Recognition Using the Multi-PDM Method and Hidden Markov Models

Chung-Lin Huang and Ming-Shan Wu
Institute of Electrical Engineering
National Tsing -Hua University
Hsin-Chu, TAIWAN, ROC
e-mail clhuang@ee.nthu.edu.tw

ABSTRACT

This paper intoruduces a gesture interpretation based on a multi-Principal-Distribution-Model (PDM) and Hidden Markov Models (HMMs). To track the hand-shape, it uses the PDM model which is built by learning patterns of variability from a training set of correctly annotated images. For gesture recognition, we need to deal with a large variety of hand-shape. Therefore, we divide all the training hand shapes into a number of similar groups, with each group trained for an individual PDM shape model. Finally, we use the HMM to determine model transition among these PDM shape models. From the model transition sequence, it can identify the continuous gestures denoting one-digit or two-digit numbers.

1. Introduction

Gestures have been widely used by human being. Gesture input aims to exploit this natural expertise for human-computer interface. If the machine can understand the human gesture either static or dynamic effectively, then it will greatly benefit us human being. In the last several years, there has been an increased interest in trying to introduce human-machine interaction through human body motion that coincides with a growing interest in a closely related field - virtual reality. Pavlovic *et al.* [1] presented a review of the most recent works related to hand gesture interface techniques: glove-based technique[2] and vision-based technique[3-9]. The vision-based technique is the most natural way of constructing a human-computer interface which has many applications. However, it has difficulties in (1) segmentation of the moving hands; (2) tracking and analyzing the hand motion; and (3) recognition.

This paper presents a multi-PDM-based method for hand tracking and handshape extraction, and then generates an ordered sequence of model transitions by using the hidden Markov Model(HMM). The PDM-based hand shape extraction is resistant to complex background influence, and the model transition is invariant to the non-uniform changes in speed and viewing direction. Our method has the advantage that the gesture recognition depends on how the system makes the PDM model transition instead of how exactly it reaches a certain position in 3-D space. Our goal is to convert the variances of the gesture in the spatio-temporal space into a sequence of PDM model transitions as a gesture symbolical representation.

The gesture recognition technique includes tracking the object of interest and identifying the non-rigid hand-shape. The major assumption for a successful tracking algorithm is that the 2-D shape of the moving hand-shape changes smoothly between two consecutive frames. The system has two stages: (1) multi-PDM-based hand-shape tracking and measurement, (2) HMM-based PDM model transition determination. First, we find that the PDM method can only fit new hand examples similar to shapes of the corresponding training set. Since there are so many different hand shapes with lots of varieties, we need to divide all the hand shapes into a number of similar groups, with each group trained for an individual PDM model. Second, for each frame, with the observation of the fitness function, we apply HMM to determine the PDM model transition. The model transition is required when the current flexible model is no longer suitable for a large variation of the hand-shape in the following frames.

2. Shape Model and Feature Points Interaction

Here, we modify the Active Shape Model[10](or Point Distribution Model(PDM)) method to extract the hand shapes. For PDM, the average example is calculated and the deviation of each example from the mean is established. A principal component analysis of the covariance matrix of deviations reveals the main mode of variation. Usually only a small number of model parameters is required to

reconstruct the training examples.

We may generate new examples of the shape, which will be similar to those in the training set, by varying the parameters within certain limits. The mean shape model is placed in the image, and is allowed to interact dynamically until it fits to the location of a newly suggested position for each model point based on the matching of the local intensity model. By varying the shape parameters that are consistent with the training set, we can find the best shape model fitted with the real shape in the image.

We manually locate the feature points on the training set images by following some rules to ensure that each point plays an essential role on the boundary of the images (see Fig. 1). This will ensure the coherence of points on the different features. We call these points "landmark points".

