Interpretation of Complex Scenes Using Dynamic Tree-Structure Bayesian Networks ⋆

Sinisa Todorovic [∗,1]

Computer Vision and Robotics Laboratory
Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign, Urbana, IL 61801, U.S.A.

Michael C. Nechyba [1]

Pittsburgh Pattern Recognition, Inc.
40 24th Street, Suite 240, Pittsburgh, PA 15222, U.S.A.

Abstract

This paper addresses the problem of object detection and recognition in complex scenes, where objects are partially occluded. The approach presented herein is based on the hypothesis that a careful analysis of visible object details at various scales is critical for recognition in such settings. In general, however, computational complexity becomes prohibitive when trying to analyze multiple sub-parts of multiple objects in an image. To alleviate this problem, we propose a generative-model framework – namely, dynamic tree-structure belief networks (DTSBNs). This framework formulates object detection and recognition as inference of DTSBN structure and image-class conditional distributions, given an image. The causal (Markovian) dependencies in DTSBNs allow for design of computationally efficient inference, as well as for interpretation of the estimated structure as follows: each root represents a whole distinct object, while children nodes down the sub-tree represent parts of that object at various scales. 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. This property leads to a strategy for recognition of objects as a whole through recognition of their visible parts. Our experimental results demonstrate that this approach remarkably outperforms strategies without explicit analysis of object parts.

Key words: generative models, Bayesian networks, dynamic trees, variational inference, image segmentation, object recognition

1 Introduction

This paper addresses the problem of object detection and recognition in complex scenes, where objects are partially occluded. A number of factors contribute to the difficulty of this problem including variations in camera quality and position, wide-ranging illumination conditions, and extreme scene diversity with partial occlusions [1–5]. A review of the literature offers various approaches that usually address only a subset of the outlined problems. For instance, the majority of research efforts is focused exclusively on either image segmentation (i.e., detection) [6–8], or image classification (i.e., recognition) [2,4,5] of scenes with occlusions. Moreover, related work (e.g., [7–9]), usually considers auxiliary information provided, for example, by image sequences or stereo views of the same scene.

In contrast, we seek a framework that is sufficiently expressive to cope with uncertainty in images, jointly addresses object detection and recognition in a unified manner, and represents a viable solution for scenes with occlusions. To this end, the probabilistic framework proposed formulates the object recognition problem as inference of structure and conditional distributions of the generative statistical model — more specifically, dynamic tree-structure belief networks (DTSBNs) — representing a given image.

DTSBNs are directed acyclic graphs, where edges indicate statistical Markov dependencies between nodes, which in turn represent hidden and observable random variables, as illustrated in Fig. 1a. As with other dynamic trees, the DTSBN is characterized by a joint distribution over image-class labels (associated with each node) and the structure of the network [10–13]. Consequently, in inference, in addition to finding posteriors of image-class labels, the network structure is optimized for the given image. The main differences between the DTSBN and the model investigated in our prior work [13], referred to as dynamic tree (DT) and depicted in Fig. 1b, concern the treatment of observable random variables and the inference algorithm. In DTs [13], observables exist at all hierarchical levels of the model; these observables can change along with the iterative modifications of the model's structure in inference. Such model design was found suitable to address the challenges of *unsupervised* image segmentation and matching, as reported in [13]. In contrast, for

[⋆] This work was supported in part by grants from NASA Langley, the Air Force Office of Sponsored Research, the U.S. Air Force and U.S. Special Operations Command.

[*] corresponding author
Email addresses: sintod@uiuc.edu (Sinisa Todorovic),
michael@pittpatt.com (Michael C. Nechyba).
URLs: http://vision.ai.uiuc.edu/∼sintod (Sinisa Todorovic),
http://www.pittpatt.com (Michael C. Nechyba).

[1] The work reported in this paper was conducted while the authors were affiliated with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, U.S.A.

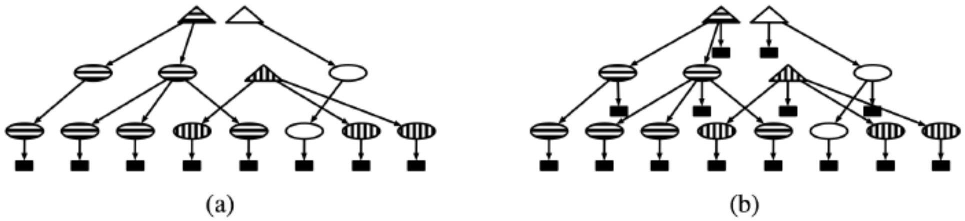


Fig. 1. (a) DTSBN; (b) DT as in [13]. The DTSBN consists of a forest of subtrees, each of which segments the 1-D signal into regions marked by distinct shading; round- and square-shaped nodes indicate hidden and observable variables; triangles indicate roots.

