Interpretation of Complex Scenes Using Generative Dynamic-Structure Models

Sinisa Todorovic and Michael C. Nechyba
ECE Department, University of Florida, Gainesville, FL 32611
{sinisha, nechyba}@mil.ufl.edu

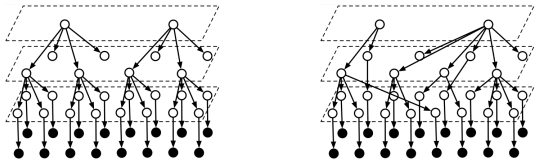
Abstract

We propose a generative modeling framework – namely, Dynamic Tree Structured Belief Networks (DTSBNs) and a novel Structured Variational Approximation (SVA) inference algorithm for DTSBNs – as a viable solution to object recognition in images with partially occluded object appearances. We show that it is possible to assign physical meaning to DTSBN structures, such that root nodes model whole objects, while parent-child connections encode component-subcomponent relationships. Therefore, within the DTSBN framework, the treatment and recognition of object parts requires no additional training, but merely a particular interpretation of the tree/subtree structure. As such, DTSBNs naturally allow for multi-stage object recognition, in which initial recognition of object parts induces recognition of objects as a whole. As our reported experiments show, this explicit, multi-stage treatment of occlusion outperforms more traditional object-recognition approaches, which typically fail to account for occlusion in any principled or unified manner.

1. Introduction

In this paper, we address the problem of object recognition in images, where some or all objects may be partially occluded;[1] for the purposes of this paper, we assume that objects are opaque and rigid, and are statistically independent. Generally speaking, object recognition entails three related components: (1) localization, (2) detection and, finally, (3) recognition of objects. A number of factors contribute to the difficulty of this problem including variations in camera quality and position, wide-ranging illumination conditions, extreme scene diversity, and the randomness of object appearances and locations in scenes. Partial occlusion of objects substantially complicates the problem further [1]–[3]. Thus, there is inherent uncertainty in how observed visual evidence in images should be attributed to infer object types and their relationships.

We seek a framework that is sufficiently expressive to cope with this uncertainty, jointly addresses the three sub-problems for the object recognition problem in a unified manner, and extends seamlessly to partially occluded settings. As we discuss further below, generative dynamic-structure models – specifically, Dynamic Tree-Structured Belief Networks (DTSBNs) [4], [5] – offer such a framework. DTSBNs can locate and detect objects simultaneously through a structure search over dynamic forests of trees, such that root nodes model whole objects. As illustrated in Fig. 1, they differ from fixed-structure TSBNs, and overcome the "blocky segmentation" characteristic of TSBNs [6]–[9]. Not only do root nodes in DTSBNs model whole objects, but subtrees model object parts at various scales, so that parent-child connections encode component-subcomponent relationships. Therefore, within the DTSBN framework, the treatment and recognition of object parts requires no additional training, but merely a particular interpretation of the tree/subtree structure. As such, DTSBNs offer a viable means of recognizing objects, even when partially occluded, through recognition of their constituent parts.



(a) (b)

Fig. 1. (a) Fixed-structure TSBN; (b) DTSBN; white nodes denote hidden RVs and black nodes denote observable RVs.

When considering the recognition of detected objects, one can employ a range of strategies. Traditionally, individual pixels are labeled as one of C classes, and then a majority vote decides on the class of the object as a whole. When objects are partially occluded, however, such an approach may not yield the best results. Instead, we propose a different approach, where we assign class labels not to individual pixels but rather to *object parts*, or groups of pixels, as encoded through DTSBN subtrees. Majority voting then proceeds not on pixel class labels, but over object-part class labels. We hypothesize that such an approach to recognition may be more resilient to occlusion and therefore more appropriate when considering the recognition of partially occluded objects.

Thus, our choice of a generative, dynamic-structure framework is directly driven by our image interpretation strategy and goals, and appears better suited than alternative statistical approaches, such as descriptive, pseudo-

[1]Herein, we focus on single-image interpretation. We do not rely on auxiliary information provided, for example, by image sequences of the same scene, where occlusions may be transitory.

descriptive or discriminative models [10]. Descriptive models lack the necessary structure we seek to exploit in object recognition, while discriminative approaches directly model conditional distributions of hidden variables given observables, and thereby loose the convenience of assigning physical meaning to the statistical parameters of the model.

One of the principal challenges in applying DTSBNs to image interpretation is the derivation of efficient algorithms for inference, whereby model parameters are learned. Herein, we propose a novel Structured Variational Approximation (SVA) algorithm that relaxes independence assumptions in prior work [4], [5], where positions of nodes are treated as independent of the model's structure. We, however, take into account their statistical dependence and, thus, achieve significantly faster convergence over currently available algorithms.

This paper is organized as follows. We first define DTSBNs and discuss probabilistic inference and learning for these models. Next, we report experimental results on DTSBN-based unsupervised image segmentation. We then proceed to experimental results on supervised image classification of scenes with partially occluded objects. We contrast performance of DTSBNs with Markov Random Fields (MRFs) [11], Discriminative Random Fields (DRFs) [12], and fixed-structure TSBNs [9], and show that DTSBNs, trained using SVA, outperform all these alternative modeling paradigms. Furthermore, in partially occluded settings, we demonstrate that recognition strategies conditioned on correct labeling of object parts, as identified through DTSBN learning, improves overall whole-object recognition. These results suggest that the proposed multistage recognition procedure allows for more flexible and accurate interpretation of complex scenes with occlusions.

