Journal of Korea Multimedia Society Vol. 11, No. 6, JUNE 2008(pp. 807-815)

Recognizing Hand Digit Gestures Using Stochastic Models

Bong-Kee Sin†

ABSTRACT

A simple efficient method of spotting and recognizing hand gestures in video is presented using a network of hidden Markov models and dynamic programming search algorithm. The description starts from designing a set of isolated trajectory models which are stochastic and robust enough to characterize highly variable patterns like human motion, handwriting, and speech. Those models are interconnected to form a single big network termed a spotting network or a spotter that models a continuous stream of gestures and non-gestures as well. The inference over the model is based on dynamic programming. The proposed model is highly efficient and can readily be extended to a variety of recurrent pattern recognition tasks. The test result without any engineering has shown the potential for practical application. At the end of the paper we add some related experimental result that has been obtained using a different model - dynamic Bayesian network - which is also a type of stochastic model.

Key words: hand gesture, gesture recognition, spotting network, hidden Markov model, dynamic Bayesian network

1. INTRODUCTION

The predominant mode of today's human-computer interaction is using keyboard and mouse at desk. With the advancement of computer vision and pattern recognition technologies during the past decade, there has been a strong interest in and a growing need for more intuitive, convenient, and human-friendly interfaces such as speech command, handwriting input, hand gesture command, and so on [1].

It is easy to imagine that using hand gesture for human-computer interaction can help people to communicate with computers in a more intuitive way, at least partially. The potential of gestures has already been demonstrated in applications that take hand gesture input to control a computer or a vision-capable robot, and softwares like Microsoft's Power Point™ and Media Player™[2]. Other possible applications of gesture recognition technique include computer games involving body motions or gestures, video-based teleconferencing, the manipulation of graphical objects by CAD designers and virtual reality players. In most of these areas, the use of video cameras is more natural than other dedicated acquisition devices such as data-gloves, and the development of high-performance pattern recognition algorithms are more than desired - they are indispensable. However, the technology needed to realize the interface is far more difficult and challenging than has been thought to be.

In video-based hand gesture recognition it is necessary to distinguish two aspects of hand gestures: the static hand posture or configuration and the dynamics or the trajectory of the moving hands in 3-dimensional space. This paper focuses on the analysis of the latter aspect of hand gestures.

The second major issue of the paper is the in-

※ Corresponding Author : Bong-Kee Sin, Address : (608-737) Tayon-dong 599-1, Nam-ku, Busan, Korea, TEL : +82-51-620-6491, FAX : +82-51-620-6450, E-mail : bkshin@pknu.ac.kr

Receipt date : Oct. 31, 2007, Approval date : May 29, 2008

† Division of Computer Multimedia, Pukyong National University

※ This is an extended version of the presented in MITA 2007 and it has been recommended for the publication in the Journal of Korea Multimedia Society, English Edition.

ference given an input sequence. There are two sub-problems to address when dealing with dynamic hand gesture recognition: spotting and classification, which are intertwined in many practical applications and usually one cannot do without the other. On the one hand, spotting aims at identifying the start and the end of particular gestures given a continuous stream of data. Usually, this stream of data is regarded as a random sequence of known gestures and unknown non-gestures. On the other hand, given an isolated gesture pattern, classification labels it with the class to which the gesture belongs.

In the rest of the paper, classification of hand gesture types and the related literature are reviewed first to give the overall picture and the definition of the target problem. Following these are the continuous gesture spotting and recognition model and the inference algorithm. Next comes a preliminary result from a simple experiment. At the end of the same section, a study on further elaboration of the model for two-hand gesture recognition will be presented briefly with an emphasis on the dynamic Bayesian network.

2. HAND GESTURES

There have been a number of studies on human gestures in psycholinguistic research. Stokoe [3] described gestures in four aspects including hand shape, position, orientation and movement. Kendon [4] describes a philology of gesture, which consists of gesticulation, language-like gestures, pantomimes, emblems, and sign language. Note that a sign language is characterized by a specific set of vocabulary and grammar whereas emblems are informal gestural expressions in which the meaning depend on context, convention and culture. When viewed in terms of generic applications, hand gestures can be classified into several categories such as conversational gestures, controlling gestures, manipulative gestures, and communicative gestures [5].