the *supervised* setting investigated herein, the DTSBN has a fixed set of single-layer observables extracted only once by the feature extraction module, which precedes the classifier in the proposed object recognition system. That is, unlike in the DT, observables in the DTSBN occupy its lowest level only. More importantly, for learning model parameters on training images, in general, it is assumed that the underlying image processes are stationary. Consequently, in supervised settings, observables should not be allowed to change along with dynamic changes of model structure. As such, the DTSBN can be viewed as a special case of the DT that is more appropriate for supervised settings. In addition, the proposed architecture allows for fair comparisons of DTSBNs with representatives of discriminative and descriptive models, as detailed later in this section.

Inference of DTSBNs is handled as a special case of the inference algorithm proposed in our prior work [13], which relaxes the assumptions related to the variational approximation of Storkey and Williams [12]. After inference, the DTSBN represents a forest of sub-trees, each of which segments the image. More precisely, leaf nodes that are descendants down the subtree of a given root form the image region characterized by that root, as depicted in Fig. 1. These segmented image regions can be interpreted as distinct object appearances in the image. That is, inference of DTSBN structure provides a solution to object detection. Then, for recognition of detected objects (i.e., segmented image regions), one possible approach is to label leaf nodes as one of M classes, by using the MAP classifier. Finally, a majority vote over each segmented region can be used to decide on the class of the object as a whole. Below, this approach is referred to as the *whole-object recognition* strategy.

When objects are partially occluded, however, such an approach may yield poor results, as demonstrated in the experiments reported in Section 5. Therefore, we propose a different strategy, where recognition is conducted in two stages through interpretation of *object parts*. More specifically, this strategy first seeks to identify visible object details, and then, by using this result, ultimately recognizes the object as a whole. Below, this strategy is referred to as *object-part recognition*.

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. In addition to the percentage of occlusion, which object parts are occluded is also critical for recognition. Not all components of an object are equally important for its recognition, especially when that object is partially occluded. Given two similar objects in the image, the visible parts of one object may mislead the algorithm to recognize it as its counterpart. Therefore, careful consideration should be given to the analysis of detected visible object components. The main advantage of such analysis is its flexibility to develop various recognition strategies that weigh the information obtained from the detected object parts more judiciously.

Many existing approaches addressing occlusions, however, lack explicit representation of object components at multiple scales [2, 4, 5]. A major obstacle in the treatment of image sub-classes is prohibitive computational complexity, which arises when the initial given set of classes (i.e., objects) is augmented with new classes of object parts. The problem could be alleviated by using greedy algorithms (e.g., [14]), which result, however, in suboptimal solutions. In contrast, by utilizing the generative property of DTSBNs, physical meaning can be assigned to DTSBN nodes such that they represent object parts at various scales. Therefore, within the DTSBN framework, the explicit treatment and recognition of object parts represents merely a particular interpretation of the tree/subtree structure.

To fully specify the object-part recognition strategy, it is necessary to define a criterion, which balances the complexity of interpreting all detected object sub-parts (i.e., DTSBN nodes) versus the reduced accuracy when analyzing only a subset of nodes. The considerations of such a criterion lie beyond the scope of this paper. Even a simple two-stage procedure, however, shows remarkable improvements over the whole-object recognition approach. After inference of DTSBN structure for a given image, this procedure first selects the set of immediate children under each root. These selected nodes are then treated as new roots of subtrees, which, in turn, segment the image into smaller image regions, that is, object parts. MAP classification and majority voting follow for selected image regions, thereby identifying object parts. Finally, in the second stage of our recognition strategy, these results are fused by yet another majority vote over the labels of those object parts that descend from a unique root. The block-diagram of the object-part recognition strategy is shown in Fig. 2.

The set of experiments in this paper show that scenes with partially occluded objects require a careful interpretation of visible object details. In exploiting the DTSBN's capability to explicitly represent object parts at multiple scales, significantly better recognition performance is achieved when compared with strategies where object components are not explicitly analyzed. This suggests that such analysis should be an integral part of object recognition systems for scenes with partially occluded objects.