2. Definition of DTSBN

A DTSBN is a directed graph with nodes in set V, organized in hierarchical levels from the finest (i.e., leaf level) to the coarsest (i.e., root level). The network connectivity is represented by a matrix Z, where entry $z(ij)$=1 if there is a connection between nodes i and j. Z also contains an additional zero ("root") column, where entries $z(i0)$ are set to 1 if i is a root node. Connections are established under constraints that no leaf node can be a root and that a node can connect only to the nodes at the next coarser level. We define the distribution over tree connectivity as

$$P(Z) = \prod_{i,j \in V} [\gamma(ij)]^{z(ij)} \ , \qquad (1)$$

where $\gamma(ij)$ is the probability of i being the child of j.

Next, each node is characterized by the position of the object part it represents relative to the position of its parent, thereby explicitly expressing geometric component-subcomponent relationships. The joint probability of R=$\{r_i\}$, $\forall i \in V$, is given by

$$P(R|Z) = \prod_{i,j \in V} \left[\frac{\exp\left(-\frac{1}{2}(r_i - r_j)^T \Sigma_{ij}^{-1}(r_i - r_j)\right)}{2\pi |\Sigma_{ij}|^{\frac{1}{2}}} \right]^{z(ij)} , \qquad (2)$$

where Σ_{ij} denotes the covariance matrix representing the size of object parts at various scales.

Further, each node i is characterized by a state, represented by an indicator random variable, $x(ik)$, such that $x(ik)$=1 if i is in state k. Node states denote labels of image classes in set M. The state of i is conditioned on the state of its parent j and is given by conditional probability tables P_{ij}^{kl}. The joint probability of all state variables, X=$\{x(ik)\}$, $\forall i \in V$, can be expressed as

$$P(X|Z) = \prod_{i,j \in V} \prod_{k,l \in M} \left[P_{ij}^{kl}\right]^{x(ik)x(jl)z(ij)} \ . \qquad (3)$$

Though x's, $\forall i \in V$, represent the same image classes at all levels, their positions in the tree structure additionally encode component-subcomponent relationships in an image. Thus the root node represents the whole object, and its children, object parts.

Next, node states determine the likelihood of observable random vectors, y_i, connected to the leaf nodes, denoted as V^0. The joint pdf of all observables Y=$\{y_i\}$, $\forall i \in V^0$, is given by,

$$P(Y|X) = \prod_{i \in V^0} \prod_{k \in M} p(y_i | x(ik)=1) \ , \qquad (4)$$

where $p(y_i|x(ik)=1)$ is modeled as a mixture of Gaussians with G components.

Finally, our DTSBN is fully specified by the joint distribution $P(Z, X, R, Y)$=$P(Z)P(X|Z)P(R|Z)P(Y|X)$.

3. Probabilistic Inference and Learning

Due to the complexity of DTSBNs, the exact computation of $P(X|Y)$, required, for example, for Bayesian pixel labeling, is intractable. Therefore, to compute $P(X|Y)$, we resort to approximate methods, which are generally subdivided into deterministic approximations [13] and Monte-Carlo methods [14]. Note that we need to learn our dynamic-tree model in the space of possible tree structures. Due to an intractably large number of configurations, Monte-Carlo methods require prohibitively extensive sampling, as we demonstrate for the case of binary 8×8 images in Section 4. Consequently, we propose a variational inference method – namely, Structured Variational Approximation (SVA).

Variational-approximation inference methods can be viewed as minimizing a convex cost function known as *free energy*, which measures the accuracy of an approximate

probability distribution [13], [15]. Essentially, the idea is to approximate the true intractable posterior distribution, in our case $P(Z,X,R|Y)$, by a simpler distribution $Q(Z,X,R)$ closest to $P(Z,X,R|Y)$, by minimizing the free energy $J(Q,P)$:

$$J(Q,P)=\sum_{Z,X,R} Q(Z,X,R)\log\frac{Q(Z,X,R)}{P(Z,X,R,Y)}. \quad (5)$$

We constrain the solution of the variational distribution to the form

$$Q(Z,X,R)=Q(Z)Q(X|Z)Q(R|Z)\ . \quad (6)$$

This formulation enforces that both state-indicator variables X and position variables R should be statistically dependent on the tree connectivity Z. Since these dependencies are significant in the prior, one should expect them to remain so in the posterior. Therefore, the chosen form appears to be more appropriate for approximating the true posterior than the Q function proposed by Storkey and Williams in [5] of the form $Q(Z,X,R)$=$Q(Z)Q(X|Z)Q(R)$. The R variables contribute in the search of dynamic-tree structure by favoring connections among neighboring nodes and, as such, are clearly not independent of network connectivity Z.

The approximating distributions are defined as

$$Q(Z)=\prod_{i,j\in V}\left[\delta(ij)\right]^{z(ij)}, \quad (7)$$

$$Q(X|Z)=\prod_{i,j\in V}\prod_{k,l\in M}\left[Q_{ij}^{kl}\right]^{x(ik)x(jl)z(ij)}, \quad (8)$$

$$Q(R|Z)=\prod_{i,j\in V}\left[\frac{\exp\left(-\frac{1}{2}(r_i-\mu_j)^T\Omega_{ij}^{-1}(r_i-\mu_j)\right)}{2\pi|\Omega_{ij}|^{\frac{1}{2}}}\right]^{z(ij)}, \quad (9)$$

where $\delta(ij)$ corresponds to $\gamma(ij)$, Q_{ij}^{kl} is analogous to P_{ij}^{kl}, and μ_j and Ω_{ij} are the mean and the covariance of the parent j position, respectively.

