Hand Shape Estimation Using Sequence of Multi-Ocular Images Based on Transition Network

Yasushi HAMADA, Nobutaka SHIMADA, Yoshiaki SHIRAI
Dept.of Computer-Controlled Mechanical Systems, Osaka University
2-1 Yamadaoka Suita-shi, 565-0871 Japan
hamada@cv.mech.eng.osaka-u.ac.jp

Abstract

This paper presents a method of hand posture estimation from silhouette images taken by multiple cameras. For each image, we extract a feature vector from the silhouette contour of the hand. We construct an eigenspace by the feature vectors extracted from the hands of various postures. The feature vectors projected into the eigenspace are registered as models. The matching criterion of each images is defined as the distance to the model. The hand shape is estimated by retrieving the registered model well-matching to the input. For effective matching, we define a shape complexity for each image to see how well the shape feature is represented. For a set of input images taken by multiple cameras at each time, the total matching criterion is evaluated by combining the matching criteria of the set of images using the shape complexities.

For rapid processing, we limit the matching candidate by using the constraint on the shape change. The possible shape transition is represented by a transition network. Because the network is hard to build, we apply offline learning, where nodes and links are automatically created by showing examples of hand shape sequences. We show experiments of building the transition networks and the performance of matching using the network.

1 Introduction

Recently image-based human interfaces and understanding the hand gestural languages have attracted increasing attentions as an alternative to traditional input devices like mouses or keyboards. Such attempts previously proposed are approximately divided into two categories.

The first category is the 3-D model-based approach including the model fitting methods [1] and "Estimation by Synthesis(ES)" methods [2, 3] which match possible postures generated from a given 3-D shape model and search for the postures best-matched to the input image. While these methods are effective for estimation of arbitrary hand postures, they often require much computation.

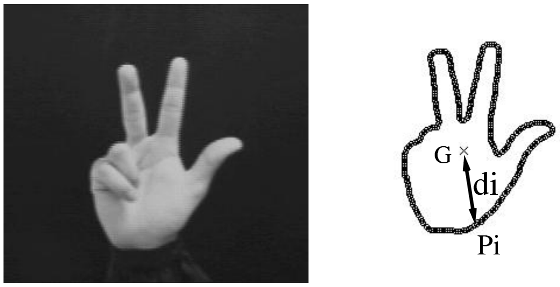
The second category directly matches the image features to those of models. The methods of this category [4, 5, 6, 7, 8] register the image appearances or the image features in the learning sequences, and then the input sequence is classified into one of the registered sequence. For estimation of a limited set of hand postures, only useful models are registered. Moreover, computation is usually less because 3-D shapes are not estimated.

For estimation of hand shapes in a gesture sequence, however, the first category is more effective because it is able to limit the search space by the constraint of the joint angles or by that of the velocity. The second category, on the other hand, has to try to match every models. This problem was solved by applying the Hidden Markov Model (HMM). However, a sequence model has to be built for every gesture sequence.

This paper proposes a method of matching a given hand posture just like the second category, while limiting the candidates by a transition network. The transition network has nodes which represent typical hand shapes and links which represent possible hand shape changes. The network alone represents the transition of all possible gestures, and is built automatically during a learning phase. In speech and gesture recognition [9], the transition network is used for integration of speech and gesture.

First, in this paper, a basic matching method is described. Because matching with images taken by monocular camera is often ambiguous, we use a set of images taken by multiple cameras. We determine the features for a set of images to estimate the hand posture. We collect various hand images to make the model of the postures. A silhouette is extracted from each image and the feature vector is computed as a sequence of the distances from the center of the silhouette to the contour points. The eigenvectors are determined from all feature vectors. The feature vectors projected into the eigenspace are registered as models.

The matching criterion of each images is defined as the distance to the nearest model. The hand shape is estimated by retrieving the registered model well-matching to the in-



(a) A silhouette

(b) Extracted contour feature

Figure 1: Feature Extraction

put. For effective matching in each frame of the image sequence, we define the shape complexity for each image to see how well the shape feature is represented. For a set of input images taken by multiple cameras at each time, the total matching criterion is evaluated by combining the matching criteria of the set of images using the shape complexities. It is easy to apply this approach for any number of cameras, because the total matching criterion does not depend on the number and the relative orientation of cameras. Thus the best-matched image is obtained for the set of images.

