STOCHASTIC MODELING OF VOLUME IMAGES WITH A 3-D HIDDEN MARKOV MODEL

Jia Li, Dhiraj Joshi and James Z. Wang

The Pennsylvania State University, University Park, PA, USA

ABSTRACT

Over the years, researchers in the image analysis community have successfully used various statistical modeling methods to segment, classify, and annotate digital images. In this paper, we propose a 3-D hidden Markov model (HMM) for volume image modeling. A computationally efficient algorithm is developed to estimate the model. The 3-D HMM is applied to volume image segmentation and tested using synthetic images with ground truth. Experiments have demonstrated that 3-D HMM outperforms Gaussian mixture model based clustering by an order of magnitude in accuracy.

1. INTRODUCTION

In recent years, we see an explosion of digital image usages in a number of application domains. Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) scanners in hospitals can produce high-resolution 3-D images of the human brain or the human body so that physicians and radiologists can look inside their patients non-invasively. At airports, 3-D CT scanners are being installed to monitor luggages checked in by travelers to detect explosive materials. Hyper-spectral imaging techniques are employed by military for field surveillance and detection of hidden targets. The amount of information generated by these volume scanners is so enormous that it becomes inevitable for computers to help analyze, segment, and classify these images.

Conventional 2-D image modeling paradigms may not always be effective in volume image analysis as there is a third dimensional linkage that they cannot capture. Researchers have strived to extend existing algorithms for modeling and analyzing large-scale multi-dimensional data. Theories and methodologies related to Markov random fields (MRF) have played important roles in the construction of many statistical image models for image segmentation and texture analysis. 3-D MRFs have been applied to medical image segmentation [6]. Among other modeling paradigms, hidden Markov models (HMM) have particularly demonstrated high effectiveness in modeling digital signals captured from physical world. Several variations of HMM have been explored as modeling techniques in speech recognition, image and video understanding [1]. To effectively account for statistical dependence in 2-D images, researchers extended 1-D HMMs to pseudo 2-D HMMs and pseudo 3-D HMMs for face recognition [2, 3]. The 2-D multiresolution HMM [4] captures the intra-scale and inter-scale statistical dependencies in 2-D images. 2-D multiresolution HMMs have been successfully used for supervised image segmentation [4] and automatic image annotation [5].

Given the great potential demonstrated by the HMM paradigm in various applications, it seems most natural to extend 2-D HMM to 3-D HMM for volume image analysis. In this paper, we construct a 3-D HMM to capture three-dimensional statistical dependence in volume images and hence enhance existing modeling techniques. 3-D HMM is an effort to bridge the gap which naturally appears when 2-D imaging solutions are applied to volume images.

In Section 2, we describe the construction of the 3-D HMM. Section 3 elaborates the proposed estimation algorithm. In Section 4, experiments and their results are provided. We present our conclusions and suggest future research directions in Section 5.

2. BASIC ASSUMPTIONS OF 3-D HMM

In this section, we present the 3-D HMM in its general form. A point (i, j, k) where i, j and k are coordinates along the X, Y and Z axes respectively will be called a *3-D point*. A three dimensional array of finite and equally spaced 3-D points (along X, Y and Z axes) in space will be referred to as *3-D grid*. Additionally, we define a *frame* as the collection of 3-D points on any plane parallel to the X-Y plane. A frame is indexed by its Z coordinate. The size of any 3-D grid that we consider will be $w \times w \times w$ and the set of all points in a grid will be denoted by $\mathcal{C}$. Figure 1(a) shows a 3-

This work was supported by the US National Science Foundation under Grant Nos. IIS-0219272 and ANI-0202007, The Pennsylvania State University, the PNC Foundation, and SUN Microsystems under grants EDUD-7824-010456-US and EDUD-7824-030469-US.

Jia Li is with the Department of Statistics and the Department of Computer Science and Engineering, Email: jiali@stat.psu.edu. Dhiraj Joshi is with the Department of Computer Science and Engineering, Email: djoshi@cse.psu.edu. James Z. Wang is with the School of Information Sciences and Technology and the Department of Computer Science and Engineering, Email: jwang@ist.psu.edu.