3.1. Update Equations

By minimizing the free energy $J(Q,P)$, we derive the update equations for the parameters of the variational distribution $Q(Z,X,R)$. For space reasons, below we summarize the final derivation results. In the extended version of this paper, we report the full derivation. In the following equations, we will use κ to denote an arbitrary normalization constant, the definition of which may change from equation to equation.

From Eq. (7) and Eq. (8), we derive the update equation for the probability m_i^k that node i is in state k as

$$\hat{m}_i^k=\sum_{j\in V}\delta(ij)\sum_{l\in M}Q_{ij}^{kl}m_j^l\ . \quad (10)$$

Then, from $\sum_{k\in M}Q_{ij}^{kl}$=1 and $\partial J(Q,P)/\partial Q_{ij}^{kl}$=0 we arrive at:

$$\widehat{Q}_{ij}^{kl}=\kappa P_{ij}^{kl}\lambda_i^k\ ,\ \forall i,j\in V\ ,\ \forall k,l\in M\ , \quad (11)$$

where auxiliary parameters λ_i^k are computed, $\forall k\in M$, as

$$\hat{\lambda}_i^k=\begin{cases}\prod_{c\in V}\left[\sum_{a\in M}P_{ci}^{ak}\lambda_c^a\right]^{\delta(ci)}, & \forall i\in V\setminus V^0\ ,\\ p(y_i|x(ik)=1) & ,\ \forall i\in V^0\ .\end{cases} \quad (12)$$

From Eq. (12), we note that the $\hat{\lambda}$'s are computed by propagating λ messages of corresponding children nodes upward. Thus, $\widehat{Q}$ can be computed by making a single pass up the tree and $\hat{m}$ can be computed by propagating state probabilities in a single pass downward. This upward-downward propagation is very reminiscent of Pearl's message passing scheme. For the special case when $\delta(ij)=1$ only for one parent j, we obtain the standard λ-π rules of Pearl's message passing scheme for TSBNs [5], [16].

The Gaussian assumption in Eq. (9) implies that $Q(R|Z)$ is fully characterized by μ_i and Ω_{ij}, $\forall i,j\in V$. Assuming that Ω_{ij} is positive definite, from $\partial J(Q,P)/\partial\Omega_{ij}$=0 and $\partial J(Q,P)/\partial\mu_i$=0, for all nodes where $\delta(ij)\neq 0$, we derive

$$Tr\{\widehat{\Omega}_{ij}^{-1}\}=Tr\{\Sigma_{ij}^{-1}\}\left(1-\sum_{p\in V}\delta(jp)\frac{Tr\{\Sigma_{ij}^{-1}\Omega_{jp}\}^{\frac{1}{2}}}{Tr\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{\frac{1}{2}}}\right)+$$
$$+\sum_{c\in V}\delta(ci)Tr\{\Sigma_{ci}^{-1}\}\left(1-\frac{Tr\{\Sigma_{ci}^{-1}\Omega_{ci}\}^{\frac{1}{2}}}{Tr\{\Sigma_{ci}^{-1}\Omega_{ij}\}^{\frac{1}{2}}}\right) \quad (13)$$

$$\widehat{\mu}_i=\left[\sum_{g,c,p\in V}\left(\delta(ci)\delta(ip)\Sigma_{ci}^{-1}+\delta(gc)\delta(ci)\Sigma_{gc}^{-1}\right)\right]^{-1}\cdot$$
$$\cdot\sum_{g,c,p\in V}\left(\delta(ci)\delta(ip)\Sigma_{ci}^{-1}\mu_p+\delta(gc)\delta(ci)\Sigma_{gc}^{-1}\mu_c\right). \quad (14)$$

Hence, both $\widehat{\mu}_i$ and $\widehat{\Omega}_{ij}$ are updated summing over children and parent nodes and, therefore, must be iterated until convergence.

Clearly, there is no unique solution to Eq. (13). To update the $\Omega's$, we assume that both the $\Sigma's$ and the $\Omega's$ are diagonal. Also, it is not guaranteed that $\widehat{\Omega}_{ij}$, as a solution of Eq. (13), is positive definite, because we assume that relative distances $(r_i-\mu_j)$ and $(r_j-\mu_p)$ are uncorrelated, where i-j-p form a node-parent-grandparent triad. However, in our extensive experimentation, this rarely ever occurs. In those seldom cases that we do encounter this problem, we "freeze" the contribution of that node to the overall belief propagation, using its old parameter values from the previous iteration. This approach

is justified in light of the incremental variant of the EM algorithm discussed by Neal and Hinton in [17].

Finally, minimizing $J(Q,P)$ with respect to connectivity probabilities, $\delta(ij)$, and accounting for the constraint $\sum_{j\in V}\delta(ij)=1$, we obtain

$$\widehat{\delta}(ij)=\kappa\gamma(ij)\ \exp(A_{ij}-B_{ij})\ ,\ \forall i,j\in V\ , \qquad (15)$$

where A_{ij} represents the contribution of the actual feature statistics (i.e., observable variables Y) to the connectivity distribution and B_{ij} represents the geometric properties of the network connectivity. These are computed, $\forall i,j\in V$, as follows:

$$A_{ij}=\sum_{l\in M} m_j^l\ \log\left(\sum_{k\in M} P_{ij}^{kl}\lambda_i^k\right)\ , \qquad (16)$$

$$B_{ij}=0.5\log|\Sigma_{ij}|/|\Omega_{ij}|+0.5\ Tr\{\Sigma_{ij}^{-1}\Omega_{ij}\}+ \\ +0.5\sum_{p\in V}\delta(jp)D_{ijp}+0.5\sum_{c\in V}\delta(ci)D_{cij}\ , \qquad (17)$$

$$D_{uvw}=Tr\{\Sigma_{uv}^{-1}\Omega_{vw}\}+Tr\{\Sigma_{uv}^{-1}(\mu_v-\mu_w)(\mu_v-\mu_w)^T\}- \\ -2Tr\{\Sigma_{uv}^{-1}\Omega_{uv}\}^{\frac{1}{2}}Tr\{\Sigma_{uv}^{-1}\Omega_{vw}\}^{\frac{1}{2}}\ . \qquad (18)$$

