Recognizing Human Action in Time-Sequential Images using Hidden Markov Model

Junji YAMATO *Jun OHYA** *Kenichiro ISHII*

NTT Human Interface Laboratories.
Take 1-2356, Yokosuka, Japan

Abstract

This paper proposes a new human action recognition method based on a Hidden Markov Model (HMM). We do not adopt model-based top-down approach because our purpose is not to reconstruct a geometric representation of the human body but to recognize human action. We use a feature based bottom up approach with HMMs that is characterized by its learning capability and time-scale invariability. To apply HMMs to our aim, one set of time-sequential images is transformed into an image feature vector sequence, and the sequence is converted into a symbol sequence by vector quantization. In learning human action categories, the parameters of the HMMs, one per category, are optimized so as to best describe the training sequences from the category. To recognize an observed sequence, the HMM which best matches the sequence is chosen. Experimental results of real time-sequential images of sports scenes show recognition rates higher than 90%. We confirm that the recognition rate is improved by increasing the number of people used to generate the training data. This indicates the possiblity of establishing a person independent action recognizer. This method can be easily applied to other fields because no domain knowledge is used.

1 Introduction

In recent years, motion related topics have been major concerns in computer vision. Considering moving objects in real scenes, human beings are very important recognition targets, since human action recognition algorithms can greatly contribute to the realization of automatic monitoring systems for various important applications. This paper introduces a new algorithm for recognizing human action from time-sequential images. The observed human action can be classified as one human action category.

Existing approaches related to human action recognition include the top-down methods based on geometric body reconstruction[1, 7, 16] and the bottom-up methods based on low-level image features[8, 4].

Most systems based on the top-down approach employ a geometrical model of the human body; human body parts are described as cylinders[3], super quadrics[1], and so on. Using spatio-temporal analysis[16], constraint propagations[7] or modal analysis[1], model parameters are determined from images. That means the posture of the human is extracted from the images and a representation is obtained. Model parameters are obtained from the sequence of images. Reconstructing the human body shape (i.e. extracting model parameters) yields rich and useful representations such as joint angle parameters if the reconstruction is successful. However, the reconstruction procedures are neither robust nor reliable for real images. This is because real images are usually too noisy to permit easy model fitting. Thus extracting successful high-level representations from images is very difficult. Therefore, almost no result on human action recognition by the model-based approach has been reported, while pattern classification techniques can be used to recognize a sequence of model parameters extracted from image sequences. Improving the robustness of reconstruction is important in this kind of approach, because failure in reconstruction prevents successful recognition. However, as the qualitative vision paradigm[5] points out, is reconstruction really necessary for human action recognition? Our purpose is to recognize human action from time-sequential images, not obtaining geometric representations of human bodies. Since we focus on recognition, we avoid reconstruction because the representation obtained by geometric reconstruction is not essential. Instead, we utilize low-level image features in a bottom-up manner.

Bottom up approaches which heuristically utilize low-level features extracted from real images have been the subject of various studies[4, 8]. Low-level image features such as the areas of human candidate regions have been used for counting the number of pedestrians in real scenes. In general, low-level features do not provide descriptions as rich as those of model based representation, but their extraction proceses are more robust than model fitting procedures.

We emphasize the robustness of the bottom-up approach , but existing studies have only been able to count the number of people, not recognizing their actions. Another problem with bottom-up approaches is that to describe action categories in a low-level representation is more difficult than in a high-level (model-based) representation. This is because relations between dimensions of the feature vector and the high-level description of category are not explicit. Furthermore, the dimension of the feature vector is usually too large to be understood intuitively.

There are two problem in applying the bottom-up approach. How can we enable the system to recognize more complicated actions? How can we describe the definitions of categories without a model-based high-level representation?

These problems can be eliminated with a learning procedure that uses low-level image features. Our goal is to classify observed low-level image feature sequences into human action categories. Usually, this kind of task is formalized as a supervised learning problem in the pattern classification field. Unlike most classical pattern classification techniques, the data to be classified is time sequential data.