To simplify the discussion of computer-based gesture recognition in this paper, the target gestures are classified into three types: data gestures conveying various kinds of messages or data which include conversation and communication, command gestures for controlling the computer or directing it to do some specific action, and pointing gestures which is highly characteristic of hand or finger-based gestures.

Among the data gestures, automatic recognition of sign language has been one of the most attractive topics that have been intensively studied since 1990s [6]. It is largely because sign languages are highly structural and thus are very suitable as a test-bed for computer vision and pattern recognition algorithms. Sign language is an important case of communicative gestures. The research of sign language may not only help the disabled to interact with computers but also facilitate our understanding of other types of human gestures and behaviors.

Command and control gestures are also an important focus of current research in vision-based interface. Many of the command gesture recognition research have aimed at instructing or controlling robots with a designated set of gestures. Some researchers tried to control Microsoft's Media Player or PowerPoint for real time presentation control [2]. Commonly they approach the problem by defining about a dozen gestures and model the patterns using hidden Markov models. Finally the manipulative gesture will serve as a natural way to interact with virtual objects. Teleoperation and virtual assembly are good examples of applications.

The last type of gesture is the pointing gesture. It is distinguished from the previous two in that it involves a finger or a hand that points to someone or some object, real or virtual, which can be located in the three dimensional space surrounding the subject. This involves the navigation gesture

in VR. Instead of using wands, the orientation of hands can be captured and interpreted as a three dimensional directional command to navigate virtual environments.

Another simple way of distinguishing gesture is using temporal or dynamic features: dynamic gestures that convey information through the movement or the trajectory of the hand, and static gestures in which the shape and pose of the hands and fingers are important. Gesture dynamics has been described hierarchically with atomic movements at the bottom, activity which may be a static configuration or a sequence of dynamic movements, and finally actions which are the high-level entities and correspond to the level at which people use to describe what is happening [7]. Dynamic models may be used to recognize activities. Action recognition may require a method of defining a complex of activities over time and a method of incorporating context.

3. GESTURE RECOGNITION MODEL

A person while he or she is alive is ever constantly in motion. The motion is often repetitive or periodic like gait but most of the time highly random and unpredictable. The good news is that the motion is not really random but quite stereotypical and patterned subject to physical constraints and past practice.

Hidden Markov model or HMM is a statistical modeling tool for highly variable sequence patterns. It consists of a number N of hidden states $S = \{1, \cdots, N\}$. They are not directly observable but can be guessed via an observation sequence $O = o_1 \cdots o_T$ which we assume is a function of some unknown sequence of hidden states $X = x_1 \cdots x_T$, $x_t \in S$ and runs synchronously parallel to the progression of state changes. One of greatest features of the HMM is that it is described with probabilistic parameters based on statistics and can be optimized using a set of exemplars. Later in this paper, the dynamic Bayesian network(DBN), a generalization of the HMM, will also be discussed to design more complex and robust models for more complex gesture patterns.

For digit gesture recognition, ten isolated hand trajectories are defined and a single HMM will be created for each digit's hand motion trajectory. Each model has five states on average with the exception of digit 1 that has one state less.

Hand trajectories in video frame are mirror reflections. They are corrected for visualization purposes. A typical hand trajectory containing several digits is shown in Figure 1. Among those digits, the segmental trajectory of the first digit 3 has been highlighted.

Each trajectory segment is converted to a sequence of direction codes quantizing the direction angle into sixteen codes. In the preliminary experiment as described later, only the direction is noted whereas the inter-point distances have been ignored.

