International Journal of Computational Vision and Biomechanics
Vol.2 No. 1 (January-June, 2016)

MUK PUBLICATIONS
Open Access Publisher

Real-Time Hand Gesture and Face Recognition via T-CombP2DHMMs

Nguyen Dang Binh

Intelligence Media Laboratory, Department of Artificial Intelligence, Kyushu Institute of Technology

This paper introduces a new Pseudo 2-D Hidden Markov Models (P2DHMMs) structure dedicated to the time series recognition (T-CombP2DHMMs). The T-CombP2DHMMs allows it to do temporal analysis, and to be used in large set of hand gestures movement and faces recognition systems in unconstrained environments. Additionally, Face detector is based upon a tree structure of boosted cascaded of weak classifiers. Furthermore, robust and flexible hand gesture tracking using an algorithm that combines two powerful stochastic modeling techniques: the first one is P2DHMMs and the second technique is the well-known Kalman filter. Our work also present a feature extraction method based on the joint statistics of a subset of discret cosine transformation (DCT) coefficients and their position on the hand. Using feature extraction method along with the T-CombP2DHMMs structure was used to develop a complete vocabulary of 36 gestures including the America Sign Language (ASL) letter spelling alphabet and digits include the person present gesture correspondence. The results of the approach are up to 99.5%.

Keywords: *Gesture recognition, hand tracking, face recognition, computer-human interaction*

1. INTRODUCTION

The efficient and robust tracking and recognition of objects in complex environments is important for a variety of applications including human-computer interaction [2, 3], video surveillance [5], autonomous driving [4]. The main challenge in object detection, tracking and recognition is a mount of variation in visual appearance. To cope with appearance variations of the target object during tracking, existing tracking approaches (e.g. [6, 7, 8]) are enhanced by adaptivness to be able to incrementally adjust to the changes in the specific tracking environment (e.g. [9, 10, 11, 12, 13, 14, 15]). In other words, invariance against the different variations is obtained by adaptive methods or representations. Visual appearance also depends on the surrounding environment. Light sources will vary in their intensity, color, and location with respect to the object. The appearance of the object also depends on its pose; that is, its position and orientation with respect to the camera. For example, a side view of a human face will look different than a frontal view (Fig. 1 and Fig. 2). The sign language is undoubtedly the most grammatically structured and complex set of human gestures. In American Sign Language (ASL), the use of hand gestures is very important to differentiate between many gestures (Fig. 3). Thus, a fast and reliable method to extract the hand postures changes from the video sequence is very important in ASL recognition systems. The hand is complex object. The basic idea lies in the real-time generation of gesture model for hand gesture recognition in the content analysis of video sequence from CCD camera. To cope with all this variation, firstly, we combine two powerful stochastic modelling techniques for hand tracking: the first one is P2DHMMs and the second technique is the well-known Kalman filter approach with hand detectors as described in Section 2. Secondly, a tree structure of boosted cascaded of weak classifiers has been improved for face detection in Section 3. Finally, we use the combined use of time “spatialization” and P2DHMMs in the proposed T-ComP2DHMMs model for hand gesture and face recognition concurrently in Section 4. The next section presents the result of experiments. The summarize contribution of this work in the conclusion section.

2. HAND TRACKING

We develop a real time hand tracking method based on the P2DHMM and Kalman filter, which is robust and reliable on hand tracking in unconstrained environments and then the hand region extraction fast and accurately.

2.1. Basic Hand Tracking Algorithm

This work propose a novel tracking method for problem using two powerful stochastic modeling techniques, namely P2DHMMs and Kalman filter. The input of the Kalman filter relies on the information provided by a complex shape model of the persons hand of which the structure has been automatically learned and acquired by the P2DHMMs. The dynamic information need for tracking is solely generated by the Kalman filter. While the Kalman filter obtains its input information from the P2DHMMs, the Kalman filter itself feeds its output information back to the P2DHMMs and improves in this way the shape detection procedure of the P2DHMMs.

Figure 1: Examples of training images for each face orientation

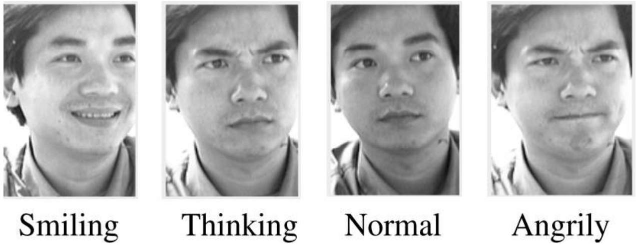


Figure 2: Examples of training image for each face expression

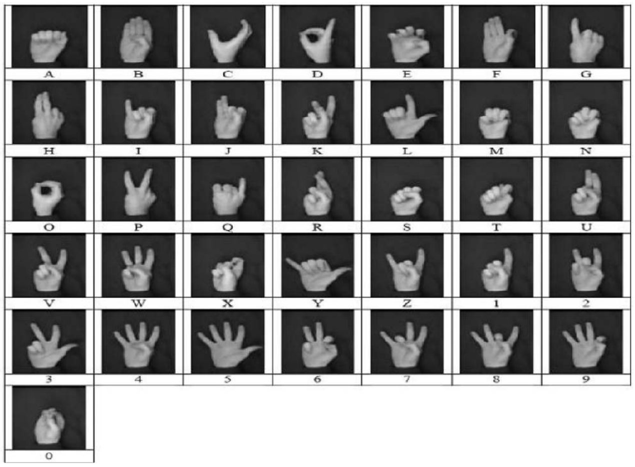


Figure 3: The ASL gesture set to be recognized [18]

This optimal feedback between these two modules is another reason for the powerful performance of the approach. By letting only Kalman filter be responsible for the dynamic information of the tracking process, and relying in the measurement process completely on shape and color information, the tracking procedure becomes entirely independent of other disturbing motions in the background.

2.2. Measurement Vector Generation with P2DHMMs

P2DHMMs generates a measurement vector that is uses as input to the Kalman filter. The components of this vector are the center of gravity of the hand person detected in the image and the width and height of the bounding box. The following steps are carried out for that purpose: firstly, the image is processed with a DCT-based feature extraction method. An overlap between adjacent sampling windows improves the ability of the HMM to model the neighborhood relations between the windows. The result of the feature extraction is two-dimensional array of vectors. This array is presented to P2DHMM as shown in Figure 4.