D grid. A lexicographic order is defined among 3-D points as follows: $(i^{'}, j^{'}, k^{'}) < (i, j, k)$ if $k^{'} < k$ or $k^{'} = k, j^{'} < j$ or $k^{'} = k, j^{'} = j, i^{'} < i$. The aim of the 3-D HMM is to model the distribution of the collection of feature vectors $\{u_{i,j,k}; (i, j, k) \in \mathcal{C}\}$ at all the 3-D points in a grid. Every point is assumed to exist in one of a finite set of states. Denote the state at (i, j, k) by $s_{i,j,k}$, which is unobservable. Let the number of states be M. The model imposes statistical dependence among $u_{i,j,k}$ through the states. As in a typical HMM, the states are assumed to follow a certain Markovian property. The observed feature vectors $u_{i,j,k}$ are conditionally independent given the states. Specifically, the following assumptions are made.

1. $P\{s_{i,j,k} = l | context\} = a_{p,m,n,l}$, where $context = \{(s_{i^{'} j^{'} k^{'}}, u_{i^{'} j^{'} k^{'}}) : (i^{'}, j^{'}, k^{'}) < (i, j, k)\}$ is the set of states and feature vectors of all points preceding (i, j, k) in the lexicographic order. In addition, $p = s_{i,j,k-1}$, $m = s_{i-1,j,k}$ and $n = s_{i,j-1,k}$. Given any point (i, j, k), the three neighboring points that affect it are shown in Figure 1 (a).

2. Given the state $s_{i,j,k}$ of a point (i, j, k), the feature vector $u_{i,j,k}$ follows a multivariate Gaussian distribution parametrized by a covariance matrix and a mean vector determined by the state. For a state l, we denote the corresponding covariance matrix and mean vector by Σ_l and μ_l. Recall that the probability density function (pdf) of a d-dimensional Gaussian distribution is

   $$b_l(u) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_l|}} e^{-\frac{1}{2}(u-\mu_l)^{'} \Sigma_l^{-1} (u-\mu_l)}.$$

3. If the state of point (i, j, k) is known, its observed feature $u_{i,j,k}$ is conditionally independent of the rest of the points in the 3-D grid.

3. PARAMETER ESTIMATION

For the 3-D HMM, we need to estimate the transition probabilities $a_{p,m,n,l}, p, m, n = 1, \cdots, M$, the mean vectors μ_l, and the covariance matrices Σ_l for each state l. The Viterbi training approach used to estimate 1-D HMM is adopted.

We denote the collection of parameters collectively as ψ and mark the iteration step by a superscript (the initial parameter set being $\psi^{(0)}$). In order to update $\psi^{(t+1)}$ from $\psi^{(t)}$, we first identify the combination of states ($s^*_{i,j,k}$) with the maximum a posteriori (MAP) probability, conditioned on the observed vectors and parameters $\psi^{(t)}$. The parameters ($\psi^{(t+1)}$) are then computed by assuming states $s^*_{i,j,k}$ are the true underlying states. If the true states were known, the maximum likelihood estimation of the parameters would be easy to obtain. The mean vector μ_l and the covariance matrix Σ_l are simply the sample mean and sample covariance matrix of all the observed vectors $u_{i,j,k}$ whose states $s^*_{i,j,k} = l$. The transition probabilities $a_{p,m,n,l}$ are computed by the empirical frequencies. In our experiment, k-means clustering is used to generate an initial set of $s^*_{i,j,k}$.

Subject to the constraint $\sum_{l=1}^{M} a_{p,m,n,l} = 1$ for any p, m, and n, the transition probabilities comprise $M^3(M-1)$ free parameters. Due to the large number of parameters even with a moderate M, we regularize the transition probabilities by a partial 3-D dependence. In particular, if the dependence along the Z axis is ignored, the model is reduced to a 2-D HMM and the transition probabilities are $\bar{a}_{m,n,l}$, where $m = s_{i-1,j,k}$, $n = s_{i,j-1,k}$ and $l = s_{i,j,k}$.

The 3-D transition probabilities are regularized towards the 2-D probabilities by a linear combination:

$$\tilde{a}^{(t+1)}_{p,m,n,l} = \alpha a^{(t+1)}_{p,m,n,l} + (1-\alpha)\bar{a}^{(t+1)}_{m,n,l} \ . \qquad (1)$$

The parameter $\alpha \in$ [0,1] controls the extent of 3-D dependence. The value α = 1 corresponds to a pure 3-D model while α = 0, a pure 2-D model. It is shown in the experiment that an intermediate value of α is often preferable.