3.2. Learning

Variational inference presumes that the parameters that specify $P(Z,X,R,Y)$ are available. In order to learn these parameters, initially, we build a balanced TSBN, where every parent has exactly four children nodes. Then, using the exact inference of Pearl's message passing scheme [16], we learn P_{ij}^{kl}, m_i^k, and parameters of the Gaussian mixture $p(y_i|x(ik)=1)$,[2] $\forall i,j\in V$, $\forall k,l\in M$. Here, we determine the number of tree levels, L, and the number of components in a mixture of Gaussians, G, by maximizing $P(Y)$ (readily available from Pearl's belief propagation) for several L and G values. Since DTSBNs are generalized TSBNs, our experimentation suggests that optimizing L and especially G with respect to the TSBN is justified. Further, we initialize $P(Z)$ such that the parameters $\gamma(ij)$ are uniform across all possible parents of i. To initialize $P(R|Z)$, we equate Σ_{ij} to the area of corresponding dyadic squares. As for the parameters of the variational distribution $Q(Z,X,R)$, we first assign to μ_i the coordinates of corresponding dyadic squares. Finally, we initialize all the variational parameters to the values of the corresponding parameters of $P(Z,X,R,Y)$, such that $\delta(ij)=\gamma(ij)$, $Q_{ij}^{kl}=P_{ij}^{kl}$, $\Omega_{ij}=\Sigma_{ij}$.

After initialization, we optimize the parameters of the variational distribution according to the given update equations. First, we fix $Q(Z)$ and update the rest of the parameters. Then, we use these new parameter values to update the $\delta's$. Once optimized, the $\delta's$ specify the most likely tree connectivity. Unlike in [5], we find the maximum probability $\delta(ij)$, $\forall i,j\in V$, and establish only that connection, deleting other candidate connections with lower probability. In this manner, we build a forest of new TSBNs that are not balanced any more, yet preserve their tree structure. Finally, we close the learning loop, again performing belief propagation for each subtree using Pearl's message passing scheme.

[2]For learning the parameters of a mixture of Gaussians, which we assume equal for all nodes, we employ the EM algorithm [17].

4. Experiments and Discussion

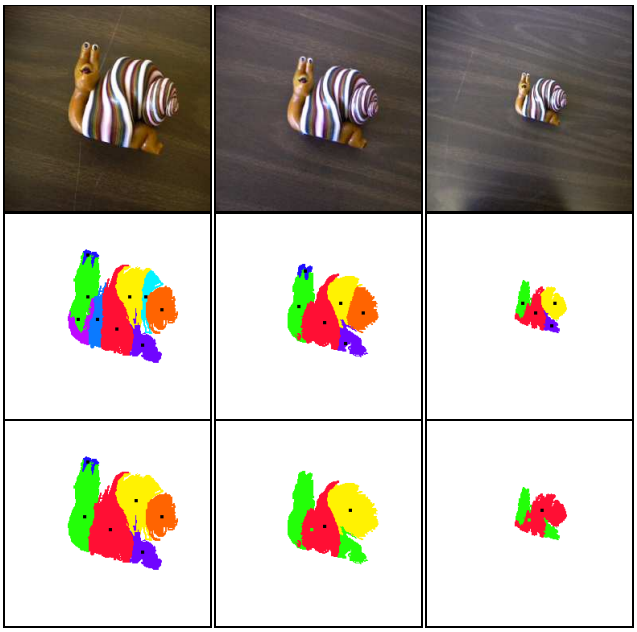
4.1. Image Segmentation

We first discuss the clustering capabilities of DTSBNs[3] in unsupervised settings on two data sets: 50 binary 8×8 images, a typical sample of which is depicted in Fig. 4, and 50 256×256 images, examples of which are shown in Figs. 2 and 6. Here, we specify observables, Y, for DTSBNs as binary or RGB color values, respectively. Despite this relatively weak description of image features, the chosen feature space proves successful for DTSBN-based segmentation of the given set of test images. This fact demonstrates good localization and detection capabilities for DTSBN models, which clearly could be enhanced by using more powerful image features (e.g., SIFT descriptors [18]). From the results presented in Figs. 2, 3 and 4, we observe that DTSBNs, trained with SVA, are able to correctly assign one subtree per "object" in an image, where a cluster of pixels descending from a root corresponds to the whole object, and clusters descending from higher level nodes underneath the root correspond to object parts.

From Figs. 2 and 3, we observe that DTSBNs preserve tree structure for objects across images subject to affine transformations (translation, rotation and scaling). In Fig. 2, note that the level-4 clustering for the largest-object scale (Fig. 2a, bottom) corresponds to the level-3 clustering for the medium-object scale (Fig. 2b, middle); similarly, the level-4 clustering for the medium-object scale (Fig. 2b, bottom) corresponds to the level-3 clustering for the small-object scale (Fig. 2c, middle). In other words, as the object transitions through scales, the tree structure changes by eliminating the lowest-level layer; however, higher-order structure remains intact.