To realize this learning capability for time-sequential image data, we employ the Hidden Markov Model (HMM) [15], which

*current adress:ATR Communication Systems Research Laboratories. Seika-cho Soraku-gun Kyoto, Japan

0-8186-2855-3/92 $3.00 ©1992 IEEE

can deal with time-sequential data and can provide time-scale invariability as well as learning capability for recognition. Although HMMs have been successfully used in speech recognition, HMMs have been applied to only a few problems in computer vision field: planar shape classification[2, 14], handwritten word recognition[6] and modeling eye movement[10]. In other words, HMMs have never been applied to motion recognition.

In our proposed approach, time-sequential images expressing human action are transformed to an image feature vector sequence by extracting a feature vector from each image. In our current implementation, the mesh feature[13] is used as the low-level image feature. Each feature vector of the sequence is assigned to a symbol which corresponds to a codeword in the codebook created by Vector Quantization[15]. The feature vectors of the sample data set for training are vector quantized. Consequently, the time-sequential images are converted to a symbol sequence. In the learning phase, the model parameters of the HMM of each category are optimized so as to best describe the training symbol sequences from the categories of human actions to be recognized. For human action recognition, the model which best matches the observed symbol sequence is chosen as the recognized category.

Section 2 details the HMMs and the recognition and learning procedures. Section 3 illustrates how a set of time-sequential images are converted to a symbol sequence that the HMMs can process. Section 4 gives experimental results using tennis action scenes. Section 5 summarizes our method.

2 Hidden Markov Model

2.1 Outline

HMMs, which have recently been applied with particular success to speech recognition, are a kind of stochastic state transit model[9]. HMMs make it possible to deal with time-sequential data and can provide time-scale invariability in recognition. Moreover, HMMs are characterized by their learning ability which is achieved by presenting time-sequential data to a HMM and automatically optimizing the model with the data.

A HMM consists of a number of states each of which is assigned a probability of transition from one state to another state. With time, state transitions occur stochastically. Like Markov models, states at any time depend only on the state at the preceding time. One symbol is yielded from one of the HMM states according to the probabilities assigned to the states. HMM states are not directly observable, and can be observed only through a sequence of observed symbols. To describe a discrete HMM[1] , the following notations are defined.

T = length of the observation sequence.
$Q = \{q_i, q_2, \ldots, q_N\}$:set of states.
N = number of states in the model.
$V = \{v_1, v_2, \ldots, v_M\}$:set of possible output symbols.
M =number of observation symbols.
$\mathbf{A} = \{a_{ij} | a_{ij} = \Pr(s_{t+1} = q_j | s_t = q_i)\}$: state transit probability, where:
a_{ij} is the probability of transiting from state q_i to state q_j.
$\mathbf{B} = \{b_j(k) | b_j(k) = \Pr(v_k | s_t = q_j)\}$: Symbol output probability, where:
$b_j(k)$ is the probability of output symbol v_k at state q_j.
$\pi = \{\pi_i | \pi_i = \Pr(s_1 = q_i)\}$ Initial state probability.
$\lambda = \{\mathbf{A}, \mathbf{B}, \pi\}$ complete parameter set of the model.
Using this model, transitions are described as follows:
$S = \{s_t\}, \ t = 1, 2, \ldots, T$: State s_t is the t th state (unobservable)
$O = O_1, O_2, ..., O_T$: Observed symbol sequence (length=T)

Figure1 illustrates the concept of the a with a transition graph. There are four states in this example indicated as circles. Each directed line is a transition from one state to another, where the transition probability is indicated by the character alongside the line.

Note that there are also transition paths from states to themselves. These paths can provide the HMM with time-scale invariability because they allow the HMM to stay in the same state for any duration.

Each state of the HMM stochastically outputs a symbol. In state q_j, symbol v_k is output with a probability of $b_j(k)$. If there are M kinds of symbols, $b_j(k)$ becomes an $N \times M$ matrix. The HMM outputs the symbol sequence $O = O_1, O_2, \ldots, O_T$ from time 1 to T. We can observe the symbol sequences output by the HMM but we can not observe the HMM states. The initial state of the HMM is also determined stochastically by the initial state probability π. A HMM is characterized by three matrices: state transit probability matrix $\mathbf{A}$, symbol output probability matrix $\mathbf{B}$, and initial state probability matrix π.

The parameters of $\mathbf{A}, \mathbf{B}$, and π are determined during the learning process described in 2.3. As described in 2.2, one HMM is created for each category to be recognized. Recognizing time-sequential symbols is equivalent to determining which HMM produced the observed symbol sequence. 2.2 and 2.3 explain the recognition and learning procedures are explained.

