A New Variational Framework for Rigid-Body Alignment

Tsuyoshi Kato[1], Koji Tsuda[2,1], and Kentaro Tomii[1] and Kiyoshi Asai[3,1]

[1] AIST Computational Biology Research Center,
2-43, Aomi, Koto-ku, Tokyo, 135-0064 Japan
[2] Max Planck Institute of Biological Cybernetics
Spemannstr. 36, 72076 Tübingen, Germany
[3] Graduate School of Frontier Sciences, The University of Tokyo
5-1-5, Kashiwanoha, Kashiwa, 277-8562, Japan
{kato-tsuyoshi,koji.tsuda,k-tomii,asai-cbrc}@aist.go.jp

Abstract. We present a novel algorithm for estimating the rigid-body transformation of a sequence of coordinates, aiming at the application to protein structures. Basically the sequence is modeled as a hidden Markov model where each state outputs an ellipsoidal Gaussian. Since maximum likelihood estimation requires to solve a complicated optimization problem, we introduce a variational estimation technique, which performs singular value decomposition in each step. Our probabilistic algorithm allows to superimpose a number of sequences which are rotated and translated in arbitrary ways.

1 Introduction

In the most simple form, the protein structure is represented as a sequence of 3-dimensional vectors, each of which indicates the position of C_α atom of an amino acid [6]. A large amount of structure data are readily available e.g. in the Protein Data Bank. However, it is not easy to compare protein structures because they are translated and rotated in arbitrary ways. A set of proteins have to be superposed correctly to measure meaningful similarities among them. Here one has to estimate the *rigid-body transformation* (i.e. rotation and translation) of each protein correctly [1]. Superposition of protein structures has been a central issue in computational biology, and many methods have been proposed (e.g. [3, 11, 1]). However, most works employ ad hoc or physically-motivated approaches, and probabilistic models (e.g. HMMs) are rather out of focus. One of the reasons would be that the probabilistic models for estimating 3-dimensional rigid-body transformation get so complicated that direct maximization of likelihood e.g. by gradient descent is almost hopeless (we will show details later). However, there

[1] Notice that estimating rigid-body transformation is more difficult than estimating affine transformation [9], because we have to constrain the rotation matrix to be orthogonal. Affine transformation allows rescaling, which is obviously inappropriate for protein structures.

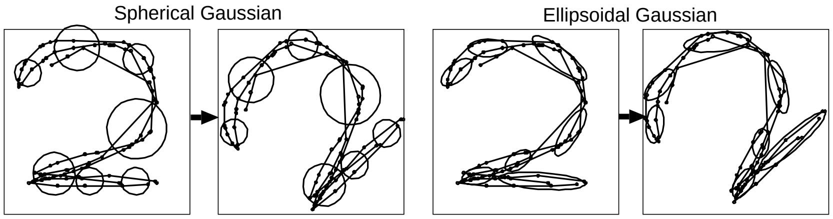

Fig. 1. Comparison of the shape models based on spherical (left) and ellipsoidal (right) Gaussians. When rotated, the covariance matrix of each Gaussian stays the same in the spherical case, but it changes nonlinearly in the ellipsoidal case. This fact makes it difficult to estimate the rotation and transformation by means of the ellipsoidal Gaussians. However, the ellipsoidal Gaussians are much better to describe string-like shapes (e.g. proteins). We will adopt a hierachical model, which combines the best of both worlds.

are crucial advantages of employing probabilistic models. For example, one can attach confidence levels on the estimated rotation and translation. Also one can embed the probabilistic model as one node of a Bayesian network for higher-level inference.