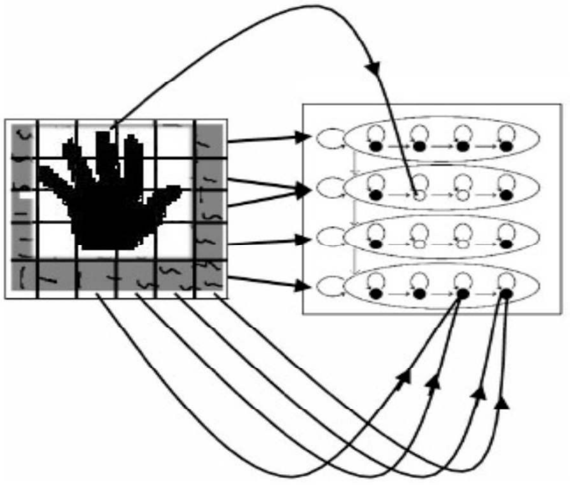


Figure 4: Stochastic model of a two-dimensional object use a P2DHMMs

Such a P2DHMM can be considered as a 2-D stochastic model of an object in an image. The center of gravity of the hands are computed from the segmentation result obtained from Viterbi algorithm by simply calculating the appropriate moment from the blocks inside the black marked area indicating the hand (as show in Fig. 4). The coordinates of this center of gravity, denoted as x and y, and the size of the bounding box of the segmentation, denoted as w (width) and h (height) serve as the measurement input the Kalman filter.

2.3. Combination of P2DHMM Output with Kalman Filter

In order to describe the moving hands and to represent the result of the tracking procedure, we use Kalman filter to predict hand location in one image frame based on its location detected in the previous frame. First, we measure hand location and velocity in each image frame in which, we define the state vector as x_t:

$$x_t = (x(t), y(t), v_x(t), v_y(t), w(t), h(t))^T \qquad (1)$$

$$x = \begin{bmatrix} x(t): & \text{x-coordinate of COG of hand} \\ y(t): & \text{y-coordinate of COG of hand} \\ v_x(t): & \text{Horizontal velocity of COG of hand} \\ v_y(t): & \text{Vertical velocity of COG of hand} \\ w(t): & \text{Width of bounding box} \\ h(t): & \text{Height of bounding box} \end{bmatrix}$$

Where $x(t), y(t), v_x(t), v_y(t)$ shows the location of hand $(x(t), y(t))$, the velocity of hand $(v_x(t), v_y(t))$ and the width and height of hand $w(t)$, $h(t)$ in the tth image frame. We define the observation vector y_t to present the location of the center of the hand detected in the tth frame. The measurement vector **y-**$_t$ consists of the location of the center of the hand region .The state vector x_t and observation vector y_t are related as the following basic system equation:

$$x_t = \Phi x_{t-1} + G w_{t-1} \tag{2}$$

$$y_t = H x_t + v_t \tag{3}$$

Where Φ is the state transition matrix, G is the driving matrix, Φ is the observation matrix, w_t is system noise added to the velocity of the state vector x_t and v_t is the observation noise that is error between real and detected location. Here we assume approximately uniform straight motion for hand between two successive image frames because the frame interval ΔT is short. Then Φ, *G*, and *H* are given as follows:

$$\Phi = \begin{bmatrix} 1 & 0 & \Delta T & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} G = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

with the measurement matrix H and the measurement noise v_t which is resulting from the measurement errors. The Kalman-filter computes the reconstruction of the state vector **x** from the mesurement **y** . The measurement vector y is in this case:

$$y = [x(t), y(t), w, h]^T \tag{4}$$

From the input information of the P2DHMMs, contained in the vector *y*, the system estimates the state vector x and predicts in that way the information about bounding box, contained in the last two dimension of *x*. The third and fourth dimension of x deliver the velocity of the hand and mainly serve as variables supporting the mathematical model of the hand's motion and the stability of the system.

The (*x*, *y*) coordinates of the state vector x_t coincide with those of the observation vector y_t defined with respect to the image coordinate system. Also, we assume that both the system noise w_t and the observation noise v_t are constant Gaussian noise with zero mean. Thus the covariance matrix for w_t and v_t become $\sigma_w^2 I_{4x4}$ and $\sigma_v^2 I_{4x4}$ respect, where I_{4x4} represent a 4×4 identity matrix. Finally, we formulate a Kalman filter as

$$K_t = \overline{P}_t H^T (H \overline{P}_t H^T + I_{4x4})^{-1} \tag{5}$$

$$\overline{x}_t = \Phi \{ \overline{x}_{t-1} + K_{t-1}(y_{t-1} - H\overline{x}_{t-1}) \} \tag{6}$$

$$\overline{P}_t = \Phi(\overline{P}_{t-1} - K_{t-1} H \overline{P}_{t-1}) \Phi^T + \frac{\sigma_w^2}{\sigma_v^2} Q_{t-1} \tag{7}$$

Where $\overline{x}_t$ equal $\cancel{\overline{x}}_{t|t-1}$, the estimated value of x_t from y_0,.., $\mathrm{y}_{\text{t-1}}$, $\overline{P}_t$ equals $\overline{\Sigma}_{t|t-1} / \sigma_v^2, \overline{\Sigma}_{t|t-1}$ represents the covariance matrix of estimate error of $\overline{x}_{t|t-1}$, K_t is Kalman gain, and *Q* equals GG^T. Then the predicted location of the hand in the *t* + 1th image frame is given as (*x*(*t*+1), *y*(*t*+1)) of $\overline{x}_{t+1}$. If we need a predicted location after more than one image frame, we can calculate the predicted location as follows:

$$\overline{x}_{t+m|t} = \Phi^m \{ \overline{x}_t + K_t (y_t - H\overline{x}_{t-1}) \} \tag{8}$$

$$\overline{P}_{t+m|t} = \Phi^m (\overline{P}_t - K_t H \overline{P}_t) \left(\Phi^T \right)^m + \frac{\sigma_w^2}{\sigma_v^2} \sum_{k=0}^{m-1} \Phi^k Q \left(\Phi^T \right)^k \tag{9}$$

Where $\overline{x}_{t+m|t}$ is the estimated value of $\overline{x}_{t+m}$ from y_0,..,y_t, $\overline{P}_{t+m|t}$ equals $\overline{\Sigma}_{t+m|t} / \sigma_v^2$, $\overline{\Sigma}_{t+m|t}$ represents the covariance matrix of estimate error $\overline{x}_{t+m|t}$.

2.4. Interaction Between Kalman Filter and P2DHMMs

An important points the fact that - while vector x is constructed from the vector y in the Kalman equations. The update of the vector *x* is used in return as input to the P2DHMM in order to improve estimation of the vector *y*, thus resulting into a cooperative feedback between the Kalman filter and the P2DHMM. The complete interaction procedure between P2DHMM and Kalman filter is illustrated in Fig. 5: on left site up site, a moving hand has been segmented, and the coordinates of the center of gravity serve as measurement signal for the Kalman filter which predicts a new state vector from this measurement input and the motion equation. On the right upper side, this leads to a new bounding box, which can be derived from the updated state vector (inner black rectangle). This area is enlarged and thus yields an image fraction shown on the right lower side (black-white bold rectangle), which serves as search area for the P2DHMM. From there, the loop is closed by yielding a new segmentation which generates the new measurement signal in the upper left part of Fig. 5.

