provides a mechanism to extract unique features for each gesture which are independent of human hand size and light illumination which are uncorrelated with minimum entropy. As in the use of compression, K-L transform provides the best representation of a unique feature vector that

Fig. 4.21 K-L transform based feature extraction based on [23]



can be classified for gesture detection. Figure 4.22 shows hand gesture image along with the Eigen vectors obtained using K-L Transform. They managed to develop the system to recognize 10 different hand gestures with a recognition rate of 96%.

4.4 Features Described by Histograms

Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs as it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

Dalal and Triggs were the researchers who first described Histogram of Oriented Gradient descriptors in 2005 [27]. In this work they focused their algorithm on the problem of pedestrian detection in static images, although since then they expanded their tests to include human detection in film and video, as well as to a variety of common animals and vehicles in static imagery. Figure 4.23 shows the use of histogram of oriented gradient descriptor used in human detection as described by Suard et al. [28]. Figure 4.24 shows the histograms with different bin resolution of the region shown in a square of Fig. 4.24. What is observed here is that gradient orientation around an edge should be more significant than the one of a point in a nearly uniform region. It also highlights that the larger the number of bins, the more detailed the histogram is.

In the context of object recognition, the use of edge orientation histogram has gained significant popularity [29–32]. However, the concept of dense and local histograms of oriented gradients (HOG) is a method introduced by Dalal et al. [27]. The aim of such a method was to describe an image by a set of local histograms. These histograms count occurrences of gradient orientation in a local part of the image [28].

Freeman and Roth were the pioneering researchers to test whether the use of histogram of local orientation would be useful as a feature in hand gesture recognition

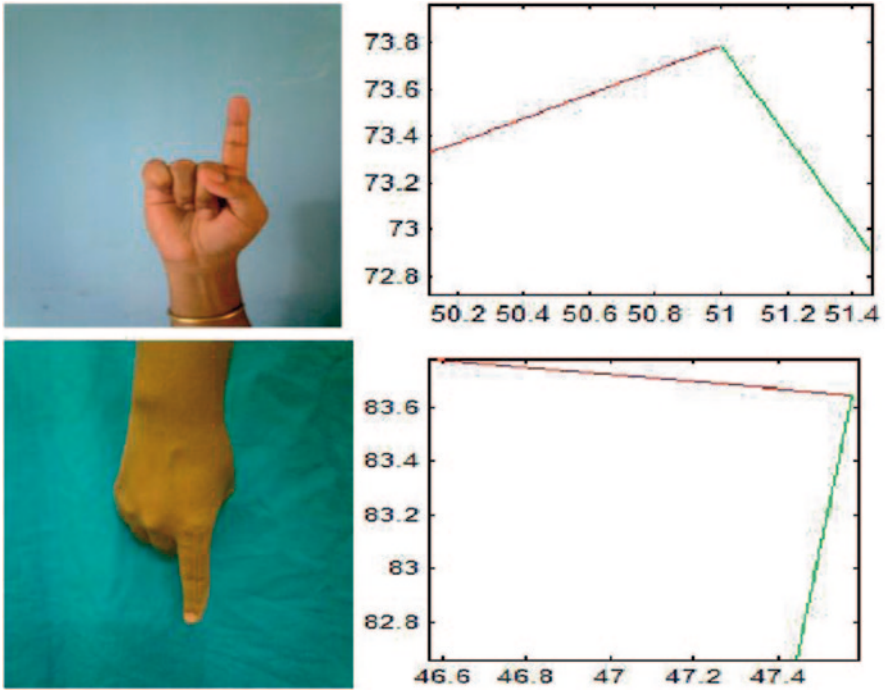


Fig. 4.22 Features extracted for gesture ‘UP’ and ‘DOWN’ and their Eigen vector plots (Right). (Courtesy of [26])

