GRAPHICAL MODELING OF CONDITIONAL RANDOM FIELDS FOR HUMAN MOTION RECOGNITION

Chih-Pin Liao and Jen-Tzung Chien

Department of Computer Science and Information Engineering
National Cheng Kung University, Tainan, Taiwan 70101, ROC
{cpliao, chien}@chien.csie.ncku.edu.tw

ABSTRACT

Modeling and understanding human motions are challenging in computer vision areas because the similar motions often occur at various time moments. The long-term dependences in observation data should be modeled to improve motion recognition performance. The conditional random field (CRF) is a powerful mechanism for large-span data modeling. In this paper, we present a new graphical model approach to effectively and efficiently implement CRF. Specifically, we integrate the dependent variables of a graph into a clique and build the junction tree for complex CRF structure with cycles. Using this approach, a tree inference algorithm is developed for finding the joint probability of all variables in the clique tree. In the implementation, we specify the continuous-valued hidden Markov model (HMM) parameters as the feature functions and evaluate the proposed junction tree CRF (JT-CRF) by using CMU Graphics Lab Motion Capture Database. The experimental results show that JT-CRF achieves the highest classification accuracies compared to the HMM, the maximum entropy Markov model and the linear-chain CRF.

Index Terms—Conditional random field, graphical model, junction tree, tree model inference, human motion recognition

1. INTRODUCTION

Human motion recognition is a key technology in many applications including automatic surveillance, video archival retrieval, sports analysis and human-computer interaction, etc. However, the human motion understanding is a challenging topic in computer vision areas because the ambiguity is seriously existed in non-rigid body articulation, loose clothing, mutual occlusion and the image noise due to shadow or illumination. The other problem is that the motions have concurrent structures which can be represented by some basic action units. To overcome these problems, many researchers have been working on establishing the robust understanding system, extracting the appropriate features and modeling the human motion [2][7]. In this paper, we focus on modeling the contexture information of human motions by graphical modeling of conditional random fields.

As we know, the hidden Markov model (HMM) has been widely developed for modeling human motions. The inevitable drawback is that HMM models have the strict assumption that the sequence of observations is mutually independent in temporal domain. But, in real-world applications, similar motions often occur at various time moments. The long-term dependences among observations are meaningful and should be modeled. For this consideration, the conditional random field (CRF) has been applied for large-span modeling of mutually dependent observations [5]. Also, a generative HMM model should enumerate all possible observation sequences and calculate the joint probability over observation sequence and state sequence. But, in real world, the set of observations is not always enumerable. In contrast to generative models, CRF is referred as the discriminative models where the competing information is involved. CRF constructs a global model over entire sequence for prediction of the state sequence given the observation sequence rather than estimating the maximum likelihood parameters from the observations as done by HMM.

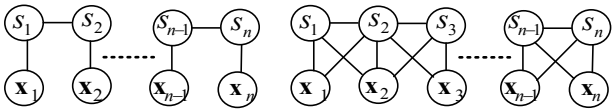
Recently, CRF has been applied for classification of human motions and gestures [9][11]. CRF is advantageous for modeling contextual information due to the reasons of (1) *avoiding the independence assumptions* and (2) *accommodating the long range interaction among observations*. The simplest CRF graph is the linear-chain structure where the optimal inference algorithm exists. However, when the random variables are highly correlated as observed in human motions, a complex graph with loops or cycles should be constructed. The exact inference of such complex CRF structure is computationally expensive. In this paper, we present a novel graphical model for rapid implementation of CRF. Importantly, we employ the junction tree algorithm to deal with the complex CRF graph with cycles. The idea of junction tree is based on a tree-like structure where the nodes in the tree are clique nodes rather than single nodes. This structure is also called hypergraph [4]. We can apply a sum-product algorithm to infer the marginal probabilities for not only a single variable but also all variables that belong to the same clique node. We merge the dependent nodes into a clique and build the junction tree. The cycle structure of CRF is converted to the clique tree. Using this procedure, a tree inference algorithm is presented for estimation of the joint probability for all variables in the graph. In the implementation of CRF, we specify the continuous-valued HMM parameters as the features and explore the relationships between observations and HMM states. Using these relationships, we construct the potential functions of CRF for model training and adopt the trained models for human motion recognition.