Gesture sequence spotting is not easy since we do not know how many and where gestures are made in a video session. The task is further complicated by the presence of numerous non-gesture motions partially similar to the legal gestures. It

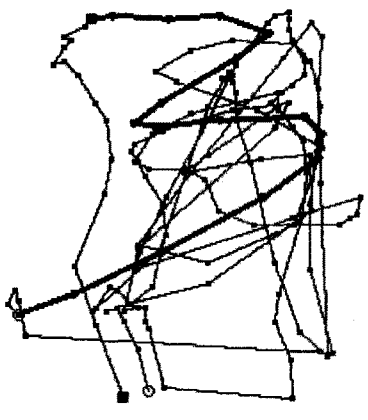


Fig. 1. A hand trajectory sample starting from the bottom (the solid square) and ending by dropping the hand to the bottom (the empty circle). A bold segmental trajectory corresponds to a digit gesture for '3'.

is often the case that typical non-gestures are embedded in gestures and vice versa to produce numerous false positives and cause oversegmentation.

The gesture recognition model designed in the work is a circular network of isolated gesture HMMs as shown in Figure 2. Basically the network models the entire sequence of human subject's motion. Non-gesture patterns are explained by the filler or garbage model denoted by 'F' in the picture. There is only one type of non-gestures and thus only one filler HMM currently. The structure of the network can be described by the following generative rule:

<GestureMotion> := <F> (<G> <F>)*
<G> := <G0> | <G1> | ··· |<G9>
<F> := <F1> | ··· | <Fk>

where <Gi> represents a gesture model for digit i and <Fi> represents a filler model where i = 1, ···, k and k is the number of filler models. According to the above definition, we can build a network of HMMs as shown in Figure 2.

4. INFERENCE

The spotter network models a small, focused set of key patterns. It is used to locate only those patterns of interest while ignoring the rest the non-gestures of no interest.

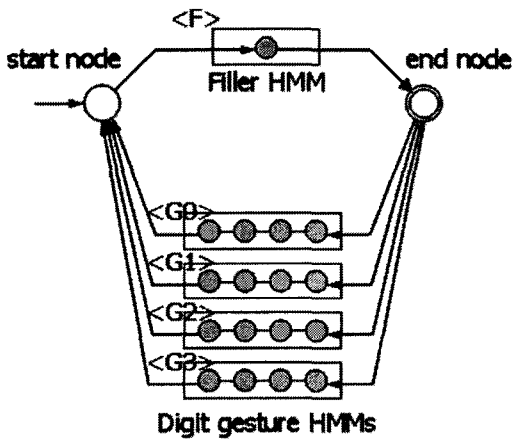


Fig. 2. Key gesture spotting network

With the network model G of Figure 2, the inference problem given a signal (an a sequence of vectors) $O = o_1 \cdots o_T$ where T is the length of the sequence is to defined as the maximization the joint probability of the input sequence and a sequence of models(HMMs) $W = w_1 \cdots w_K$, $0 < K < T$, and a state sequence $Q = q_1\ q_2 \cdots q_T$ therein. Two sequences Q and O correspond to the source and the output respectively and are paired in time. There is a two-fold maximization as follows:

$$\max_{W,Q} P(X,Q,W \mid G) = \max_{W,Q} P(X \mid Q,W,G)P(Q,W \mid G)$$

Since Q is defined within the sequence of models W which in turn is identified within the spotting network G, we can safely drop the latter variables from the above equation. The resulting equation is

$$\max_{W,Q} P(O \mid Q,W)P(Q,W \mid G) = \max_{W,Q} P(O \mid Q,W)P(Q \mid W)P(W \mid G)$$