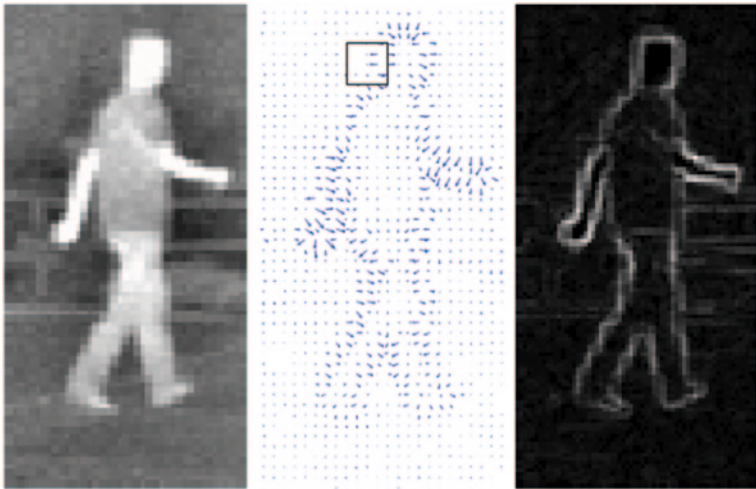


Fig. 4.23 The gradient computation of an image. (left) is the original image, (middle) shows the direction of the gradient, (right) depicts the original image according to the gradient norm [28]

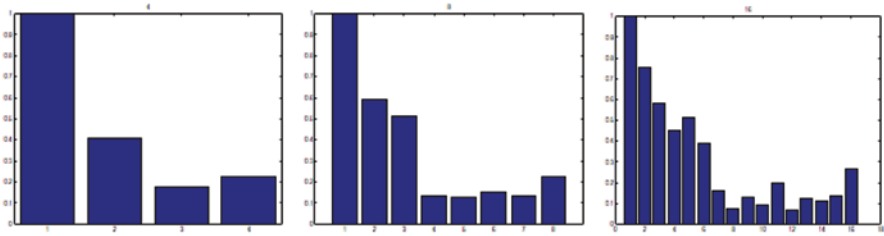


Fig. 4.24 This figure shows the histograms of gradient orientation for (left) 4 bins, (middle) 8 bins (right) 16 bins [28]

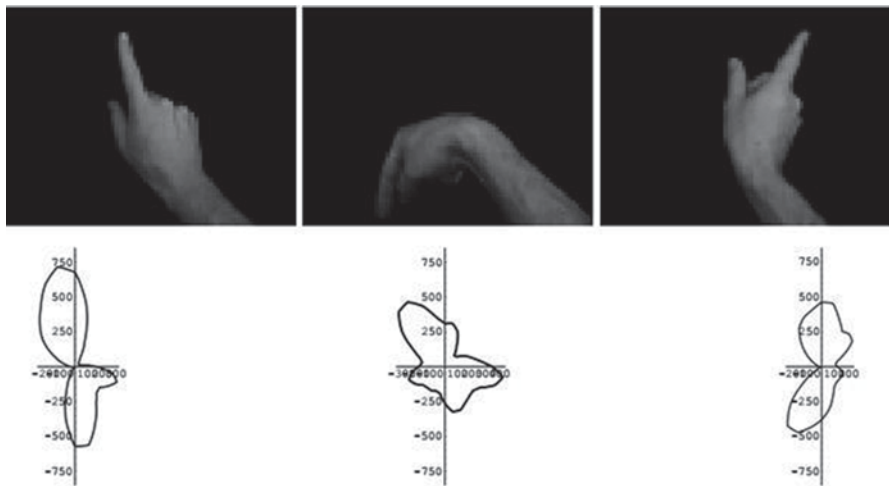


Fig. 4.25 Top row: Up down and right gestures and their orientation histograms shown on the bottom row. (Courtesy of [33])

[33]. They developed a training set that contained up to 15 histograms with their local orientation of various gestures. In their test phase, they compared another histogram of another gesture as shown in Figs. 4.25 and 4.26. The vector in the training database that was closest to the test vector was selected as the gesture was made. Even though their system was restricted to few gestures in today’s standards, their goal was to develop a fast and a robust system that could be implemented on a desktop (in 1994) with invariability to moderate illumination changes. The selection of orientation histogram as a feature vector to represent hand gestures offered robustness to lighting changes and translational invariance of the hand position. Furthermore, the histogram can be calculated very quickly.