2. RELATED WORK

2.1 Conditional random field

Although the HMM models have been extended to human motion recognition task [12], the assumption that observations are independent was made so that the recognition performance was limited in real-world applications. Accordingly, the conditional random field was proposed to relax this assumption by modeling the dependencies between observations and their Markov states. Owing to this advantage, CRF was widely extended to solve the problems in natural language processing and recently applied to model the contextual information in human motions. A CRF can be defined as an undirected graph with each node representing a variable and obeying the property of Markov chains. The state label is not only determined by the observation but also by its neighbors in the graph. The undirected graphical structure can be factorized into a normalized product of potential functions. As shown in Figure 1, the structure of CRF can be a simple graph as a linear chain or a complex graph which was used for human motion

recognition [9]. Similar to the notations in HMM, the observation sequence and state sequence are denoted by $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ and $\mathbf{S} = \{s_1, \cdots, s_n\}$, respectively.



(a) linear-chain CRF structure (b) complex CRF structure

Figure 1 Different structures of CRF

Different from HMM, the state s_t of CRF in Figure 1(b) is decided by considering its neighbors s_{t-1}, $\mathbf{x}_{t-1}$, $\mathbf{x}_t$, $\mathbf{x}_{t+1}$ and s_{t+1}. In [9], the graph of a linear-chain structure was considered. The conditional probability was defined by

$$p(\mathbf{S} \mid \mathbf{X}, \lambda) = \frac{\prod_{t=1}^{n} \exp(F(s_{t-1}, s_t, \mathbf{X}, \lambda))}{Z(\mathbf{X}, \lambda)}, \quad (1)$$

where $Z(\mathbf{X}, \lambda)$ was a normalization term and the term $\exp(F(s_{t-1}, s_t, \mathbf{X}, \lambda))$ expressed the potential function of two successive states and

$$F(s_{t-1}, s_t, \mathbf{X}, \lambda) = \sum_{a=1}^{A} \alpha_a g_a(s_t, \mathbf{X}) + \sum_{b=1}^{B} \beta_b f_b(s_{t-1}, s_t, \mathbf{X}). \quad (2)$$

Their feature functions in potential function were specified by

$$f_b(s_{t-1}, s_t, \mathbf{X}) = f_b(s_{t-1}, s_t) = \mathrm{I}[s_{t-1} = m_1 \wedge s_t = m_2], m_1, m_2 \in \mathbf{S}, \quad (3)$$

$$g_a(s_t, \mathbf{X}) = \mathrm{I}[s_t = m]\mathbf{x}_{t-j}[i], m \in \mathbf{S}, i \in \{1, \cdots, d\}, j \in [-W, W]. \quad (4)$$

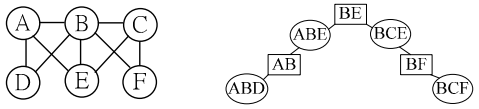
In (2), (3) and (4), A depended on the state numbers and B depended on the size of observation dimension and state number. Also, the function f was defined as an identity function $\mathrm{I}[\cdot]$ and its value was not equal to 0 when the states m_1 and m_2 were concurrent. The function g indicated whether the d dimensional observation $\mathbf{x}$ was assigned to the state m at time t. The temporal context window $2W+1$ was used to identify the range size W around the current observation and the context information influencing the same state. We estimate the parameter set $\lambda = \{\alpha_a, \beta_b\}$ by differentiating the log-conditional probabilities with respect to all parameters. The generalized iterative scaling (GIS) algorithm [3] is feasible to solve this estimation problem.