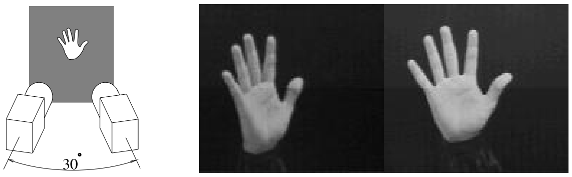
Next, an effective hand posture matching of a gesture sequence is described. For a given application, we may be able to limit the matching candidate by the constraint on the shape change. That is, the next shape is confined to a set of possible models. The possible transition is represented by a transition network. Because the transition network is hard to build, we apply offline learning, where nodes and links are automatically created by showing examples of hand shape sequences. It is important to merge similar nodes in different image sequences so that the transition obtained in a sequence can be used at the similar node in other sequences.

2 Feature Extraction

2.1 Contour Feature

For simplicity, the hand region is assumed to be brighter than the background and the clothes so that the hand region is easily obtained (Figure 1(a)).

We use a set of hand images taken by multiple cameras fixed laterally in front of the user (Figures 2(a) and 2(b)). For each image, hand region is extracted and then its area S and center of gravity G are computed. Then 256 points $P_i (i = 1 \cdots 256)$ are sampled on the contour of the region so that they are placed at a constant interval (Figure 1(b)).



(a) Camera layout

(b) Captured images

Figure 2: Multiple Cameras Environment

Scale-normalized distance r_i is obtained by

$$r_i = \frac{d_i}{\sqrt{S}} \quad (1)$$

where d_i denotes the distance between G and P_i. This is the shape feature independent of the translation and scale change.

Because the sequence of features depends on rotation, the realignment of the elements is necessary. We select the most significant peak or valley as the start point of r_i. Then realigned $\{r_i\}, x = \{r_{a_1}, \cdots, r_{a_{256}}\}^T$, is obtained as the feature vector. Figure 3 shows extracted feature vectors for two similar hand images.

2.2 Building of Eigenspace

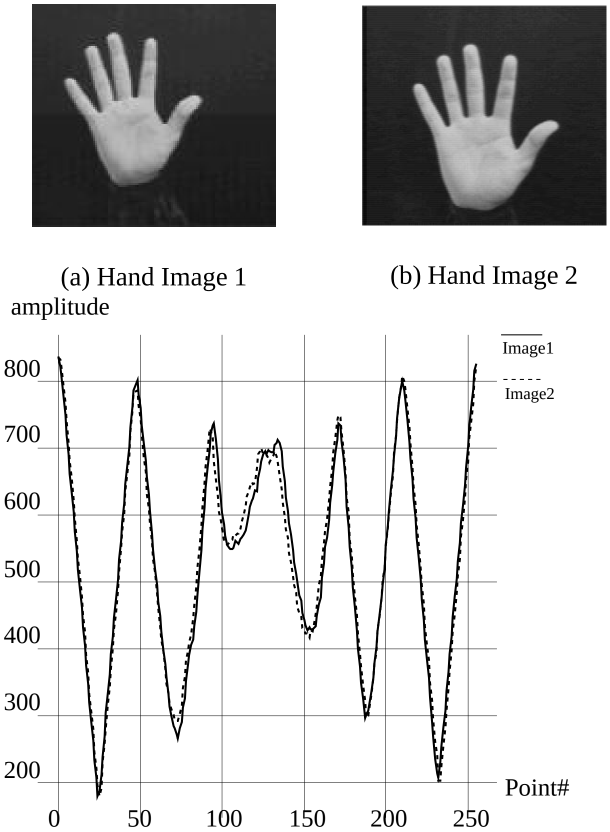
In the offline learning phase, hand shape images taken by cameras are registered as the model images. For compressing feature vectors and reducing computation, the eigenspace of the feature vectors is constructed. The bases of the eigenspace are computed by selecting k principal eigenvectors $E = [e_1, \cdots, e_k]$ obtained by Principal Component Analysis. The compressed feature vectors $g_n = E^T(x_n - \bar{x})(n = 1, \cdots, M)$ are stored in the database.