In 2004, Zhou et al. proposed a static hand gesture recognition system based on local orientation histogram features [34]. In general, orientation histograms cannot be directly applied to hand gestures as the hand does not provide sufficient texture

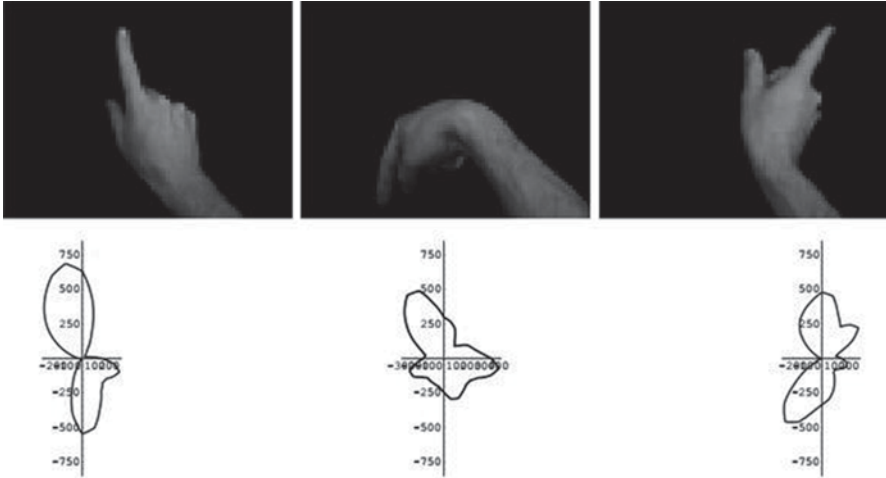


Fig. 4.26 Another instance of information similar to the ones shown in Fig. 4.25. The orientation histograms in this figure highlights that the gestures may be slightly different in each instance but their trajectory is unique to the gesture. (Courtesy of [33])

[35]. Since orientation histograms show the frequency of edges aligned in a certain angle, there might not be enough information available inside the hand area in order to uniquely describe a hand gesture. According to [33], the main problem that might arise is that hand gestures which look different to a human being, might have almost identical orientation histograms. Similar looking hand gestures due simply to rotation yield very different orientation histograms. However, in [34], it is found that the boundary of the hand shape contains enough information to uniquely describe the feature of a specific gesture. Therefore, the idea of local orientation histograms consists of creating overlapping subwindows, containing at least one pixel which lies inside the hand shape. For each of these subwindows, an orientation histogram is created, which is then added to the feature vector. Beside the local orientation histograms, subwindow positions are also added to the feature vector. These positions are measured relative to the median value of all pixel positions that were determined to be in the hand region. Clearly, the advantage of this technique lies in the improved robustness since using relative positions allow in-plane translations.

Misra et al. proposed a hand gesture recognition system that employed the techniques developed for pedestrian detection to recognize a small vocabulary of 7 hand gestures using Histogram of Oriented Gradients as the descriptors [36]. They claimed to use Partial Least Square (PLS) as a ‘class aware’ method of dimensionality reduction which performs better than Principle Component Analysis (PCA) and preserved significant discriminative information in the lower dimensions. Three sets of databases consisting of training as well as testing image sets with varying degree of positional variation were developed to analyse the

importance of using multi-level HOG features for robust human hand gesture recognition. They demonstrated that using only low level HOG features were not adequate for high detection rate. They attained marginal degree of accuracy of detection of human hand gestures and the performance degraded due to the trade-offs between the accuracy and positional variation of the hand. This was also due to the fact that simple brute-force implementation that they relied on using the k -nearest neighbor search algorithm to classify gestures was not effective. Their vocabulary of gestures were confined to only seven hand gestures as they were simply evaluating the feasibility of HOG descriptors and PLS reduction for human hand gesture recognition.

Many techniques exist that uses features derived from edge and gradient based descriptors for hand gesture recognition [37, 38]. Cluttered backgrounds with multiple users and skin-tone regions have hampered hand gesture recognition using such features as gradient based descriptors are only useful in simple uncluttered backgrounds. Dalal and Triggs [27] have demonstrated that for robust visual object recognition, Histogram of Gradients (HOG) descriptors can outperform many other gradient-based feature sets. The HOG descriptors are obtained using different block sizes on the same image and the blocks are contrast normalized to remove the illumination variance. These descriptors are then concatenated to realize the final image descriptors. The HOG features are computed several times for each block in the image, resulting in multiple contributions to the final descriptor, with each cell being normalized with respect to a different block [27].

The HOG based method by Misra et al. uses the edge and gradient based techniques developed for human detection for the problem of hand sign recognition. Similar features have been reported by other research [27, 37, and 38]. Some have used an array of moving spots [39], to recognize hand gestures, [40] presented a glove free solution to this problem.