Now let us consider a particular model sequence $W = w_1 \cdots w_K$ and a T-long complete state sequence $Q = Q_1 \ldots Q_K = (Q(w_1), \ldots, Q(w_K))$ where $Q(w_k) = q_{\tau_{k-1}+1} \ldots q_{\tau_k}$, $q_t \in S_{w_k} = \{1, \ldots, N_{w_k}\}$, is a partial or segmantal state sequence from time $\tau_{k-1}+1$ to τ_k (aligned to k-th model w_k) satisfying

$$1 \le \tau_1 \le \tau_2 \le \cdots \le \tau_k \le \cdots \le \tau_K = T.$$

Note that w_k is an HMM with N_{w_k} states and comes after $k-1$ models to model the partial input sequence from time $\tau_{k-1}+1$ to τ_k. Taking all these considerations and the conditional independence assumptions into account, we can write the above joint as

$$\max_{W,Q} P(O \mid Q(w_1) \ldots Q(w_K))P(Q \mid W)P(W \mid G) = \max_{W,Q,\tau} \prod_{k=1}^{K} P(O_{\tau_{k-1}+1}^{\tau_k} \mid Q(w_k))P(Q(w_k) \mid w_k) \times P(W \mid G)$$

where $\tau = (\tau_1, \tau_2, \ldots, \tau_K)$ is a sequence partition. The first two factors in the right hand side are none other than the formulae of HMM evaluation while the last factor corresponds to the language model score. Figure 3 shows how an input sequence signal

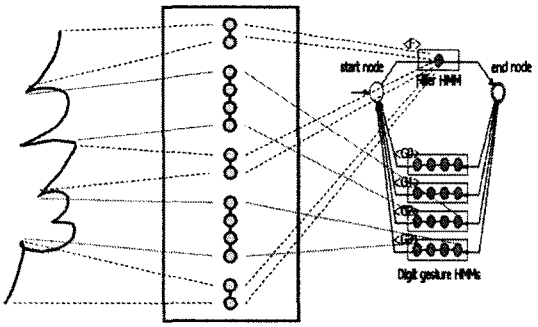


Fig. 3. An alignment between an input digit gesture and network state sequence

O (to the left) is aligned to the right G (network) via the partial state sequences $Q(w_1)$, ..., $Q(w_K)$ top to bottom in the center. Note that the state sequence comes from a circular path from the network. Now we are left with the problem of efficient computation of the above equation.

One efficient search is the one stage dynamic programming based on Viterbi algorithm. It is a modified version for the cyclic network proposed in the previous section. The algorithm proceeds sequentially in the frame synchronous fashion. For each state j of model m at time t we compute

$$\delta_t(m,j) = \max_{\{i \in S_m, l_m\}} \left\{ \delta_{t-1}(m,i) a_{ij}^m,\ \Delta_{t-1}(l_m) \pi_i^m \right\} b_j^m(o_t),$$
$$j = 1,\ldots,N_m,\ m = 1,\ldots,M,\ t = 1,\ldots,T \tag{1}$$

$$\varepsilon_t(m,j) = \begin{cases} \varepsilon_t(m,\hat{i}) + 1 & \hat{i} \in S_m \\ 1 & \hat{i} \notin S_m \end{cases} \tag{2}$$

where $\delta_t(m,j)$ is the likelihood of the partial state sequence reaching state j of model m at time t while observing the partial sequence $O_{1,t} = o_1 \cdots o_t$, and $\varepsilon_t(m,j)$ denotes the time duration since the Markov chain has entered (the initial state of) the current model m, S_m the state space of model m, and $\hat{i}$ the most likely state (at time t-1) leading to state j at time t. Finally $\Delta_t(l_m)$ is the likelihood of the node l_m – either the start node or the end node – at time t and is the node preceding the model m. It is computed as

$$\Delta_t(l) = \max_{\substack{m \\ m \to l}} \delta_t(m, N_m), \qquad l \in \{1,2\},\ t = 1,\ldots,T \tag{3}$$

where l is either the start node (l = 1) or the end node (l = 2). For each time and node we also keep the back pointer to the best model m^* and the preceding node l (which is deterministic in the current case) in

$$\Psi_t(l) = \arg\max_{\substack{m \\ m \to l}} \delta_t(m, N_m), \qquad l \in \{1,2\},\ t = 1,\ldots,T \tag{4}$$