Ultimately, what allows us to overcome obstacles in analyzing scenes with occlu-

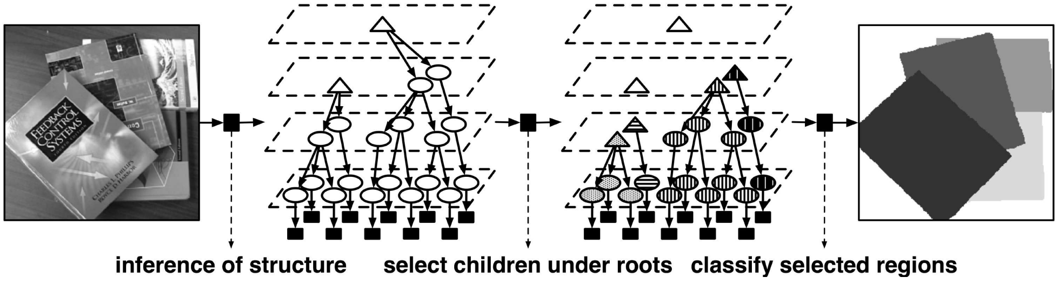


Fig. 2. Object-part recognition strategy: after inference of DTSBN structure, select the roots' children as new roots, and classify image regions underneath them; shading indicates four distinct subtrees under four selected children nodes; triangles indicate roots.

sions in a computationally efficient and intuitively appealing manner is the proposed generative-model framework. This framework provides an explicit representation of objects and their sub-parts at various scales, which, in turn, constitutes the key factor for improved interpretation of scenes with partially occluded objects, as demonstrated in Section 5. Thus, our choice of a generative model is directly driven by our image interpretation strategy and goals, and appears better suited than alternative statistical approaches, such as descriptive, or discriminative models [15, 16]. Descriptive models lack the necessary structure, 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.

This paper is organized as follows. Section 2 first defines DTSBNs while Section 3 discusses their probabilistic inference. Then, Section 4 explains how to learn the parameters of the joint prior distribution for DTSBNs. Next, Section 5 first reports experimental results on DTSBN-based unsupervised image segmentation, and then proceeds to results on supervised image classification for scenes with partially occluded objects. Performance of DTSBNs is also contrasted with Markov Random Fields (MRFs) [17], Discriminative Random Fields (DRFs) [18], and fixed-structure TSBNs [19]. This comparison demonstrates that DTSBNs, trained using SVA, outperform all these alternative modeling paradigms. Furthermore, in experiments with occlusions, recognition strategies conditioned on correct identification of object parts significantly improve overall recognition performance.

2 DTSBNs

DTSBNs can be viewed as generalized tree-structured belief networks (TSBNs), [2] which have been applied extensively in the image processing and computer vision

[2] In this paper, the terms quad-trees and TSBNs are used interchangeably to denote the same model; this convention departs somewhat from the literature, where the term TSBN has been associated with more general tree structures.

literature [19–25]. For example, TSBNs have been applied to multiscale document segmentation [21], simultaneous image denoising and segmentation [24], and medical applications [25]. There are several variants of very efficient inference algorithms for TSBNs [19, 22]; in this paper, our inference algorithm for DTSBNs is compared to the most prominent of these – namely, Pearl's λ-π message passing scheme, also known as belief propagation [26].

Despite the powerful expressiveness of TSBNs and the efficiency of their inference algorithms, TSBN-based segmentation/classification suffers from boundary artifacts. Due to the fixed structure of TSBNs, dependencies between TSBN nodes may be inadequately modeled, causing blocky discontinuities. In the literature, several approaches have been proposed to address this problem, including overlapping tree models with distinct nodes corresponding to overlapping parts in the image [27], random cascades on trees of multiresolution coefficients [28], and two-dimensional hierarchical models with nodes mutually dependent both at any particular layer through a Markov-mesh and across resolutions [29]. In these approaches, the descriptive component of the models is improved at some increased computational cost, leading to superior segmentation results when compared to standard TSBNs. However, these approaches do not alleviate the main cause of blocky discontinuities – that is, the fixed-tree structure of TSBNs. This problem is more explicitly addressed in research concerning dynamic/irregular tree structures. Thus, for example, Montanvert et al. [30] have explored irregular multiscale tessellation that adapts to image content. Also, Williams' group of researchers have introduced several variants of dynamic and position-encoding dynamic trees [10–12]. Finally, we have previously investigated dynamic trees in unsupervised settings, concluding that the model's random structure is critical to the superior segmentation performance of dynamic trees over TSBNs [13].

2.1 Definition of DTSBN

DTSBNs are most closely related to position-encoding dynamic trees [12], where observables are fixed and present only at the lowest model level. By contrast with DTSBNs, the dynamic trees in our previous work [13] comprise two disjoint sets of random variables, one of which represents multi-layered observable data at all model levels that change as a function of the dynamic model's structure. In our brief definition of DTSBNs below, differences between prior models and DTSBNs are highlighted where appropriate.