Given a current shape position $\mathbf{X}$ (with centroid $\mathbf{X}_c$), we need to adjust the global shape variation(including the translation $d\mathbf{X}_c = (dX_c, dY_c)$, rotation $d\theta$, the scale ds) and the local shape variation $\mathbf{db}$ to find the next fitting position $\mathbf{X}+d\mathbf{X}$ as,

$$\mathbf{X}+d\mathbf{X}=(\mathbf{X}_c+d\mathbf{X}_c)+\mathbf{M}((s+ds),\ (\theta+d\theta))\cdot[\bar{\mathbf{x}}+\mathbf{P}\cdot(\mathbf{b}+d\mathbf{b})] \quad (1)$$

where $\mathbf{M}(s, \theta)$ is a 2×2 rotation matrix. By finding gray-level profiles of every point j on $\mathbf{X}+d\mathbf{X}$ ($\mathbf{x}_j \in \mathbf{X}+d\mathbf{X}$) as $\mathbf{g}_j$, we calculate the gray-level profile fitness value $F(\mathbf{x}_j)$ and find the overall F values (i.e., $\Sigma_j F(\mathbf{x}_j)$ for $\mathbf{x}_j \in \mathbf{X}+d\mathbf{X}$) of all landmark points. If the $\Sigma_j F(\mathbf{x}_j)$ is minimized then the position $\mathbf{X}+d\mathbf{X}$ indicates the best fitted shape. In the following, we illustrate a modified PDM-based fitting process.

1) **Initial Hand Model Position Estimation.** In the hand-shape extraction process, we may encounter the problem that if the positions of some fitting points are too far away from the actual positions, then the adjustment may require a lot of iterations to pull the landmarks points to the proper place. Therefore, we apply frame difference operation to find the moving regions one of which is supposed to be the moving hand. From these extracted regions, we can roughly estimate the position of the hand to place the initial PDM shape model.

2) **Shape Adjustment Process.** Here, we apply the two-step estimations for the global shape variation parameters (i.e., the translation $d\mathbf{X}_c$, the rotation $d\theta$, the scale ds) and the local shape variation parameter (i.e., $\mathbf{db}$). First, we assume that the current global shape is $\mathbf{X}$, then we can do the global shape variation for the new global shape as $\mathbf{X}+d\mathbf{X} = \mathbf{M}(s+ds, \theta+d\theta)\cdot[\mathbf{x}]+(\mathbf{X}_c+d\mathbf{X}_c)$, where $\mathbf{M}$ is a 2×2 rotation matrix, $\mathbf{x}$ represents the aligned shape, and $\mathbf{X}_c$ represents the central point of current shape. Second, we may also deform the current local shape $\mathbf{x}$, by changing local shape parameter $\mathbf{db}$ to generate the new local shape as $\mathbf{x}+d\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}(\mathbf{b}+d\mathbf{b})$.

3. Multi-PDM Model Transition using Hidden Markov Models

If the hand shapes undergo enormous shape changes in the image sequence (the variance of the cloud of each corresponding model point of aligned shapes is very large), then we need to divide the training set of all the possible hand shapes into several similar shape groups. The variance of each cloud of aligned shapes in each group has to be small for tracking the variable hand shapes. Then each group is treated as an individual training set and trained as a different PDM shape model.

If the hand shape extraction by using current PDM shape model is no longer effective, the specific HMM can be found to determine when to replace it by another PDM model that is called PDM model transition (see Fig. 3). The measurements $\{F(x_i)\}$ for certain landmark points are used as an observation sequence for the system to determine which HMM has the highest model probability that indicates the most appropriate PDM model transition.

3.1 Hidden Markov Model

Here, we create one HMM for each possible PDM transition between two consecutive frames. We use the observations, $\mathbf{O}=\{F_i\}$, from current frame, to estimate the optimum parameters for each HMM, i.e., we obtain the model parameter λ_p, for the p^{th} HMM. Given the measurement $\mathbf{O}=\{F_i\}$ of current frame and a HMM, which may indicate certain unknown model transition, we calculate $P(\mathbf{O}|\lambda)$. The $P(\mathbf{O}|\lambda)$ can be calculated by summing the probability over all the possible state sequence $S=(s_0, s_1, \ldots s_T)$, where $s_t \in \{1,2,\ldots N\}=Z_N$, in a HMM model for the observation sequence:

$$P(\mathbf{O}|\lambda) = \sum_{allS} \pi_{s_0} \prod_{t=1}^{T} a_{s_{t-1}s_t} b_{s_t}(O_t) \quad (2)$$

The objective in maximum likelihood estimation is to maximize $P(\mathbf{O}|\lambda)$ over all parameters λ for a given observation. The above maximum likelihood estimation can be effectively solved by Baum-Welch algorithm [22]. Here we consider different optimization criterion for estimating the parameters of HMM. Instead of using the likelihood function (10), we apply the following function as the optimization objective (it is called the state-optimized likelihood):

$$\max_{s} P(O, S|\lambda) = \max_{s} \pi_{s_0} \prod_{t=1}^{T} a_{s_{t-1}s_t} b_{s_t}(O_t) \qquad (3)$$

Then we may apply the segmental K-means algorithm[21] for estimating the parameters of the HMM's. Then, we choose the best HMM u^* (indicating the appropriate PDM model transition) by finding the highest model probability, i.e., $u^* = \text{argmax}_{1\le u\le U} [P_u]$ where $P_u = P_u(\mathbf{O}, S^* | \lambda_p)$, and λ_p makes $max_\lambda P(\mathbf{O}, S^* | \lambda)$.