In this paper, we model protein structures by an HMM where each state outputs a 3-dimensional vector subject to an ellipsoidal Gaussian. [2] The mean vectors and covariance matrices of Gaussians have parameters corresponding to rotation and translation. The rigid-body transformation is basically estimated by maximum likelihood with respect to these parameters. The main difficulty is that the covariance matrices are highly nonlinear functions of the rotation parameter (Fig. 1, right). In order to alleviate the computational problem, we replace the ellipsoidal Gaussian with the hierarchical model, that is, a spherical Gaussian distribution whose mean is subject to an ellipsoidal Gaussian. Here we have a new set of hidden variables, that is, the means of spherical Gaussians. Fixing these hidden variables, the tranformation parameters are easily obtained [2], because the covariance matrix of a spherical Gaussian does not change by rotation and translation (Fig. 1, left). Now the estimation of transformation parameters amounts to maximize the expected log-likelihood with respect to the hidden variables, which is tractably solved by a variational technique [5].

The organization of this paper is as follows. In section 2 we describe an HMM shape model for representing a sequence of vectors. In section 3, we provide an efficient algorithm for estimating rigid-body transformation. Section 4 explains how to learn the HMM from a set of sequences. We will show several experiments in section 5 before concluding in section 6.

[2] Typically superposition is helped by side information such as amino acid sequences (i.e. `Leu-Thr-Ser-Ile-`$\cdots$). However, this paper considers more challenging setting that only a sequence of 3-dimensional vectors is available.

2 Shape Models

First of all, let us formulate the shape model without rotation/translation parameters. Let us define the sequence of d-dimensional vector sequence as $\mathbf{X} = [\mathbf{x}(1), \cdots, \mathbf{x}(L)] \in \mathcal{R}^{d \times L}$, where L denotes the length of sequence. In the case of protein structure, L is the number of residues. We use the continuous density hidden Markov model(HMM) as the shape model. The HMM has the following latent variables: $\mathbf{S} = [s(1), \cdots, s(L)]$ where $s(r) \in \{1, \cdots, M\}$ indicates the state at residue r. We use a d-dimensional Gaussian as the output distribution: $p(\mathbf{x}(r)|s(r) = j) \sim \mathcal{N}(\mathbf{m}_j^0, \mathbf{C}_j)$ where $\mathcal{N}()$ denotes a Gaussian density function and $\mathbf{m}_j^0$, $\mathbf{C}_j$ are the mean vector and the covariance matrix of state j, respectively. The density function of an observed sequence $\mathbf{X}$ is given by $f(\mathbf{X}|\Theta) \equiv \sum_{\mathbf{S}} p(\mathbf{S}|\Theta) \prod_{r=1}^{L} p(\mathbf{x}(r)|s(r), \Theta)$ where $\sum_{\mathbf{S}}$ denotes summing over all possible $\mathbf{S}$. For simplicity, let us describe all the parameters by Θ which consists of the parameters of Gaussian, $\mathbf{m}_j^0, \mathbf{C}_j$, as well as the state transition probabilities, a_{ij}, and the initial state probabilities π_i.

The density function of the rotated and translated model is described as

$$p(\mathbf{X}|\Theta, \mathbf{U}, \mathbf{c}) = f(\mathbf{U}\mathbf{X} + \mathbf{c}\mathbf{1}_{1\times L}|\Theta) \tag{1}$$

where $\mathbf{U} \in \mathcal{R}^{d \times d}$ is a rotation matrix, $\mathbf{c} \in \mathcal{R}^{d \times 1}$ is an offset vector and $\mathbf{1}_{1\times L}$ is the $1 \times L$ matrix with all elements equal to one. The rotation matrix $\mathbf{U}$ has to satisfy $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$ for orthonormality and $\det(\mathbf{U}) = 1$ for preserving orientation. Assuming that Θ is known, our task is to estimate $\mathbf{U}$ and $\mathbf{c}$ by maximum likelihood:

$$\{\hat{\mathbf{U}}, \hat{\mathbf{c}}\} = \mathrm{argmax}_{\mathbf{U},\mathbf{c}} \log p(\mathbf{X}|\Theta, \mathbf{U}, \mathbf{c}). \tag{2}$$