The dimensionality of the final descriptors increases due to redundancy which needs to be curtailed for classical machine learning algorithms such as the k -nearest neighbor search algorithms to be discussed in the next chapter. Misra et al. used Partial Least Square regression technique for dimensionality reduction as it models relations between a set of observations by means of latent variables, and is aware of the classes into which the observations are classified [41]. They demonstrated that their PLS outperforms PCA in terms of classification of the training data into various hand gestures. They further demonstrated that PLS as the preferred method of dimensionality reduction. PLS is known to have a lower execution time than PCA which saves time in the learning phase [42]. HOG descriptors characterize the articulated gestures by the distributions of local intensity gradients. The feature extraction begins with the gradient computation for all the pixels of the image, with the largest of the gradient of three channels chosen as the gradient of the pixel. Each 'cell' in the image has a histogram which is constructed using the directions and the magnitudes of pixel gradients in the cell. The features are accumulated over a block and are then normalized.

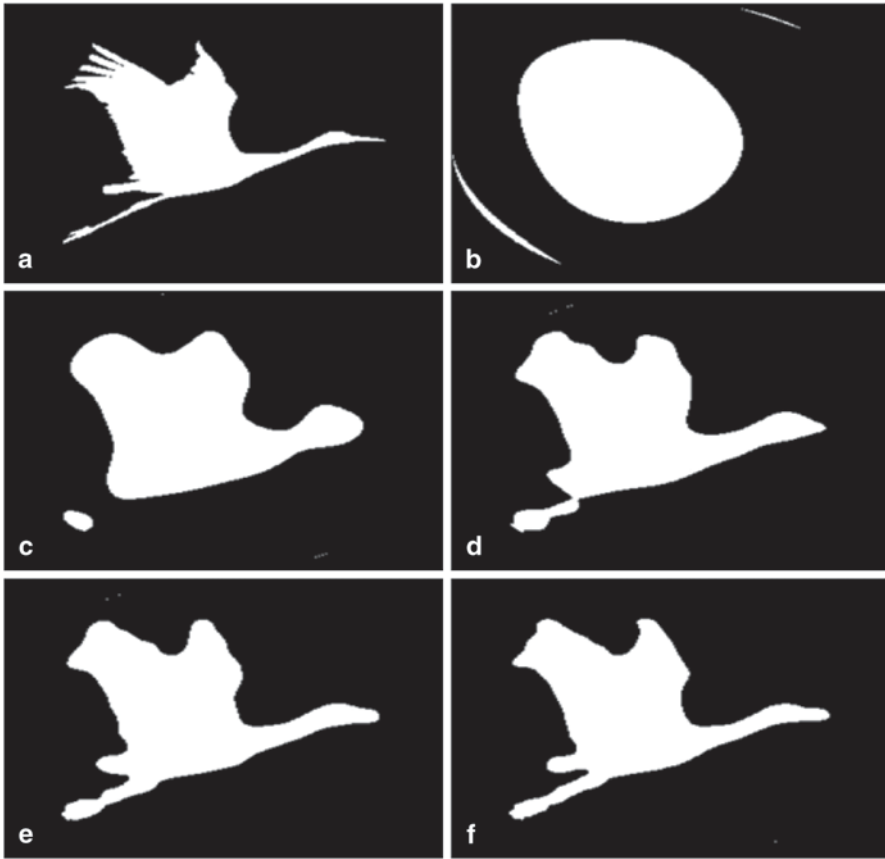


Fig. 4.27 Image reconstruction with Zernike moments. Starting with (b), image is reconstructed gradually using higher Zernike moments

4.5 Zernike Moments

Zernike polynomials are a sequence of polynomials developed by a Nobel laureate mathematician Frits Zernike in 1934 [43]. These sequences are orthogonal on the unit disk and play an important role in beam optics. Zernike moments have been used in image construction as shown in Fig. 4.27.

Moments have been used in image processing and classification type problems since Hu introduced them in his groundbreaking publication on moment invariants [44]. In 1962, Hu mathematically demonstrated that geometric moments can be made to be translation and scale invariant. Since then more powerful moment techniques have been developed. A notable example is Teague's work on Zernike Moments (ZM) as a pioneer to use the Zernike polynomials (ZP) as basis functions for the moments [45]. ZM's have been used in a multitude of applications with great success and some with 99% classification accuracy [46].

The use of ZP's as a basis function is theoretically beneficial because they are orthogonal polynomials which allows for maximum separation of data points, given that it reduces information redundancy between the moments. Their orthogonal properties make them simpler to use during the reconstruction process as well. Furthermore, the magnitude of ZM's is rotationally invariant, which is crucial for certain image processing applications, such as classifying shapes that are not aligned.