3.2 HMM Training

Since our decision rule is based on the state-optimized likelihood function, the estimated parameter λ' should be such that $\Pr(\mathbf{O} | \lambda')$ is maximized over the training set. The training problem is the crucial one for most applications of HMM's. It allow us to optimally adapt model parameters to the observed training data, and then create the best models for real phenomena. In this paper, we define the observation sequence in terms of spatial order (for each input frame) as $\mathbf{O}=(O_1, O_2, O_3, O_4, O_5)$, where $O_1=\{F(x_i)\}$ The central point of $\{x_i,\}$ is located on the finger-tip of the thumb, the index finger, the middle finger, the ring finger, and the little finger, respectively. Each observation vector O_t may be assigned to one of the three different states: bending (S_b), half-bending (S_h), and straight (S_s) indicating the status of each finger.

We start with a training sequence consisting of a number of repetitions of the gesture frames (made by many gesture-makers). For each HMM model, we first adjust the model parameters λ so that $\Pr(\mathbf{O} | \lambda)$ is maximized. Then we use Viterbi algorithm to find the optimal state sequence associated with the given observation sequence. The results are used to re-estimate the model parameter λ'. The initial model defines a critical point of the likelihood function, in which $\lambda'=\lambda$. Baum-Welch algorithm [12] has been proposed to re-estimate a new model λ' which is more likely in a sense that $\Pr(\mathbf{O} | \lambda') > \Pr(\mathbf{O} | \lambda)$. The model λ' indicates that the observation sequence is more likely to be produced. Instead of finding the λ_p that minimizes $P(\mathbf{O} | \lambda)$ (i.e., $\max_{\lambda_p} P(\mathbf{O} | \lambda)$), which requires summing all possible state sequences (see (2)), we focus on the most likely state sequence(see (3)), and apply the segmental K- mean algorithm [11]

4. System Implementation Criteria

To make a single-digit number gesture, we start the gesture-making operation from holding our fist, then raise certain fingers to indicate the specific number (see Fig. 4), and finally bend those fingers to return to fist-holding state. If one want to make gesture indicating two-digit number, then he may repeat the above operation. However, if we want to make a gesture indicating a single-digit '0', then we may differentiate the beginning/ending fist-holding gesture from the gesture indicating digit '0'. Therefore, we use the forward translation motion between the beginning fist-holding gesture and the gesture indicating digit '0' and then use the reverse translation motion between the gesture indicating digit '0' and the ending fist-holding gesture.

For each frame, we can track the hand gesture by using the most appropriate PDM models (applied to the previous frame) to calculate the $\{F(x_i)\}$ as an observation sequence. Using the observations of current frame, we apply all possible related HMMs and find the best HMM with the highest state-optimized likelihood that indicates the most appropriate PDM model for the current frame. In our system, we have trained two different categories of HMMs. The first one has 10 HMM's (HMM_i, i=0,1,..9) indicating no PDM model transition. The second one consists of 45 HMMs (HMM_{ij}) corresponding to a PDM model transition, from current PDM model m_i to the other PDM model m_j. We assume that the measurement statistics $\{F(x_i)\}$ corresponding to HMM_{ij} representing the transition from PDM model m_i to PDM model m_j and the other HMM_{ji} indicating the transition from PDM model m_j to PDM model m_i are trained as the same HMM. Given an observation sequence, we need to find the optimal HMM which indicates whether there is an PDM model transition or not. If there is a PDM model transition, then what kind of PDM model transition may occur. During the training process, given as many known input frames as possible, we train 55 different HMMs individually for our system. The best trained HMM is the one indicating no PDM model transition. Since the measurement statistics $\{F(x_i)\}$ of most of the frames in the image sequence favor the first category HMM.