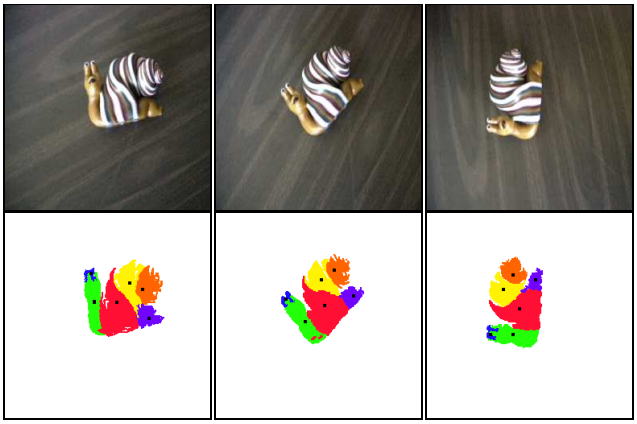
We also note that the estimated positions of higher-level hidden variables are very close to the center of mass of object parts, as well as of whole objects. We estimate the error of root-node positions (x_r,y_r) as a distance in pixels from the actual center of mass of hand-labeled objects, $d_{err}=\sqrt{(x_r-x_{CM})^2+(y_r-y_{CM})^2}$. The obtained averaged error values are $d_{err}^{8\times8}=1.8$ (22% of the image size), and $d_{err}^{256\times256}=11.4$ (4% of the image size),

[3]Each DTSBN node defines one cluster composed of those DTSBN leaf nodes (pixels) that are that node's descendants.



(a) (b) (c)

Fig. 2. DTSBN-based pixel clustering in RGB color space: scale invariance. (top row) 256×256 images; (middle row) pixel clusters with the same parent at level 3; (bottom row) pixel clusters with the same parent at level 4; points mark the position of parent nodes; DTSBN structure is preserved through scales.



(a) (b) (c)

Fig. 3. DTSBN-based pixel clustering in RGB color space: rotation invariance. (top row) 256×256 images; (bottom row) pixel clusters with the same parent at level 3. DTSBN structure is preserved over rotations.

for the 8×8 and 256×256 images, respectively. The error significantly decreases as the image size increases, because in summing node positions over parent and children nodes, as in Eq. (13) and Eq. (14), more statistically significant information contributes to the position estimates.

Next, we compare convergence of SVA with the following inference algorithms: Gibbs sampling [19], mean-field variational approximation (MFVA) proposed in [4], and

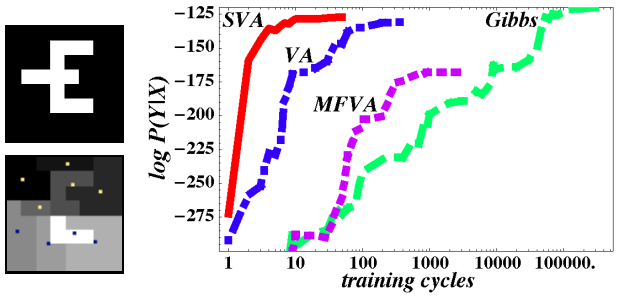


Fig. 4. DTSBN-based pixel clustering: (top left) 8×8 binary image; (down left) 9 clusters of pixels specified by 9 parents at level 1; points mark the position of parent nodes; (right) averaged log-likelihoods $\log P(Y|X)$ for the binary image to the left; SVA converges significantly faster than other inference methods.

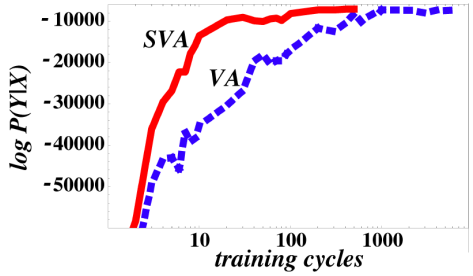


Fig. 5. Averaged log-likelihoods $\log P(Y|X)$ for 256×256 image in Fig. 2a; SVA is more stable and converges significantly faster than VA.

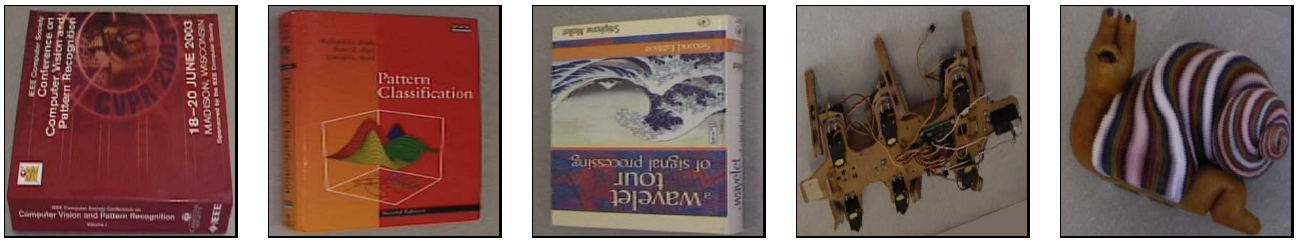
variational approximation (VA)[4] discussed in [5]. In Figs 4 and 5, we plot the averaged log-likelihood, $\log P(Y|X)$, obtained after the specified number of iteration steps for images in Fig. 4(left) and Fig. 2a. Iteration was stopped when segmentation of the image stabilized. We averaged $\log P(Y|X)$ over the number of training cycles for a given logarithmic scale. Obviously, Gibbs sampling yields the greatest $\log P(Y|X)$, albeit at a huge computational price. With the increase in image size, Gibbs sampling becomes unfeasible and MFVA exhibits very poor performance. Therefore, for 256×256 images, we report results only for SVA and VA. SVA converges in the fewest number of iterations, an order of magnitude faster than the second-place VA.