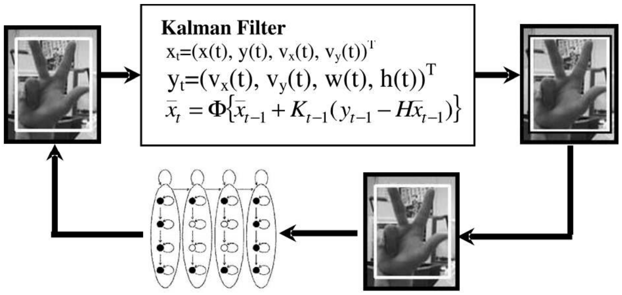

Kalman Filter
$x_t=(x(t), y(t), v_x(t), v_y(t))^T$
$y_t=(v_x(t), v_y(t), w(t), h(t))^T$
$\overline{x}_t = \Phi\{\overline{x}_{t-1} + K_{t-1}(y_{t-1} - H\overline{x}_{t-1})\}$

Figure 5: Scheme of interaction between P2DHMM and Kalman filter

3. FACE DETECTION

Our detector is based upon a tree structure of boosted cascaded of weak classifiers. The head of the tree forms the general face detector and its sole purpose is to find all possible face hypotheses in the image. Successful hypotheses are the passed onto the branches of the tree where specific cascades designed only to detect faces of a specific pose are used to determine the exact pose of the face in the image. To build shape specific detectors the data set must be broken up or clustered into similar shapes that are specific, yet contain sufficient variation in pose to allow the classifier to generalize. To do this we perform an intelligent selection of training images on the training data [3]. In order to detect faces in an image, we first perform an exhaustive detection across all possible position and scales. We use a heuristic coarse-to-fine strategy to speed up this process. While this may sound very computationally taxing, we note that a majority of the positions and scales would not contain faces. The structure of the detector cascades means many parameterizations will be rejected in the first few layers of the top strong classifier, which required only a very small a mount of computation. We exploit this to build a rough cumulative image that highlights areas that have many detections.

The cumulative image is then threshold to remove weak error detections in the background. Finally, a connected components analysis is performed on the threshold image to detect the size and positions of the faces. We also note that the false detections in the background were rejected by this method. In order to detect face pose, the sub-images in the areas where the face was detected are given to the set of hand pose detectors on the second layer. We choose the pose corresponding to the detector that has the highest output. The results of the pose detection can be seen in the experiments section.

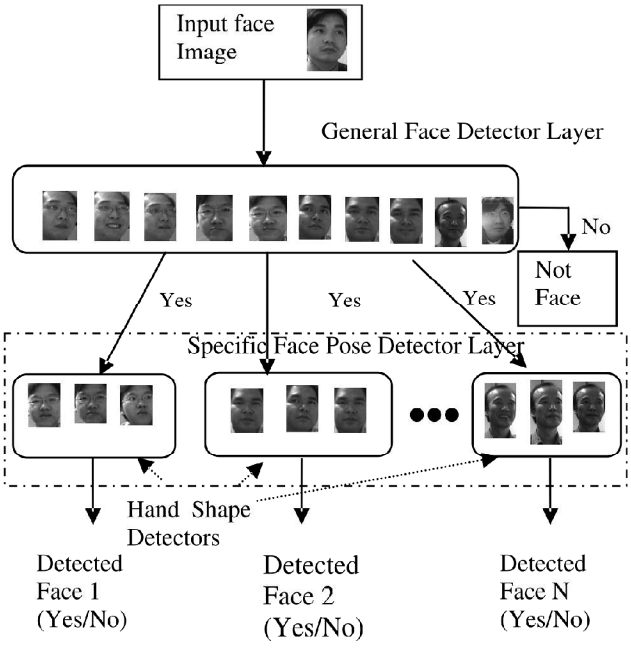


Figure 6: The framework of a tree of face detectors

4. RECOGNITION HAND GESTURE AND FACE VIA T-COMBP2DHMMS MODEL

We have developed the pseudo 2D Hidden Markov Models (P2DHMMs) model to deal with real-time gesture recognition system [17]. The advantages of the improved P2DHMMs structure is simplification, and efficient 2-D model that retains all of the useful HMMs features and intelligent selection of training images of training stage, reducing the number of local minimum in P2DHMMs training. One that the input space is pre-classified. A big problem is divided in several small ones. This philosophy allows a problem with a large number of classes to be solved easily, reducing the training time and/or permitting a very good solution to be found, increasing the recognition rate. Taking these advantages and transporting it to a time varying space, we propose the T-CombP2DHMMs for hand gesture and and face recognition shown by the Fig. 7.

An input vector X in a space S is fed in the Stem and one or more output are chosen according to some similarity criterion. The branches associated with the Stem output receive the same input vector X and do a refined processing. The final output is selected based on the scores obtained by the input vector in the Stem (SM) and branch (SB). The main advanced of CombP2DHMMs structure is the simplification of the training stage. It reduces the number of location minimums in the branch P2DHMMs training. Since the input space is previously classified, a big problem is divided in several small ones. This principle allows a problem with a large number of classes to be solved easily, reducing the training time and/or permitting a very good solution to be found, increasing the recognition rate. Taking these advantages and transporting it to a time varying space, we propose the T-CombP2DHMMs shown by the Figure 7. T-CombP2DHMMs is composed of a Stem and branch P2DHMMs to allow the model to carry out efficiently temporal analysis. The major structure is the inclusion of a Time Normalization preprocessing step and the use of P2DHMMs in the branch layers. However, the main advance is the use of different input spaces in Stem and Branch P2DHMMs, which allows the P2DHMMs to specialize in analyzing a determined subspace.