2.2 Recognition

To recognize observed symbol sequences, we create one HMM for each category. For a classifier of C categories, we choose the model which best matches the observations from C HMMs $\lambda_i = \{\mathbf{A}_i, \mathbf{B}_i, \pi_i\}, i = 1 \ldots C$. This means that when a sequence of unknown category is given, we calculate $\Pr(\lambda_i | O)$ for each HMM λ_i and select λ_{c^*}, where

$$c^* = \arg\max_i (\Pr(\lambda_i | O)) \tag{1}$$

Given the observation sequence $O = O_1, \ldots O_T$ and the HMM λ_i, according to the Bayes rule, the problem is how to evaluate $\Pr(O | \lambda_i)$, the probability that the sequence was generated by HMM λ_i. This probability is calculated by using the 'forward algorithm'[15].

The forward algorithm is defined as follows:

$$\alpha_t(i) \equiv \Pr(O_1, O_2, \ldots, O_t, s_t = q_i | \lambda). \tag{2}$$

$\alpha_t(i)$ is called the forward variable and can be calculated recursively as follows:

$$\alpha_t(j) = \{\sum_i \alpha_{t-1}(i) a_{ij}\} b_j(O_t) \tag{3}$$

$$\alpha_1 = \pi_i b_i(O_1) \tag{4}$$

Then

$$\Pr(O | \lambda) = \sum_{i \in S_F} \alpha_T(i) \lambda_{c^*}, c^* = \arg\max_i (\Pr(\lambda_i | O)) \tag{5}$$

[1] There are several types of HMM according to the formulations used. This paper refers only to discrete HMMs which output symbol sequences. In continuous HMMs, which output continuous feature vectors, quantization error caused by the vector quantization procedure can be avoided, but unfortunately, the learning process is more complicated.

We can calculate the likelihood of each HMM using the above equation and select the most likely HMM as the recognition result. Since the likelihood is calculated from the entire pattern length as described above, time scale variance, time shifts and some failure in vector quantization have little influence on the accuracy of determining the likelihood. The advantage of HMMs for time-sequential pattern recognition, which is robust to time scale variance and shift, results from this factor.

2.3 Learning

In the learning phase, each HMM must be trained so that it is most likely to generate the symbol patterns for its category. Training an HMM means optimizing the model parameters (A, B, π) to maximize the probability of the observation sequence $\Pr(O|\lambda)$. The Baum-Welch algorithm is used for these estimations.

Define:

$$\beta_t(i) \equiv P(O_{t+1}, \ldots, O_T | s_t = q_i, \lambda) \qquad (6)$$

$\beta_t(i)$ is called the backward variable and can also be solved inductively in a manner similar to that used for the forward variable $\alpha_t(i)$.

$$\begin{aligned} \gamma_t(i) &\equiv P(s_t = q_i | O_1, \ldots, O_T, \lambda) && (7) \\ &= \frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)}. && (8) \end{aligned}$$

$$\begin{aligned} \xi_t(i,j) &\equiv P(s_t = q_i, s_{t+1} = q_j | O_1, \ldots, O_T, \lambda) && (9) \\ &= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)}. && (10) \end{aligned}$$

Using these equations, HMM parameters λ can be improved to $\bar{\lambda}$. The re-estimation equations from $\lambda = (A, B, \pi)$ to $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$ are:

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}; \qquad (11)$$

$$\bar{b}_i(k) = \frac{\sum_{t \in \{t | O_t = v_k\}} \gamma_t(i)}{\sum_{t=1} T \gamma_t(i)}; \qquad (12)$$

$$\bar{\pi}_i = \gamma_1(i). \qquad (13)$$

Learning converges if $\bar{\lambda} = \lambda$. The Baum-Welch algorithm does not always find the global maximum, but it does find the local maximum of $\Pr(O|\lambda)$. Our experience shows that this is not a significant problem.