The average clustering error over the 50 binary test images is 8% for VA and 6% for SVA, and, over the 50 256×256 images, is 8% for VA and 4% for SVA. Here, the reported results account only for "real objects" (i.e. not the background), and errors are relative to hand-labeled ground truth.

4.2. Image Classification

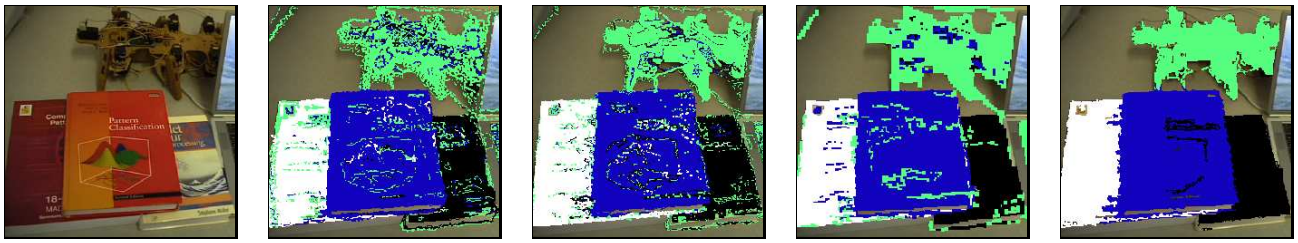
Here, we consider a finite number of objects representing image classes. Our training data set comprises 20 carefully chosen images per object to account for variability in

[4]As we noted before, the inference method proposed in [5] is also structured variational approximation. To differentiate that method from ours, we slightly abuse the notation.



O_1 O_2 O_3 O_4 O_5

Fig. 6. Training images of objects, which we refer to as O_1-O_5.



(a) 256×256 (b) MRF (c) DRF (d) TSBN (e) DTSBN

Fig. 7. Comparison of classification results for various statistical models; pixels are labeled with a color specific for each object; non-colored pixels are classified as background.

illumination and/or camera position, as illustrated in Fig. 6. Image classification is conducted on 50 test images, which contain a total of 162 partially occluded objects, examples of which are shown in Figs. 7a and 9a; Herein, we compare classification performance of DTSBNs, learned using SVA, with MRFs, DRFs, and TSBNs being representatives of descriptive, discriminative and simple-structure generative models, respectively.

For MRFs, we assume that the label field $P(X)$ is a homogeneous and isotropic MRF, given by the generalized Ising model with only pairwise nonzero potentials [11]. The likelihoods $P(y_i|x_i)$ are assumed conditionally independent given the labels. Thus, the posterior energy function is given by

$$U(X|Y) = \sum_{i \in V^0} \log P(y_i|x_i) + \sum_{i \in V^0} \sum_{j \in \mathcal{N}_i} V_2(x_i, x_j) \quad (19)$$

$$V_2(x_i, x_j) = \begin{cases} \beta_{MRF} & , if \quad x_i = x_j \ , \\ -\beta_{MRF} & , if \quad x_i \neq x_j \ . \end{cases} \quad (20)$$

where $\mathcal{N}_i$ denotes the neighborhood of i, $P(y_i|x_i)$ is a G-component mixture of Gaussians, and V_2 is the interaction parameter. DRFs are learned as proposed in [12]; for these models, the posterior energy function is given by

$$U(X|Y) = \sum_{i \in V^0} A_i(x_i, Y) + \sum_{i \in V^0} \sum_{j \in \mathcal{N}_i} I_{ij}(x_i, x_j, Y) \quad (21)$$

where $A_i = \log \sigma(x_i W^T y_i)$ and $I_{ij} = \beta_{DRF}(K x_i x_j + (1 - K)(2\sigma(x_i x_j V^T y_i) - 1))$ are the unary and pairwise potentials, respectively. Since the above formulation deals only with binary classification (i.e. $x_i \in \{-1, 1\}$), when estimating parameters $\{W, V, \beta_{DRF}, K\}$ for an object, we treat that object as a positive example, and all other objects as negative examples. Training of TSBNs is conducted as specified in [20].

Next, we briefly describe image features used for observables Y; a more detailed treatment of these features is given in [20]. We account for both color and texture. To extract color features, we choose the generalized RGB color space, $r = R/(R+G+B)$, and $g = G/(R+G+B)$, which effectively normalizes variations in brightness. For texture analysis, we choose the complex wavelet transform (CWT), due to its inherent representation of texture at different scales, orientations and locations. The CWT's directional selectivity is encoded in six subimages of coefficients oriented at angles $\pm 15°$, $\pm 45°$, and $\pm 75°$. To limit the dimensionality of our feature space, we apply our algorithm for adaptive feature selection proposed in [20], selecting only the two most discriminative subimages from the CWT feature set. Thus, to each pixel at position i we assign a feature (observable) vector y_i containing corresponding r, g, and CWT values.