4.1. Stem Network

The Stem network receives an input vector and selects a subspace from the input space according to a similarity criterion. In a new T-CombP2DHMMs structure the input space of the Stem network (S'_0) is defined as a modified subspace of original input space S, and the P2DHMM input space S_1 is the complementary subspace. This strategy permits a more efficient use of the spatial analysis capability of the layers. Given X a time sequence of N

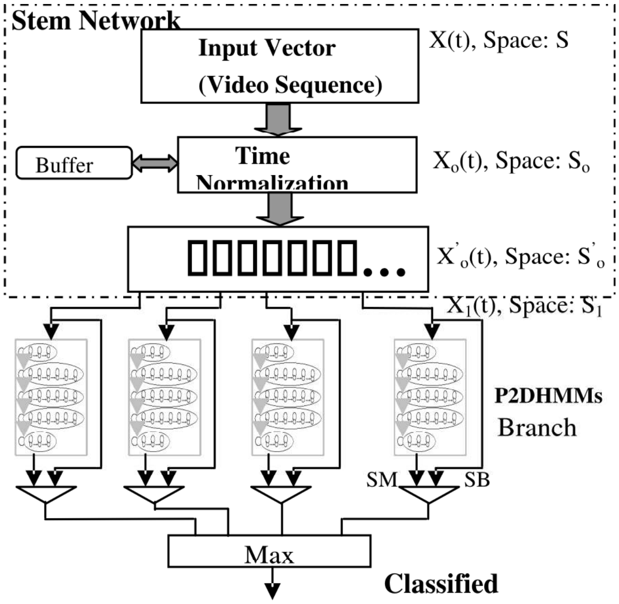


Figure 7: The T-ComP2DHMMs structure

dimensional vectors $x(t)$ belonging to the space S, with time length $T(X)$ samples. $x_o(t)$ is defined as sub-vector of $x(t)$, $x_o(t) \in S_o$; $x_1(t)$ is a sub-vector of x,

$$x_1(\mathrm{t}) \in S_1 \text{ and } S_o \subset S \text{ and } S_1 \subset S|S = S_o\ x\ S_1 \qquad (10)$$

As we are dealing with temporal series, we need to create a space S'$_o$ able to describe completely the time variations existing in the chosen subspace S$_o$. So we propose the application of a Time Normalization procedure on the time independent space S'$_o$. This procedure is needed because a fixed dimension vector is due in the input of the LVQ that composes the Stem network. The dimension of the Stem input layer L is related to the original input time sequence X by

$$L = \mathrm{Dim}(S_o) \times \bar{T};\ T' = N \mid N \in N \text{ and } N \geq E\{T(X)\} \quad (11)$$

where Dim(S_0) is the dimension of the selected subspace S_0 and $\bar{T}$ is the first interger greater than the expected value of the time length $T(X)$. To obtain the vector X^i_0 from the original time sequence X, we suggest the modeling of the time sampling series by Dim(S$_0$) continuous functions using cubic spline interpolation procedure. Given a N$_o$ dimensional time series $X^i_0 = \{x^i_0(t_1), x^i_0(t_2), ..., x^i_0(t_{T(X^i_0)})\}$ and defining a set of Dim(S$_0$) continuous associated function $f_i(t)$ = CubicSpline(X^i_o) for i = 1, 2,..., Dim(S$_0$). Resampling each continuous function $f_i(t)$ in T' sample points by $\bar{x}^i_o(n) = f_i(n \quad x \quad T_o)$, where $n = 1, 2,...,\bar{T}$ and T$_o$ is sample period in an appropriate time basis. The system input vector

$$\begin{aligned} \mathrm{X}^{\mathrm{i}}_0 = [&\vec{\mathrm{x}}^1_0(1), \vec{\mathrm{x}}^2_0(1),...,\vec{\mathrm{x}}^{\mathrm{Dim(S_0)}}_0(1), \vec{\mathrm{x}}^1_0(2), \vec{\mathrm{x}}^2_0(2),..., \\ &\vec{\mathrm{x}}^{\mathrm{Dim(S_0)}}_0(2), \vec{\mathrm{x}}^1_0(\vec{\mathrm{T}}), \vec{\mathrm{x}}^2_0(\vec{\mathrm{T}}),...,\vec{\mathrm{x}}^{\mathrm{Dim(S_0)}}_0(\vec{\mathrm{T}})] \end{aligned} \qquad (12)$$

The selection of the subspaces So and S1 is very important in the T-CombP2DHMMs structure. The feature selection can be based on a class separability criterion that is evaluated for all of possible combinations of the input features. An Interclass Distance Measure criterion based on the distance between DCT vectors as follows:

$$J_s = \frac{1}{2}\sum_{i=1}^{m} P(w_i) \sum_{j=1}^{m} P(w_j) \frac{1}{N_i N_j} \sum_{k=1}^{N_i} \sum_{l=1}^{N_j} \delta(\vec{\xi}_{ik}, \vec{\xi}_{jl}) \qquad (13)$$

where m is the number classes, $P(w_i)$ is the probability of the ith class, N_i is the number of pattern vectors belonging to the class w_i and $\delta(\vec{\xi}_{ik},, \vec{\xi}_{jl})$ is the DCT distance based measure from the k$^{\text{th}}$ candidate pattern of the class i to the l$^{\text{th}}$ candidate pattern of the class j defined by

$$\delta(\vec{\xi}_k, \vec{\xi}_l) = \left|\vec{\xi}_k - \vec{\xi}_l\right| = \sqrt{\sum_{j=1}^{d} \xi^j_k - \xi^j_l)^2} \qquad (14)$$

where d is the dimension of candidate space. In the T-CombP2DHMMs context, it is needed to optimize the joint class separability of the Stem and P2DHMMs. From the equation (13), the values of the class probability $P(w_i)$ are possible to estimate only for the P2DHMMs output. The classification criterion is based on the probability of error obtained for the P2DHMMs. It can be modeled according:

$$J_T = J_{\delta S} + \sum_{p=1}^{m} J_{\delta B_p} \qquad (15)$$

where $J_{\delta S}$ is the interclass distance measure for the Stem network and $J_{\delta B_p}$ for the p$^{\text{th}}$ P2DHMMs, defined as

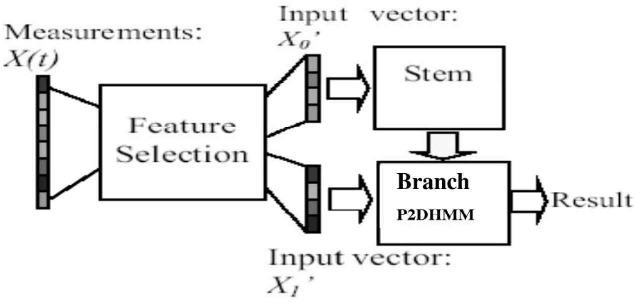


Figure 8: Applying feature selection to T-CombP2DHMMs

$$J_{\delta S} = \frac{1}{2}\sum_{i=1}^{m} P(\psi_i) \sum_{j=1}^{m} P(\psi_j) \frac{1}{N_i N_j} \sum_{k=1}^{N_i} \sum_{l=1}^{N_j} \delta(\vec{x}_{0_{ik}}, \vec{x}_{0_{jl}}) \qquad (16)$$

where $P(\psi_i)$ is the priori probability obtained for the i$^{\text{th}}$ pseudo class designed by the Stem network training algorithm, $\vec{x}_{0_i}$ are the input vectors in the S'$_o$ space, and m is the total number of vectors.