Let us analyze the difficulty of solving this problem. Consider an easier problem when $\mathbf{S}$ is known, i.e. maximize

$$\log p(\mathbf{X}|\mathbf{S}, \Theta, \mathbf{U}, \mathbf{c}) = -\frac{1}{2}\sum_{r=1}^{L}(\mathbf{U}\mathbf{x}(r) + \mathbf{c} - \mathbf{m}_{s(r)}^0)^\top \mathbf{C}_{s(r)}^{-1}(\mathbf{U}\mathbf{x}(r) + \mathbf{c} - \mathbf{m}_{s(r)}^0) + \mathrm{const}. \tag{3}$$

subject to $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$ and $\det(\mathbf{U}) = 1$. Basically, this problem has a quadratic objective function and a set of quadratic constraints, thus it is significantly more complicated than the quadratic programming (i.e. quadratic objective function and linear constraints) [7]. Efficient algorithms such as interior point methods are not straightforwardly applicable for this problem, so typically one has to use general purpose nonlinear optimizers (e.g. gradient descent, Newton methods), which are not so efficient and prone to local minima. Here we do not insist on finding a good approximation algorithm of solving Eq. (2), but rather decompose the covariance of the shape model:

$$\mathbf{C}_j = \mathbf{V}_j^0 + \sigma^2 \mathbf{I}. \tag{4}$$

Then, using the property that the convolution of two Gaussians is also a Gaussian, we have the following hierarchical model:

$$p(\mathbf{x}(r)|\mu(r),\Theta) \sim \mathcal{N}(\mu(r),\sigma^2\mathbf{I}), \qquad p(\mu(r)|s(r),\Theta) \sim \mathcal{N}(\mathbf{m}^0_{s(r)},\mathbf{V}^0_{s(r)}) \qquad (5)$$

where $\mu(r)$ is a new hidden variable. Now the density function is rewritten as

$$p(\mathbf{X}|\Theta,\mathbf{U},\mathbf{c}) = \sum_{\mathbf{S}} p(\mathbf{S}|\Theta)\prod_{r=1}^{L}\int p(\mu(r)|s(r),\Theta)p(\mathbf{U}\mathbf{x}(r)+\mathbf{c}|\mu(r),\Theta)d\mu(r), \quad (6)$$

where Θ is redefined by $\Theta \equiv \{\mathbf{m}^0_j,\mathbf{V}^0_j,a_{1j},\cdots,a_{Mj},\pi_j\}_{j=1}^M \cup \sigma^2$. Fixing hidden variables $\mathbf{S}$ and $\mu(r)$, the optimization problem can be solved analytically using the singular value decomposition (SVD) [2]. As we see in the next section, this property allows us to maximize the *negative free energy functional*, which is the lower bound of the log-likelihood.

3 Variational Estimation

We will discuss how to maximize the likelihood in Eq. (6) approximately by the variational EM algorithm [5]. For any distribution $q(\mathbf{S},\{\mu(r)\}_{r=1}^L)$, the following inequality holds:

$$\log p(\mathbf{X}|\Theta,\mathbf{U},\mathbf{c}) \geq \left\langle \log p(\mathbf{X},\mathbf{S},\{\mu(r)\}_{r=1}^L|\Theta,\mathbf{U},\mathbf{c})\right\rangle_{q(\mathbf{S},\{\mu(r)\}_{r=1}^L)} + \mathcal{H}\left(q(\mathbf{S},\{\mu(r)\}_{r=1}^L)\right). \quad (7)$$

where $\mathcal{H}(\cdot)$ denotes the entropy function which is defined by: $\mathcal{H}(p(x)) = -\int_x dxp(x)\log p(x)$. We maximize the lowerbound by setting up a parametric model on q and optimize q and $\mathbf{U}$, $\mathbf{c}$, alternately. Typically, q is assumed to be factorized as

$$q(\mathbf{S},\{\mu(r)\}_{r=1}^L) = q(\mathbf{S})\prod_{r=1}^{L} q(\mu(r)). \quad (8)$$