2.2 Junction tree and joint probability

It is known that the inference problem on tree graph can be solved exactly. But for a graph with cycles, we need to cluster the dependent nodes and form a clique tree. Such clique tree guarantees the consistent computation of the join probability because the same variable may appear at different cliques. The junction tree was used to infer the complex graph with cycles [8][10]. In general, a graph has a junction tree if and only if it is a triangulated graph [4]. Here, we shorten the discussion by assuming that the graph was already triangulated as Figure 2(a). For this example, we can build junction tree as displayed in Figure 2(b). In this junction tree, the original nodes in a clique are merged into a clique node, e.g. node ABD, ABE BCE, and BCF. The separator nodes AB, BF and BE separating the neighboring clique nodes are also created. Finally, an exact tree-based inference algorithm can be run on the transformed graph. We obtain the joint probability of all variables as

$$p(A,B,C,D,E,F) = \frac{\varphi(A,B,D)\varphi(A,B,E)\varphi(B,C,E)\varphi(B,C,F)}{\varphi(A,B)\varphi(B,E)\varphi(B,F)}, \quad (5)$$

where $\varphi(\cdot)$ is the potential function of a node.



(a) Triangulated Graph (b) Corresponding junction tree

Figure 2 Junction tree of a triangulated graph

3. CRF GRAPHICAL MODELING

In general, the relations between observations and states are complex in human motion recognition. This complexity goes beyond the capability that the linear-chain structure can handle. To develop a sophisticated CRF framework, we consider the complex CRF structure with cycles and use the junction tree algorithm for CRF graphical modeling. Using the transformed tree structure, the algorithm of finding joint probability in a graph is constructed. Also, we assign different feature functions from those in conventional CRF so as to fit the continuous variables and control the size of parameter set.

3.1. Building junction tree for CRF model inference

In this study, we take into account the complex CRF structure in Figure 1(b). Each Markov state is affected by three sequential observations. First, we transform this CRF structure into a junction tree graph. Because the original graph has been already a triangulated graph, we merge the variables in each clique into a node. Then, the original graph is converted to a tree-like structure as given in Figure 3.

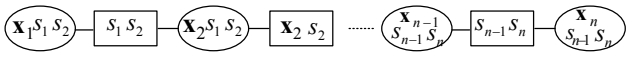


Figure 3 Junction tree for CRF

Given this junction tree, we can calculate the joint probability $p(\mathbf{X}, S)$ of the observation sequence and the state sequence by

$$\frac{\varphi(\mathbf{x}_1, s_1, s_2)\varphi(\mathbf{x}_2, s_1, s_2)\varphi(\mathbf{x}_2, s_2, s_3)\varphi(\mathbf{x}_3, s_2, s_3)\cdots\varphi(\mathbf{x}_n, s_{n-1}, s_n)}{\varphi(s_1, s_2)\varphi(\mathbf{x}_2, s_2)\varphi(s_2, s_3)\cdots\varphi(s_{n-1}, s_n)} \quad (6)$$

We can rewrite it as

$$\prod_{t=1}^{n-1}\psi(\mathbf{x}_t, \mathbf{x}_{t+1}, s_t, s_{t+1}) = \prod_{t=1}^{n-1}\frac{\varphi(\mathbf{x}_t, s_t, s_{t+1})\varphi(\mathbf{x}_{t+1}, s_t, s_{t+1})}{\varphi(s_t, s_{t+1})\varphi(\mathbf{x}_{t+1}, s_{t+1})}. \quad (7)$$