The above two equations comprise the core of the modified Viterbi algorithm for model decoding given the input sequence $O_{1,t}$. The boundary conditions for the computation include

$$\Delta_0(1) = 1,\ \Delta_1(2) = 0 \tag{4}$$

for the start node and the end node respectively, and

$$\delta_0(m,i) = 0 \quad m = 1,\ldots,M \text{ and } i = 1,\ldots,N_m \tag{5}$$

where M is the number of HMMs comprising the network and N_m the number of states of model m.

```
Algorithm RecognizeGesture( input: O, output: { wstar_k : k = 1 ··· K} )
 1  Initialization: perform Equations (3) and (4).
 2  T = |O|, M = # models, Nw = #nodes = 2
 3  for t = 1 to T
 4     for w = 1 to M
 5        for j = 1 to Nw
 6           Compute Equations (1) and (2)
 7        end
 8     end
 9     for g = 1 to 2
10        Compute Equations (3) and (4)
11     end
12 end
13 Backtracking:
14 t = T, q = 2        // the end node
15 while g ≠ null,     // or t > 0
16    [g, wstar] = Ψ_t(g), t = t − ε_t(wstar, N_wstar)
      // wstar = filler model, skipped
17    [g, wstar] = Ψ_t(g), t = t − ε_t(wstar, N_wstar)
      // wstar = gesture model, desired
18    print wstar
19 end
```

It is noted that K, the number of gestures, is not known a priori but determined probabilistically. But we can limit the number when prior knowledge

is available.

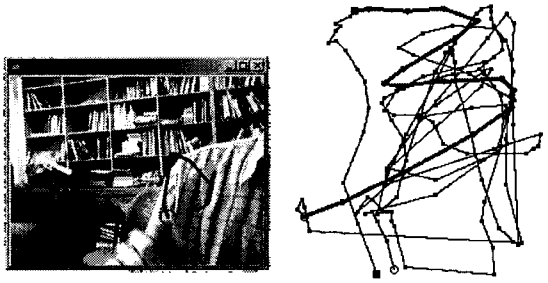
5. EXPERIMENTAL RESULTS

In the preliminary experiment a simple test of system prototype has been carried out. The dataset includes, for each of the ten digit gestures, on average 21.6 isolated gesture trajectory samples each encoded into a chain code. The test set consists of eight sequences each of which includes occasional gestures at random.

The spotting and recognition results were summarized in Table 1. Among the 24 gesture occurrences, 23 have been correctly spotted and recognized. But the overall degradation of performance was due to oversegmentation of one gesture and several false positives. Hence the proposed method performed 95.8% recall whereas the spotting precision dropped to 79.3%. It is considered that the recall rate is adequate considering the difficulty of the task. But the precision score poses a block in real world application. In fact the result is more or less anticipated due to the use of HMMs with maximum likelihood estimation-based learning. Further research, if any, should be made in this direction.

Gesture digit recognition problem is different from that of online handwriting recognition due to the absence of the concept of stroke and hence the stroke boundaries. Figure 5 shows a sample gesture of finger trajectory and the spotting result from the proposed method. The finger motion trajectory contains three digit gestures 0, 1, and 2 in order. The decoding result correctly locates the three digits in time as marked at the top of the trellis with segmental subsequences and a number of rectangles on the trellis. The horizontal axis corresponds to time flowing to the right. The sequence of blocks, when linked from left to the right, makes a complete path of state sequence. Each block in the figure simply defines the temporal region of corresponding digit gesture. The results are correct.