In the online shape estimation, scale-normalized distances $\{r_i\}$ are similarly obtained. For normalization of the rotation, we select start point candidates as the significant peaks and valleys. For robust normalization, we select L candidates and evaluate each of them. For the jth candidate $(1 \leq j \leq L)$, feature vector $y_j = \{r_{b_{j1}}, \cdots, r_{b_{j256}}\}^T$ is generated as the jth realigned $\{r_i\}$.

Each y_j is projected into the eigenspace and then the compressed feature vector of the input is computed as

$$h_j = E^T(y_j - \bar{x}). \quad (2)$$

All candidates are matched to the model features to determine the best-matched model.



(a) Hand Image 1

(b) Hand Image 2

(c) Extracted Feature Vectors: They are well-matched together.

Figure 3: Feature Vectors of Similar Images

3 Appearance Matching Criterion

The basic matching criterion for L feature vectors and the model image n is

$$d_n = \min_{j=1,\cdots,L}(||h_j - g_n||). \tag{3}$$

The best-matched model is determined as $\arg\min_{n=1,\cdots,M}(d_n)$.

We use a set of images taken by T cameras. The matching scheme for T images is described in this section.

3.1 Matching based on Shape Complexity

Since T input feature vectors are obtained from a set of T images, matching criteria are computed by equation(3). Note that some hand shapes are difficult to discriminate from a single silhouette.

A problem is how to integrate them to determine the best model.

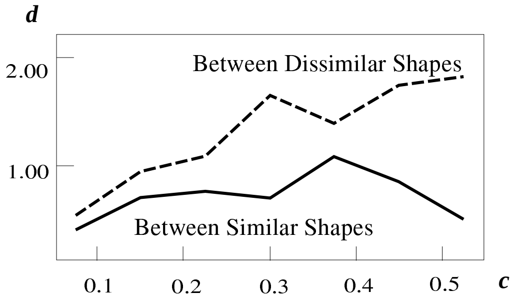


Figure 4: Relation between the matching criterion and the shape complexity

A simple method is to use the average of the criteria. This method, however, may be influenced by a silhouette which is not suitable to determine a unique shape.

The more clearly observed the contour of fingers is, the more effective for matching the image is. Detecting each finger has the problems of detection failure and much computation. An approach [10] selects a camera which is most perpendicular to the hand palm. However such a camera does not always exist.

We conjecture that the more clearly observed the contour of fingers is, the more complex the shape is. The complexity is represented by the sum of the curvature of the contour, because the curvature is large along the contour of a finger tip and of a valley between fingers. In this paper, the complexity of the shape feature is defined as

$$c = \sum_{i=1}^{256} \frac{|r_{i+2a} - 2r_{i+a} + r_i|}{a^2} \tag{4}$$

where $r_i = r_{i-256}$ if $i > 256$, and a is an experimentally determined constant. $a = 10$ was used in the experiments.

Figure 4 shows the relation between the matching criterion d and the complexity c for 1026 inputs and 260 models. A solid line shows the criterion of similar shapes, and a broken line shows that of dissimilar shapes. If the complexity is large, the difference of two criteria is large. And if the complexity is small, the difference of them is small. This means that the larger the complexity is, the more reliably the correct model is determined.

If the complexity of one image is much more than others, only the former may be used for matching. In general, each of the images is assigned to a weight $w_t(t = 1,\cdots,T)$ according to the complexity, and the best model is determined by the weighted average of the criteria.