A DTSBN is a directed, acyclic graph with nodes in set V, organized in hierarchical levels, V^ℓ, $\ell{=}\{0, 1, ..., L{-}1\}$, where V^0 denotes the leaf level. The number of nodes is identical to that of the quad-tree, such that $|V^\ell|{=}|V^{\ell-1}|/4{=}...{=}|V^0|/4^\ell$. Each node is characterized by a set of random variables, the first of which are network connectivity indicators.

Any node at level ℓ can be a root, or connect only to the nodes at the next $\ell{+}1$ level. A node at level ℓ can have only one parent; however, all the nodes at $\ell{+}1$ are candidates to become the parent of that node. The event that two nodes i and j are connected is represented by an indicator random variable $z(ij)$. The set of $z(ij)$'s over all nodes forms a random matrix Z, which is further augmented with an additional zero ("root") column, where entries $z(i0)$ are equal to 1 if i is a root node. The distribution over connectivity is defined as

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

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

Next, the position of each node i, $\boldsymbol{r}_i$, is random and takes continuous values in the image plane. The distribution of $\boldsymbol{r}_i$ is conditioned on the position of its parent $\boldsymbol{r}_j$ using the normal distribution

$$P(\boldsymbol{r}_i|\boldsymbol{r}_j, z(ij){=}1) \triangleq \frac{\exp(-\frac{1}{2}(\boldsymbol{r}_i{-}\boldsymbol{r}_j)^T\Sigma_{ij}^{-1}(\boldsymbol{r}_i{-}\boldsymbol{r}_j))}{2\pi|\Sigma_{ij}|^{\frac{1}{2}}} \tag{2}$$

where Σ_{ij} is a diagonal matrix with elements $\sigma_{ij}^{(\mathrm{x})}$ and $\sigma_{ij}^{(\mathrm{y})}$, which represent the order of magnitude of object size along "x" and "y" image coordinates, respectively. In this fashion, the model explicitly expresses geometric component-subcomponent relationships through multiple scales in the image. The joint probability of $R \triangleq \{\boldsymbol{r}_i|\forall i{\in}V\}$, is given by

$$P(R|Z) \triangleq \prod_{i,j\in V} [\,P(\boldsymbol{r}_i|\boldsymbol{r}_j, z(ij))\,]^{z(ij)} \tag{3}$$

At the leaf level, V^0, node positions are fixed to pixel locations. Therefore, $P(Z, R'|R^0)$ is used as the prior over positions and connectivity, where $R^0 \triangleq \{\boldsymbol{r}_i|\forall i{\in}V^0\}$, and $R' \triangleq \{\boldsymbol{r}_i|\forall i{\in}V\backslash V^0\}$.

Further, each node i is associated with an image-class label x_i, and an image-class indicator random variable $x(ik)$, such that $x(ik){=}1$ if $x_i{=}k$, where $k{\in}M$, and M represents the set of image classes, which is assumed finite. The image class k of node i is conditioned on image class l of its parent j and is given by conditional probability tables P_{ij}^{kl}. Thus, the joint probability of $X{\triangleq}\{x(ik)|i{\in}V, k{\in}M\}$ is conditioned on network connectivity and given by

$$P(X|Z) \triangleq \prod_{i,j\in V}\prod_{k,l\in M} \left[P_{ij}^{kl}\right]^{x(ik)x(jl)z(ij)} \ . \tag{4}$$

Finally, leaf nodes are characterized by observable random vectors $\boldsymbol{y}_i$, where $Y \triangleq \{\boldsymbol{y}_i|\forall i{\in}V^0\}$. Observables $\boldsymbol{y}_i$ represent image-feature vectors comprising image texture and color cues in the neighborhood of node $i{\in}V^0$. The observables $\boldsymbol{y}_i$ are assumed to be conditionally independent given the corresponding $x(ik)$:

$$P(Y|X, R^0) \triangleq \prod_{i\in V^0}\prod_{k\in M} P(\boldsymbol{y}_i|x(ik))^{x(ik)}, \tag{5}$$

where $P(\boldsymbol{y}_i|x(ik){=}1)$ is a mixture of Gaussians:

$$P(\boldsymbol{y}_i|x(ik){=}1) \triangleq \textstyle\sum_{g=1}^{G_k} \pi_k(g) N(\boldsymbol{y}_i; \boldsymbol{\nu}_k(g), \Xi_k(g)) \ , \tag{6}$$

The Gaussian-mixture parameters can be grouped in the following set:

$$\theta \triangleq \{(G_k, \pi_k(g), \boldsymbol{\nu}_k(g), \Xi_k(g)) \mid \forall k \in M\} \ .$$

For large G_k, a Gaussian-mixture density can approximate any probability density [31].