4.5.1 Hu Moment Invariants

Hu demonstrated the utility of moment invariants through a simple pattern recognition experiment. The first two moment invariants were used to represent several known digitized patterns in a two-dimensional feature space [47]. An unknown pattern could be classified by computing its first two moment values and finding the minimum Euclidean distance between the unknown and the set of well-known pattern representations in feature space. If the minimum distance was not within a specified threshold, the unknown pattern was considered to be of a new class, given an identity, and added to the known patterns. A similar experiment was performed using a set of twenty-six capital letters as input patterns. When plotted in two-dimensional space, all the points representing each of the characters were distinct. It was observed, however, that some characters that were very different in image shape were close to each other in feature space. In addition, slight variations in the input images of the same character resulted in varying feature values that in turn lead to overlapping of closely spaced classes. Hu concluded that increased image resolution and a larger feature space would improve object distinction [47].

Moment invariants algorithm has been known as one of the most effective methods to extract descriptive feature for object recognition applications. The algorithm has been widely applied in classification of aircrafts, ships, ground targets, etc [48–56]. Essentially, the algorithm derives a number of self-characteristic properties from a binary image of an object. These properties are invariant to rotation, scale and translation. Let $f(i, j)$ be a point of a digital image of size $M \times N$ ($i=1, 2, \dots, M$ and $j=1, 2, \dots, N$). The two dimensional moments and central moments of order $(p+q)$ of $f(i, j)$, are defined as:

$$m_{pq} = \sum_{i=1}^M \sum_{j=1}^N i^p j^q f(i, j)$$

$$U_{pq} = \sum_{i=1}^M \sum_{j=1}^N (i - \bar{i})^p (j - \bar{j})^q f(i, j)$$

Where

$$\bar{i} = \frac{m_{10}}{m_{00}} \quad \bar{j} = \frac{m_{01}}{m_{00}}$$

From the second order and third order moments, a set of seven moment invariants are derived as follows [44]:

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

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

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

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

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 \right] \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right]$$

$$\phi_6 = (\eta_{20} - \eta_{02}) \left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 \right] \\ - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right]$$

Where η_{pq} is the normalised central moments defined by:

$$\eta_{pq} = \frac{U_{pq}}{U_{00}^r}$$

where $r = [(p+q)/2] + 1$ and $p+q = 2, 3, \dots$

4.5.1.1 Example of Invariant Properties of Hu Moments

Figure 4.28 shows images containing letter 'A', rotated and scaled, translated and noisy versions of letter 'A' and Fig. 4.29 shows letter 'L'. Their respective moment invariants calculated using the moment invariants are shown in Tables 4.1 and 4.2. It is obvious from Table 4.1 that the algorithm produces the same result for the first three orientations of letter 'A' despite the different transformations applied upon them. There is only one value, i.e. Φ_1 displays a small discrepancy of 5.7% due to the difference in scale. The other values of the three figures are effectively the same for $\Phi_2, \Phi_3, \Phi_4, \Phi_5, \Phi_6$ and Φ_7 . The last letter, however, reveals the drawback of the algorithm: it is susceptible to noise. Specifically, the added noisy spot in the letter has changed the entire moment invariants set. This drawback suggests that moment invariants can only be applied on noise-free images in order to achieve

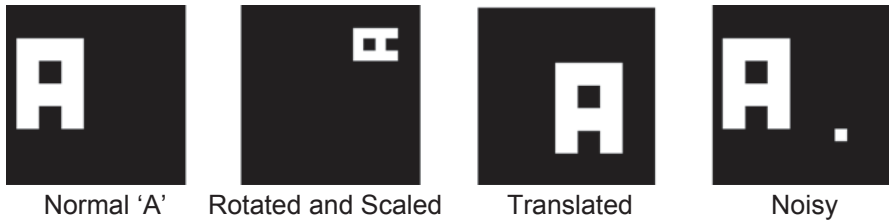


Fig. 4.28 Letter 'A' in different orientations



Fig. 4.29 Letter 'L' in different orientations

Table 4.1 Moment invariants of the different orientations of letter 'A'