$$J_{\delta B_p} = \frac{1}{2}\sum_{i=1}^{m_p} P(w_{ip} \mid \psi_p) \sum_{j=1}^{m_p} P(w_{jp} \mid \psi_p) \frac{1}{N_i N_j} \sum_{k=1}^{N_i} \sum_{l=1}^{N_j} \delta(\vec{x}_{1_{ik}}, \vec{x}_{1_{jl}}) \qquad (17)$$

where $P(w_{ip} | \psi_p)$ is the conditional probability obtained for the ith class of the pth P2DHMMs given the Stem. $\vec{x}_{1_{ik}}$ is the input vectors allocated for the class i in the S_1 space by the Stem network and m_p is the number of classes allocated to the Stem network. The set of features of $\vec{x}_0$ and $\vec{x}_0$ that maximize the class separability criterion given by equation (15) has chose to define the spaces S_0, S_1, and S'_0. Doing this can required a huge amount of computation time to obtain an optimum solution, since the equation (15) must be evaluated for every $C_{N_0}^{N}$ combination of the input space dimensions N.

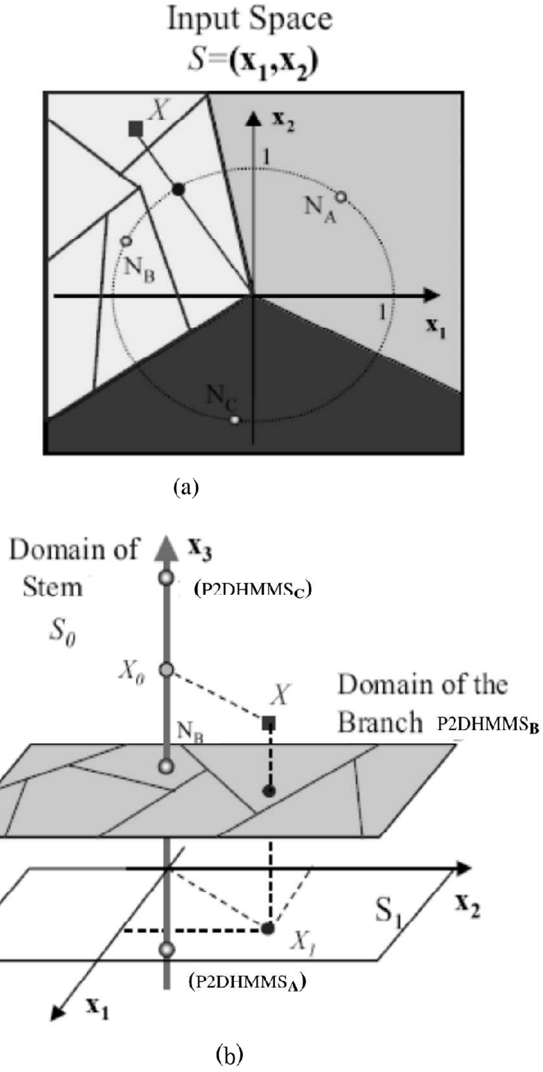
4.2. Space Splitting

Besides the structural modifications, the T-CombP2DHMMs model adds value by using the Space Splitting technique. In the T-CombP2DHMMs, the input space of the Stem S is the same space of the input layer in the branch P2DHMMs, meaning that the same input vector $\vec{x}$ is analyzed twice for different P2DHMMs. Obviously, it is performed in different levels of accuracy, as we can see in Figure 9 (a), but almost the same information is used by the Stem and Branch P2DHMMs, becoming in a redundant processing. In the new T-CombP2DHMMs, the Stem and Branch P2DHMMs work in different input spaces S_0 and S_1. These subspaces are derived from the original input space S. Figure 9(b) presents a simple example of Space Splitting technique. In this example, the original input space S is a 3-D space composed by x1; x2; x3 axis. We chose the subspace composed by the axis x3 to define the space S_0 and the subspace x1; x2 to define the space S_1. Therefore, an input vector X in the space S is first projected into the subspace S_0, in this example axis x3, generating the vector X_0 to be used by the Stem. In the next step, the input vector X is projected into subspace S_1, represented by the plan x1; x2 in this example, generating the vector X_1 which is analyzed by the Branch P2DHMMs.

The use of Space Splitting permits each P2DHMMs networks to specialize in analyzing a determined set of input features efficiently. So, uncorrelated input features can be analyzed separately by the P2DHMMs, reducing the training time and improving the quality of the solution. The method used to define the subspaces S_0 and S_1 given a training set in the input space S is the most important point in Space Splitting. We will present here two approaches to do the selection of subspaces.

4.3. Temporal Pattern Recognition with T-CombP2 DHMMs

In order to select which components of input space S will compose S_1 and S_2, the most natural way is using the engineer's analysis and experience.



(a)

(b)

Figure 9: (a) The input space analyzed by P2DHMMs, (b) Space Splitting-the input space analyzed by T-Comb P2DHMMs

The use of this input space can lead to some theoretical analysis. A hand gesture can be decomposed in two uncorrelated components:

- The translation movement of the hand;
- The time variation of the hand posture.

These two features are characterized by different components in the proposed input space S. To select the features which define the S_0 subspace, the interclass separability measurement optimization procedure describe in the section 4.1 can be applied. In this work, we selected from the input vector, the two coordinates corresponding to the $\vec{P}(t)$ hand position on the screen to compose the S_0 subspace, generating two-space S_0 and its complementary S_1 space, to be used by T-CombP2DHMMs model. It is efficient due to the natural uncorrelation existing in the hand posture, describe by S_1, and the hand trajectory, described by S'_0. To obtain invariance of the motion to the camera relative possition, we use the normalized velocity measurement $\vec{P}(t)$ instead of absolute possition, defined

$$\vec{P}(t_i) = \frac{\vec{P}(t_i) - \vec{P}(t_{i-1})}{\left|\vec{P}(t_i) - \vec{P}(t_{i-1})\right|} \quad and \quad \vec{P}(t_0) = 0 \tag{18}$$

Where $\vec{P}(t_i)$ is the 2-D vector of the obsolute position of the hand palm centroid inthe screen at time t_i .Using these definitions we are assigning the stem layer to analyze a normalized trajectory and the P2DHMMs to analyze fine hand postures variation for pre-selected trajectory.

Each P2DHMMs is trained by hand gesture or face in the database obtained from the training set of each of the gesture using the Baum-Welch algorithm due to pre-analysis achieved by Stem network, which reduces the complexity of the problem. Then, the proposed T-CombP2DHMMs structure has expected to be general and easily trainable.

