Relative Hidden Markov Models for Evaluating Motion Skills

Qiang Zhang and Baoxin Li
Computer Science and Engineering
Arizona State Univerisity, Tempe, AZ 85281
qzhang53,baoxin.li@asu.edu

Abstract

This paper is concerned with a novel problem: learning temporal models using only relative information. Such a problem arises naturally in many applications involving motion or video data. Our focus in this paper is on video-based surgical training, in which a key task is to rate the performance of a trainee based on a video capturing his motion. Compared with the conventional method of relying on ratings from senior surgeons, an automatic approach to this problem is desirable for its potential lower cost, better objectiveness, and real-time availability. To this end, we propose a novel formulation termed Relative Hidden Markov Model and develop an algorithm for obtaining a solution under this model. The proposed method utilizes only a relative ranking (based on an attribute of interest) between pairs of the inputs, which is easier to obtain and often more consistent, especially for the chosen application domain. The proposed algorithm effectively learns a model from the training data so that the attribute under consideration is linked to the likelihood of the inputs under the learned model. Hence the model can be used to compare new sequences. Synthetic data is first used to systematically evaluate the model and the algorithm, and then we experiment with real data from a surgical training system. The experimental results suggest that the proposed approach provides a promising solution to the real-world problem of motion skill evaluation from video.

1. Introduction

Understanding human motion is an important task in many fields including sports, rehabilitation, surgery, computer animation and dance. One key problem in such applications is the analysis of skills associated with body motion. In domains such as dance, sports and surgery, the motion of experts differs considerably from that of novices. Sensory data that capture such motion may be analyzed to provide a computational understanding of such differences, which may in turn be used to facilitate tasks such as skill evaluation and training. For example, [4] utilized control trajectories and motion capture data for human skill analysis, [20] reported motion skill analysis in sports using data from motion sensors, [18] studied computational skill rating in manipulating robots, and [15] considered hand movement analysis for skill evaluation in console operation.

Among those fields, surgery is one domain where motion expertise is of the primary concern. Often a surgeon has to go through lengthy training programs that aim at improving his/her motion skills. As a result, simulation-based training platforms have been developed and widely adopted in surgical education. For example, the Fundamentals of Laparoscopic Surgery (FLS) Box (`www.flsprogram.org`) has practically become a standard training platform for minimally-invasive surgery. Accordingly, computational approaches have been developed for motion skill analysis on such training platforms. Recognizing the sequential nature of motion data, many analysis approaches utilize state-transition models, such as the Hidden Markov Model (HMM). For example, [14] provided an HMM-based method to evaluate surgical residents' learning curve. The method first constructs different HMMs for each different levels of expertise, and then calculates a probability distance between the expert and a novice resident. The magnitude of the probability distance is used to rate the level of the novice resident. HMM was also adopted in [7] to measure motion skills in surgical tasks, where the video is first segmented into basic gestures based on velocity and angle of movement, with segments of the gestures corresponding to the states of an HMM.

One practical difficulty in these approaches is that they require the skill labels for the training data since the HMMs are typically learned from data of each skill level. Labeling the skill of a trainee is currently done by senior surgeons, which is not only a costly practice but also one that is subjective and less quantifiable. Thus it is difficult, if not impossible, to obtain sufficient and consistent skill labels for a large amount of data for reliable HMM training. This problem has also been encountered in other fields such as image classification. For example, in [12], it was argued that using binary label to describe the image is not only too restrictive but also unnatural and thus relative visual attributes were used and classifiers were trained based on such features. Relative information has also been used in other applications, e.g., distance metric learning [16], face verification [9], and human-machine interaction [13].

In this paper, we propose a novel formulation termed *Relative Hidden Markov Model* and develop an algorithm for obtaining a solution under this model. The proposed method utilizes only a relative ranking (based on an attribute of interest, or motion skill in the surgical training application) between pairs of the inputs, which is easier to obtain and often more consistent. This is especially useful for the applications like video-based surgical training, where the trainees go through a series of training sessions with their skills get improved over time, and thus the time of the sessions would already provide a natural relative ranking of the skills at the corresponding time. The proposed algorithm effectively learns a model from the training data so that the attribute under consideration (i.e., the motion skill in our application) is linked to the likelihood of the inputs under the learned model. The learned model can then be used to compare new data pairs. For evaluation, we first design synthetic experiments to systematically evaluate the model and the algorithm, and then experiment with real data captured on a commonly-used surgical training platform. The experimental results suggest that the proposed approach provides a promising solution to the real-world problem of motion skill evaluation from video.

The key contribution of the work lies in the novel formulation of learning temporal models using only relative information and the proposed algorithm for obtaining solutions under the formulation. Additional contributions include the specific application of the proposed method to the problem of video-based motion skill evaluation in surgical training, which has seen increasing importance in recent years.

2. Related Work

In this section, we review two categories of existing work, discriminative learning for hidden Markov models and learning based on relative information, which are most related to our effort. Distinction between our proposed method and the reviewed work will be briefly stated.