The DTSBN can be viewed a special case of the dynamic tree investigated in our earlier paper [13], where observables at all levels depend on node positions as $P(Y|X, R^0, \boldsymbol{\rho}') = \prod_{i \in V} \prod_{k \in M} \left[P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k, \boldsymbol{\rho}(i))\right]^{x_i^k}$, and where $\boldsymbol{\rho}(i)$ is a suitably defined function of i's random position. Setting $\boldsymbol{\rho}(i) = i$ for all i's that belong to $\ell = 0$ level leads to the formulation for DTSBNs.

Speaking in generative terms, for a given set of V nodes, first $P(Z)$ is defined using Eq. (1) and $P(R|Z)$ using Eq. (3) to give us $P(Z, R)$. Leaf level node positions are then fixed to pixel locations to obtain $P(Z, R'|R^0)$. Combining Eq. (4) with $P(Z, R'|R^0)$ results in $P(Z, X, R'|R^0){=}P(X|Z)P(Z, R'|R^0)$. Finally, from Eq. (5), it follows that a DT is fully specified by the joint prior

$$P(Z, X, R', Y|R^0) = P(Y|X, R^0)P(X|Z)P(Z, R'|R^0) \ . \tag{7}$$

All the parameters of the joint prior can be grouped in the following set:

$$\Theta \triangleq \{\gamma(ij), \Sigma_{ij}, P_{ij}^{kl}, \theta\}, \ \forall i, j \in V, \ \forall k, l \in M.$$

3 Probabilistic Inference

One of the principal challenges in applying the DTSBN to image interpretation is the derivation of efficient algorithms for its inference, that is, for computing posterior probabilities of Z, X, and R' given Y and R^0. As for many complex-structure models, the exact inference for DTSBNs is intractable, which makes us consider inference approximation methods. In variational approximation, averaging phenomena in the model are exploited, such that a given set of variables is assumed approximately independent of the rest of the network. The idea is to approximate the true intractable distribution $P(Z, X, R'|Y, R^0)$, by a simpler distribution $Q(Z, X, R'|Y, R^0)$. In our discussion below, the conditioning on Y and R^0 is omitted to simplify notation. The approaches proposed in prior work range from a factorized approximating distribution over hidden variables $Q(Z, X) = Q(Z)Q(X)$ – the method known as mean field variational approximation (MFVA) [10] – to more structured solutions $Q(Z, X, R') = Q(Z)Q(X|Z)Q(R')$, where dependencies among hidden variables are enforced [12].

In variational approximation, the goal is to find $Q(Z, X, R')$ closest to $P(Z, X, R'|Y, R^0)$. This is achieved by minimizing *free energy* [32], $J(Q, P)$, specified as

$$\begin{aligned} J(Q, P) &\triangleq KL(Q\|P) - \log P(Y|R^0) - \log P(R^0), \\ &= \int_{R'} dR' \sum_{Z,X} Q(Z, X, R') \log \frac{Q(Z, X, R')}{P(Z, X, R, Y)}, \end{aligned} \tag{8}$$

where $KL(Q\|P)$ denotes Kullback-Leibler (KL) divergence between $Q(Z, X, R')$ and $P(Z, X, R'|Y, R^0)$ [33], defined as

$$KL\ (Q\|P) \triangleq \int_{R'} dR' \sum_{Z,X} Q(Z, X, R') \log \frac{Q(Z, X, R')}{P(Z, X, R'|Y, R^0)}. \tag{9}$$

From Eq. (8), it follows that minimizing $J(Q, P)$ amounts to minimizing KL divergence. As a direct corollary of Jensen's inequality [33], $KL(Q\|P)$ is non-negative for any two distributions Q and P, and $KL(Q\|P){=}0$ if and only if $Q{=}P$. Consequently, minimizing free energy $J(Q\|P)$ with respect to $Q(Z, X, R')$ guarantees a unique global solution to $Q(Z, X, R')$.

In this paper, inference of DTSBNs is carried out through our structured variational approximation (SVA) algorithm [13], in which the approximating variational distribution is specified as

$$Q(Z, X, R') \triangleq Q(Z)Q(X|Z)Q(R'|Z)\ . \tag{10}$$

This formulation enforces that both state-indicator variables X and position variables R' be statistically dependent on the tree connectivity Z, unlike the Q function proposed by Storkey and Williams [12] that has a simpler form $Q(Z, X, R') = Q(Z)Q(X|Z)Q(R')$. Moreover, the approximating distributions are defined as