3 Applying HMM to time-sequential images

To apply HMMs to time-sequential images $I = \{I_1, I_2, \ldots I_T\}$, the images must be transformed into symbol sequences O(**Figure2**) in the learning and recognition phases. From each frame I_i of an image sequence, a feature vector $f_i \in R^n$, $(i = 1, 2, \ldots, T, n$ is the dimension of the feature vector) is extracted, and f_i is assigned to a symbol v_j chosen from the symbol set V. It is necessary to associate the symbol set V with the feature space R^n. For this, the feature space is divided into clusters by a pattern classification technique, and the symbols are assigned to the clusters. We used vector quantization[15] here, as is usual in HMM applications. For vector quantization, codewords $g_j \in R^n$, which represent the centers of the clusters in the feature R^n space, are needed. Codeword g_j is assigned to symbol v_j . Consequently, the size of the code book equals the number of HMM output symbols.

Each feature vector f_i is transformed into the symbol which is assigned to the codeword nearest to the vector in the feature space. This means f_i is transformed into symbol v_j if $j = \arg\min_j d(f_i, g_j)$ ($d(x, y)$:distance between x and y).

Image I_i is transformed into the symbol which is assigned to the codeword nearest to feature vector f_i, and $O_i(= v_j)$ is yielded.

We use mesh features[13] as the feature vectors because they were successfully applied to complex 2D patterns such as multifont characters. **Figure3** explains the calculation of mesh features. Each binarized image ($M_I \times N_I$ pixels) is divided into $M_M \times N_M$ pixel meshes. The ratio of black pixels in each mesh becomes an element of the feature vector,

$$f(i,j) = number\ of\ black\ pixels(i,j)/(M_M \times N_M) \qquad (14)$$

Thus, the feature vector sequence F extracted from an image sequence I is transformed to a symbol sequence O by extracting the mesh feature vector of each frame and vector quantizing them.

These transformations have the merit that they are applicable to various images without manual tuning because after extracting the humän area, this method uses no parameters except mesh size, which can be decided from the size of the human.

As described in Section 2, the symbol sequences obtained from the above procedures are used for both the recognition and learning phases. In recognition, the symbol sequence O obtained from an observed image sequence is substituted into Eqs.(1)-(5), and the recognition result is obtained as the category C^* which maximizes Eq.(5). In learning, the symbol sequences are obtained from training with time-sequential image data and HMM's parameters are optimized for each category by Eqs.(11)-(13).

4 Experimental results and discussions

4.1 Experimental conditions

We tested our algorithm on real time-sequential images. Our experiment used tennis actions. The categories to be recognized were six tennis strokes: 'forehand stroke','backhand stroke','forehand volley','backhand volley', 'smash', and 'service'.

Three persons performed each of the six tennis actions 10 times. The performances were captured by a TV camera (NTSC, 30 frames/second) and digitized into 200 × 200 pixel 256 gray-level images. **Figure4** shows an original image sequence of the tennis action: 'forehand volley', where every fourth frame is displayed.

As shown in **Figure4**, the original image sequence contains a complicated background, and the positions of the player moves during sequence. Thus, human area extraction and tracking are needed. We used the image preprocessing operations described below: 1)Preparing background image(**Figure5**-(b)) for its original action images(**Figure5**-(a)).

2)Blurring both images with a low pass filter.

3)Extracting human area (**Figure5**-(c)) by the following conditions:

$if |I_a(x,y) - I_b(x,y)| < th,\ I_e(x,y) = 0$
$else\ \ I_e(x,y) = I_a(x,y).$

I_e:human extracted image, I_a:original image
I_b:background image, th:threshold

We then binarized the extracted images so that the white and black pixels corresponded to human areas and background respectively. **Figure7** shows binarized image examples of the six tennis actions.

The mesh feature extraction described in Section 3 was performed on the binarized images. The size $M_M \times N_M$ of the mesh feature vector f_i in Eq.(14) was 8 × 8pixels; consequently, the dimension n of f_i was 625. As the positions and sizes of human areas varied in the binarized images, the mesh center was placed at the center of gravity of each human area to avoid the influence of grossbody displacement, in other words, to normalize the positions. The body size was also normalized by scaling each frame of the binarized images so as to equalize the mean radii of human areas.

For vector quantization, we selected 12 images for each category as codewords in the codebook for the experiment. Thus, the number of HMM symbols was 72; symbols 0 to 11 represent 'backhand stroke', 12 to 23 the 'backhand volley', 24 to 35 the 'service', 36 to 47 the 'smash', 48 to 59 the 'forehand stroke', and 60 to 71 the 'forehand volley'.