Discriminative learning for HMM: Maximum-likelihood methods for learning HMM (e.g., the forward-backward algorithm) in general do not guarantee the discrimination ability of the learned models. To this end, several discriminative learning methods for HMM have been proposed. In [3], a discriminative training method for HMM was proposed based on perceptron algorithms. The methods iterates between the Viterbi algorithm and the additive update of the models. Hidden Markov Support Vector Machine (HM-SVM) was proposed in [1], which combines SVM with HMM to improve the discrimination power of the learned model. These methods are "supervised" in nature, and thus the labeling of the state sequence is required for the training data, which limits their practical use. In [17], another discriminative learning method for HMM was proposed, which only requires the labels of the training sequences. The method initializes the HMMs with maximum-likelihood method and then updates the models with SVM. One drawback is that, the updated models do not always lead to valid HMMs, which could be problematic for a physics-driven problem where the model states have real meanings (like the gesture elements in [7]). Our proposed method requires neither the labeling of the states nor the class label for the training sequences, which are difficult to obtain or even not accessible in applications. Instead, only a relative ranking of the training data is used, and the resultant model is a valid HMM.

Learning with relative information: Several methods for learning with relative information have been proposed recently. In [16], a distance metric is learned from relative comparisons. Considering the limited training examples for object recognition, [19] proposes an approach based on comparative objective similarities, where the learned model scores high for objects of similar categories and low for objects of dissimilar categories. In [9], comparative facial attributes were learned for face verification. The method of [12] learns relative attributes for image classification and the problem is formulated as a variation of SVM. Similar idea was also been used in [13] for the purpose of human-machine interaction. In [8], relative attributes feedback, e.g., "Shoe images like these, but sportier", is used to improve the performance of image search. Relative information between scene categories has also been used to enhance the performances of scene categorization in [6]. These approaches are mostly for image-based attributes, whereas our current task is on modeling sequential data, for which it is natural to assume that the most relevant attributes (e.g., motion skills) are embedded in a temporal structure. This is what our proposed method attempts to address.

3. Basic Notations of HMM

In this section, we briefly describe HMM and introduce some basic notations that will be used later. An HMM can be defined by a set of parameters: the initial transition probabilities $\pi \in \mathbb{R}^{K\times 1}$, the state transition probabilities $A \in \mathbb{R}^{K\times K}$ and the observation model $\{\phi_k\}_{k=1}^{K}$, where K is the number of states. There are two central problems in HMM: 1) learning a model from the given training data; and 2) evaluating the probability of a sequence under a given model, i.e., the decoding problem.

In the **learning problem**, one learns the model (θ) by maximizing the likelihood of the training data ($\mathbb{X}$):

$$\theta^* : \max_\theta \prod_{\mathbf{X}^i \in \mathbb{X}} p(\mathbf{X}^i|\theta) \sim \max_\theta \sum_{\mathbf{X}^i \in \mathbb{X}} \log p(\mathbf{X}^i|\theta) \quad (1)$$

where $\mathbb{X}$ is the set of i.i.d. training sequences.

One efficient solution to the above problem is the well-known Baum-Welch algorithm [2]. Another scheme, namely the segmental K-means algorithm [5], may also be used to seek a solution, and it has been shown that the likelihoods under models estimated by either of the two algorithms are very close [5]. When the training data include sequences of multiple categories, multiple models would be learned and each model will be learned from data of each category independently.

In the **decoding problem**, given a hidden Markov model, one needs to determine the probability of a given sequence $\mathbf{X}$ being generated by the model. Generally we are more interested in the probability associated with the optimal state sequence ($\mathbf{z}^*$), i.e., $p(\mathbf{X}, \mathbf{z}^*|\theta) = \max_{\mathbf{z}} p(\mathbf{X}, \mathbf{z}|\theta)$. The optimal state path can be found via the Viterbi algorithm. To use HMM in classification, we first compute the probability of the given sequence drawn from each model, then we choose the model yielding the maximal probability.

4. Proposed Method

Based on the previous discussion, we are concerned with a new problem of learning temporal models using only relative information. This is a problem arising naturally in many applications involving motion or video data. In the case of video-based surgical training, the focus is on learning to rate/compare the performance of the trainees from recorded videos capturing their motion. To this end, in recognition of some fruitful trials of HMMs in this application domain, we propose to formulate the task as one of learning a *Relative Hidden Markov Model*, which not only maximizes the likelihood of the training data, but also maintains the given relative rankings of the input pairs. In its most basic form, the proposed model can be formally expressed as (following the notations defined in Eqn. (1))

$$\begin{aligned} \theta \quad &: \quad \max_\theta \prod_{\mathbf{X}^i \in \mathbb{X}} p(\mathbf{X}^i|\theta) \qquad (2) \\ \text{s.t.} \quad &: \quad F(\mathbf{X}^i, \theta) > F(\mathbf{X}^j, \theta), \forall (i,j) \in \mathbb{E} \end{aligned}$$

where $F(\mathbf{X}, \theta)$ is a score function for data $\mathbf{X}$ given by model θ, which is introduced to maintain the relative ranking of the pair $\mathbf{X}^i$ and $\mathbf{X}^j$, and $\mathbb{E}$ is the set of given pairs with prior ranking constraint. Different score functions may be defined, as described in the following subsections.

From this formulation, the difference between the proposed method and any of the existing HMM-based methods is obvious. In an existing HMM-based method, a set of models is trained using the training data of each category independently. That is, explicit class labels are required for each training sequence. The proposed model has the following unique features:

- The model does not require explicit class labels. What needed is only a relative ranking.
- The model explicitly considers the ranking constraint between given data pairs, whereas independently-trained HMMs in existing methods can't guarantee it.
- Only one model is learned for the entire set of data. There are two benefits: more data for training and less computation during testing.