For each time step, the potential functions $\varphi(\cdot)$ can be obtained as

$$\varphi(\mathbf{x}_t, s_t, s_{t+1}) = \exp(\alpha_{t1} g(\mathbf{x}_t, s_t) + \beta_{t1} g(\mathbf{x}_t, s_{t+1}) + \gamma_{t1} f(s_t, s_{t+1}))$$

$$\varphi(\mathbf{x}_{t+1}, s_t, s_{t+1}) = \exp(\alpha_{t2} g(\mathbf{x}_{t+1}, s_t) + \beta_{t2} g(\mathbf{x}_{t+1}, s_{t+1}) + \gamma_{t2} f(s_t, s_{t+1}))$$

$$\varphi(s_t, s_{t+1}) = \exp(\gamma_{t3} f(s_t, s_{t+1}))$$

$$\varphi(\mathbf{x}_{t+1}, s_{t+1}) = \exp(\alpha_{t3} g(\mathbf{x}_{t+1}, s_{t+1}))$$

The posterior probability is finally calculated by

$$p(\mathbf{S} \mid \mathbf{X}, \lambda) = \frac{1}{p(\mathbf{X})}\prod_{t=1}^{n-1}\psi(\mathbf{x}_t, \mathbf{x}_{t+1}, s_t, s_{t+1})$$

$$= \frac{1}{p(\mathbf{X})}\prod_{t=1}^{n-1}\frac{\left\{\begin{array}{l}\exp(\alpha_{t1} g(\mathbf{x}_t, s_t) + \beta_{t1} g(\mathbf{x}_t, s_{t+1}) + \gamma_{t1} f(s_t, s_{t+1})) \\ \cdot \exp(\alpha_{t2} g(\mathbf{x}_{t+1}, s_t) + \beta_{t2} g(\mathbf{x}_{t+1}, s_{t+1}) + \gamma_{t2} f(s_t, s_{t+1}))\end{array}\right\}}{\exp(\gamma_{t3} f(s_t, s_{t+1})) \cdot \exp(\alpha_{t3} g(\mathbf{x}_{t+1}, s_{t+1}))} \quad (8)$$

$$= \frac{1}{p(\mathbf{X})}\prod_{t=1}^{n-1}\left\{\begin{array}{l}\exp(\alpha_{t1} g(\mathbf{x}_t, s_t) + \beta_{t1} g(\mathbf{x}_t, s_{t+1}) + \alpha_{t2} g(\mathbf{x}_{t+1}, s_t) \\ + (\beta_{t2} - \alpha_{t3}) g(\mathbf{x}_{t+1}, s_{t+1}) + (\gamma_{t1} + \gamma_{t2} - \gamma_{t3}) f(s_t, s_{t+1}))\end{array}\right\}$$

And $p(\mathbf{X})$ is obtained from all possible state sequences as

$$p(\mathbf{X}) = \sum_{all\ \mathbf{S}} p(\mathbf{X},\mathbf{S}) = \sum_{all\ \mathbf{S}} \prod_{t=1}^{n-1} \psi(\mathbf{x}_t, \mathbf{x}_{t+1}, s_t, s_{t+1}) . \quad (9)$$

3.2. Parameter estimation

In (2) and (8), f functions are used to express the features of two neighboring states and g functions are used to express those of observation-state. At each time t, we have four g functions and one f function in (8). For ease of expression, we use the notation of parameter vectors $\mathbf{a}_t$ and $\mathbf{b}_t$ to replace different α, β, γ at time t. Then (8) can be expressed by

$$\frac{1}{p(\mathbf{X})} \prod_{t=1}^{n-1} \exp(\mathbf{a}_t^T \cdot \mathbf{f}(s_t, s_{t+1}) + \sum_{i=1}^{4} \mathbf{b}_{ti}^T \cdot \mathbf{g}_i) . \quad (10)$$