As shown in **Figure6**, symbols were correctly assigned to the binarized images, where each assigned symbol is one of the symbols representing 'forehand volley'.

In the HMMs used in our experiment, the number N of states Q was set to 36. As the durations of the actions were not constant, the length of the observed symbol sequences varied between 23 and 70. In training the HMMs, while the likelihood converged after about 100 iterations, we performed 150 iterations for each HMM.

4.2 Experiment 1

4.2.1 Experiment details

We recognized the actions performed by three people using HMMs which were trained by the data of the subject. Three subjects(person A, B, and C) performed each of the six actions ten times. Five sequences were used to train the HMMs while the remaining five sequences were used to test recognition performance.

4.2.2 Results

The recognition results are shown in **Table1** and **Table2**. **Table1** shows the likelihood of 'forehand volley' with six HMMs trained with the data of subject C. The likelihood of the correct category is the highest.

Table2 shows the recognition rates of the three subjects with 10 different combinations of five training sequences out of the 10 trials; that is, for each person, there are 300(= 10 × 5 × 6) test data. The average for the three subjects was 96.0%.

Recognition performance worsens as the number of training patterns decreases. We also tested the recognizer constructed by HMMs trained by three training patterns in ten combinations. For one combination, the recognition rate was 100%(42/42) but for some other combinations it fell to 78.5%.

The performance of our recognition system depends not only on the number of training patterns but also how well the training patterns represent the category. When the number of training patterns is small, the recognition rate becomes more unstable. This is because the HMM parameters critically depend on the selection of training patterns.

To construct a robust recognition system, appropriate training patterns are important. This means training patterns should cover the maximum test pattern scatter possible. If covering the scatter with one HMM is difficult, we should divide the category into sub-categories.

4.3 Experiment 2

4.3.1 Experiment details

In experiment (2), training and test subjects were completly different. The HMMs were trained with the sequences of one or two subjects and tested with those of other subjects. The HMMs were trained using 10 mixed data sets collected from one or two subjects, and tested with those of other subjects.

In this experiment, learning subjects and recognition subjects were different not only in HMM processing but also vector quantization. The code book of vector quantization was constructed using only training data. This experiment is more appropriate for estimating the usefulness of this method in real situations because a real world recognition system should be person independent. To achieve this, data of many people must be collected for learning. This reduced-scale experiment used mixed training data collected from more than one person to recognize another person's actions.

4.3.2 Results

In experiment (2), as shown in **Table3**, the recognition rate of the HMMs were not as good as recorded in experiment(1). This is because test pattern subjects and training pattern subjects were different. Each person has some uniqueness in his actions, but the variance range is limited such that humans can recognize. Thus we can improve the recognition rate by using mixed training data. The recognition rate of the HMMs trained with the data of two subjects was improved to 70.8%. Thus performance can be improved by collecting more training patterns which are suitable for representing the category.

Our current implementation uses the mesh feature for its simplicity. However, this does not match the human posture space well because it is sensitive to position displacement. The HMM learning capability is fairly well developed but we should further improve the feature extraction technique.

5 Conclusion

This paper has presented a Hidden Markov Model based approach for human action recognition from a set of time-sequential images. In our algorithm, a mesh feature vector sequence extracted from time-sequential images is converted to a sequence of symbols which correspond to codewords in the codebook created by vector quantization. In learning, symbol sequences obtained from training image sequence data are used to optimize HMMs for action categories. In recognition, a symbol sequence from an observed image sequence is processed by HMMs, and the recognition result is determined as the category which best matches the observed sequence.

The main experimental results using tennis actions performed by three subjects are as follows. When training data and test data are those of the same subject, a recognition rate of over 90% was achieved. On the other hand, when training data and test data are those of different subjects, the performance drops. However, the recognition rate was improved by mixing the data from two subjects for learning.

These results show that our method is promising to realize human action recognition for various applications such as finding shoplifters in department stores and dangerous behavior in a kindergarten. To improve our current implementation, we will try a large scale experiment and further refine feature extraction.