One the biggest problem in the gesture recognition is correct identification of non-gestures

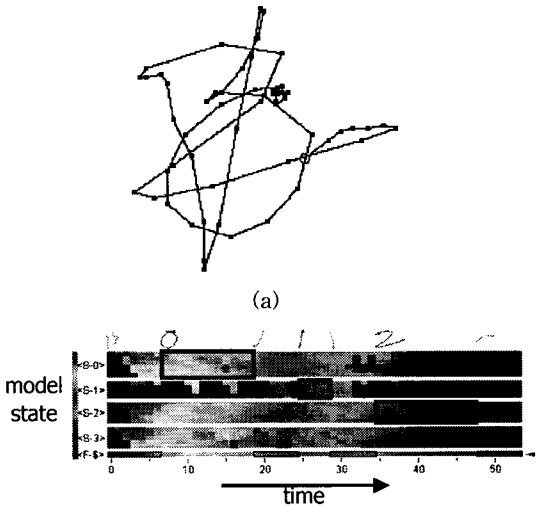


(a) (b)

Fig. 4. (a) Capturing a video, and (b) visualization of the history of the fingertip trace where a bold solid line segment corresponds to a gesture digit 3.

Table 1. Test results of digit spotting and recognition. (Substitution error: labeling error or confusion, Insertion error: introduction of nonexistent gestures, Deletion error: missing of legal gestures)

	Correct hits	Substitution error	Insertion error	Deletion error
%	76.7	3.3	10.7	3.3



(a)

(b)

Fig. 5. Sample input sequence and network decoding result.

which can be very ambiguous and often mistaken for a spurious gestures and leads to many oversegmentations. One typical source of error is show in Figure 6 where the last digit gesture '3' has been cut in the middle reporting gesture '2' and an extended filler longer than the reality.

Figure 7 shows an example of spotting failures due to two missegmentations, one in the middle (nonexistent gesture) and the other at the end (oversegmentation where the digit '2' was cut in the middle). Those problems of missegmentation error are not serious in speech recognition, but in real world gesture recognition it is a big issue in dealing with extremely variable and ambiguous and even confusing non-gestures.

The experiments thus far has been concerned about one-hand gesture recognition under the assumption that only single hand appears and the camera is so positioned as to capture only the one hand in its frame. In the follow-up research we extended the scope of recognition and tried to process both one and two hand gestures using a network of dynamic Bayesian networks (DBN) [8]. This work will be separately presented elsewhere. But a brief description of it will be given here for reference.

The DBN is a temporal extension of the Bayesian network and a generalization of the HMM with additional random variables for flexible and efficient modeling of complex real world dynamics. For the background theory of DBN, you may want to refer to several review or introductory papers on Bayesian network(BN) and DBN [8].

The DBN defined in our work consists of three hidden variables and five observable ones (Figure 8). This is in contrast to the HMM that has one hidden and one output variables. Each of the hidden variables corresponds to left hand, right hand, and the spatial configuration of the two, all of them are not explicitly encoded in the input data and thus are not observed.

In total ten artificial gestures were defined, five of them are one hand gestures and the rest two hand gestures as in Figure 9. We used similar feature vectors where some are direction codes.

Fig. 6. Typical case of segmentation error.

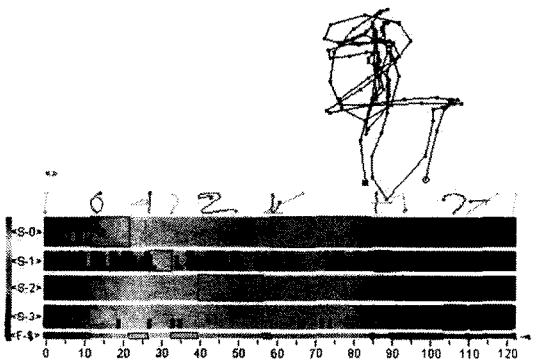


Fig. 7. Segmentation error cases: between t = 59 and 84, and at t = 109.

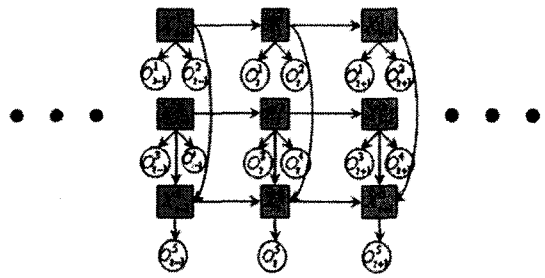


Fig. 8. The gesture DBN defined in our team.