	A1	A2	A3	A4
Φ_1	0.2165	0.2165	0.204	0.25153
Φ_2	0.001936	0.001936	0.001936	0.002161
Φ_3	3.69×10^{-5}	3.69×10^{-5}	3.69×10^{-5}	0.004549
Φ_4	1.64×10^{-5}	1.64×10^{-5}	1.64×10^{-5}	0.002358
Φ_5	-4.03×10^{-10}	-4.03×10^{-10}	-4.03×10^{-10}	7.59×10^{-6}
Φ_6	7.21×10^{-7}	7.21×10^{-7}	7.21×10^{-7}	7.11×10^{-5}
Φ_7	0	0	0	1.43×10^{-6}

Table 4.2 Moment invariants of the different orientations of letter 'L'

	L1	L2	L3
Φ_1	0.34028	0.31944	0.31944
Φ_2	0.043403	0.043403	0.043403
Φ_3	0.023148	0.023148	0.023148
Φ_4	0.002572	0.002572	0.002572
Φ_5	-5.56×10^{-6}	-5.56×10^{-6}	-5.56×10^{-6}
Φ_6	-0.00015	-0.00015	-0.00015
Φ_7	1.91×10^{-5}	1.91×10^{-5}	1.91×10^{-5}













Hand Gesture						
Classification accuracy	44%	50%	74%	82%	97%	90%
Hand Gesture						
Classification accuracy	95%	96%	70%	95%	47%	89%

Table 4.3 Some hand gestures and their corresponding classification scores

the best results. Since the algorithm is firmly effective against transformations, a simple classifier can exploit these moment invariants values to differentiate as well as recognise the letter ‘A’ from other letters, such as the letter ‘L’.

4.5.1.2 Application of Moment Invariants in Hand Gesture Recognition

The example in the previous section proved that moment invariants can be used for object recognition applications since it is rigidly invariant to scale, rotation and translation. The following account summarizes the advantages of moment invariants algorithm for gesture classification.

For each specific gesture, moment invariants always give a specific set of values. These values can be used to classify the gesture from a sample set. The set of chosen gestures have a set of unique moments.

- Moment invariants are invariant to translation, scaling and rotation. Therefore, the user can issue commands disregarding orientation of the hand.
- The algorithm is susceptible to noise. Most of this noise, however, is filtered at the gesture normalisation stage.
- The algorithm is moderately easy to implement and requires only an insignificant computational effort from the CPU. Feature extraction, as a result, can be progressed rapidly and efficiently.
- The first four moments, Φ_1 , Φ_2 , Φ_3 , and Φ_4 are adequate to represent a gesture uniquely and hence result in a simple feature vector with only four values.

In 2005, the author successfully used moment invariants for classifying hand gestures to control consumer electronics with extremely high accuracy. This was partly due to the fact that selection of specific ten gestures resulted in a distinctive set of gestures which achieved good classification scores with Hu moments. The system was classified using a Neural Network approach [57]. Table 4.3 highlights the recognition accuracy for different hand gestures.

Feature extraction plays the most prominent role in any classification problem. Hand gesture recognition is no exception. Over the years, researchers have used basic Fourier descriptors to exotic versions of Fourier descriptors such as Elliptic Fourier descriptors to modified Fourier descriptors to remove the limitations of feature extractions. Yet, poor results in classification further drove them to HOG to KL transform in an effort to robustly classify gestures. The authors' personal involvement in developing a feature extraction method based on Hu moments improved the classification of hand postures significantly that resulted in a pioneering gesture controlled interface for home entertainment.