For recognition, the double-Viterbi algorithm is used to determine the probability of each hand model and face model. The image is recognized as the hand gesture or face, whose model has the highest production probability. Due to the structure of the P2DHMMs, the most likely state sequence is calculated in two stages. The first stage is to calculate the probability that rows of the individual images have been generated by one-dimensional HMMs, that are assigned to the super-stages of the P2DHMMs. These probabilities are used as observation probabilities of the super-states of the P2DHMMs. Finally, on the level second Viterbi algorithm is executed.

4.4. Pseudo 2D HMM Construction

4.4.1. Description

Pseudo 2DHMMs in this paper are realized as a vertical connection of horizontal HMMs (λk,). However it is not the only one. In order to implement a continuous forward search method and sequential composition of gesture models, the former type has been used in this research. There are three kinds of parameters in the P2DHMMs. However, since the hand image is two-dimensional, we further divided the Markov transition parameters into super-state transition and state transition probabilities; each is denoted as

$$\vec{a}_{kl} = P(r_{t+1} = l \,|\, r_t = k),\ 1 \le k, l \le N \ \text{ and}$$

$$\vec{a}_{ij} = P(q_{t+1} = j \,|\, q_t = i),\ 1 \le i, j \le M \tag{19}$$

where r_t denotes a super-state which corresponds to a HMMs λ_k, and q_t denotes a state observing at time t. The mode has N super-states and the HMMs λ_k, is defined as standard HMM consisting of M states.

4.4.2. Evaluation Algorithm

Let us consider a t^th horizontal frame, observation future vector $\vec{O}_t = \vec{o}_{1t}, ..., \vec{o}_{st}$, $1 \le t \le T$. This is a one-dimension feature sequences like that of $\vec{O}$ in Eq. (13). This is modeled by a HMM λ_k, with likelihood $P(\vec{O}_t | \lambda_k)$. Each HMM λ_k may be regarded as a super-state whose observation is a horizontal frame of states.

$$P_{r_t}(\vec{O}_t | \lambda_t) = \sum_{allQ} \Pr(\vec{O}_t, Q | \lambda_{r_t}) = \sum_{q_1, q_2, .., q_T} \pi_{q_1} b_{q_1}(\vec{o}_{1t}) \prod_{s=2}^{S} a_{q_{t-1} q_t} b_{q_t}(\vec{o}_{st}) \tag{20}$$

Now let us consider a hand region image, which we define as a sequence of such horizontal frames as $\vec{O} = \vec{O}_1, \vec{O}_2, .., \vec{O}_T$. Each frame will be modeled by a super-state or a HMM. Let Λ be a sequential concatenation of HMMs. Then the evaluation of Λ given feature sequence $\vec{O}$ of the sample image X is

$$P(\vec{O} | \Lambda) = \sum_{R} P_1(\vec{O}_1) \prod_{t=2}^{T} \vec{a}_{r_{t-1} r_t} P_{r_t}(\vec{O}_t) \tag{21}$$

where it is assumed that super-state process starts only one from the first state. The P_{r_t} function is the super-state likelihood. Note that both of the Eqs. (20) and (21) can be effectively approximated by the Viterbi score. One immediate goal of the Viterbi search is the calculation of the matching likelihood score between $\vec{O}$ and HMM. The objective function for an HMM is defined by the maximum likelihood as

$$\Delta(\vec{O}_t, \lambda_k) = \max_{Q} \prod_{s=1}^{S} a_{q_{s-1} q_s} b_{q_s}(\vec{o}_{st}) \tag{22}$$

where $Q = q_1, q_2, \ldots, q_s$ is a sequence of states of λ_k, and $a_{q_0 q_1} = \pi_{q_1} . \Delta(\vec{O}_t, \lambda_k)$ is the similarity score between two sequences of different length. The basic idea behind the efficiency of DP computation lies in formulating the expression into a recursive form

$$\delta_S^k(j) = \max_i \delta_{S-1}^k(i) a_{ij}^k b_j^k(\vec{o}_{st}),\ j = 1, ..., M_k, s = 1, ..., S, k = 1, ..., K \tag{23}$$

where $\delta_s^k(j)$ denotes the probability of observing the partial sequence $\vec{o}_{1t}, ..., \vec{o}_{st}$ in model k along the best state sequence reaching the state j at time/step s. Note that $\Delta(\vec{O}_t, \lambda_k) = \delta_S^k(N_k)$ where N_k is the final state of the state sequence. The above recursion constitutes the DP in the lower level structure of the P2DHMM. The remaining DP in the upper level of the network is similarly defined by

$$D(\vec{O},\Lambda) = \max_{k} \prod_{t=1}^{T} \vec{a}_{r_{t-1}r_t} \Delta(\vec{O}_t, \lambda_{r_t}) \qquad (24)$$

that can similarly be reformulated into a recursive form. Here denotes the probability of transition from super-state r_1 to r_2. According to the formulation described thus far, a P2DHMM add only one parameter set, i.e., the super-state transitions, to the conventional HMM parameter sets. Therefore it is simple extension to conventional HMM.

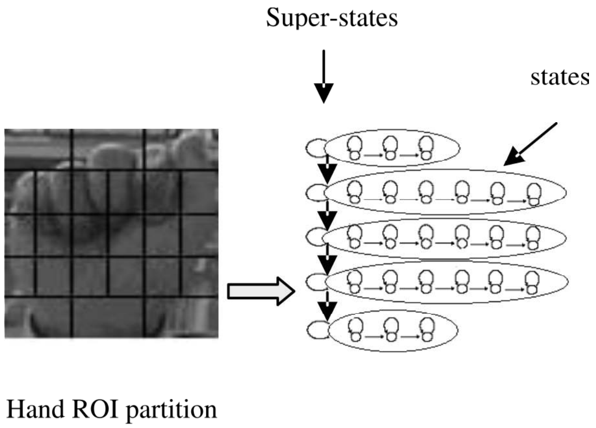


Figure 10: **The P2DHMM for hand gesture**

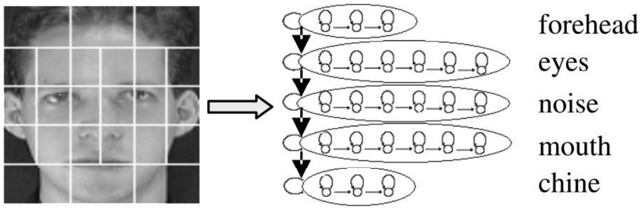


Figure 11: **The P2DHMMs for face**

For each gesture there is a P2-DHMM, Fig. 10 and Fig. 11 show a P2DHMMs model consists of 5 super-states and their states in each super-state that model the sequence of rows in the image. The topology of the super-state model is a linear model, where only self transitions and transitions to the following super-states are possible. Inside the super-states, these are linear one dimension hidden Markov model to model each row. The state sequence in the rows is independent of the state sequences of neighboring rows.