Here, we assume that the PDM model transition can also be determined if the hand movement is tracked by measuring the displacement of the centroid of the extracted hand shapes in two consecutive frames. Therefore, to make a gesture indicating digit '0' is made, we apply a hand translation motion to indicate the PDM model transition from the initial conjunctive model m_0 to the sign model m_0. A input image sequence of a gesture indicating a single-digit number 'n', will be processed and described by three consecutive PDM models m_0, m_n, and m_0. Hence, the PDM model m_0 plays two different roles: (1) m_0 is a conjunctive PDM model, if some sort of translation motion is detected and the hand has moved away from the original position. (2) m_0 is a sign PDM model, if no translation motion is found for a small time interval and then the hand has returned to the original position.

To give a more specific illustration of how to interpret the gesture through the PDM model transition sequence, we illustrate the following examples.

Example one: As illustrates in Fig. 4, to make a gesture indicating two-digit number 'jk', we can use a so-called *the gesture with translation motion*. This gesture can be described successfully by four PDM model transitions as: $m_0 \rightarrow m_j \rightarrow m_0 \rightarrow m_k \rightarrow m_0$.

Examples two: As shown in Fig. 5, to make another gesture indicate the same two-digit number 'jk', we can use a so-called *the gesture without translation motion*. This gesture can also be depicted by another PDM model transition sequence as: $m_0 \rightarrow m_j \rightarrow m_k \rightarrow m_0$. Here, the hand translation motion is unnecessary to imply the PDM model transition from m_j to m_k.

Example three: If we want to recognize a gesture of a double-digit number 'nn', then we may find the intermediate conjunctive PDM model m_0 between two sign PDM models m_n. The corresponding PDM model transition sequence is represented as $m_0 \rightarrow m_n \rightarrow m_0 \rightarrow m_n \rightarrow m_0$. There is only one kind of gesture, "*the gesture with translation motion*", that can be used to indicate a double-digit number.

Example four: However, we can only use one type of gesture (*the gesture with translation motion*) to represent the same number 'n0'. We may find the intermediate model m_0 between two sign models m_n and m_0, since there is noticeable hand movement between the sign model m_0 and the intermediate (or end) model m_0. This example can be represented by the PDM model transition sequence as $m_0 \rightarrow m_n \rightarrow m_0 \rightarrow m_0 \rightarrow m_0$, in which the second PDM model m_0 acts as an intermediate model.

From the above examples, we may find that we can use two kinds of gestures (with/without motion) to indicate the one-digit or two-digit numbers. However, for the double-digit number 'nn' or the number with digit '0', we can only apply *the gestures with translation motion* to avoid the misunderstanding between the sign model m_0 and the conjunctive model m_0. The rules can also be applied to other gestures indicating multi-digit numbers.

5. Experimental Results

We have developed a system to recognize a gesture representing any one-digit or two-digit number. First, we take 30 typical frames for training each HMM which indicates a specific PDM transition. There are five vectors (T=5) in each observation sequence indicating current information of the five fingers and three different states (N=3) for each model indicating the bending, half-bending and straight status of each finger.

We have tested four image sequences for each gesture. Most of the input gesture can be identified accurately. We have made the gestures, including the single-digit gestures, two-digit gestures with/without hand translation motion. These gestures are made in front of three different complex backgrounds (i.e., Fig. 3). The feature extraction results for the gestures of single-digit number (see Fig. 3) are very accurate that makes the corresponding recognition rate the highest. Since there are fewer model transitions in the transition sequence, the selected HMM's have better chance to indicate the correct PDM model transitions, and the new PDM models can be used to extract the features more precisely.

The results for the gestures of two-digit number without translation (see Fig. 6), and the two-digit number with translation (see Fig. 7) are not as good as the single-digit ones (see Table 1). However, they are acceptable. On the average, the identification rate of our gesture recognition system is about 85%. The translation information provides the system a very important additional information of determining the correct PDM model transition. Therefore, the recognition rate of the one-digit (or two-digit) gestures without translation is lower than the one-digit (or two-digit) gestures with translation. The reasons for miss-identification are (1) the pre-trained gray-level profiles stored in the database are not sufficient for coping with every new input gesture, (2) the number of principal components taken from the gray-level profile are not sufficient for all the unknown input gestures.