References

1. Zahn, C.T., Roskies, R.Z.: Fourier descriptors for plane close curves. *IEEE Trans. Comput.* **C-21**, 269–281 (1972)
2. Granlund, G.H.: Fourier preprocessing for hand print character recognition. *IEEE Trans. Comput.* **C-21**, 195–201 (1972)
3. Bertrand, O., Queval, R., Maître, H.: Shape interpolation by Fourier descriptors with application to animation graphics. *Signal Process. Chapter 4*, 53–58 (1981)
4. Persoon, E., Fu, K.: Shape discrimination using Fourier descriptors. *IEEE Trans. Syst. Man Cybern.* **7**(3), 170–179 (1977)
5. Zhang, D., Lu, G.: Shape-based image retrieval using generic Fourier descriptor. *Signal Process. Image Commun.* **17**(10), 825–848 (2002)
6. Bober, M.: MPEG-7 visual shape descriptors. *IEEE Trans. Circuits Syst. Video Technol.* **11**(6), 716–719 (2001)
7. Zhang, D., Lu, G.: Generic Fourier descriptor for shape-based image retrieval. *International Conference on Multimedia and Expo*, 425–428 (2002)
8. Lin, C.S., Hwang, C.L.: New forms of shape invariants from elliptic Fourier descriptors. *Pattern Recognit.* **20**(5), 535–545 (1987)
9. Lin, C.S., Jungthirapanich, C.: Invariants of three-dimensional contours. *Pattern Recognit* **23**(8), 833–842 (1990)
10. Harding, P.R.G., Ellis, T.J.: Recognizing hand gesture using Fourier descriptors. *Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)* 4, (2004)
11. Conseil, S., Bourennane, S., Martin, L.: Comparison of Fourier descriptors and Hu moments for hand posture recognition. *European Signal Processing Conference*, (2007)
12. <http://www.idiap.ch/resource/gestures/>
13. Utsumi, A., Miyasato, T., Kishino, F.: Multi-camera hand pose recognition system using skeleton image. *IEEE International Workshop on Robot and Human Communication*, 219–224 (1995)
14. Licsár, A., Szirányi, T.: Dynamic training of hand gesture recognition system. *Proceedings of the 17th International Conference on Pattern Recognition vol. 4*, (2004)
15. Rui, Y., She, A., Huang, T.S.: A modified Fourier descriptor for shape matching in MARS. *Image Databases and Multimedia Search*, 165–180 (1998)
16. Ng, C.W., Ranganath, S.: Real-time gesture recognition system and application. *Image Vis. Comput.* **20**, 993–1007 (2002)
17. Chen, F.S., Fu, C.M., Huang, C.L.: Hand gesture recognition using a real-time tracking method and hidden Markov models. *Image Vis. Comput.* **21**, 745–758 (2003)
18. Malima, A., Ozgur, E., Cetin, M.: A fast algorithm for vision-based hand gesture recognition for robot control. *IEEE 14th Signal Processing and Communications Applications*, (2006)
19. Hasan, M.M., Mishra, P.K.: HSV brightness factor matching for gesture recognition system. *Int. J. Image Process. (IJIP)*. **4**(5), 456–467 (2010)

20. Li, X.: Gesture recognition based on fuzzy C-means clustering algorithm. Department of Computer Science, The University of Tennessee, Knoxville (2003)
21. Wachs, J., Kartoun, U., Stern, H., Edan, Y.: Real-time hand gesture telerobotic system. Proceedings of WAC, Florida (2002)
22. Segen, J., Kumar, S.: Fast and accurate 3d gesture recognition interface. Proceedings of the 14th International Conference on Pattern Recognition, vol. 1, 86 (1998)
23. Chang, C.C., Chen, I.-Y., Huang, Y.-S.: Hand pose recognition using curvature scale space. IEEE International Conference on Pattern Recognition (2002)
24. Mokhtarian, F., Mackworth, A.K.: Scale-based description and recognition of planar curves and two dimensional shapes. IEEE Trans. Pattern Anal. Machine Intell. **PAMI-8**(1), 34–43 (1986)
25. Mokhtarian, F., Mackworth, A.K.: A theory of multiscale, curvature-based shape representation for planar curves. IEEE Trans. Pattern Anal. Machine Intell. **14**(8), 789–805 (1992)
26. Singha, J., Das, K.: Hand gesture recognition based on Karhunen–Loeve transform. Mobile & Embedded Technology International Conference 2013, pp. 365–371 (2013)
27. Dalal, D., Triggs, B.: Histograms of oriented gradients for human detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2, (2005)
28. Suard, F., Rakotomamonjy, A., Bensrhair, A., Broggi, A.: Pedestrian detection using infrared images and histograms of oriented gradients. IEEE Intelligent Vehicles Symposium, 206–212 (2006)
29. Bineesh, T.R., Simon, S.: Fast pedestrian detection using smart ROI separation and integral image based feature extraction. Int. J. Comput. Sci. Eng. **4**(11), 1771–1779 (2012)
30. Zhu, Q., Avidan, S., Yeh, M., Cheng, K.: Fast human detection using a cascade of histograms of oriented gradients. CVPR (2006)
31. Hu, B., Wang, S., Ding, X.: Pedestrian detection based on hybrid features. Second International Symposium on Intelligent Information Technology Application, 321–325 (2008)
32. Shashua, A., Gdalyahu, Y., Hayon, G.: Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. Proceedings of IEEE Intelligent Vehicles Symposium, (2004)
33. Freeman, W., Roth, M.: Orientation histograms for hand gesture recognition. Mitsubishi Research Laboratory Report (1994)
34. Zhou, H., Lin, D.J., Huang, T.S.: Static hand gesture recognition based on local orientation histogram feature distribution model. Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'04) (2004)
35. Messer, T.: Static hand gesture recognition. University of Fribourg, Switzerland, (2009)
36. Misra, A., Okatani, T., Deguchi, K.: Hand gesture recognition using histogram of oriented gradients and partial least squares regression. MVA2011 IAPR Conference on Machine Vision Applications, 479–482 (2011)
37. Freeman, W.T., Roth, M.: Orientation histograms for hand gesture recognition. Intl. Workshop on Automatic Face- and Gesture- Recognition, IEEE Computer Society, Zurich, Switzerland, 296–300 (1995)
38. Lee, H.-J., Chung, J.-H.: Hand gesture recognition using orientation histogram. Proc. IEEE Region 10 Conf. **2**, 1355–1358 (1999)
39. Tartter, V.C., Knowlton, K.C.: Perception of sign language from an array of 27 moving spots. Nature **289**, 676–678 (1981)
40. Fukumoto, M., Mase, K., Suenaga, Y.: Real time detection of pointing actions for a glove free interface. Workshop on Machine Vision Applications, Tokyo, (1992)
41. Wold, H.: Partial least squares. In: Kotz S., Johnson, N. (eds.) Encyclopedia of Statistical Sciences, vol. 6, John Wiley: New York, pp. 581–591. (1985)
42. Schwartz, W.R., Kambhavi, A., Harwood, D., Davis, L.S.: Human detection using partial least squares analysis. Computer Vision, 2009 IEEE 12th International Conference on, (2009)
43. Zernike, F.: Beugungstheorie des Schneidverfahrens und Seiner Verbesserten Form, der Phasenkontrastmethode. Physica **1**(8), 689–704 (1934)