Our method is also different from the existing work on learning with relative attributes in that it models sequential data and the relative ranking information is capsulated in a temporal dynamic model of HMM (albeit new algorithms are thus called for), which has demonstrated performance in modeling physical phenomena like human movements.

In the following subsections, we present two instantiations of the general model expressed in Eqn. (2), and develop the corresponding algorithms in each case. It will become clear that the first model (Sec. 4.1), while being intuitive, has some practical difficulties, which motivated us to develop the improved model of Sec. 4.2. Both models/algorithms are presented (and evaluated later in Sec. 5) for the progressive nature of the methods and for facilitating the understanding of the improved model and algorithm of Sec. 4.2, which is the recommended solution.

4.1. The Baseline Model

One intuitive choice of the score function in Eqn. (2) is the data likelihood, i.e., $F(\mathbf{X}^i, \theta) = p(\mathbf{X}^i|\theta)$. With this, the formulation in Eqn. (2) can be rewritten as

$$\begin{aligned} \theta \quad &: \quad \max_\theta \prod_{\mathbf{X}^i \in \mathbb{X}} p(\mathbf{X}^i|\theta) \qquad (3) \\ \text{s.t.} \quad &: \quad p(\mathbf{X}^i|\theta) > p(\mathbf{X}^j|\theta), \forall (i,j) \in \mathbb{E} \end{aligned}$$

It has been proved in [11] that, the marginal likelihood is dominated by the likelihood with the optimal path and their difference decreases exponentially with regarding to the length (number of frames) of sequence. This idea was used in segmental K-means algorithm and similarly we can approximate the marginal data likelihood $p(\mathbf{X}|\theta)$ by the likelihood with optimal path $p(\mathbf{X}, \mathbf{z}^*|\theta)$ (when there is no ambiguity, we will use $\mathbf{z}$ for $\mathbf{z}^*$), which can be written as:

$$\begin{aligned} \log p(\mathbf{X}, \mathbf{z}|\theta) \quad &= \quad \log p(\mathbf{X}_1|\phi_{\mathbf{z}_1}) + \log \pi(\mathbf{z}_1) \\ &+ \quad \sum_{t=2}^{T} [\log p(\mathbf{X}_t|\phi_{\mathbf{z}_t}) + \log \mathbf{A}(\mathbf{z}_t|\mathbf{z}_{t-1})] \quad (4) \end{aligned}$$

If we assume a multinomial observation model, i.e., $p(\mathbf{X}_t|\phi_{\mathbf{z}_t}) = \prod_{d=1}^{D} \phi_{\mathbf{z}_t}(l)^{\mathbf{X}_t(l)}$, where D is the dimension of each frame, $\mathbf{X}_t(l)$ is the l_{th} dimension of $\mathbf{X}_t$ and $\phi_{\mathbf{z}_t}$ is the parameters of observation model with State $\mathbf{z}_t$. We further define the following variables for each sequence $\mathbf{X}^i$:

$$\begin{aligned}
\mathbf{n}^i \in \mathbb{R}^{K\times 1} \quad &: \quad \mathbf{n}^i(k) = \delta(\mathbf{z}_1^i = k) \\
\mathbf{O}^i \in \mathbb{R}^{K\times D} \quad &: \quad \mathbf{O}^i(k,d) = \sum_{t:\mathbf{z}_t = k} \mathbf{X}_t^i(d) \\
\mathbf{M}^i \in \mathbb{R}^{K\times K} \quad &: \quad \mathbf{M}^i(k,l) = \sum_{t=2}^{T} \delta(\mathbf{z}_{t-1}^i = k)\delta(\mathbf{z}_t^i = l)
\end{aligned}$$

where $\delta(\cdot)$ is Dirac Delta function. Then the log likelihood with the optimal path can be written as:

$$\begin{aligned}
\log p(\mathbf{X}^i, \mathbf{z}^i|\theta) &= \sum_l \mathbf{n}^i(l)\log\pi(l) + \sum_{k,l}\mathbf{M}^i(k,l)\log\mathbf{A}(k,l) \\
&+ \sum_{k,d}\mathbf{O}^i(k,d)\log\phi_k(d) \\
&= \psi^T\mathbf{y}^i \qquad (5)
\end{aligned}$$

where $\psi = [\log\pi; \mathrm{vec}(\log\mathbf{A}); \mathrm{vec}(\log\phi)]$, $\mathbf{y}^i = [\mathbf{n}^i; \mathrm{vec}(\mathbf{M}^i); \mathrm{vec}(\mathbf{O}^i)]$ and vec converts matrix to vector. With these, Eqn. 3 can be finally written as

$$\begin{aligned}
\psi \quad &: \quad \max_{\psi\in\Omega} \psi^T \sum_{i:\mathbf{X}^i\in\mathbb{X}} \mathbf{y}^i \qquad (6) \\
\text{s.t.} \quad & \quad \psi^T\mathbf{y}^i \geq \psi^T\mathbf{y}^j + \rho\,\forall(i,j)\in\mathbb{E}
\end{aligned}$$

where $\rho \geq 0$ defines the required margin between the logarithms of likelihood for a pair of data and Ω defines the set of valid parameters for the hidden Markov model, i.e.:

$$\psi(i) \leq 0 \quad ; \quad \sum_{i:\psi(i)\in\log(\pi)} e^{\psi(i)} = 1 \qquad (7)$$
$$\sum_{i:\psi(i)\in\log(\mathbf{A}_j)} e^{\psi(i)} = 1 \quad ; \quad \sum_{i:\psi(i)\in\log(\phi_j)} e^{\psi(i)} = 1$$

where $i : \psi(i) \in \log(A_j)$ is the set of the indexes which corresponds to the j_{th} row of matrix A.