Let the complexity of the images be c_t, and let the criterion of them be $d_{n,t}$. The weighted average is defined as

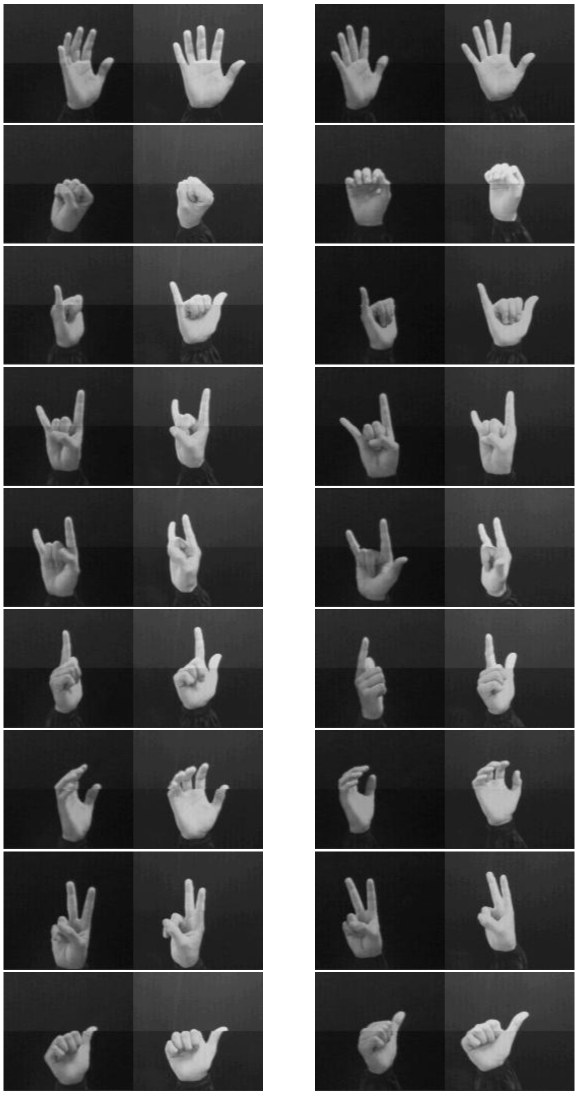


Figure 5: Matching results (from left side, input images(left, right) and matched model images(left and right)

$$d_n = \sum_{t=1}^{T} w_t d_{n,t} \tag{5}$$

where $w_t = c_t / \sum_{t=1}^{T} c_t$.

3.2 Estimation Experiment using Binocular Images

We used a pair of images taken by two cameras in the experiments.

First the eigenspace is built using 260 image pairs of typical hand shapes. By out experiments, the performance saturates with 12 eigenvectors. In the following sections, 12 dimensional eigenspace is used. 260 image pairs are registered as 260 models.

By a estimation experiment with 1026 input images, estimation rate 88.3% was obtained.

Examples of input images and the estimation results are shown in Figure 5.

4 Transition Network

For recognition of gestures or a hand sign language, a sequence of hand shapes should be recognized. For a given set of gestures, a limited set of shape changes is allowed.

For efficient matching, possible shape changes are stored in a transition network, where nodes represent typical shapes and links represent possible transitions. Generally such a transition network is difficult to build because it takes much efforts to teach all possible transitions.

This section describes a method to build a transition network by showing a limited number of gesture sequences and effective tracking of a shape sequence using the network.

4.1 Building of Transition Network

The transition network is represented in the eigenspace which is built for estimation of hand shapes described in the previous sections.

For learning the network, sample sequences are taken and the network is incrementally built. For a given sample sequence, a sequence of feature vectors is first created in the eigenspace. Each vector is then matched to model nodes using the criterion d described in section 3.

If d is less than a threshold d_{thres}, it is matched to the model node. This threshold is determined so that most of the images which correspond to a model have the criterion value less than this threshold (d_{thres} = 1.6 in this paper so that 97% of them satisfy this condition). If the matched node is the same as the previous node, the hand shape is regarded as the same as the previous one. In this case, no transition takes place.

If the matched node is different from the previous one, the matched node is linked to the previous one. By this operation, a new possible transition is automatically created without actual samples. Figure 6 depicts this case.

If d is greater than d_{thres}, it is regarded as a new shape. Then the new shape becomes a model node, and the node is linked to the previous one.

By repeating this operation for sample sequences, typical hand shapes and possible transitions are represented in the transition network. Note that each node corresponds to the set of images taken by multiple cameras and the shape features (in eigenspace).