$$\begin{aligned} Q(Z) &\triangleq \textstyle\prod_{i,j\in V} \left[\xi(ij)\right]^{z(ij)}, & (11)\\ Q(X|Z) &\triangleq \textstyle\prod_{i,j\in V} \prod_{k,l\in M} \left[Q_{ij}^{kl}\right]^{x(ik)x(jl)z(ij)}, & (12)\\ Q(R'|Z) &\triangleq \textstyle\prod_{i,j\in V'} \left[Q(\boldsymbol{r}_i|z(ij))\right]^{z(ij)}, & (13)\\ Q(\boldsymbol{r}_i|z(ij){=}1) &\triangleq \frac{1}{2\pi|\Omega_{ij}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})^T\Omega_{ij}^{-1}(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})\right), & (14) \end{aligned}$$

where $\xi(ij)$ corresponds to $\gamma(ij)$, Q_{ij}^{kl} is analogous to P_{ij}^{kl}, and $V'{\triangleq}V\backslash V^0$. Note that covariances Ω_{ij} and mean values $\boldsymbol{\mu}_{ij}$ form the set of Gaussian parameters for a given node i over its candidate parents $j{\in}V$. Thus, the sampling of $\boldsymbol{r}_i$ is determined by the pair $(\boldsymbol{\mu}_{ij}, \Omega_{ij})$, provided there is a connection between i and j. In our implementation, the Ω's are diagonal, since the the node positions along the "x" and "y" image axes are assumed uncorrelated. The diagonal elements of Ω_{ij} are denoted as $\omega_{ij}^{(\mathrm{x})}$ and $\omega_{ij}^{(\mathrm{y})}$.

From the derivation steps reported in [13], it is straightforward to obtain the inference equations for DTSBNs, by taking into account the constraint that observables in the DTSBN cannot be modified in inference, and that they exist only at the $\ell = 0$ level. For completeness, Fig. 3 summarizes the final derivation results, where the sign $\propto$ is used to denote that the right-hand side should be normalized, such that the term on the left-hand side represents a probability. Note that the upward-downward propagation, specified by Steps (2.2) and (2.3) in Fig. 3, is similar to belief propagation for TSBNs [19,22,26]. In the special case, when $\xi(ij)=1$ only for one parent j, that is, the set of candidate parents is reduced to only one node, the algorithm reduces to the standard λ-π rules of Pearl's message passing scheme for TSBNs. Also, in Step (2.6) in Fig. 3, A_{ij} represents the influence of observables Y on the connectivity distribution, and B_{ij} represents the contribution of the geometric properties of the network to the connectivity distribution.

In [13], convergence of SVA was compared with the following inference algorithms: Gibbs sampling [34], mean-field variational approximation proposed in [10], and structured variational approximation discussed in [12]. The reported empirical results demonstrate that Gibbs sampling becomes unfeasible as image size grows, and that the mean-field variational approximation exhibits very poor performance. In summary, SVA converges to the largest likelihoods, in the fewest number of iterations, an order of magnitude faster than the second-place structured variational approximation proposed by Storkey and Williams [12].

3.1 Inference Algorithm

For the given set of parameters Θ that fully specify the joint prior of the DTSBN, the Bayesian formulation of the inference problem amounts to minimizing the expectation of a cost function $\mathcal{C}$:

$$(\hat{Z}, \hat{X}, \hat{R}') = \arg\min_{Z,X,R'} \mathbb{E}\{\mathcal{C}((Z,X,R'),(Z^*,X^*,R'^*))|Y,R^0,\Theta\}, \tag{15}$$

where $\mathcal{C}(\cdot)$ penalizes the difference between the estimated, (Z,X,R'), and the true configuration (Z^*,X^*,R'^*). As in [13], the following cost function is used:

$$\begin{aligned}\mathcal{C}((Z,X,R'),(Z^*,X^*,R'^*)) \triangleq\ & \textstyle\sum_{i,j\in V}[1-\delta(z(ij)-z^*(ij))] \\ &+ \textstyle\sum_{i\in V}\sum_{k\in M}[1-\delta(x(ik)-x^*(ik))] \\ &+ \textstyle\sum_{i\in V'}[1-\delta(\boldsymbol{r}_i-\boldsymbol{r}_i^*)]\ ,\end{aligned} \tag{16}$$

where * stands for true values, and $\delta(\cdot)$ is the Kronecker delta function. By using the variational distribution instead of true posterior, $P(Z, X, R'|Y, R^0) \approx Q(Z)Q(X|Z)Q(R|Z)$, it follows from Eqs. (15) and (16) that:

$$\hat{Z} = \arg\min_Z \sum_Z Q(Z) \sum_{i,j\in V}[1-\delta(z(ij)-z^*(ij))], \tag{17}$$

$$\hat{X} = \arg\min_X \sum_{Z,X} Q(X,Z) \sum_{i\in V} \sum_{k\in M}[1-\delta(x(ik)-x^*(ik))], \tag{18}$$

$$\hat{R}' = \arg\min_{R'} \int_{R'} dR' \sum_Z Q(R,Z) \sum_{i\in V'}[1-\delta(\boldsymbol{r}_i-\boldsymbol{r}_i^*)]. \tag{19}$$