Given the chosen feature representation, we perform MAP image classification; ground truth for each image is determined through hand-labeling of pixels. Here, we say that an object is recognized as that object if the majority of assigned pixel labels are equal to the true labeling. We refer to this strategy as traditional.

In Fig. 7, we illustrate an example of pixel labeling for a complex-scene test image, while in Fig. 8, we report the confusion matrix for DTSBNs over the entire test set. For the given set of test images, we detect all objects, although swapped object identities do occur in recognition, as represented by non-zero off-diagonal elements in the confusion matrix. Swapped-identity errors range from 21% for MRFs (worst) to 10% for DTSBNs (best); misclassified-pixel

	O_1	O_2	O_3	O_4	O_5
O_1	31	0	1	0	0
O_2	0	25	2	1	1
O_3	0	2	37	1	1
O_4	0	0	0	16	1
O_5	1	1	3	1	37

Fig. 8. The confusion matrix for DTSBN-based MAP classification of 50 test images containing a total of 32 O_1, 28 O_2, 43 O_3, 19 O_4, and 40 O_5 partially occluded objects; columns indicate ground truth.

errors averaged over the 50 test images and 162 objects are 17% for TSBNs and MRFs, 16% for DRFs, and 14% for DTSBNs.

From these results, we note that while DTSBNs outperform the other three models, in general, recognition performance suffers substantially when an image contains occlusions. To some extent, the results could have been improved had we employed more discriminative image features. However, better image features will not alleviate the fundamental problem of the traditional strategy: to train statistical models on full-appearance objects and to use these models for recognition of partially occluded objects. As such, we now focus on recognition strategies beyond majority rule that may prove to be more optimal for images with occlusion. We begin with a detailed two-class case study to motivate our approach.

From the confusion matrix in Fig. 8, we note that mislabeling of object O_3 as O_5 contributes more than any other single error to the reported swapped-identity error. Therefore, we further investigate classification of these two objects in greater detail. In Fig. 9, we present MAP classification results, assuming only two possible image classes – namely, O_3 and O_5. Next, in Fig. 10, we plot ROC curves for various decision boundaries between likelihoods of O_3 and O_5, where pixels labeled as O_5 are considered true positives, while pixels labeled as O_3 are considered true negatives. From Fig. 10, we observe that higher true positive rates come at the substantial cost of large increases in the false positive rate. On the other hand, biasing the decision boundary to favor true negatives, we obtain better recognition of O_3 than for MAP classification, with a permissible loss of sensitivity for O_5. For example, with a modified decision criterion (MDC), yielding a 90% true positive rate for the image in Fig. 9, we are able to reduce the swapped-identity error for DTSBNs from 10% to 8%, as can be observed from the new confusion matrix for DTSBNs in Fig. 11a over the same 50 test images with 162 partially occluded objects.

As the two-class example illustrates, majority-voting under MAP classification may not be the most appropriate recognition strategy for scenes with partial occlusion. However, even a better suited classification criterion does

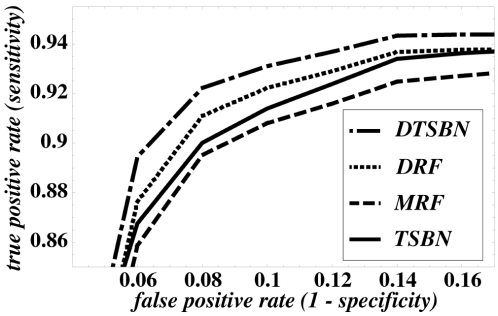


Fig. 10. ROC curves for the image in Fig. 9a with DTSBNs, TSBNs, DRFs and MRFs.

not yield satisfactory classification, as compared to standard recognition results for non-occluded object appearances [10], [20]. Therefore, it is necessary to introduce more radical changes to the traditional strategy. We speculate that in the face of the occlusion-problem, *recognition of object parts* is critical and should condition recognition of the object as a whole. Thus, we now propose another recognition strategy that proceeds in two stages. First, each "small" image region is classified as part of an object if the majority of its pixel labels, assigned through MAP or some other classifier, are equal to the true labeling of that object. That is, in the first stage, "small" image regions are classified as object parts. In the second stage, the whole object is recognized as that object if the majority of its object parts are recognized as components of that object.

Recall from Section 4.1 that DTSBNs are capable of capturing component-subcomponent structures at various scales, such that DTSBN root nodes represent the center of mass of distinct objects, while children nodes down the subtrees represent object parts. As such, DTSBNs provide a natural and seamless framework for identifying candidate image regions as object parts, requiring no additional training for such identification. Thus, the first stage of recognition now begins by treating children nodes (i.e. subtrees) of each root in a DTSBN as new roots of the corresponding image regions. We then assign labels to all pixels that are descendants of these new roots, and proceed with majority voting to label image regions (corresponding to subtrees) as particular object parts. Note that the treatment of subtrees here is exactly the same as whole-object trees before. Finally, the original root nodes inherit the class labels that are shared by the majority of their corresponding subtrees.

In Fig. 11b, we report the confusion matrix for this two-stage recognition strategy over the same 50 test images with 162 partially occluded objects. Note that the swapped-identity error has been reduced from the original 10% (and 8% for the MDC) to 6%, and that this improvement did not require any class-specific fine-tuning of the decision criterion (as was the case in the MDC example).

(a) 256×256 (b) MRF (c) DRF (d) TSBN (e) DTSBN

Fig. 9. Two-class pixel labeling using different statistical models.