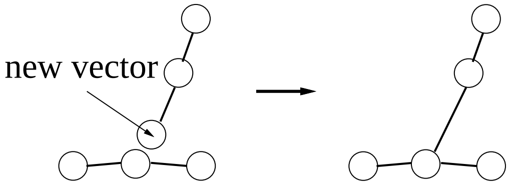


Figure 6: Transition network

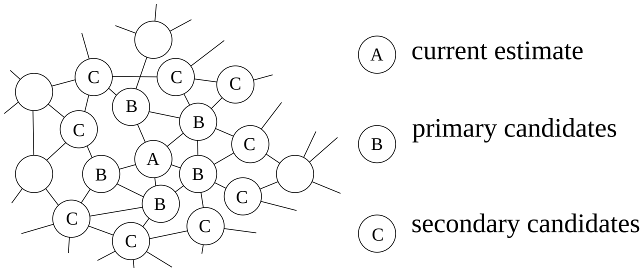


Figure 7: Limitation of matching candidate with transition network

4.2 Shape tracking using Transition Network

In shape tracking, the transition network is utilized to limit the shape candidates. Given the previous estimation result, the previous node and its neighbors(for example, node A and Bs in Figure 7) are selected as the primary shape candidates for the succeeding image . We search the primary candidates for the matching model. If the matching model is not found, the neighbor nodes of the primary candidates are selected as the secondary candidates (Cs in the figure). This operation is repeated until the matching model is found. If no matching models are found, all nodes are selected as the matching candidates of the next shape. Thus the computation cost is much reduced.

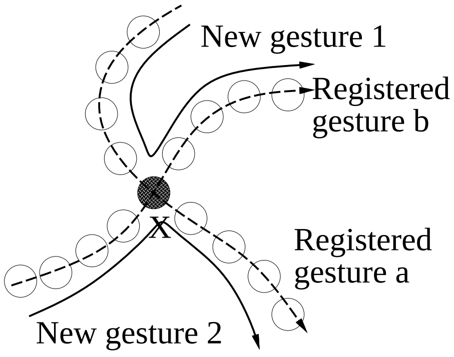
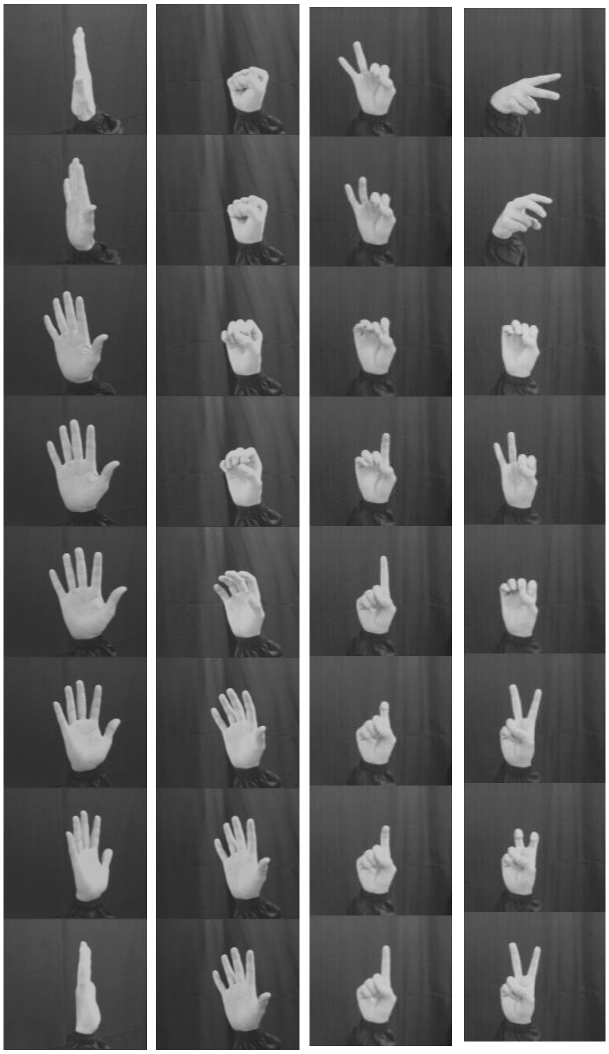


Figure 8: Junction node of transition network

Because the junction node (with more than two links) is automatically generated during the learning phase, a new sequence can be tracked. In figure 8, for example, gesture **a** and **b** have been shown and junction X has been generated. If a part of gesture **a** followed by a part of gesture **b** (gesture 1 in the figure) is shown in the estimation phase, it is successfully tracked in the network. Also gesture 2 can be tracked similarly.