The key computational issue in Viterbi training is to solve the MAP states $\{s^*_{i,j,k} : (i, j, k) \in \mathcal{C}\}$ under a given set of parameters. For 1-D HMM, the MAP sequence of states can be solved by the Viterbi algorithm. For 3-D HMM, there are M^{w^3} possible combinations of states. The Viterbi algorithm enables us to avoid exhaustive search along the Z axis. However, it is still necessary to consider all the possible combinations of states in every frame. Thus the computational complexity of searching for the optimal set of states using the Viterbi algorithm is at least $\Omega(wM^{2w^2})$. To address the computational difficulty, we propose the following locally optimal algorithm to search for states with the maximum a posteriori probability.

Proposed Algorithm

Define the set of points with a fixed Y and Z coordinate in a 3-D grid as a row denoted by $R_{j,k} = \{(i, j, k) : 0 \leq i \leq (w-1)\}$. Let $\mathcal{D} = \{(j, k) : 0 \leq j \leq w-1, 0 \leq k \leq w-1\}$. Denote the sequence of states and observed vectors in row $R_{j,k}$ by $\mathbf{s}_{j,k}$ and $\mathbf{u}_{j,k}$ respectively. Rows are processed in the lexicographic order defined as: $(j', k') < (j, k)$ if $k' < k$ or $k' = k$, $j' < j$. In the following algorithm, we denote the states (in row $R_{j,k}$) obtained at pass t by $\mathbf{s}^t_{j,k}$. The approach is illustrated in Figure 1(b).

1. Initialize $t \leftarrow 0$.

2. Initialize $k \leftarrow 0$, $j \leftarrow 0$.

3. If ($t = 0$) $\mathcal{S}_{j,k} = \{\mathbf{s}^t_{j',k'}, (j', k') < (j, k)\}$.
   $\mathcal{F}_{j,k} = \{\mathbf{u}_{j',k'}, (j', k') \leq (j, k)\}$.
   If ($t > 0$)
   $\mathcal{S}_{j,k} = \{\mathbf{s}^t_{j',k'}, (j', k') < (j, k)\} \cup \{\mathbf{s}^{(t-1)}_{j',k'}, (j', k') > (j, k)\}$.
   $\mathcal{F}_{j,k} = \{\mathbf{u}_{j',k'}, (j', k') \in \mathcal{D}\}$.

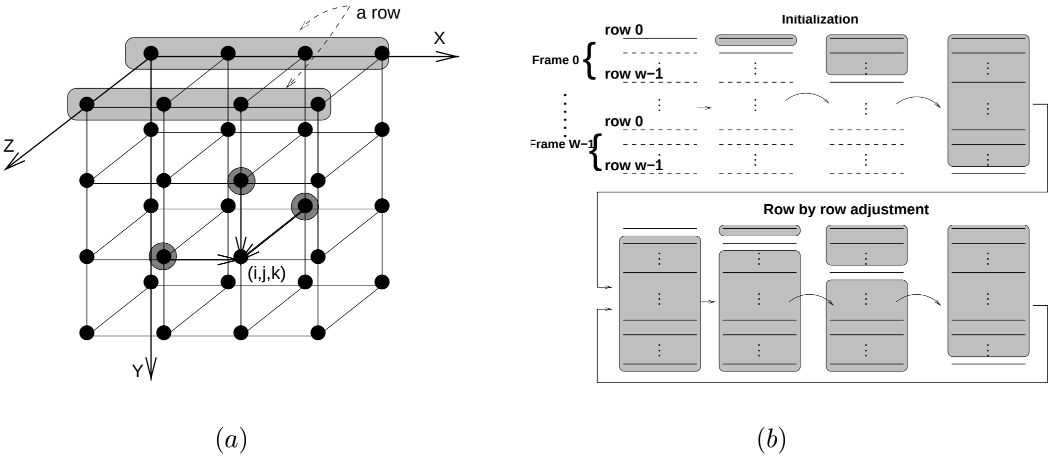

Fig. 1. (a) A 3-D grid. Given the states of all the points that precede point (i, j, k), only the states of the three indicated neighboring points affect the distribution of the state at (i, j, k). (b)The process of updating the sequence of states in each row recursively. States and observed vectors in the shaded rows are included in the condition for solving the sequence of states in a current row that has the maximum a posteriori probability. After initialization, the scan through all the rows can be repeated in several passes.