This method basically deals with 2D images but it can be extended to 3D object actions using , for example, aspect graphs in assigning symbols or making HMMs. HMMs can be applied to action recognition in other various ways. Ishikawa[11] uses HMMs to recognize words by analyzing sound and the height of lip data. This shows the applicability of HMMs to multi-modal time-sequential pattern recognition. In other words, sensor fusion problems. Future improvements in recognition accuracy and recognizing complicated actions without any reduction in robustness are now being sought.

Acknowledgements

The authors would like to thank Dr.Yukio Kobayashi and Takaya Endo of NTT Human Interface Laboratories for their encouragement and support for this work. The authors also thank Akihiro Imamura of NTT Human Interface Laboratories for his helpful advice on HMMs. Finally, thanks are due to laboratory members for maintaining a positive research environment and for serving as the models.

References

[1] B.Horowitz and A. Pentland. "Recovery of Non-Rigid Motion and Structure". In *Proceeding of CVPR*, pp. 325–330, 1991.

[2] Yang He and Amlan Kundu. "Planar Shape Classification using Hidden Markov Model". In *Proc.CVPR*, pp. 10–15, 1991.

[3] David Hogg. "Model-based vision:a program to see a walking person". *Image and vision computing*, Vol. 1, No. 1, pp. 5–20, Feb 1983.

[4] B. W. Hwang and S. Takaba. "Real-Time Measurement of Pedestrian Flow Using Processing of ITV Images". *Trans. IEICE*, Vol. J66, No. 8, pp. 917–924, 1983. (in Japanese).

[5] J.Aloimonos. "Purposive and Qualitative Active Vision". In *Proceeding of ICPR*, pp. 346–360, 1990.

[6] A. Kunda, Y. He, and P. Bahl. "Handwritten Word Recognition:a Hidden Markov Model Based Approach". *Pattern Recognition*, pp. 283–297, may 1989.

[7] J. O'Rourke and N.Badler. "Model-Based Image Analysis of Human Motion Using Constraint Propagation". *IEEE Trans. PAMI*, Vol. PAMI-2, No. 6, pp. 522–536, Nov 1980.

[8] E. Oscarsson. "TV-Camera Detecting Pedestrians for Traffic Light Control". In *Technological and Methodological Advances Measurement (ACTA IMEKO)*, volume 3, pp. 275–282, 1982.

[9] L.R. Rabiner and B.H. Juang. "An Introduction to Hidden Markov Models". *IEEE ASSP MAGAZINE*, pp. 4–16, Jan 1986.

[10] R.D.Rimey and C.M.Brown. "Selective Attention as Sequential Behavior:Modeling Eye Movements with an Augmented Hidden Markov Model". In *Proc. DARPA Image Understanding Workshop*, pp. 840–849, 1990.

[11] T.Aono and M.Ishikawa. "Sensor Fusion using Stochastic Process". In *2nd Symposium of Autonomous Distributed Systems*, pp. 115–118, 1991. (in Japanese).

[12] D. Terzopoulos and D. Metaxas. "Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics". *IEEE Trans. on PAMI*, Vol. 13, No. 7, pp. 703–714, 1991.

[13] M. Umeda. "Recognition of Multi-Font Printed Chinese Characters". In *Proc. 6th ICPR*, pp. 793–796, 1982.

[14] W.D.Mao and S.Y. Kung. "An Object Recognition System Using Stochastic Knowledge Source and VLSI Parallel Architecuture". In *Proc. 10th ICPR*, pp. 832–836, 1990.

[15] X.D.Huang, Y. Ariki, and M.A.Jack. *"Hidden Markov Models for Speech Recognition"*. Edingurgh Univ. Press, 1990.

[16] M. Yamamoto. "Human Motion Analysis Based on A Robot Arm Model". In *Proceeding of CVPR*, pp. 664–665, 1991.