For the model in Eqn. 3, we assumed that every pairwise ranking constraint provided in the data is correct (or valid). However, in real data, there may be outliers in such training pairs. To handle this, we further introduce some slack variables ϵ, and relax the pair-wise ranking constraint as $\psi^t y^i + \epsilon_{ij} \geq \psi^t y^j + \rho, \forall(i,j)\in\mathbb{E}$. Accordingly Eqn. 6 can be written as following:

$$\begin{aligned}
\psi \quad &: \quad \max_{\psi\in\Omega} \psi^T \sum_{\mathbf{X}^i\in\mathbb{X}} \mathbf{y}^i - \gamma \sum_{(i,j)\in\mathbb{E}} \epsilon_{ij} \qquad (8) \\
\text{s.t.} \quad & \quad \psi^T\mathbf{y}^i + \epsilon_{ij} \geq \psi^t\mathbf{y}^j + \rho\,\forall(i,j)\in\mathbb{E} \\
& \quad \epsilon_{ij} \geq 0\,\forall(i,j)\in\mathbb{E}
\end{aligned}$$

where γ is the weight for the penalty term $\sum_{(i,j)\in\mathbb{E}}\epsilon_{ij}$. For initialization, we can set $\epsilon_{ij} = 0$. Now, we are ready to describe the proposed learning algorithm:

```
The Baseline Algorithm
Input: $\mathbb{X}$, $\mathbb{E}$, $\rho$, $\gamma$
Output: $\theta$
Initialization: Initialize $\theta$ (and $\psi$) via ordinary HMM learning algorithm;
while NOT terminated
    Compute the optimal path $z$ for each sequence;
    Update the model $\psi$ according to Eqn. 8;
end
Convert $\psi$ to $\theta$;
```

After the model is learned, it can be used to a testing pair: For each sequence we evaluate the data likelihood via the Viterbi algorithm and use the logarithm of the data likelihood as the score of the data. By definition, the obtained scores can be used to compare the pair.

4.2. The Improved Model

In the model described in Eqn. 8, we compare the logarithm of the data likelihood, which is, according to Eqn. 4, roughly proportional to the length of the data. Thus a shorter sequence is likely to have a larger score. This means that the learned model would be biased towards the shorter sequences. If the observation describes a long, periodic event, e.g., repeating an action multiple times within a sequence, we may consider normalizing the logarithm of the data likelihood by the number of frames of the observation. However, this cannot be applied directly for non-periodic observations.

To overcome the above practical problem, we consider an improved version. Recall that in HMM, we classify a sequence based on the model with which the sequence gets the maximal likelihood, i.e., it is the ratio of data likelihood with different models that decides the label of the data. For example, if $\log\frac{p(\mathbf{X},\hat{\mathbf{z}}|\theta_1)}{p(\mathbf{X},\tilde{\mathbf{z}}|\theta_2)} > 0$, then we assign $\mathbf{X}$ to Model θ_1. Thus we propose to use the ratio of the data likelihoods of two HMMs as the score function, i.e., $F(\mathbf{X},\theta) = \log\frac{p(\mathbf{X},\hat{\mathbf{z}}|\theta_1)}{p(\mathbf{X},\tilde{\mathbf{z}}|\theta_2)}$, where we "partition" the original model into two models (or, effectively, we train a pair of HMMs simultaneously). This results in the following improved model:

$$\begin{aligned}
\theta_1,\theta_2 \quad &: \quad \max_{\theta_1,\theta_2} \sum_{i\in\Xi_1} \log p(\mathbf{X}^i, \hat{\mathbf{z}}^i|\theta_1) + \sum_{j\in\Xi_2} \log p(\mathbf{X}^j, \tilde{\mathbf{z}}^j|\theta_2) \\
& \quad -\gamma \sum_{(i,j)\in\mathbb{E}} \epsilon_{ij} \\
\text{s.t.} \quad &: \quad \log\frac{p(\mathbf{X}^i, \hat{\mathbf{z}}^i|\theta_1)}{p(\mathbf{X}^j, \tilde{\mathbf{z}}^j|\theta_2)} + \epsilon_{ij} \geq \log\frac{p(\mathbf{X}^j, \hat{\mathbf{z}}^j|\theta_1)}{p(\mathbf{X}^j, \tilde{\mathbf{z}}^j|\theta_2)} + \rho \\
& \quad \epsilon_{ij} \geq 0\,\forall\,(i,j)\in\mathbb{E} \qquad (9)
\end{aligned}$$

where Ξ_1 is the set of data associated with Model θ_1 (Ξ_2 for Model θ_2), $\hat{\mathbf{z}}^i$ is the optimal path for sequence