Denote by $\mathcal{F}(\mathbf{U},\mathbf{c},q|\Theta,\mathbf{X})$ the right hand side of Eq. (7) where the parametric model is plugged in. In terms of statistical physics, $\mathcal{F}$ is often called the negative free energy functional. Then the variational EM algorithm [5] is represented as follows:

$$q(\mathbf{S}) := \mathrm{argmax}_{q(\mathbf{S})}\,\mathcal{F}(\mathbf{U},\mathbf{c},q|\Theta,\mathbf{X}), \quad (9)$$

$$q(\mu(r)) := \mathrm{argmax}_{q(\mu(r))}\,\mathcal{F}(\mathbf{U},\mathbf{c},q|\Theta,\mathbf{X}), \quad \forall r \quad (10)$$

$$\{\mathbf{U},\mathbf{c}\} := \mathrm{argmax}_{\mathbf{U},\mathbf{c}}\,\mathcal{F}(\mathbf{U},\mathbf{c},q|\Theta,\mathbf{X}) \quad (11)$$

The first two belong to the E-step while the last one belongs to the M-step. Let us solve the first one in Eq. (9). Using the variational method and keeping the other parameters fixed, the current optimal posteriors $q(\mathbf{S})$ are given by: $q(\mathbf{S}) \propto \left(\prod_{r=1}^T b_{s(r)}(r)\right)\pi_{s(1)}\left(\prod_{r=1}^{L-1} a_{s(r)s(r+1)}\right)$, where we define $b_{s(r)}(r) \propto$

$\mathcal{N}(\mathbf{m}_j(r)|\mathbf{m}_j^0, \mathbf{V}^0\mathrm{J}) \exp\left(-0.5 \,\mathrm{tr}\left((\mathbf{V}_j^0)^{-1}\mathbf{V}_j(r)\right)\right)$. In the other two steps, the function $q(\mathbf{S})$ is not fully needed, but only the following statistics are referred: $\gamma_i(r) \equiv q(s(r)=i) = \sum_{\mathbf{S}} \delta_{i,s(r)} q(\mathbf{S})$ where $\delta_{\cdot,\cdot}$ denotes Kronecker's delta. The statistics $\gamma_i(r)$ can be computed efficiently by applying the forward-backward algorithm [8] as follows. Computing the variables, $\alpha_i(r)$ and $\beta_i(r)$, as

$$\alpha_i(r) = \begin{cases} \pi_i b_i(1) & \text{if } r = 1, \\ b_i(r)\sum_j \alpha_j(r-1)a_{ji} & \text{if } r > 1, \end{cases} \quad \beta_i(r) = \begin{cases} 1 & \text{if } r = L, \\ \sum_j \beta_j(r+1)a_{ij}b_j(r+1) & \text{if } r < L, \end{cases}$$

we have $\gamma_i(r) \propto \alpha_i(r)\beta_i(r)$. We can also obtain a by-product of this procedure: $\xi_{ij}(r) = \sum_{\mathbf{S}} \delta_{i,s(r)}\delta_{j,s(r+1)}q(\mathbf{S}) \propto \alpha_i(r)a_{ij}b_j(r+1)\beta_j(r+1)$. The statistics $\xi_{ij}(t)$ are utilized in the next section.

Also, the second one in Eq. (10) is solved analytically as

$$q(\mu(r)) \sim \mathcal{N}(\mathbf{m}(r), \mathbf{V}(r)) \tag{12}$$

where

$$\mathbf{V}(r) = \left(\sigma^{-2}\mathbf{I} + \sum_j \gamma_j(r)(\mathbf{V}_j^0)^{-1}\right)^{-1}, \quad \mathbf{m}(r) = \mathbf{V}(r)\left(\sigma^{-2}\mathbf{x}(r) + \sum_j (\mathbf{V}_j^0)^{-1}\mathbf{m}_j\right). \tag{13}$$

Finally we will show how to solve the M-step in Eq. (11). Removing the terms which do not depend on $\mathbf{U}$ and $\mathbf{c}$ from $\mathcal{F}$, we have the following:

$$\mathcal{F}_0(\mathbf{U}, \mathbf{c}|\Theta, \mathbf{X}) = -\frac{1}{2\sigma^2}\sum_{r=1}^{L} \|\mathbf{m}(r) - (\mathbf{U}\mathbf{x}(r) + \mathbf{c})\|^2 . \tag{14}$$