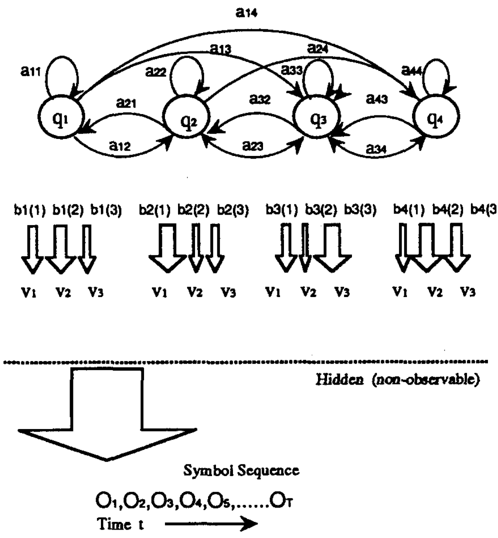


Figure 1: HMM Concept

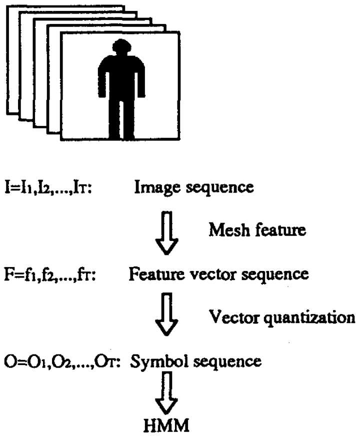
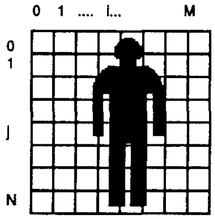

I=I1,I2,...,IT: Image sequence
Mesh feature
F=f1,f2,...,fT: Feature vector sequence
Vector quantization
O=O1,O2,...,OT: Symbol sequence
HMM

Figure 2: Processing flow



$f=(a_{00},a_{01},\ldots,a_{ij},\ldots a_{MN})$

a_{ij}=number of black mesh(ij)/M_mN_m

Figure 3: Mesh feature

Table 1: Likelihood (backhand volley)

HMM	log-likelihood
backhand volley(correct cat.)	-29.8715
backhand stroke	-431.7681
forehand volley	-233.9752
forehand stroke	-442.1949
smash	-221.7908
serve	-466.5597

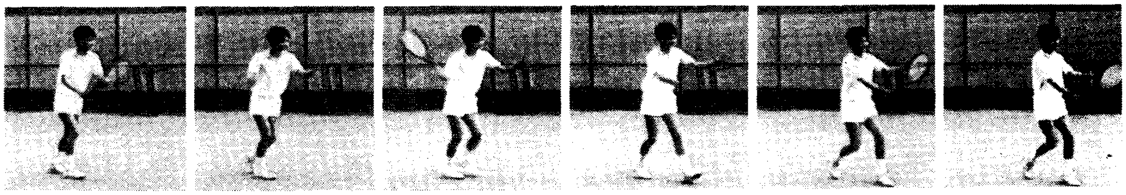


Figure 4: Example of tennis action (forehand volley)

Table 2: Recognition rate (experiment 1)

Player	rate(%)
player A	90.66 (272/300)
player B	97.33 (292/300)
player C	100.00 (300/300)
Average	96.00

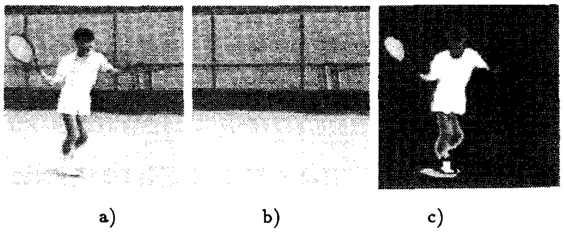
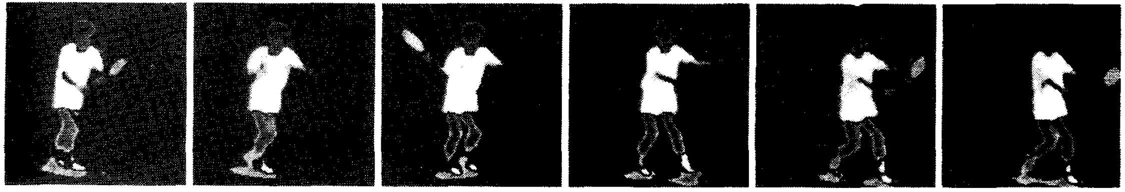