In our experiments, most of the model transitions detected by HMMs are accurate. The incorrect PDM model transitions are identified when (1) the observation vector (provided by the PDM-based hand-shape extraction process) is not accurate, (2) the movements of the raising or bending fingers are not coherent. For instance, the gesture of number '2', normally, requires both the index finger and the middle finger raised up-right almost at the same time. If the middle finger is raised faster by one frame or two, then the selected HMM may not indicate the correct PDM model transition. The error will influence the selection of all possible HMMs tested for the succeeding frames. If the current selected HMM is not correct, then the correct HMM for the next frame is normally not in the set of possible HMMs. The recognition rate of using HMM in the experiments to test the 120 image sequences (30 frames/sequence) is illustrated in Table 1.

6. Conclusions

We have developed a recognition system to extract the shape feature and recognize the gestures. In the experiments, we have proved that our method is more reliable than the previous methods when dealing with the problems of recognizing gestures before non-stationary backgrounds, complex backgrounds, and similar-intensity occlusion. We may easily extend our system to recognize the gestures indicating more-than-two-digit numbers.

Table 1. The overall gesture recognition rate.

Gesture Types	Number of test sequences	Recognition rate
Single-digit gestures with translation	120	93%
Single-digit gestures without translation	120	91%
Double-digit gestures with translation	120	84%
Double-digit gestures without translation	120	81%

REFERENCES

1. V.I. Pavlovic, et al., "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Trans. on PAMI, Vol.19, No.7, pp.677-695, 1997.
2. T. Takahashi et al., "A Hand Gesture Recognition Method and Its Application," Systems and Computers in Japan, vol. 23, No.3, pp.38-48, 1992.
3. C. L. Huang et. al., "Sign Language Recognition using Model-based Tracking and 3-D Hopfield Network" *Machine Vision and Application*, vol.10, pp.292-307, 1998.
4. A. Wilson et. al., "Recovering the Temporal Structure of Natural Gesture," ICAFGR, pp.66-71, Vermont, 1996.
5. J. Davis and M. Shah, "Visual Gesture Recognition," *IEE Proc.-Vis. Image Signal Process*, Vol.141, No.2, April 1994.
6. J. M. Rehg et al.,, "DigitEyes: Vision-based hand tracking for human-computer interaction,"Proc. of Int. Workshop on AFGR, 1995, Zurich, Switzerland.
7. T. Darrell and A. Pentland, "Space-Time Gestures" Proc. IEEE Conf. CVPR-93.
8. L. Campbell at al., "Invariant Conariant festures for 3-D gesture recognition," ICAFGR, pp.157-162, Vermont, 1996.
9. Y. Cui et al."Hand Sign Recognition from Intensity Image Sequences with Complex Backgrounds" Int. Conf. on AFGR, pp.259-264, Vermont, 1996..
10. T. F. Cootes et al.,"Active Shape Models - Their Training and Application,"Computer Vision and Image Understanding, Vol. 61, no.1, pp.38-59, 1995.
11. B. H. Juang and L. R. Labiner, "The Segmentation K-mean algorithm for estimation parameters of Hidden Markov Model," IEEE Trans. ASSP, Vol. ASSP-38, No. 9, pp.1639-1641, Sept. 1990.
12. L. R. Rabiner and B. H. Juang, "An introduction to hidden Markov models," IEEE ASSP Mag., vol.3, no.1, pp.4-16, Jan.1986.

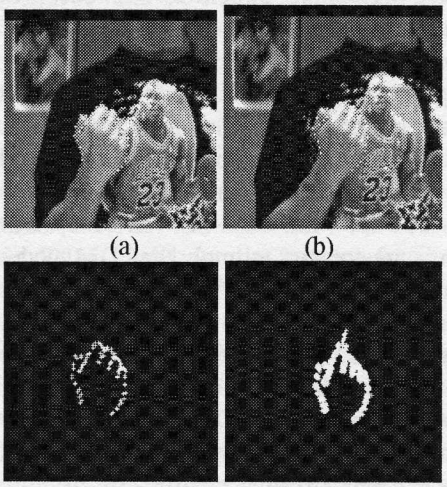


Figure 1.(a)(b) illustrate the hand shapes with labeled points. (c) shows the result that (b) is aligned with (a). (d) shows the aligned shape of a training set.