Furthermore, the minimization in Eq. (17) is equivalent to finding parents:

$$(\forall\ell)(\forall i{\in}V^\ell)(Z_i{\neq}0)\ \ \hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}} \xi(ij), \tag{20}$$

where $\xi(ij)$ is computed as in Step (2.6) in Fig. 3. Here, Z_i denotes i-th column of Z, and $Z_i{\neq}0$ indicates that there is at least one non-zero element in column Z_i (i.e., i has children). The global solution to Eq. (20) is intractable leading us to resort to a stage-wise optimization in which the consecutive selection of parents is made in a bottom-up pass. Thus, starting from the leaf level $\ell{=}\{0, 1, ..., L{-}1\}$, optimal parents at $V^{\ell+1}$ are selected as:

$$(\forall i{\in}V^\ell)(\hat{Z}_i{\neq}0)\ \ \hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}} \xi(ij), \tag{21}$$

where $\hat{Z}_i$ denotes i-th column of estimated $\hat{Z}$, and $\hat{Z}_i{\neq}0$ indicates that i has already been selected as the optimal parent when optimizing the previous level V^ℓ.

Next, from Eq. (18), the Bayesian estimation of image-class labels reads

$$(\forall i{\in}V)\ \hat{x}_i = \arg\max_{k\in M} \sum_Z Q(x(ik){=}1|Z)Q(Z) = \arg\max_{k\in M} m_i^k\ . \tag{22}$$

where the approximate posterior probability m_i^k that node i is assigned to image class k is computed as in Step (2.3) in Fig. 3.

Finally, from Eq. (19), the Bayesian estimation of node positions is conducted as

$$(\forall\ell{>}0)(\forall i{\in}V^\ell)\ \hat{\boldsymbol{r}}_i = \arg\max_{\boldsymbol{r}_i} \sum_Z Q(\boldsymbol{r}_i|Z)Q(Z) = \sum_{j\in V^{\ell+1}} \boldsymbol{\mu}_{ij}\xi(ij), \tag{23}$$

where $\boldsymbol{\mu}_{ij}$ is computed as in Step (2.4.2) in Fig. 3.

Our inference algorithm for DTSBNs is summarized in Fig. 3.