x^i with Model θ_1 and $\tilde{\mathbf{z}}^i$ for optimal path with Model θ_2. The model in Eqn. 9 can also be written as the standard form in Eqn. 8 with similar technique as described in Sec. 4.1, and thus the details are omitted. The corresponding improved algorithm is given below:

```
Improved Algorithm
Input: $\mathbb{X}$, $\mathbb{E}$, $\rho$, $\gamma$, $\Xi_1$, $\Xi_2$
Output: $\theta_1$ and $\theta_2$
Initialization: Initialize $\theta_1$ and $\theta_2$ via ordinary HMM learning algorithm with data from $\Xi_1$ and $\Xi_2$ accordingly;
while NOT terminated
  Compute the optimal path $\hat{\mathbf{z}}$ and $\tilde{\mathbf{z}}$ for each sequence with $\theta_1$ and $\theta_2$;
  Update the model $\theta_1$ and $\theta_2$ according to Eqn. 9;
end
```

After we learn the model with the improved algorithm, we can apply it to a given pair by first computing their likelihoods with respect to the ”sub-models” given by θ_1 and θ_2 (with the Viterbi algorithm), and then we use the logarithm of the ratio of the data likelihoods as the score to rank/compare the pair.

The learned models θ_1 and θ_2 serve as a unified model to rank the data. We may view them as the centers of two clusters, where the distances of the data to those two centers can be related to the ranking score.

It needs to be emphasized that the improved model is not equivalent to a supervised HMM with two classes. In a 2-class HMM setting, two models will be independently trained with their respective training sets. Here, the proposed model trains two ”sub-models” jointly with only relative ranking constraints. Specifically, if there is no further information for Ξ, we could assume that $\Xi_1 = \{i|(i,j) \in \mathbb{E}, \forall j\}$ and $\Xi_2 = \{j|(i,j) \in \mathbb{E}, \forall i\}$, and thus there could be overlaps between Ξ_1 and Ξ_2 (which will become clear in the experiment with synthetic data in Sec. 5). This situation not even allowed by a supervised HMM setting. We don't require any extra properties for Ξ_1 and Ξ_2, e.g., balances.

4.3. Discussion

Eqn. 8 (similarly for Eqn. 9) can be written in a more standard form:

$$\begin{aligned} \mathbf{x} &: \min_{\mathbf{x}} \mathbf{f}^T \mathbf{x} \qquad (10) \\ \text{s.t.} &: \mathbf{A}\mathbf{x} \leq \mathbf{b}; \mathbf{x} <= 0; \mathbf{C}e^{\mathbf{x}} = 1 \end{aligned}$$

Eqn. 10 is a nonlinear programming problem (due to the nonlinear equality constraint). To solve this problem, we use primal-dual interior point method. The dimension of this problem is $K(1+K+D)+|\mathbb{E}|$ (or $2K(1+K+D)+|\mathbb{E}|$) with $2|\mathbb{E}|+K(1+K+D)$ (or $2|\mathbb{E}|+2K(1+K+D)$) linear inequality constraints and $1+K+D$ (or $2(1+K+D)$) nonlinear equality constraints for the baseline model (or the improved model). However, the Hessian ($\mathbf{H}$) of the problem is a diagonal matrix and can be computed as $\mathbf{H} = \Lambda(e^{\mathbf{x}} \cdot (\mathbf{C}\lambda))$, where $\Lambda(\cdot)$ converts a vector to a diagonal matrix, $\cdot$ is element-wise product and λ the Lagrange multipliers for the nonlinear constraints. Thus the problem can still be solved quickly.

The algorithm is terminated when at least one of the following condition satisfied: the maximal number of iterations is achieved; all of the training pair get correctly ranked; the model (i.e., the value of objective function) doesn't change.

The problem in Eqn. 10 (i.e., Eqn. 8 and 9) is not convex, due to the nonlinear equality constraint. Thus we can only found local optimal solutions. While there is no guarantee on the convergence, empirically it was found that after a certain number of iterations the learned model starts to deliver reasonable results (in terms of the percentage of the training pairs getting correctly-maintained ranking).

5. Experiments

In this section, we evaluate the proposed methods, including the baseline method and the improved method, using both synthetic data (Sec. 5.1) and realistic data collected from the surgical training platform FLS box (Sec. 5.2). The performance of the proposed methods is compared with a supervised 2-class HMM. (Lacking a comparative approach in the literature that is both unsupervised and works with only relative rankings, this is believed to be a reasonable way of a reference point to assess the proposed methods.)

5.1. Evaluation with Synthetic Data

To evaluate the proposed method, we generate synthetic data: we first generate six different HMMs (θ_1 to θ_6, which are referred as data-generating models), from each of which we draw 200 sequences, with the length being uniformly distributed between 80 to 120. Each data-generating model has five states. For the sequences from each data-generating model, we randomly assign 50 of them to the training set and the remaining to the testing set. We assume there exists a score function such that $F(\mathbf{X}^i) > F(\mathbf{X}^j)$ if and only if $\mathbf{X}^i \sim \theta_k$, $\mathbf{X}^j \sim \theta_l$ and $k < l$. That is, the sequences from a data-generating model with a lower index are viewed to have a higher score (or ranking) than those from a data-generating model with a higher index. A set of pairs $\{(i,j)|\mathbf{X}^i \sim \theta_k, \mathbf{X}^j \sim \theta_{k+1}, k = 1, \cdots, 5\}$ are then formed accordingly, some of which are then randomly selected as the training pairs $\mathbb{E}$.