4.5. Decomposition of Appearance in Space, Frequency, and Orientation

Our approach is to jointly model visual information that is localized in space, frequency, and orientation. To do so, we decompose visual appearance a long these dimensions. Below we explain this decomposition and in the next section we specify our visual attributes based on this decomposition. First, we decompose the appearance of the object into "parts" whereby each visual attribute describes a spatially localized region on the object. We would like these parts to be suited to the size of the features on each object. However, since important cues for hands at may size, we need multiple attributes over a range of scales. We will define such attributes by making a joint decomposition in both space and frequency. We would like these parts to be suited to the size of the features on each object. Finally, by decomposing the object spatially, we do not want to discard all relationships between the various parts. We believe that the spatial relationship of the parts is an important cue for recognition. With this representation, each feature vectors now becomes a joint distribution of attribute and attribute position.

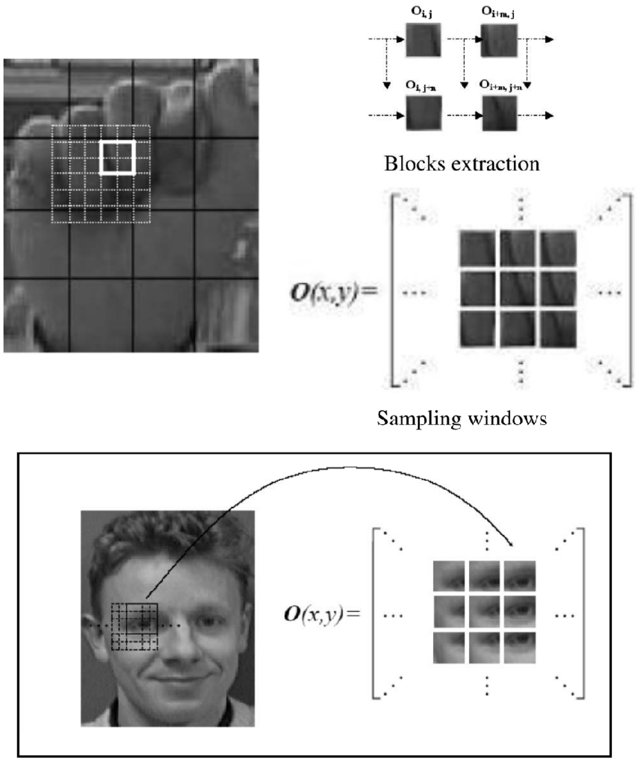


Figure 12: **The features extraction**

4.6. Representation of Visual Attributes by Subsets of DCT Coefficients

Using DCT coefficient as features instead of gray values of the pixels in the shift window where most of the image energy is found. They tend to be insensitive to image noise as well as image rotations or shifts, and changes in illumination. To create visual attributes that are localized in space, frequency, and orientation, we need to be able to easily select information that is localized along these dimensions. In particular, we would like to transform the image into a representation that is jointly localized in space, frequency, and orientation. To do so, we perform a DCT of the image. The DCT transform is not the only possible decomposition in space, frequency, and orientation. In our work, instead of use an overlap of 75% between adjacent sampling windows, we have to consider the neighboring sampling of a sampling window.

5. EXPERIMENTAL RESULTS

5.1. Hand Tracking

For our experiments, example is shown in the video clip that accompanies the paper and is available from the first author's website.[1] A few frames detector ran real-time from CCD camera are also shown in Fig. 13.

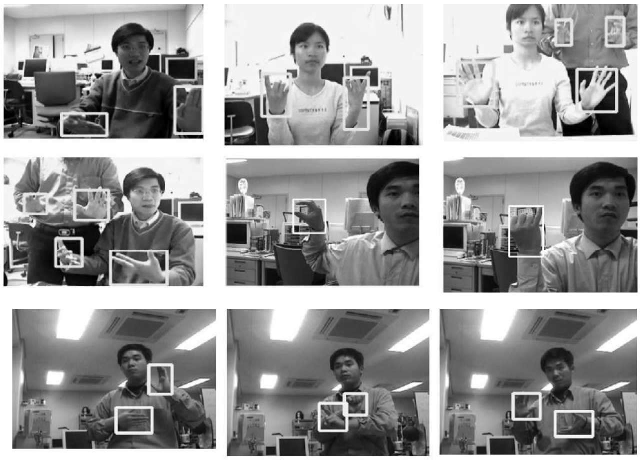


Figure 13: The results of the hand tracking

5.2. Results of T-ComP2DHMM Based Hand Gesture Recognition

The training set consists of 36 hand gestures from vocabulary of 36 gestures including the ASL letter spelling alphabet and digits. Each one of the 36 gestures was performed 60 times by 20 persons to create a database. The images of the same gesture were taken at different times. Thus, each set has composed of 1080 images. The combined using of time "spatialization" and P2DHMM in the proposed T-CombP2DHMMs model overcoming the classical approaches achieving a 99.5% of correct recognition rate.

The example is shown in the video clip that accompanies the paper and is available from the author's website.[2]

5.3. Results of T-CombP2DHMMs Based Hand Gesture and Face Recognition

For face, we gathered a total of 5015 face images from various video sequences of different people signing and from Feret database.[3] We selected 2405 examples for training and similarly 2012 face images from different sequences were retrained for testing. Some of the results of hand detector ran over the test data sequences real-time from CCD camera can be seen in Fig. 15. The example is shown in the video clip that accompanies the paper and is available from the author's website.[4]