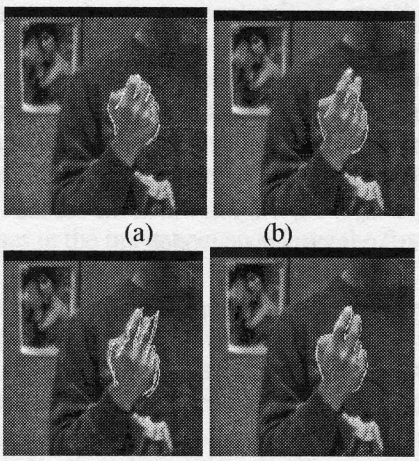
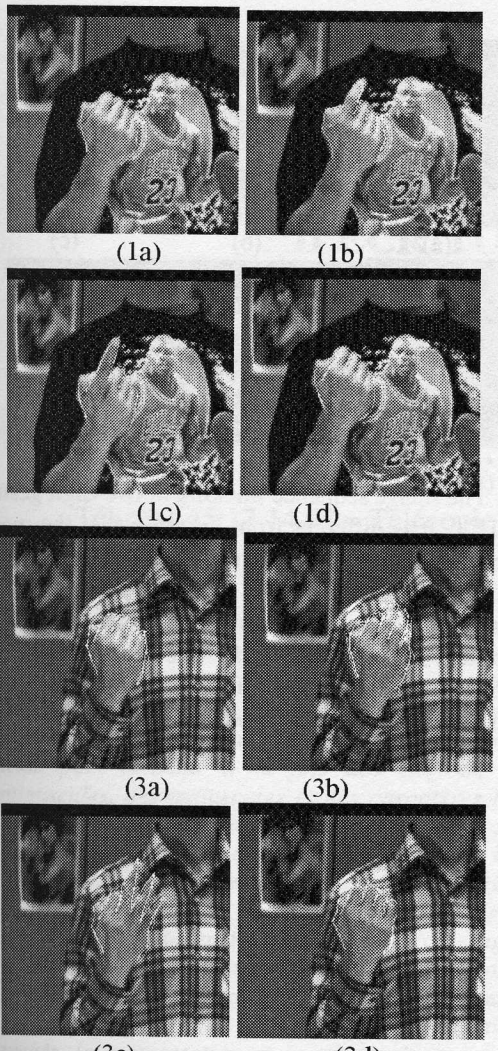


Figure 2. Illustration of the process of model transition. (a) shows the fitting of ith image frame using the model gesture-0, (b) when the flexible model meets the (i+1)th frame, the current model can not fit the hand shape exactly, (c) given an initial hand-shape, the model transition occurs, (d) (i+1)th frame is fitted exactly using the newly suggested flexible model.



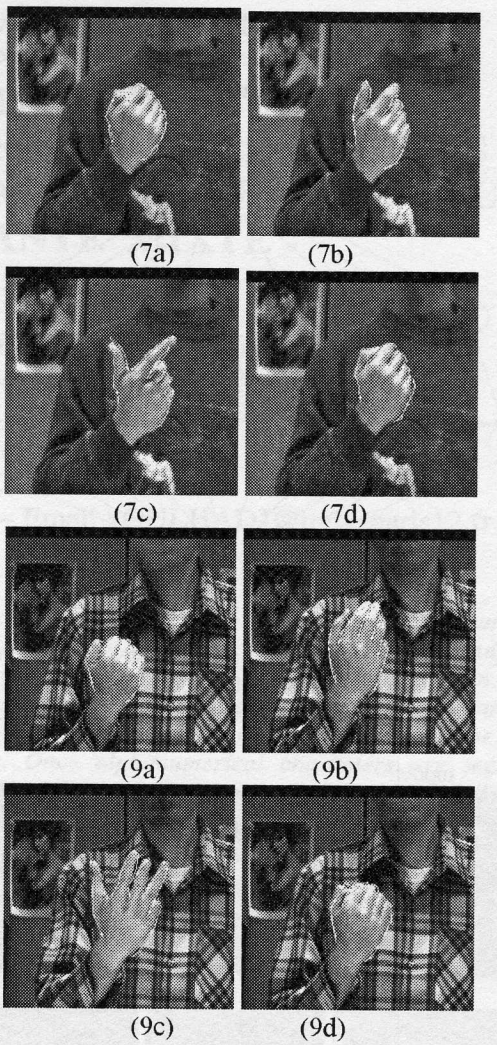


Figure 3. The image sequence tracking of the single-digit gestures "1", "3", "5", "7" and "9", the PDM model transition starts from m_0 to m_i, and finally returns m_0.