For all three methods, we assume that the maximal number of states is ten. For the HMM algorithm and the improved method, we initialize the two sets as $\Xi_1 = \{i|(i,j) \in \mathbb{E}, \forall j\}$ and $\Xi_2 = \{j|(i,j) \in \mathbb{E}, \forall i\}$. Note, the data generated from data-generating Models $\theta_2 \sim \theta_5$ could be included in both Ξ_1 and Ξ_2. Thus existing discriminative learning methods for HMM could not be applied here.

The learned models are then used to evaluate the testing set, i.e., how many testing pairs that they rank the same as ground truth. The result of the methods with different number of training pairs is summarized in Fig. 1, where due to the computational time it takes, we don't have the results for the baseline method when there are more than 3750 training pairs.. From Fig. 1, we can find that the improved method achieves the best results on both the training set and the testing set; and the HMM method gives the worse result. In addition, the performance of both of the proposed methods stabilized after certain number of training pairs. However the performance of the HMM method drops dramatically when the number of training pairs reaches about 6250. It can be explained by that the two HMMs share a lot of common data (for those generated by $\theta_2 \sim \theta_5$) and the models are trained independently without consideration of their discrimination ability. Normalizing the logarithm of data likelihood does not improve the performance of baseline method, which could be explained by that, all the sequences have roughly the same length, i.e., $80 \sim 120$. Fig. 2 shows the logarithm of the data likelihood ratio with the models learned by the improved method, when about 1250 training pairs are provided. This clearly demonstrates that, although we formed the training pairs only with data from data-generating models of adjacent indices (i.e., i and $i+1$), the learned model is able to recover the strict ranking of the original data.

For empirically understanding the convergence behavior of the improved method, we plot in Fig. 3 the objective value in the model as a function of the number of iterations. We can find that the improved method converges fairly quickly (within about 14 iterations) and the value of the objective function monotonically increases. The time complexity for the improved algorithm is roughly $O(|\mathbb{E}|^2)$ (e.g., about 60 seconds for about 600 constraints in Matlab on a quad-core PC platform).

It is obvious from this experiment that the sequences are different from (or similar to) each other only because they are from different (or the same) data-generating models, whereas their relative ranking can be arbitrarily defined. In the end, the proposed methods will learn a temporal model to reflect the defined rankings. This suggests that, as long as we can assume there are some data-generating models for the given sequential data, we can use the proposed methods to learn a relative HMM. This is the basis for applying the approach to the surgical training data in the following sub-section, where it is reasonable to assume that movement patterns of subjects with different skill levels may be modeled by different underlying HMMs while the ranking can be based on the time of training, which reflects the skill level of the subject at the time.

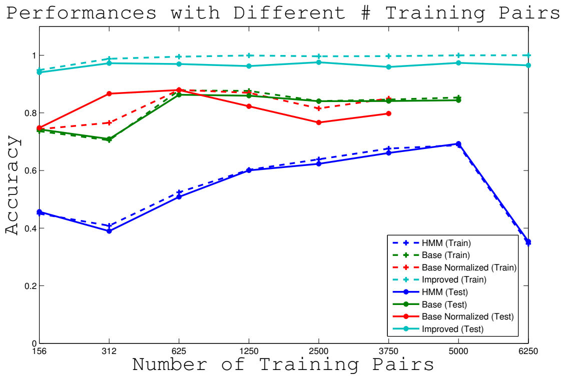

Figure 1. The results of four methods on training set (dashed curve) and testing set (solid curve) with different numbers of training pairs.

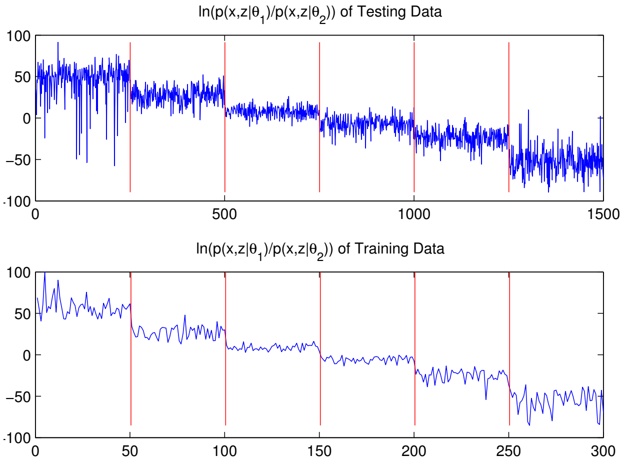

Figure 2. The logarithm of the data likelihood ratio with the models learned by the improved method. Top: the result for the testing set. Bottom: the result for the training set. The data are grouped (as the section partitioned by the red lines) according to the data generation model from which they are synthesized.