44. Hu, M. K.: Visual pattern recognition by moment invariants. *IRE Trans. Inf. Theory* **8**(2), 179–187 (1962)
45. Teague, M. R.: Image analysis via the general theory of moments. *Opt. Soc. Am.* **70**(8), 920–930 (1979)
46. Khotanzad, A., Hong, Y. H.: Invariant image recognition by zernike moments. *IEEE* **12**(5), 489–497 (1990)
47. Prokop, R.J., Reeve, A.P.: A survey of moment-based techniques for unoccluded object representation and recognition. Survey Cornell University <http://www.via.cornell.edu/ece547/text/survey.pdf>. Accessed Oct. 29, 2013
48. Zhongliang, Q., Wenjun, W.: Automatic ship classification by superstructure moment invariants and two-stage classifier. *ICCS/ISITA '92. 'Communications on the Move'*, 544–547 (1992)
49. Teague, M.R.: Image analysis via the general theory of moments. *J. Opt. Soc. Am.* **70**(8), 920–930 (1980)
50. Dudani, S.A., Breeding, K.J., McGhee, R.B.: Aircraft identification by moment invariants. *IEEE Trans. Comput.* **C-26**(1), 39–46 (1977)
51. Sluzek, A.: Using moment invariants to recognize and locate partially occluded 2D objects. *Pattern Recognit. Lett.* **7**(4), 253–257 (1988)
52. Gilmore, J.F., Boyd, W.W.: Building and bridge classification by invariant moments. *SPIE* **292**, pp. 256–263 (1981)
53. Wong, R.Y., Hall, E.L.: Scene matching with invariant moments. *Comput. Graphics Image Process.* **8**, 16–24 (1978)
54. Smith, F.W., Wright, M.H.: Automatic ship photo interpretation by the method of moments. *IEEE Trans. Comput.* **C-20**, 1089–1095 (1971)
55. Boyce, J.F., Hossack, W.J.: Moment invariants for pattern recognition. *Pattern Recognit. Lett.* **1**(5, 6), 451–456 (1983)
56. Yin, B.H., Mack, H.: Target classification algorithms for video and FLIR imagery. *SPIE* **0302**, pp. 134–141 (1981)
57. Premaratne, P., Safaei, F., Nguyen, Q.: Moment invariant based control system using hand gestures: Book intelligent computing in signal processing and pattern recognition. *Book Series Lect. Notes Control Inf. Sci* **345**, 322–333 (2006)