4. Search for $\mathbf{s}^*_{j,k}$ with MAP conditioned on $\mathcal{S}_{j,k} \cup \mathcal{F}_{j,k}$.
5. $\mathbf{s}^t_{j,k} \leftarrow \mathbf{s}^*_{j,k}$, $j \leftarrow j + 1$.
6. If $j < w$, go back to step 3. Otherwise,
   (a) $j \leftarrow 0$, $k \leftarrow k + 1$.
   (b) If $k < w$, go back to step 3. Otherwise,
      i. $t \leftarrow t + 1$.
      ii. If $t < P_{max}$, go to step 2. Otherwise, stop.

The value P_{max} is the pre-selected number of passes the procedure will scan the 3-D grid. As seen above, the initial states are obtained by a greedy technique. For each row, the sequence of states with MAP conditioned on the states in all the preceding rows and the observed vectors in the preceding and current rows is selected. The difference between the search for states during initialization and succeeding steps, lies in the conditioned information as elaborated in the algorithm. 1-D Viterbi is used to search for the MAP states $\mathbf{s}^*_{j,k}$ at step 4 given by:

$$\begin{aligned}&\arg\max_{\mathbf{s}_{j,k}} [\\ &\textstyle\sum_{i=0}^{w-1} \left(\log b_{s_{i,j,k}}(u_{i,j,k}) + \log a_{\bar{s}_{i,j,k-1}, s_{i-1,j,k}, \bar{s}_{i,j-1,k}, s_{i,j,k}}\right)\\ &\qquad + \log a_{\bar{s}_{i,j+1,k-1}, \bar{s}_{i-1,j+1,k}, s_{i,j,k}, \bar{s}_{i,j+1,k}}\\ &\qquad + \log a_{s_{i,j,k}, \bar{s}_{i-1,j,k+1}, \bar{s}_{i,j-1,k+1}, \bar{s}_{i,j,k+1}}]\end{aligned}$$

The conditioned states (assumed given) are denoted by $\bar{s}_{\cdot,\cdot,\cdot}$ in order to distinguish from the states $s_{i,j,k}$ to be optimized. Due to lack of space we have omitted the proof of the above equation. The complexity of using 1-D Viterbi to search the optimal states of a row is $O(wM^2)$. All w^2 rows of a 3-D grid are processed in $O(M^2w^3)$ time. Thus the proposed algorithm runs in *polynomial time* (in both the number of states and problem size) compared to using unconstrained Viterbi to search states globally which would run in *exponential time*, $\Omega(wM^{2w^2})$ as shown before.

α	$\Gamma_1(\gamma_1)$	$\Gamma_2(\gamma_2)$
0.0	35.67(0.32)	1.35(0.54)
0.2	35.67(0.32)	1.33(0.54)
0.4	35.67(0.32)	1.29(0.52)
0.6	35.67(0.32)	1.49(0.54)
0.8	35.67(0.32)	1.81(0.53)
1.0	35.67(0.32)	1.92(0.55)

Table 1. A comparison study of a 3-D image (black sphere in a white cube, w=100, σ = 0.9) segmentation with varying α. The values $\Gamma_i(\gamma_i)$ are mis-segmentation rates explained in the text.

4. EXPERIMENTS

We applied 3-D HMM to segment synthetically generated volume images. Images representing black spheres inside a white cube and some with a third shade (gray) were generated as follows. Each color voxel, black (μ = 0), white (μ = 1) (and gray μ = 0.5 for 3 class images), was perturbed by an additive Gaussian noise $\sim N(0, \sigma^2)$ and the voxel values were truncated to lie in the interval $[-2\sigma, 1 + 2\sigma]$. For the purpose of displaying images, voxel values in the interval $[-2\sigma, 1 + 2\sigma]$ were scaled to $[0, 255]$.

Segmentation using 3-D HMM and Gaussian Mixture Model (GMM) based clustering have been compared in Figure 2. For results shown in the figure, w = 100 and 2-D frames with z = $w/4$, $w/2$ and $4w/5$ respectively are shown for the original as well as segmented images. Table 1 compares numerical results of 3-D HMM and GMM based segmentation. Also included are results after k-means cluster-