In a general expression, we rewrite (10) as

$$\sum_{j=1}^{B} \boldsymbol{\lambda}_j{}^T \cdot \mathbf{F}_j(\mathbf{S},\mathbf{X}) = \sum_{t}^{n-1} \mathbf{a}_t^T \mathbf{f}(s_t, s_{t+1}) + \sum_{t}^{n-1} \sum_{i=1}^{A} \mathbf{b}_{ti}^T \cdot \mathbf{g}_i , \quad (11)$$

where B is number of states in the model and

$$\mathbf{F}_j(\mathbf{S},\mathbf{X}) = \sum_{t=1}^{n-1} \mathrm{I}[s_t = j] \cdot (\mathbf{a}_t^T \cdot \mathbf{f}(s_t, s_{t+1}) + \sum_{i=1}^{A} \mathbf{b}_{ti}^T \cdot \mathbf{g}_i) . \quad (12)$$

Our goal is to estimate CRF parameters by maximizing the objective function

$$L(\boldsymbol{\lambda}) = \sum_{k=1}^{N} \left[\log \frac{1}{p(\mathbf{X}^{(k)})} + \sum_{j=1}^{B} \boldsymbol{\lambda}_j{}^T \cdot \mathbf{F}_j(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}) \right], \quad (13)$$

with all observation-state pairs $\{(\mathbf{S}^{(1)}, \mathbf{X}^{(1)}) \ldots (\mathbf{S}^{(N)}, \mathbf{X}^{(N)})\}$ in training set.

Here, the GIS algorithm [3] is applied to find CRF solution. We differentiate (13) with respect to each parameter [5] and obtain

$$\begin{aligned} &\sum_{\mathbf{X},\mathbf{S}} \tilde{p}(\mathbf{X},\mathbf{S}) \cdot \mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k) - \sum_{\mathbf{X},\mathbf{S}} \tilde{p}(\mathbf{X}) p(\mathbf{S} \mid \mathbf{X}, \lambda) \cdot \mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k) \\ &= E_{\tilde{p}(\mathbf{X},\mathbf{S})}[\mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k)] - E_{p(\mathbf{S}|\mathbf{X},\theta)}[\mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k)]. \end{aligned} \quad (14)$$

The objective function is optimal when the empirical expectation equals to the expectation of true model

$$E_{\tilde{p}(\mathbf{X},\mathbf{S})}[\mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k)] = E_{p(\mathbf{S}|\mathbf{X},\theta)}[\mathbf{F}_j(\mathbf{S}^k, \mathbf{X}^k)] . \quad (15)$$

The notation $\tilde{p}(\cdot)$ means the empirical probability function. Then we update parameters by the following steps:

1. Initialize $\lambda^{(0)} = 1$.
2. Calculate the empirical expectation F_s.
3. Determine the expectation for true model E_s
4. Update the parameter until convergence

$$\lambda^{(j+1)} = \lambda^{(j)} + \eta \log(F_s / E_s^{(j)}) . \quad (16)$$

In GIS algorithm, η is a learning rate which is decreased by iterations so as to guarantee the convergence of learning. Different from conventional CRF developed for discrete variables, we adopt the probability parameters as the continuous features. We start from a HMM structure. A well-trained HMM is prepared for finding the states corresponding to the observations. Next, the log-likelihoods and the transition probabilities are calculated for representing the observation-to-state feature functions g and state-to-state feature functions f [1]. After GIS algorithm converges, we estimate the CRF parameters. We can recognize an observation sequence by calculating the posterior probability for class c_i given the observation sequence $\mathbf{X}$. The decision function is calculated by

$$P(C = c_1 \mid \mathbf{X}) = \frac{\sum_{s \in c_1} \exp(\lambda_s \cdot F_s(\mathbf{S},\mathbf{X}))}{\sum_{all\ c_i} \sum_{s' \in c_i} \exp(\lambda_{s'} \cdot F_{s'}(\mathbf{S},\mathbf{X}))} , \quad (17)$$

where c_i is class i and s' expresses all possible state sequences in i. The classification output is determined according to this posterior probability.