(a) gesture A (b) gesture B (c) gesture C (d) gesture D

Figure 9: Gesture sequences used for building transition network.

4.3 Estimation Experiment with Transition Network

We made an experiment of building a transition network and recognizing gesture sequences.

In learning, we prepared 8 kinds of gestures each of which consists of 5 to 35 sample sequences. Each sequence consists of approximately 200 frames with the sampling rate of 15 frames per second. Figure 9 shows 4 kinds of gesture (only 8 typical frames of the left images are shown). Gesture A and B show simple gestures of the wrist rotation and the hand grasp. Gesture C represents ”110”: the 1st image represents ”100”, and 5th,6th,7th and 8th represent ”10”. Gesture D represents ”220”: 1st represents ”2”, 4th represents ”100”, and 6th,7th and 8th represent ”20”.

In total, 155 sequences and 31000 frames are shown and a transition network with 354 nodes is generated. The number of the nodes is not very large because many frames correspond to one node.

Figure 10 shows the generated network where the node and the link is represented by a point and a line. Only two principal components in the 12 dimensional eigenspace are shown.

Next, an experiment with combined gesture sequences is performed. Figure 11(a) shows an input sequence which consists of multiple learned gestures. This sequence represents ”210”. The first part of the sequence is similar to a part of learned gesture D, and the last part of the sequence is similar to learned gesture C. Two parts are connected by the junction node which represents ”100”.

The result of tracking is shown in Figure 11(b). Although this sequence is not learned explicitly, the system traced the transition network successfully.

We compared the efficiency of this method with that of an exhaustive search.

For the total 2630 frames which includes the above sequences, the average number of matching trials per image was 31.6 by using the transition network. 99.8% of the matching models was found in the primary candidates, and the others were found in the secondary candidates. The average of the processing time for one frame except image capturing is approximately 30.7 m-sec on Pentium-III 600 PC.

On the other hand, an exhaustive search requires 354 matching trials for every frame. The computation of the method with the transition network is reduced to 8.9% of an exhaustive search.

5 Conclusion

This paper presented a method of the hand posture estimation of gesture sequence from silhouette images taken by multiple cameras. In the offline learning phase, we construct an eigenspace of image features. In the online estimation phase, the complexities of the images are first evaluated, and the best-matched model is determined by integrating the matching criteria of the multiple images based on the complexities.

For efficient estimation of gesture sequences, the shape transition network is proposed. In the learning phase, the network is automatically generated from the gesture sequences. In the estimation phase, the transition network is utilized to limit the shape candidates. An experiment proved that the computation is reduced to 8.9% of an exhaustive search.

The proposed method just estimates hand shape. A future work is to recognize a sequence of hand shapes as a meaningful unit such as a word in a sign language.