	O_1	O_2	O_3	O_4	O_5
O_1	**31**	**0**	**1**	**0**	**0**
O_2	**0**	**25**	**1**	**1**	**1**
O_3	**0**	**3**	**40**	**1**	**2**
O_4	**0**	**0**	**0**	**16**	**1**
O_5	**1**	**0**	**1**	**1**	**36**

(a)

	O_1	O_2	O_3	O_4	O_5
O_1	**31**	**0**	**0**	**0**	**0**
O_2	**0**	**26**	**1**	**1**	**1**
O_3	**0**	**1**	**41**	**1**	**1**
O_4	**0**	**0**	**0**	**17**	**1**
O_5	**1**	**1**	**1**	**0**	**37**

(b)

Fig. 11. Confusion matrices for DTSBNs over 50 test images containing a total of 32 O_1, 28 O_2, 43 O_3, 19 O_4, and 40 O_5 partially occluded objects, using: (a) the modified classification criterion (MDC); (b) two-stage strategy, based on object parts; columns indicate ground truth.

5. Conclusion

In this paper, we advocate generative, dynamic-structure models for object recognition in images where objects may be partially occluded. We first defined Dynamic Tree Structure Belief Networks (DTSBNs) as the underlying framework, and developed an inference algorithm for these models that (1) relaxes poorly justified independence assumptions, and (2) is shown to converge an order of magnitude faster than competing algorithms. Next, we demonstrated the capability of DTSBNs to capture important component-subcomponent structures in unsupervised image segmentation. Finally, we reported supervised image classification experiments for partially occluded appearances that (1) demonstrate better performance of the proposed generative framework (i.e. DTSBNs) compared with other statistical models; and (2) suggest novel multi-stage recognition strategies based on classification of object parts.

The results presented in this paper raise a number of interesting points for further research. For instance, the two-stage strategy outlined above is certainly not the only one possible (let alone optimal) within the DTSBN framework; for example, the treatment of object parts need not be confined to subtrees one level below root nodes. No matter what parts-based strategy is implemented, however, it will still require successful identification of object parts through DTSBN learning.

References

[1] B. J. Frey, N. Jojic, and A. Kannan, "Learning appearance and transparency manifolds of occluded objects in layers," in *Proc. 2003 IEEE Computer Soc. Conf. Computer Vision Pattern Rec.*, vol. 1, 2003, pp. 45–52.

[2] Z. Ying and D. Castanon, "Partially occluded object recognition using statistical models," *Int'l J. Computer Vision*, vol. 49, no. 1, pp. 57–78, 2002.

[3] G. J. III and B. Bhanu, "Recognition of articulated and occluded objects," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 7, pp. 603–613, 1999.

[4] N. J. Adams, A. J. Storkey, Z. Ghahramani, and C. K. I. Williams, "MFDTs: Mean field dynamic trees," in *Proc. 15th Int'l Conf. Pattern Rec.*, vol. 3, 2000, pp. 147–150.

[5] A. J. Storkey and C. K. I. Williams, "Image modeling with position-encoding dynamic trees," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 7, pp. 859–871, 2003.

[6] C. A. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Trans. Image Processing*, vol. 3, no. 2, pp. 162–177, 1994.

[7] W. W. Irving, P. W. Fieguth, and A. S. Willsky, "An overlapping tree approach to multiscale stochastic modeling and estimation," *IEEE Trans. Image Processing*, vol. 6, no. 11, pp. 1517–1529, 1997.

[8] C. Spence, L. Parra, and P. Sajda, "Hierarchical image probability (HIP) models," in *Proc. 2000 Int'l Conf. Image Processing*, vol. 3, 2000, pp. 320–323.

[9] X. Feng, C. K. I. Williams, and S. N. Felderhof, "Combining belief networks and neural networks for scene segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 4, pp. 467–483, 2002.

[10] S. C. Zhu, "Statistical modeling and conceptualization of visual patterns," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 6, pp. 691–712, 2003.

[11] S. Z. Li, *Markov Random Field modeling in image analysis*. Tokyo: Springer-Verlag, 2001.

[12] S. Kumar and M. Hebert, "Discriminative random fields: A discriminative framework for contextual interaction in classification," in *Proc. IEEE Int'l Conf. Comp. Vision*, vol. 2, 2003, pp. 1150–1157.

[13] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999.

[14] D. J. C. MacKay, "Introduction to Monte Carlo methods," in *Learning in Graphical Models (Adaptive Computation and Machine Learning)*, M. I. Jordan, Ed. Cambridge, MA: MIT press, 1999, pp. 175–204.

[15] B. J. Frey and N. Jojic, "Advances in algorithms for inference and learning in complex probability models for vision," *IEEE Trans. Pattern Anal. Machine Intell.*, (to appear) 2004.

[16] J. Pearl, *Probabilistic reasoning in intelligent systems : networks of plausible inference*. San Mateo: Morgan Kaufamnn, 1988, ch. 4, pp. 143–236.

[17] R. M. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed. Kluwer Academic Publishers, 1998, pp. 355–368.

[18] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. Seventh IEEE Int'l Conf. Computer Vision*, vol. 2, 1999, pp. 1150–1157.

[19] Z. Ghahramani, "An introduction to hidden Markov models and

Bayesian networks," *Int'l J. Pattern Rec. Artificial Intell.*, vol. 15, no. 1, pp. 9–42, 2001.

[20] S. Todorovic and M. C. Nechyba, "Towards intellignet mission profiles of Micro Air Vehicles: multiscale Viterbi classification," (accepted to) 8th European Conf. Computer Vision, 2004.