4. EXPERIMENTS

4.1. Experimental setup

In the experiments, we carried out the proposed method by using image features of human body which were extracted from video frames directly rather than reconstructing 2D or 3D model of human motion. We evaluated the performance of HMM, MEMM, linear-chain CRF (LC-CRF) and the proposed junction-tree CRF (JT-CRF) on human motion recognition by using two public-domain motion databases; CMU Graphic Lab Motion Capture Database (http://mocap.cs.cmu.edu/) and TwoHandManip Gesture Database from IDIAP (http://www.idiap.ch/resources/twohanded/). We selected 12 single body's motions contained in the CMU database: Walk, Run, Jump, Climb down, Climb up, Place Tee, Swing, Putt, Boxing, Wash window, Stand up and Sit down. Each motion had 5 section videos with duration at most 4 seconds long at resolution 240x320. The sampling rate was 30 frames per second. Using the TwoHandManip database, we adopted 7 gesture classes: Front, Back, Push, Up, Down, Left and Right in the evaluation. We selected 70 section videos for all gestures which were captured by the same camera from original database. These videos were captured with 25 frames per second and duration of 1 or 2 seconds long at resolution 240x320. On both databases, we randomly selected 3 sections for each motion class to train the models and the remaining sections were used as test data. There were 24 and 49 sections in CMU and IDIAP databases, respectively. We applied a preprocessing stage of extracting the features from the motion video frames for a view-dependent human motion recognition system. This stage is shown as Figure 4 on different databases. The background information was provided on CMU database that we detected the foreground and extracted the human body exactly. On TwoHandManip Database, we detected both hands by using the color information and preserved the spatial information of two hands in the whole frame. After extracting the interested parts from the video frames, we down-sampled the resolution of images and transformed the images to ones with 96 dimensions for reducing the computation cost.

(a) Human motion (b) Gesture

Figure 4 (a) Body in CMU Graphic Lab Motion Capture Database; (b) Gesture in IDIAP TwoHandManip Database

Also, we modeled each motion with a HMM model which was constructed by three states. We used a single Gaussian distribution to model each state in a HMM model. The diagonal covariance matrices were adopted. The transitions of states were limited as left to right and the skipping of state was not allowed in the model. After training HMM models for all motions, we run Viterbi algorithms to obtain log-likelihood of each training sample. Sequentially, we gathered all the frames assigned to the same state for CRF models training. The iteration number in GIS algorithm was fixed as 300 in our experiments. In the test phase, the test

frame sequences were aligned by running Viterbi algorithm with the well-trained HMM model. We accordingly had the features to calculate the posterior probabilities for each CRF model. We retrieve the most likely class with the highest posterior probability.

4.2. Experimental results

In Table 1, we compare HMM and LC-CRF with different window size W by evaluating them on CMU Motion database. We find that considering the context information does improve the recognition accuracy. The accuracy is increased when the window size is enlarged. The improvement is saturated at the case of $W = 2$.

Table 1 Classification accuracies of HMM and LC-CRF on using CMU database

	Accuracy
HMM	62.5%
LC-CRF (W =0)	75.0%
LC-CRF (W =1)	79.2%
LC-CRF (W =2)	79.2%

On using CMU and IDIAP databases, we compare the recognition accuracies of HMM, LC-CRF and the proposed JT-CRF in Figure 5. If the state in JT-CRF is affected by 3 observations, we denote it as JT-CRF3. We find that JT-CRF outperform other methods. Furthermore, we extend JT-CRF to the case (JT-CRF5) that each state is affected by 5 observations. We even obtain good performance on using IDIAP database.