Figure 5: Human area extraction
a)original, b)background, c)extracted

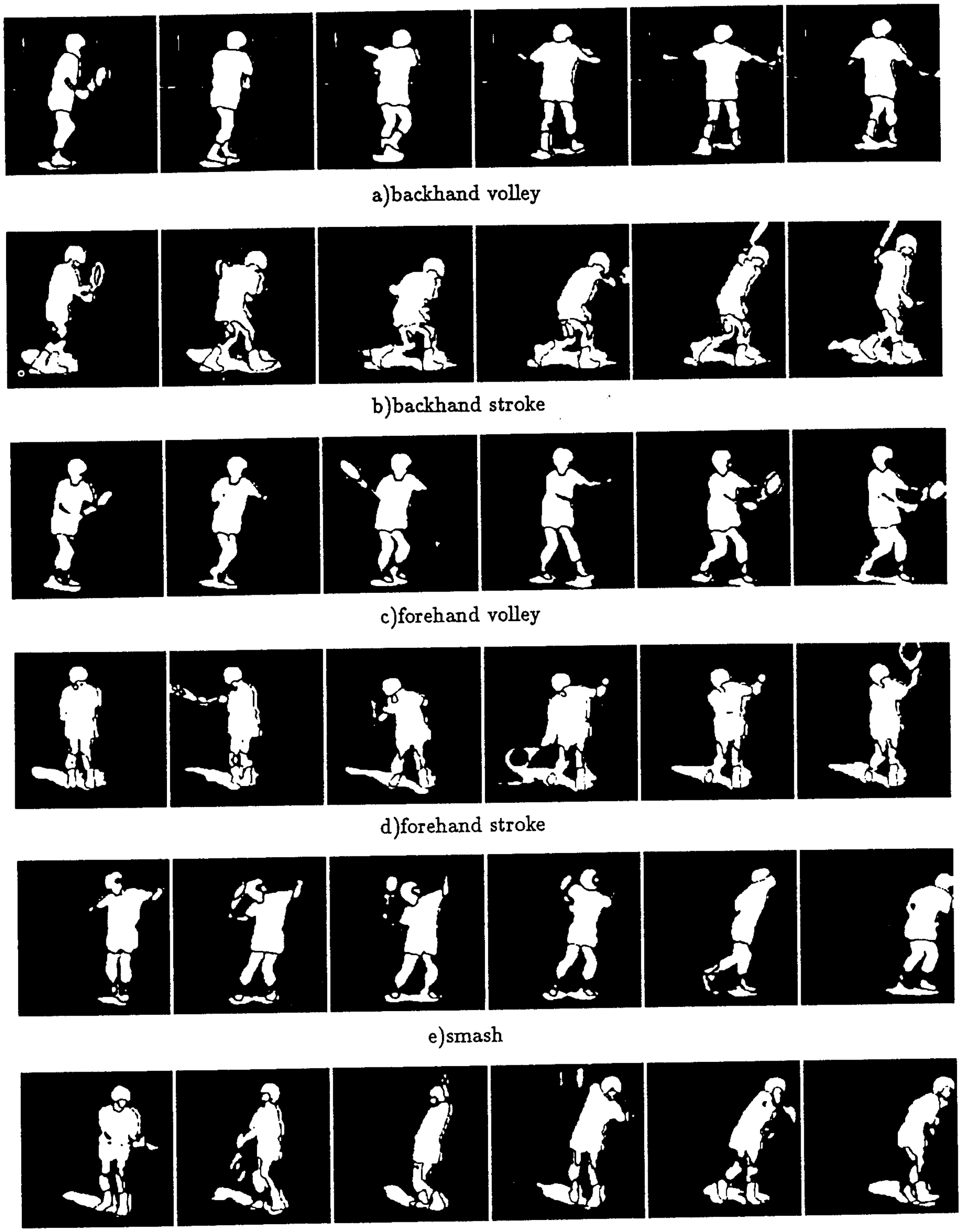
Table 3: Recognition rate (%) (experiment 2)

	Training data player			
Test data player	A	B	A+B	C
C	61.2	66.8	70.8	100.0



Symbol sequence	$\underline{60}$ 61 61 62 $\underline{62}$ 62 63 63 $\underline{64}$ 64 65 66 $\underline{66}$ 66 67 68 $\underline{68}$ 69 69 70 $\underline{70}$ 70 71 71

Figure 6: Example of extracted tennis action and symbol sequence(forehand volley).
(Underlined symbol is assigned to frame of above figure.)



a)backhand volley

b)backhand stroke

c)forehand volley

d)forehand stroke

e)smash

f)service

Figure 7: Sample actions