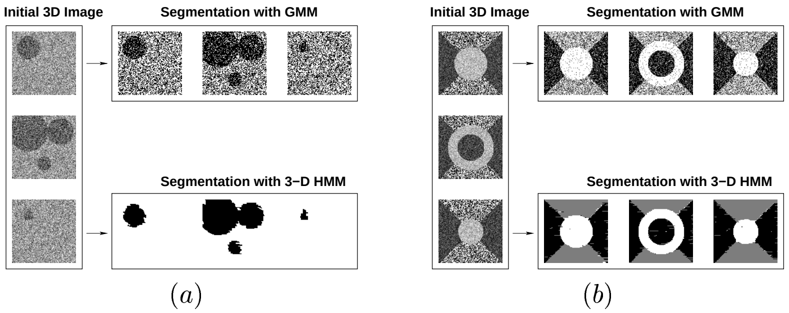

Fig. 2. Compare the segmentation performance of 3-D HMM algorithm and clustering using Gaussian Mixture Model. The value of σ used for Experiment (a) is 0.9 and for Experiment (b) is 0.3

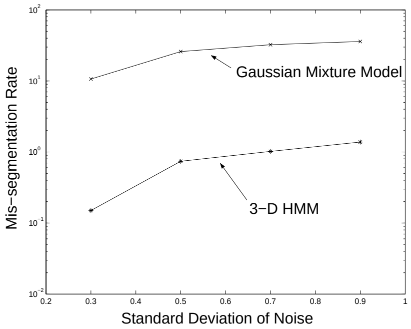

Fig. 3. A comparison study of a 3-D image (2 black spheres in a white cube, w=100, α = 0.5) segmentation with increasing σ.

ing. In the table, Γ_1 and Γ_2 denote mis-segmentation rates with k-means and 3-D HMM respectively while γ_1 and γ_2 denote boundary mis-segmentation rates. As expected, a trade-off value of α (here 0.4) yields best segmentation performance. The mis-segmentation rate for the same 3-D image with GMM based clustering is found out to be 37.22. We see that 3-D HMM outperforms GMM by a very large margin which is expected as GMM ignores spatial dependency information. Interestingly, for 3-D HMM, nearly 30 to 50 $\%$ mis-segmentation occurs among boundary points each time. We argue that boundary points could be considered as belonging to either class. Mere rounding of distances (for geometrical shapes generated) puts them in the class where they belong. Figure 3 shows the performance of 3-D HMM with increasing variance of noise. Comparison with GMM has been made. It is evident that segmentation using 3-D HMM is consistent even for large noise. About 50 passes over a 3-D image were sufficient to achieve a stable segmentation performance. The computer time for a single pass over a $100 \times 100 \times 100$ image was estimated as nearly 2 seconds on a 2.6 GHz Xeon based processor running Linux. All the results shown were obtained using the proposed method of estimating parameters. We did not use unconstrained Viterbi as the computational complexity was found out to be exponential in Section 3.

5. CONCLUSIONS

We proposed 3-D HMM and suggested a fast parameter estimation technique. Next, we demonstrated its performance on synthetic 3-D images. Performance over a large range of noise variance was found to be consistent. In the future, we wish to incorporate 3-D HMM based modeling and learning into real-world multi-dimensional image applications. Several manifestations of the third dimensional information exist in the form of spatial, spectral or temporal as in MRI, hyper-spectral images and video, respectively.

6. REFERENCES

[1] J. Boreczky, L. Wilcox, “A hidden Markov model framework for video segmentation using audio and image features,” *Proc. IEEE Conf. on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 3741-3744, 1998.

[2] S. Eickeler, S. Muller, G. Rigoll, “Improved face recognition using pseudo 2-D hidden Markov models,” *Proc. Workshop on Advances in Facial Image Analysis and Recognition Technology in conjunction with ECCV*, Freiburg, Germany, June 1998.

[3] F. Hulsken, F. Wallhoff, G. Rigoll, “Facial expression recognition with pseudo-3D hidden Markov models,” *Proc. DAGM-Symposium, Lecture Notes in Computer Science*, vol. 2191, pp. 291-297, 2001.

[4] J. Li, R. M. Gray, R. A. Olshen, “Multiresolution image classification by hierarchical modeling with two dimensional hidden Markov models,” *IEEE Trans. Information Theory*, vol. 46, no. 5, pp. 1826-41, August 2000.

[5] J. Li, J. Z. Wang, “Automatic linguistic indexing of pictures by a statistical modeling approach, ” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1075-1088, 2003.

[6] M. Li’evin, N. Hanssen, P. Zerfass, E. Keeve, “3D Markov random fields and region growing for interactive segmentation of MR data,” *Proc. MICCAI*, 2001.