5.2. Skill Evaluation Using Surgical Training Video

We now evaluate the proposed method using real videos captured from the FLS trainer box, which has been widely used in surgical training. The data set contains 546 videos captured from 18 subjects performing the "peg transfer" operation, which is one of the standard training tasks a resident surgeon needs to perform and pass. The number of frames in each video varies from 1000 to 6000 (depending on the trainees' speed in completing a training session). In the training, the subject needs to lift six objects (one by one) with a grasper by the non-dominant hand, transfer the object midair to the dominant hand, and then place the object on a peg on the other side of the board. Once all six objects are transferred, the process is reversed, and the objects are to be transferred back to the original side of the board. The videos capture the entire process inside the trainer box, showing

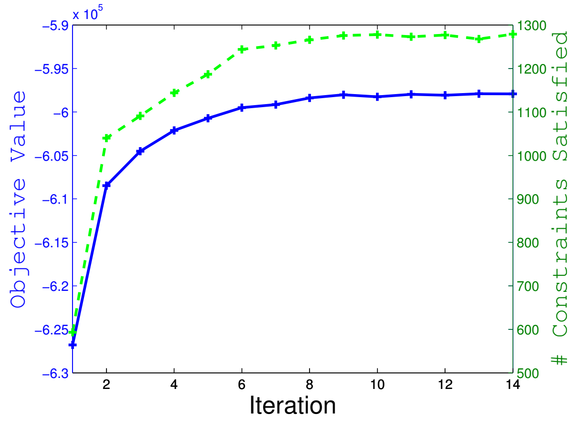

Figure 3. The convergence behavior of the improved method, where around 1250 training pairs were used. The blue curve/axis shows the value of the objective function, and the green curve/axis shows the number of constraints satisfied.

how the tools and objects are moved by the subject. In the existing practice, senior surgeons rate the performance of the trainees based on such videos. Our goal is to perform the rating automatically with the proposed model. The data set covers a training period of four weeks, with every trainee performing three sessions each week.

The time of recording is used to rank the recorded videos **within each subjects' corpus** (i.e., a later video is associated with a better skill) based on the reasonable assumption that the trainees improve their skills over time (which is the whole point of having the resident surgeons going through the training before taking the exam). Other than this relative ranking, there are no other labels assumed for the video, e.g., there is no rank information between videos of different subjects (which would be hard to obtain anyway, since there is no clearly-defined skill levels for a group of trainees with diverse background). Based on this, we randomly pick 300 pairs as the training pairs, similarly as in the experiment using synthetic data.

We use the "bag of words" approach for feature extraction from the videos as follows. The spatiotemporal interest point detector [10] is applied to obtain the histogram-of-gradient (HoG) features. K-means ($k = 100$) is then used to build a codebook for the descriptors of the interest points. Finally, the codebook is used to obtain a histogram of interest points for each frame, and thus each video is represented as a sequence of histograms. This representation, compared with the existing way of using bag of words in action recognition, i.e., transforming each video into a single histogram, can better capture the temporal information of the data.

After learning the models from the training data, we compute the score of the test data as the logarithm of data likelihood (for the baseline method) or the logarithm of the data likelihood ratio (for the improved method and the HMM). We compare these scores for each pair of the testing data (within each subject) and compute the percentage of correctly labeled pairs (recall that, we use their time of recording as ground truth). The result is summarized in Tab. 1, where the improved method obtained a significantly better result than the other approaches. Surprisingly, the baseline method even performed slightly worse than the HMM method. This is largely due to the wide range of variations of the length of the input sequences. Fig. 4 shows the computed scores with the learned models, where for better illustration purpose we group them by their subject ID and within each subjects' corpus we sort the videos by their recording time. From the figure, we can find that the improved method (bottom) reveals a more clear trend for the data than both the HMM method (top) and the baseline method (middle), i.e., the scores of the data increase over times (X-axis) for each subject (segments within the red lines). It is worth emphasizing that only one joint model is learned from ranked pairs of subjects with potentially varying skill levels. Still the learned model is able to recover the improving trend, independent of the underlying skill levels.

Method	HMM	Baseline	Improved
# Pairs	6363	6215	**6993**
Accuracy	79.39%	77.54%	**87.25%**

Table 1. The result for experiment on evaluating surgical skills. There are 8015 pairs in total (only 300 for training), excluding the comparisons among data of different subjects.

It is also interesting to look at what the jointly-learned models look like in the proposed approach. Fig. 5 depicts the two models learned by the improved method in this real-data based experiment. From the figure, we can see that the two models have different transition patterns. For example, the transition from State 8 to States 2 and 5 are only observed in Model 1. This may be linked to different motion patterns for data of different surgical skills.

6. Discussions and Conclusions

In this paper, we presented a new formulation for the problem of learning temporal models using only relative information. Algorithms were developed under the formulation, and experiments using both synthetic and real data were performed to verify the performance of the proposed method. In essence, the proposed method attempts to learn an HMM with relative constraints. Such a setting is useful for many practical applications where relative attributes are easier to obtain while explicit labeling is difficult to get. The application of video-based surgical training was the focus of this study, and the evaluation results using realistic data suggests that the proposed method provides a promising solution to the problem of motion skill evaluation from videos. For future work, we plan to extend the proposed method to cover different observation models so that other types of applications may be handled.