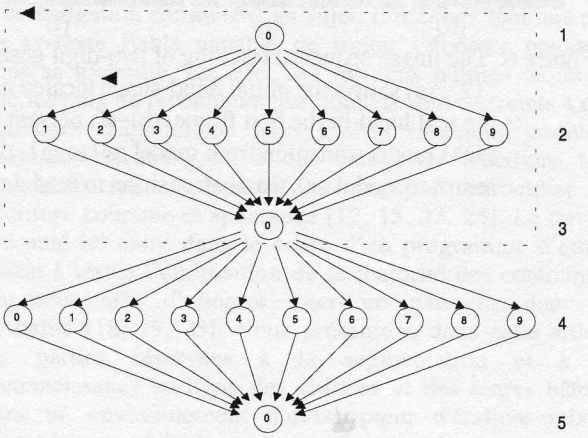


Figure 4. Illustration of the gesture recognition with the model transition having a global motion. The level 1 represents the initial model, the levels 2

and 4 represent the active model, the level 3 is the intermediate model, and the level 5 represents the final model.

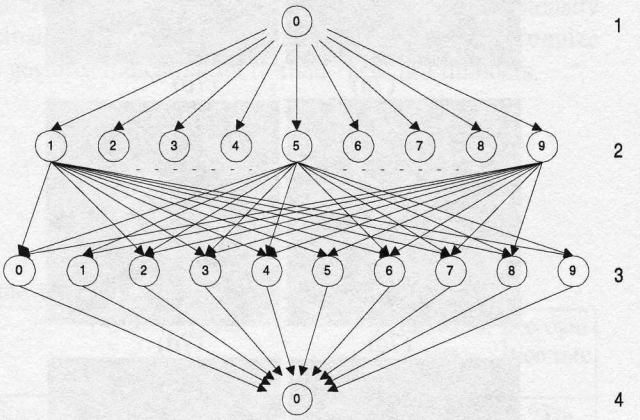


Figure 5. Illustration the gesture recognition without intermediate state of continuous gesture model transition. The level 1 represents the initial model, the level 2 and 3 represent the active model, and the level 4 represents the final model.

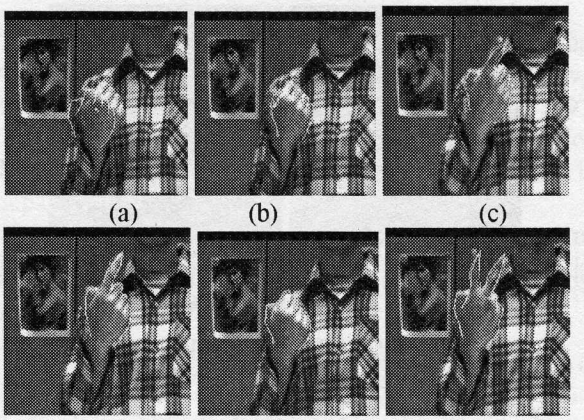


Figure 6. The image sequence tracking of two-digit gesture "12"; (a) shows the initial hand-shape located near the real hand in the first frame, (b)~(f) present the PDM model transition from model m0 to m1, then return to model m0, finally transition to model m2.

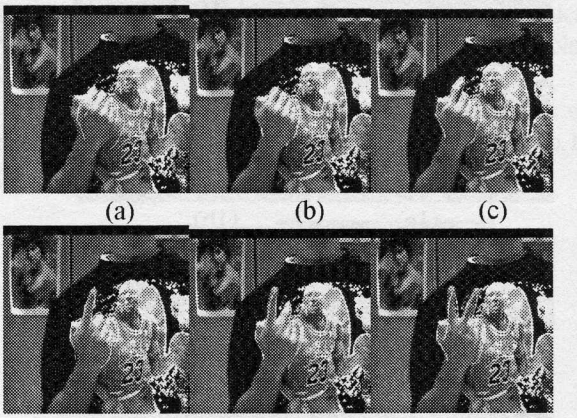


Figure 7. The image sequence tracking of two-digit gesture "12". It is different from Figure 6 that the model transition does not be return to m0 and the middle finger is straightened directly which can be described by m2. (a) shows the initial hand shape located near the real hand in the first frame, (b)~(f) present the state transition from model m0 to model m1, then transition to model m2.