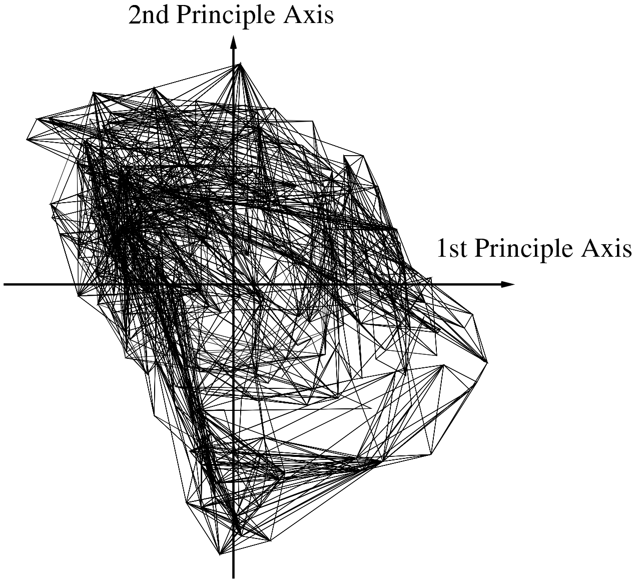


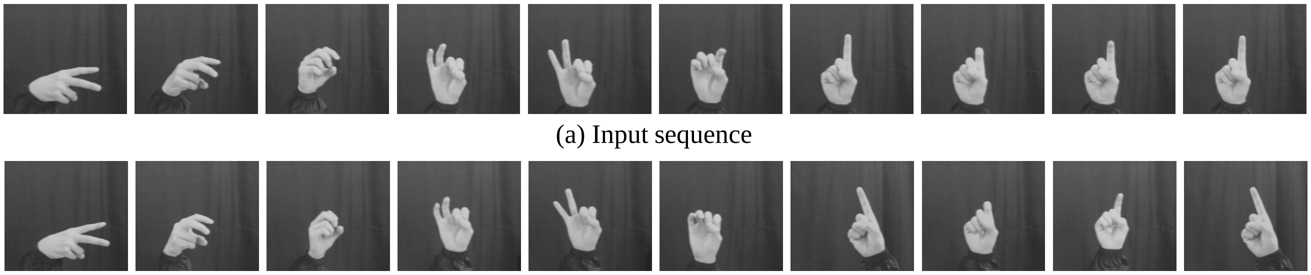
Figure 10: Generated transition network

References

[1] J.M.Rehg and T.Kanade. ”Visual Tracking of High DOF Articulated Structures: an Application to Human Hand Tracking,” ECCV’94, pp.35-46, 1994.

[2] J.J.Kuch and T.S.Huang, ”Virtual Gun: A Vision Based Human Computer Interface Using the Human Hand, ” In MVA’94, pp.196-199, 199

[3] N.Shimada, Y.Shirai, Y.Kuno, and J.Miura, ”Hand Gesture Estimation and Model Refinement using Monocular Camera, ” In Proc. of 3rd Int. Conf. on



(a) Input sequence

(b) Matching model

Figure 11: Posture estimation result by shape tracking with transition network (left images only)

Automatic Face and Gesture Recognition, pp.268-273, 1998.

[4] Y.Cui and J.Weng, ”Learning-based Hand Sign Recognition, ” Proc.of Int. Workshop on Automatic Face and Gesture Recognition, pp.201-206, 1995.

[5] A.D.Wilson and A.F.Bobick, ”Configuration States for the Representation and Recognition of Gesture, ” Proc. of Int. Workshop on Automatic Face and Gesture Recognition, pp.129-136, 1995.

[6] B.Moghaddam and A.Pentland, ”Maximum Likelihood Detection of Faces and Hands, ” Proc. of Int. Workshop on Automatic Face and Gesture Recognition, pp.122-128, 1995.

[7] T.Nishimura, T.Mukai, and R.Oka. ”Spotting Recognition of Human Gesture performed by People from a Single Time-Barying Image, ” In Proc. of IROS’97 vol.2, pp.967-972, 1997.

[8] M.J.Black and A.D.Jepson. ”Eigen Tracking: Robust Matching and Tracking of Articlated Objects Using a View-Based Representation, ” Int. J. of Computer Vision 26(1), pp.63-84, 1998.

[9] S.Nagaya, Y.Itoh, T.Endo, J.Kiyama, S.Seki, and R.Oka. ”Information Integration Architecture for Agent-Based Computer Supported Cooperative Work System, ” IEICE TRNAS. INF. & SYST., Vol.E81-D, No.9, 1998.

[10] A.Utsumi, T.Miyasato, F.Kishino, and R.Nakatsu. ”Hand Gesture Recognition System Using Multiple Cameras, ” Proc. of ICPR’96 vol.1, pp.667-671, 1996.