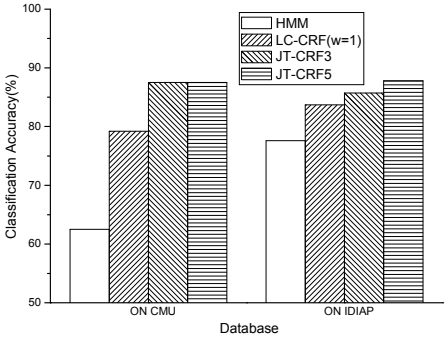


Figure 5 Classification accuracies of HMM, CRF and JTCRF on using two databases

We also carry out the maximum entropy Markov model (MEMM) [6] on using IDIAP database. In HEMM algorithm, the normalization is considered at each time moment rather than the whole time sequence. In the results, we obtain the performance of MEMM (77.6%) and JT-CRF3 (87.5%) which is better than HMM (75.5%). In addition, the definition of feature functions affects the classification performance. Basically, the use of likelihood function as feature has much smaller size of parameters than that of raw image data [9]. We find that the classification accuracy of CRF using the traditional raw data as feature is 61.2% and that using the HMM parameters as feature is 83.7%. This is because that the dimension of raw image data is too high to obtain sufficient training data for reliable CRF modeling.

5. COCLUSIONS

In this study, we presented a complex CRF model rather than the traditional linear-chain CRF. This model was able to describe complex dependence of several variables. By using the junction tree algorithm from graphical theory, we performed transformation and inference for the graph structure with cycles. We developed a new JT-CRF algorithm for human motion recognition. Also, we established the HMM-based feature function so as to fit the continuous variables and data segmentations without tagging manually. The definition of feature functions was different from those in conventional CRF methods. We improved the performance especially when the training data was limited in the collected databases. Compared to HMM, HEMM and LR-CRF, better performance were archived by JT-CRF in the experiments on two public-domain motion databases. In the future, we are developing the variational Bayesian inference for CRF modeling. Also, we are extracting robust features for CRF and integrating multi-streams where each stream models the motion of a part of human body. A larger database will be adopted for evaluation of human motion recognition.

7. REFERENCES

[1] C.-H. Chueh and J.-T. Chien, "Maximum entropy modeling of acoustic and linguistic features", *Proc. of ICASSP*, vol. 1, pp. 1061-1064, 2006.

[2] R. Cucchiara, C. Grana, G. Tardini and R. Vezzani, "Probabilistic people tracking for occlusion handling", *Proceedings of International Conference on Pattern Recognition*, vol.1, pp. 132-135, 2004.

[3] J. N. Darroch and D. Ratcliff, "Generalized Iterative Scaling for Log-Linear Models", *The Annals of Mathematical statistics*, vol. 43, no.5, pp. 1470-1480, 1972.

[4] M. Jordan, "Graphical models", *Statistical Science*, vol. 19, pp. 140–155, 2004.

[5] J. Lafferty, A. McCallum and F. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data", *Proceeding of 18th International Conference on Machine Learning*, 2001.

[6] A. McCallum, D. Freitag and F. Pereira, "Maximum entropy Markov models for information extraction and segmentation", *Proceedings of International Conference on Machine Learning*, 2000.

[7] M. M. Rahman and S. Ishikawa, "Robust appearance-based human action recognition", *Proceedings of International Conference on Pattern Recognition*, pp. 165-168, 2004.

[8] S. Sakti, K. Markov and S. Nakamura, "An HMM model incorporating various additional knowledge source", *Proc. of INTERSPEECH*, pp. 2117-2120, 2007.

[9] C. Sminchisescu, A. Kanaujia, Z. Li and D. Metaxas, "Conditional models for contextual human motion recognition", *Proceedings of International Conference on Computer Vision*, pp. 1808-1815, 2005.

[10] M. J. Wainwright and M. I. Jordan, "A variational principle for graphical models", *Chapter 11 in New Directions in Statistical Signal Processing*, *MIT Press*, 2005.

[11] S. B. Wang, A. Quattoni, L. P. Morency, D. Demirdjian and T. Darrel, "Hidden conditional random fields for gesture recognition", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1521-1527, 2006.

[12] J. Yang, Y. Xu and C. S. Chen, "Human action learning via hidden Markov model", *IEEE Transactions on Systems, Man and Cybernetics*, pp. 34-44, 1997.