Thus maximization of $\mathcal{F}_0$ is a least squares problem, which is known to be solved by SVD [2]. Let us define a matrix $\mathbf{\Sigma} = \frac{1}{L}\sum_{r=1}^{L}(\mathbf{m}(r) - \mu_b)(\mathbf{x}(r) - \mu_a)^\top$. where $\mu_a = \frac{1}{L}\sum_{r=1}^{L}\mathbf{x}(r), \mu_b = \frac{1}{L}\sum_{r=1}^{L}\mathbf{m}(r)$. Then decompose $\mathbf{\Sigma} = \mathbf{V}\mathbf{D}\mathbf{W}^\top$ by SVD, where $\mathbf{V}$ and $\mathbf{W}$ are the matrices of left and right singular vectors and $\mathbf{D}$ is the diagonal matrix of singular values. The optimal values of $\mathbf{U}$ and $\mathbf{c}$ are obtained as

$$\mathbf{U} := \mathbf{V}\mathbf{P}\mathbf{W}^\top, \quad \mathbf{c} := \mu_b - \mathbf{U}\mu_a, \tag{15}$$

where

$$\mathbf{P} = \begin{cases} \mathbf{I} & \text{if } \det(\mathbf{V})\det(\mathbf{W}) = 1, \\ \mathrm{diag}(1, \cdots, 1, -1) & \text{if } \det(\mathbf{V})\det(\mathbf{W}) = -1. \end{cases}$$

As seen in Eq. (14), each M-step finds $\mathbf{U}$ and $\mathbf{c}$ which yields the least square error between the transformations of $\mathbf{X}$ and $\mathbf{m}(r)$. So the location of $\mathbf{m}(r)$ is extremely important in this procedure. The latent variable $\mathbf{m}(r)$ can be regarded as the intermediates between the transformation $\mathbf{x}'(r)(\equiv \mathbf{U}\mathbf{x}(r) + \mathbf{c})$ and the corresponding inner Gaussian $\mathcal{N}(\mathbf{m}_{s(r)}^0, \mathbf{V}_{s(r)}^0)$. The crucial variable determining $\mathbf{m}(r)$ is σ^2. From the nature that Gaussians merely generate points outside of the circle with the variance, $\mathbf{m}(r)$ is likely to be in the circle with radius σ and centre $\mathbf{x}'(r)$ so as to explain $\mathbf{x}'(r)$ produced by $\mathcal{N}(\mu(r), \sigma^2\mathbf{I})$. Therefore, the larger σ^2 is, the closer $\mathbf{m}(r)$ is to the centre of the Gaussian and the more quickly the optimal solution is found. From these observations, we employ an annealing approach: we start with the large σ^2 and reduce the values step by step. In all simulations provided later, we used the following value of σ^2 in the t-th iteration: $\sigma^2(t) := (49\exp(-t/20) + 1)\sigma_0^2$ where σ_0^2 is the minimum of all the eigen-values in M covariance matrices. $\mathbf{V}_j^0$ are fixed at $\mathbf{V}_j^0 = \mathbf{C}_j - \sigma_0^2\mathbf{I}$. This annealing is scheduled so that Eq. (4) holds in the ∞-th iteration.