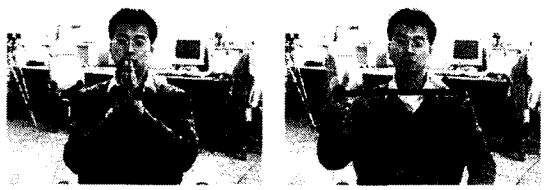


Fig. 9. Two different gestures, (left) two-hand and (right) one-hand gestures corresponding to 'open' and 'fast forward' command.

The models have been trained using Expectation -Maximization algorithm. For testing isolated gesture recognition, we employed the well-known inference methods of Inference Algorithm [8] and Junction-Tree Algorithm [9]. For gesture spotting in continuous stream, we reverted to the use of dynamic programming algorithm as described in the previous section. The initial test result was 76.4% reliability with 84.0% correct detection.

6. CONCLUSIONS

Video-based gesture recognition is a hot topic among researchers around the world. But most research results have been focused on isolated gestures and/or tried to deal with 3-dimensional features. In this paper an interesting solution to continuous stream of human motion involving occasional digit gestures was presented. The overall method is highly effective and robust. There was no effort to refine to model to deal with oversegmentation. But it is believed that the problem can be solved using state duration HMM or by introducing appropriate duration statistics into the dynamic programming computation. The presentation about the DBN at the end of the previous section is for its relevance and readers' reference. The current research direction of the field is oriented toward using the DBN. However, the HMM as the simplest type of DBN is none the less attractive because it is one of the most successful and efficient models ever used in the field of pattern recognition and computer vision.

REFERENCES

[1] Y. Wu and T. Huang, "Vision-based gesture recognition - a review," in *LNCS, Gesture Workshop*, 1999.

[2] H.K. Lee and J.H. Kim, "An HMM-based threshold model approach for gesture recognition," *IEEE Trans. Pat. Anal. and Mach. Intel.*, 21(10), 961-973, 1999.

[3] W.C. Stokoe, D. Casterline, and C.C. Cronberg, *A Dictionary of American Sign Language*, Linstok Press, Washington, DC, 1995.

[4] A. Kendon, "Current issues in the study of gestures," *The Biological Foundation of Gestures: Motor and Semiotic Aspects*, pp. 23-47, Lawrence Erlbaum Associate, Hillsdale, NJ, 1986.

[5] Y. Wu and T. Huang, "Human hand modeling, analysis and animation in the context of HCI," *IEEE Int. Conf. Image Proc.*, 1999.

[6] T. Starner, J. Weaver, and A. Pentland, "Real-time American sign language recognition using desk and wearable computer based video," *IEEE Trans. Pat. Anal. and Mach. Intel*, 1998.

[7] A. Bobick and Y. Ivanov, "Action recognition using probabilistic parsing," *IEEE Int. Conf. on Comp. Vision and Pat. Recog.*, 1998.

[8] K. P. Murphy, *Dynamic Bayesian Network: Representation, Inference and Learning*, PhD Dissertation, UC Berkeley, 2002.

[9] C. Huang and A. Darwiche, "Inference in Belief Networks: A Procedural Guide," *International Journal of Approximate Reasoning*, Vol.15, pp. 225-263, 1994.

Bong-Kee Sin

Bong-Kee Sin received B.S. degree in Mineral and Petroleum Engineering from Seoul National University, Seoul, Korea, in 1985, and M.S. degree in Computer Science from Korea Advanced Institute of Science and Technology or KAIST in 1987. Then he had worked for the Software Research Labs of Korea Telecom until February 1999. Between 1991 and 1994, he continued his study for his PhD in computer science in KAIST. In March 1999, he joined the faculty of the department of Computer Multimedia Engineering in Pukyong National University, Busan, and is now an associate professor. His general research interest includes character recognition and various applications of statistical pattern recognition methods in general and HMM in particular. His current research focus lies in the analysis and recognition of various sequential data including music signal, and video containing human activities.