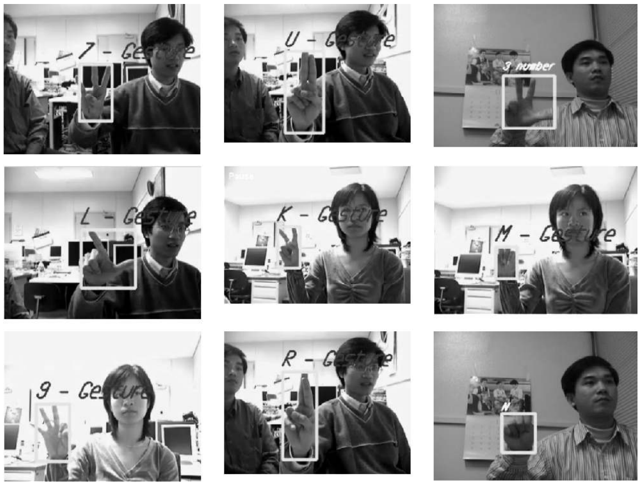


Figure 14: The results of the hand gestures recognition

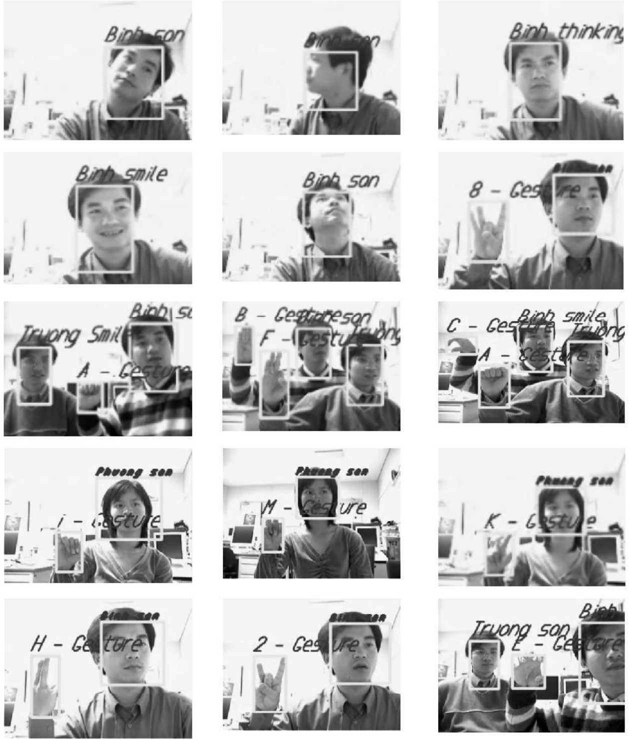


Figure 15: Some results of recognition

Table 1
Recognition rates and complexities of HMMs

	Complexity	*Recognition Rate*
Classical 1D-HMM	$N_0T_0^2$	85%
Optimized 1D-HMM	$N_0T_0^2$	92%
Classical 2D-HMM	$(\sum_{k=1}^{N_0} N_1^{(k)})^2 T_0 T_1$	96%
P2DHMM	$(\sum_{k=1}^{N_0} (N_1^{(k)})^2 T_1) T_0 + N_0^2 T_0$	98.5%
T-CombP2HMMs	$((\sum_{k=1}^{N_0} (N_1^{(k)})^2 T_1) T_0 + N_0^2 T_0)/M$	99.5%

(N_0= number of super states, $N_p^{(k)}$= number of states in the k'th super state, T_0 = number of vertical observations, T_1 = number of horizontal observations. M = number of P2DHMM in P2DHMM branch)

6. CONCLUSIONS

This work introduced a new structure T-CombP2DHMMs dedicated to the time series recognition and presented a new feature extraction method using joint statistics of a subset of DCT coefficients to hand gestures and faces. The T-ComP2DHMM structure uses a Time Normalized Learning Vector Quantization in the Stem network and P2DHMM. We build the T-ComP2DHMM model based upon the P2DHMM, which allows it to do temporal analysis and to be used in large set of human movements' recognition system. The results obtained for a set of 36 different gestures show a 99.5 % of correct recognition rate. This results demonstrate that the joint use of time "spatialization" techniques, natural time processing techniques and P2DHMM given good results.

NOTES

1. currently at *http://www.fsai.kyutech.ac.jp/~ndbinh/Research/HandTracking.wmv*
2. currently at *http://www.fsai.kyutech.ac.jp/~ndbinh/Research/HandGestureRecognition.wmv*
3. *http://www.itl.nist.gov/iad/humanid/feret/feret_master.html*
4. currently at *http://www.fsai.kyutech.ac.jp/~ndbinh/Research/HandGestureAndFaceRecognition.wmv*

REFERENCES

[1] T. Starner, and Pentland, "Real-Time American Sign Language Recognition from Video Using Hidden Markov Models", TR-375, MIT Media Lab, 1995.

[2] A. Bobick, S. Intille, J. Davis, F. Baird, C. Pinhanez, L. Campbell, Y. Ivanov, A. Schutte, and A.Wilson. The KidsRoom. Communications of the Association for Computing Machinery (ACM), **39**(3&4): 438–455, 2000.

[3] G. R. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, 1998.

[4] S. Avidan. Support vector tracking. The IEEE Pattern Analysis and Machine Intelligence (PAMI), 26: 1064–1072, 2004.

[5] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *In Proc. IEEE International Conference on Computer Vision* (ICCV), **2**, 734–741, 2003.

[6] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *The International Journal of Computer Vision* (IJCV), **26**(1): 63–84, 1998.

[7] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. *In Proc. CVPR*, **2**, 142–149, 2000.

[8] A. Elgammal, R. Duraiswami, and L. S. Davis. Probabilistic tracking in joint feature-spatial spaces. *In Proc. CVPR*, **1**, 781–788, 2003.

[9] B. Han and L. Davis. On-line density-based appearance modeling for object tracking. *In Proc. ICCV*, **2**, 1492–1499, 2005.

[10] J. Ho, K. Lee, M. Yang, and D. Kriegman. Visual tracking using learned linear subspaces. *In Proc. CVPR*, **1**, 782–789, 2004.

[11] J. Lim, D. Ross, R. Lin, and M. Yang. Incremental learning for visual tracking. In Advances in Neural Information Processing Systems 17, pp. 793–800. MIT Press, 2005.

[12] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *In Proc. CVPR*, **1**, 415–422, 2001.

[13] D. Ross, J. Lim, and M. Yang. Adaptive proballistic visual tracking with incremental subspace update. *In Proc. ECCV*, **2**, 470–482, 2004.

[14] J. Vermaak, P. Perez, M. Gangnet, and A. Blake. Towards improved observation models for visual tracking: Selective adaption. *In Proc. ECCV*, 645–660, 2002.

[15] J.Wang, X. Chen, and W. Gao. Online selecting discriminative tracking features using particle filter. *In Proc. CVPR*, **2**, 1037–1042, 2005.

[16] O. E. Agazzi and S. S. Kuo, Pseudo two-dimensional hidden markov model for document recognition. *AT&T Technical Journal*, **72**(5): 60-72, 1993.

[17] N. D. Binh, E. Shuichi, and T. Ejima "Real-Time Hand Tracking and Gesture Recognition System", Proceedings of International Conference on Graphics, Vision and Image Processing (GVIP), 362-368, 2005.

[18] R. Lockton, A. W. Fitzgibbon, "Real-Time gesture recognition using deterministic boosting", *Proceedings of British Machine Vision Conference*, 817-826, 2002.