Acknowledgement: The work was supported in part by a

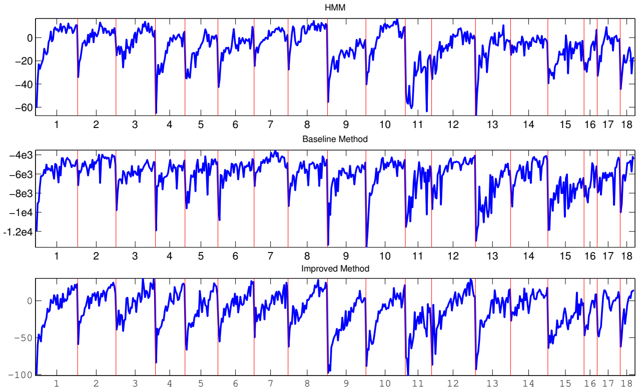

Figure 4. Top: the logarithm of the data likelihood ratio from two models learned by HMM. Middle: the logarithm of data likelihood with the model learned by the baseline method. Bottom: the logarithm of the data likelihood ratio with the models learned by the improved method. The red vertical lines separate the data of different subjects, where X-axis is the corresponding subject ID. Within each subjects' corpus, the videos are sorted according to their time of recording.

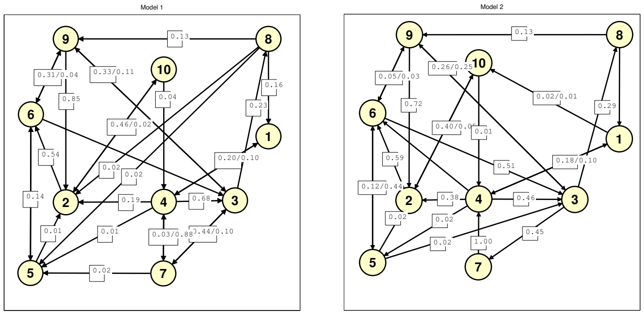

Figure 5. The two component models (Model 1 for Ξ_1 and Model 2 for Ξ_2) learned by the improved method, where we only draw the edges with a transition probability larger than 0.01 and ignore self transitions. The number attached to each edge indicates the transition probability.

grant (#0904778) from the National Science Foundation (NSF). Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

[1] Y. Altun, I. Tsochantaridis, T. Hofmann, et al. Hidden markov support vector machines. In *ICML*, volume 20, page 3, 2003.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):pp. 164–171, 1970.

[3] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 1–8. Association for Computational Linguistics, 2002.

[4] F. Duan, Y. Zhang, N. Pongthanya, K. Watanabe, H. Yokoi, and T. Arai. Analyzing human skill through control trajectories and motion capture data. In *Automation Science and Engineering, 2008. IEEE International Conference on*, pages 454–459, aug. 2008.

[5] B. Juang and L. Rabiner. The segmental¡ e1¿ k¡/e1¿-means algorithm for estimating parameters of hidden markov models. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(9):1639–1641, 1990.

[6] I. Kadar and O. Ben-Shahar. Small sample scene categorization from perceptual relations. In *CVPR 2012*, pages 2711 –2718, june 2012.

[7] K. Kahol, N. C. Krishnan, V. N. Balasubramanian, S. Panchanathan, M. Smith, and J. Ferrara. Measuring movement expertise in surgical tasks. In *ACM Multimedia*, pages 719–722, New York, NY, USA, 2006. ACM.

[8] A. Kovashka, D. Parikh, and K. Grauman. Whittlesearch: Image search with relative attribute feedback. In *CVPR 2012*, pages 2973 –2980, june 2012.

[9] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and simile classifiers for face verification. In *ICCV 2009*, pages 365 –372, 29 2009-oct. 2 2009.

[10] I. Laptev. On space-time interest points. *IJCV*, 64(2):107–123, 2005.

[11] N. Merhav and Y. Ephraim. Maximum likelihood hidden markov modeling using a dominant sequence of states. *Signal Processing, IEEE Transactions on*, 39(9):2111 –2115, sep 1991.

[12] D. Parikh and K. Grauman. Relative attributes. In *ICCV 2011*, pages 503 –510, nov. 2011.

[13] D. Parikh, A. Kovashka, A. Parkash, and K. Grauman. Relative attributes for enhanced human-machine communication. In *AAAI*, 2012.

[14] J. Rosen, M. Solazzo, B. Hannaford, and M. Sinanan. Task decomposition of laparoscopic surgery for objective evaluation of surgical residents' learning curve using hidden markov model. *Computer Aided Surgery*, 7:49–61, 2002.

[15] S. Satoshi and H. Fumio. Skill evaluation from observation of discrete hand movements during console operation. *Journal of Robotics*, 2010, 2010.

[16] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. *NIPS*, page 41, 2004.

[17] A. Sloin and D. Burshtein. Support vector machine training for improved hidden markov modeling. *Signal Processing, IEEE Transactions on*, 56(1):172 –188, jan. 2008.

[18] S. Suzuki, N. Tomomatsu, F. Harashima, and K. Furuta. Skill evaluation based on state-transition model for human adaptive mechatronics (ham). In *Industrial Electronics Society, 2004. IECON 2004. 30th Annual Conference of IEEE*, volume 1, pages 641–646. IEEE, 2004.

[19] G. Wang, D. Forsyth, and D. Hoiem. Comparative object similarity for improved recognition with few or no examples. In *CVPR 2010*, pages 3525–3532. IEEE, 2010.

[20] K. Watanabe and M. Hokari. Kinematical analysis and measurement of sports form. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 36(3):549–557, 2006.