4 Learning Shape Models

Here we describe a method for learning the shape model parameters Θ from a number of sequences. We again use the variational EM algorithm in order to estimate the shape model parameters, Θ, and the rotation and offset parameters, $\mathbf{U}_n$, $\mathbf{c}_n$, of each sequence simultaneously. Given a training set of sequences, $\{\mathbf{X}_n\}_{n=1}^N$, the objective function for learning is the following log-likelihood function:

$$\mathcal{L}(\Theta, \{\mathbf{U}_n, \mathbf{c}_n\}_{n=1}^N | \{\mathbf{X}_n\}_{n=1}^N) \equiv \sum_{n=1}^N \log p(\mathbf{X}_n | \Theta, \mathbf{U}_n, \mathbf{c}_n) \tag{16}$$

where $\mathbf{U}_n$, $\mathbf{c}_n$ are the rotation matrix and offset vector for n-th sequence $\mathbf{X}_n$, respectively. The log-likelihood function in Eq. (16) leads the following negative free energy functional:

$$\mathcal{F}_{\text{shape}}(\{\mathbf{U}_n, \mathbf{c}_n\}_{n=1}^N, q | \Theta, \{\mathbf{X}_n\}_{n=1}^N) = \sum_{n=1}^N \mathcal{F}(\mathbf{U}_n, \mathbf{c}_n, q | \Theta, \mathbf{X}_n) \tag{17}$$

by the similar variational approximation to Eq. (8), that is, $q(\mathbf{S}_n, \{\mu_n(r)\}_{r=1}^L) = q(\mathbf{S}_n) \prod_{r=1}^L q(\mu_n(r))$. We then obtain the variational EM algorithm as follows:

$$q(\mathbf{S}_n) := \text{argmax}_{q(\mathbf{S}_n)} \mathcal{F}(\mathbf{U}_n, \mathbf{c}_n, q | \Theta, \mathbf{X}_n), \quad \forall n \tag{18}$$

$$q(\mu_n(r)) := \text{argmax}_{q(\mu_n(r))} \mathcal{F}(\mathbf{U}_n, \mathbf{c}_n, q | \Theta, \mathbf{X}_n), \ \forall n, \forall r \tag{19}$$

$$\{\mathbf{U}_n, \mathbf{c}_n\} := \text{argmax}_{U_n, \mathbf{c}_n} \mathcal{F}(\mathbf{U}_n, \mathbf{c}_n, q | \Theta, \mathbf{X}_n), \quad \forall n \tag{20}$$

$$\Theta := \text{argmax}_{\Theta} \mathcal{F}_{\text{shape}}(\{\mathbf{U}_n, \mathbf{c}_n\}_{n=1}^N, q | \Theta, \{\mathbf{X}_n\}_{n=1}^N) \tag{21}$$

The E-step includes Eq. (18) and Eq. (19), whereas the M-step includes Eqs. (20), (21).

In the first problem in Eq. (18), we need not to solve $q(\mathbf{S}_n)$ completely. Here the statistics $q(s_n(r) = i)$ and $q(s_n(r) = i, s_n(r+1) = j)$ are required for solving Eq. (21), which are commonly described as $\gamma_{i,n}(r)$ and $\xi_{i,j,n}(r)$, respectively, in HMM literature (e.g. [8]). Again they can be computed by the forward-backward algorithm [8]. The second problem in Eq. (19) can be solved by the similar update equations as Eq. (12). In this case, we have to replace $\mu(r)$, $\mathbf{m}(r)$, $\mathbf{V}(r)$, $\mathbf{x}(r)$,

$\gamma_j(r)$ with $\mu_n(r)$, $\mathbf{m}_n(r)$, $\mathbf{V}_n(r)$, $\mathbf{x}_n(r)$, $\gamma_{j,n}(r)$, respectively. The third problem in Eq. (20) can be solved in the same way as Eq. (15). In the fourth problem in Eq. (21), the optimal solution of σ^2 is described as

$$\sigma^2 := \frac{\sum_{n,r} \|\mathbf{U}_n \mathbf{x}_n(r) + \mathbf{c}_n - \mathbf{m}_n(r)\|^2 + \operatorname{tr} \mathbf{V}_n(r)}{d \sum_n L_n}. \tag{22}$$

The other variables are obtained by vanishing the derivative of Eq. (21) subject to the constraints that $\sum_j a_{ij} = 1$ and $\sum_i \pi_i = 1$. The solutions are described as follows:

$$\mathbf{m}_j^0 := \frac{\sum_{n,r} \gamma_{j,n}(r) \mathbf{m}_n(r)}{\sum_{n,r} \gamma_{j,n}(r)}, \qquad \mathbf{V}_j^0 := \frac{\sum_{n,r} \gamma_{j,n}(r) \mathbf{V}_{n,r,j}}{\sum_{n,r} \gamma_{j,n}(r)}, \tag{23}$$

$$a_{ij} := \frac{\sum_{n=1}^{N} \sum_{r=1}^{L_n - 1} \xi_{i.j,n}(r)}{\sum_{n=1}^{N} \sum_{r=1}^{L_n - 1} \gamma_{i,n}(r)}, \qquad \pi_i := \frac{\sum_{n=1}^{N} \sum_{r=1}^{L_n} \gamma_{i,n}(r)}{\sum_{n=1}^{N} L_n}, \tag{24}$$

where $\mathbf{V}_{n,r,j} \equiv \mathbf{V}_n(r) + \left(\mathbf{m}_n(r) - \mathbf{m}_j^0\right) \left(\mathbf{m}_n(r) - \mathbf{m}_j^0\right)^\top$.

5 Experiments

We first tested the algorithm on on-line handwritten digits '2' and '6', where eight 2-dimensional vector sequences are superposed for each digit (Figure 2). In all simulations in this paper, we set the number of states $M = 7$. The variational EM algorithm found the almost optimal rotations and translations and the common shape in the data set, as shown in Figure 2. Next we will show the superposition of protein sequences. We used eight 3-dimensional structures from the globin family: 4HHB:A, 4HHB:B, 5MBN:-, 1ECD:-, 2LHB:-, 2LH3:-, 2HBG:-, which have also been used in [1, 14]. Although we did not use any additional information such as amino acid sequences or the position of other atoms than C_α, almost perfect superposition was achieved (Figure 3).

One crucial advantage of probabilistic modeling is that it can be used as a building block of a larger probabilistic model. For illustrating this advantage, we actually implemented the mixture of HMMs [12] and applied it to semi-supervised learning (i.e. learning from labeled and unlabeled data) [10]. We used 46 protein structures of three classes (16 Globins, 17 Ig-likes, and 13 TIM-barrels). For each class, six structures are randomly chosen as training data, where two of them are given class labels and the other four stays unlabeled. The remaining samples are used as test data. The confusion matrices averaged over 10 trials are shown in Table 1. When unlabeled samples are involved, the classification accuracy improved significantly.

6 Conclusion

In this paper, we presented a novel algorithm which estimates the rigid-body transformations from arbitrarily rotated and translated vector sequences. As

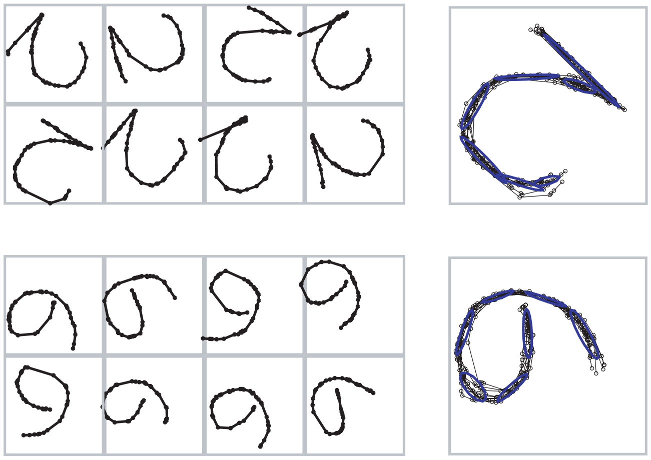

Fig. 2. Superpositions of on-line handwritten digits. Using eight on-line handwritten '2's (top left), we estimate the rotation and the offset parameters as well as common shape parameters by the variational EM algorithm described in section 3, and obtained the resultant superposition (top right). The result of superposition and common shape of eight '6's are also shown in the bottom row. In both cases, almost optimal superpositions are achieved.

	No unlabeled sequences			4 unlabeled sequences		
	Globin	Ig-like	TIM-barrel	Globin	Ig-like	TIM-barrel
Globin	80.0%	1.0%	19.0%	95.0%	1.0%	4.0%
Ig-like	0.0%	64.5%	35.5%	0.0%	82.7%	17.3%
TIM-barrel	0.0%	1.3%	98.8%	0.0%	0.0%	100.0%

Table 1. Confusion matrices from the semi-supervising experiment. The mixture of HMMs is trained by 2 labeled and 4 unlabeled sequences. Significant improvement is observed when unlabeled samples are incorporated.

partly suggested in the previous section, a large number of extensions can be developed from this algorithm due to its probabilistic nature, for example, clustering, detecting outliers, introducing prior knowledge, interpolating missing values, and so on.

One of the most attractive extensions is to combine discriminative methods such as support vector machines. The discriminative methods are often reported to be superior in classification to generative models [4]. Motivated by the fact, several methods which design kernel functions for use in discriminative methods have been proposed (e.g. Fisher kernel [4], marginalized kernel [13] etc.). We might achieve the further improvement by adopting such methods.

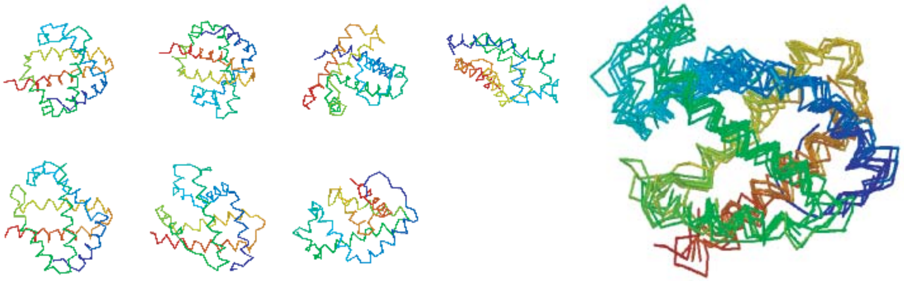

Fig. 3. Superposition of globins. We apply the variational EM algorithm described in section 3 to seven globin structures (left) and achieve almost perfect superposition (right) in spite of using only coordinates of C_α atoms.

References

1. D. Bashford and A. M. Lesk C. Chothia. Determinants of a protein fold: unique features of the globin amino acid sequences. *J. Mol. Biol.*, 196:199–216, 1987.
2. J. H. Challis. A procedure for determining rigid body transformation parameters. *J. Biomechanics*, 28:733–737, 1995.
3. Z. Weng J. D. Szustakowski. Protein structure alignment using a genetic algorithm. *Proteins: structure, function, and genetics*, 38(4):428–440, March 2000.
4. T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, volume 11, pages 487–493. MIT Press, 1999.
5. M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical methods. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 105–161. MIT Press, 1998.
6. D. W. Mount. *Bioinformatics: Sequence and Genome Analysis.* Cold Spring Harbor Laboratory Press, March 2001.
7. J. Nocedal and S.J. Wright. *Numerical Optimization.* Springer Verlag, New York, 1999.
8. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77:257–286, 1989.
9. M. D. Revow, C. K. I. Williams, and G. E. Hinton. Using generative models for handwritten digit recognition. *IEEE T. PAMI*, 18(6):592–606, July 1996.
10. M. Seeger. Learning with labeled and unlabeled data. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh, 2001.
11. I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Engineering*, 11:739–747, 1998.
12. P. Smyth. Clustering sequences with hidden markov models. In *NIPS*, volume 9, pages 648–654. The MIT Press, 1997.
13. K. Tsuda, T. Kin, and K. Asai. Marginalized kernels for biological sequences. *Bioinformatics*, 18(Suppl. 1):S268–S275, 2002.
14. T. D. Wu, T. Hastie, S. C. Schmidler, and D. L. Brutlag. Regression analysis of multiple protein structures. *J. Comput. Biol.*, 5(3):585–595, 1998.