Figure 3: SVA Inference Algorithm for DTSBNs

(1) Initialization: Assume that V, L, M, Θ, N_ε, ε, and ε_μ are given. Initialize $t{=}0$, $\forall i, j{\in}V$, $\forall k, l{\in}M$, $\xi(ij; t{=}0){=}\gamma(ij)$, $Q_{ij}^{kl}(t{=}0){=}P_{ij}^{kl}$. Set $\forall i, j{\in}V$, $\boldsymbol{\mu}_{ij}(t{=}0)$ to node locations in the corresponding quad-tree. Set diagonal elements of $\boldsymbol{\Omega}_{ij}(t{=}0)$ to be equal to the area of corresponding dyadic squares in the quad-tree.

```
(2) REPEAT Outer Loop
  (2.1) $t = t + 1$;
  (2.2) Compute in bottom-up pass for $\ell{=}0, 1, ..., L{-}1$, $\forall i, j{\in}V^{\ell}$, $\forall k, l{\in}M$

        $$\lambda_i^k(t){=}\begin{cases} P(\boldsymbol{y}_i|x(ik)) & , \quad i \in V^0, \\ \prod_{c\in V} \left[\sum_{a\in M} P_{ci}^{ak}\lambda_{ci}^{ak}(t)\right]^{\xi(ci;t-1)} & , \quad i \in V', \end{cases}$$

        and

        $$Q_{ij}^{kl}(t) \propto P_{ij}^{kl}\lambda_i^k(t),$$

  (2.3) Compute in top-down pass the approximate posterior probability $m_i^k$ that
        node $i$ is labeled as image class $k$, given $Y$ and $R^0$, for $\ell{=}L{-}1, L{-}2, ..., 0$,
        $\forall i{\in}V^{\ell}$, $\forall k{\in}M$,

        $$m_i^k(t) = \sum_{j\in V} \xi(ij; t-1) \sum_{l\in M} Q_{ij}^{kl}(t) m_j^l(t),$$

  (2.4) REPEAT Inner Loop
    (2.4.1) $t_{\text{in}} = t_{\text{in}} + 1$;
    (2.4.2) Compute $\forall i, j{\in}V'$,

        $$\boldsymbol{\mu}_{ij}(t_{\text{in}}){=}\left[\sum_{p\in V'} \xi(jp;t{-}1)\Sigma_{ij}^{-1}{+}\sum_{c\in V'}\xi(ci;t{-}1)\Sigma_{ci}^{-1}\right]^{-1}$$
        $$\cdot\left[\sum_{p\in V'}\xi(jp;t{-}1)\Sigma_{ij}^{-1}\boldsymbol{\mu}_{jp}(t_{\text{in}}{-}1){+}\sum_{c\in V'}\xi(ci;t{-}1)\Sigma_{ci}^{-1}\boldsymbol{\mu}_{ci}(t_{\text{in}}{-}1)\right],$$
        $$\frac{1}{\omega_{ij}^{(\text{x})}(t_{\text{in}})}{=}\frac{1}{\sigma_{ij}^{(\text{x})}}\left(1{+}\sum_{p\in V'}\xi(jp;t{-}1)\left[\frac{\text{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}(t_{\text{in}}{-}1)\}}{\text{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}(t_{\text{in}}{-}1)\}}\right]^{\frac{1}{2}}\right)$$
        $$+\sum_{c\in V'}\xi(ci;t{-}1)\frac{1}{\sigma_{ci}^{(\text{x})}}\left(1{+}\left[\frac{\text{Tr}\{\Sigma_{ci}^{-1}\Omega_{ci}(t_{\text{in}}{-}1)\}}{\text{Tr}\{\Sigma_{ci}^{-1}\Omega_{ij}(t_{\text{in}}{-}1)\}}\right]^{\frac{1}{2}}\right),$$

        where $c$ and $p$ denote children and grandparents of node $i$, respectively. Similarly, compute $\forall i, j{\in}V'$, $\omega_{ij}^{(\text{y})}(t_{\text{in}})$.
  (2.5) UNTIL $|\boldsymbol{\mu}_{ij}(t_{\text{in}}){-}\boldsymbol{\mu}_{ij}(t_{\text{in}}{-}1)|/\boldsymbol{\mu}_{ij}(t_{\text{in}}{-}1) < \varepsilon_{\mu}$;
  (2.6) For $\Omega_{ij} = \Omega_{ij}(t_{\text{in}})$, $\boldsymbol{\mu}_{ij} = \boldsymbol{\mu}_{ij}(t_{\text{in}})$, and $\mathcal{M}_{ijp} = (\boldsymbol{\mu}_{ij}{-}\boldsymbol{\mu}_{jp})(\boldsymbol{\mu}_{ij}{-}\boldsymbol{\mu}_{jp})^T$
        compute $\forall i, j{\in}V'$,

        $$\xi(ij)(t) \propto \gamma(ij)\ \exp\left(A_{ij}(t) - B_{ij}(t{-}1)\right),$$

        where

        $$A_{ij}(t) = \sum_{k,l\in M} Q_{ij}^{kl}(t) m_j^l(t) \log(\sum_{a\in M} P_{ij}^{al}\lambda_i^a(t)),$$
```

$$
\begin{aligned}
B_{ij}(t{-}1) = {} & \frac{1}{2}\log\frac{|\Sigma_{ij}|}{|\Omega_{ij}|} + \frac{1}{2}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\} \\
& + \sum_{p\in V'} \xi(jp;t{-}1)\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}\}^{\frac{1}{2}} \\
& + \frac{1}{2}\sum_{p\in V'} \xi(jp;t{-}1)\mathrm{Tr}\{\Sigma_{ij}^{-1}(\Omega_{jp}{+}\mathcal{M}_{ijp})\} \\
& + \frac{1}{2}\sum_{c\in V'} \xi(ci;t{-}1)\mathrm{Tr}\{\Sigma_{ci}^{-1}(\Omega_{ij}{+}\mathcal{M}_{cij})\} \\
& + \sum_{c\in V'} \xi(ci;t{-}1)\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ci}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ij}\}^{\frac{1}{2}} \ ,
\end{aligned}
$$

```
      where $c$ and $p$ denote children and grandparents of node $i$, respectively.
(3) UNTIL $|Q(Z,X,R';t){-}Q(Z,X,R';t{-}1)|/Q(Z,X,R';t{-}1){<}\varepsilon$ for $N_\varepsilon$ consecutive iteration steps ;
(4) Compute in bottom-up pass for $\ell{=}0,1,...,L{-}1$
    $(\forall i{\in}V^\ell)(\hat{Z}_i{\neq}0)$ $\hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}} \xi(ij;t)$;
(5) Compute $(\forall i{\in}V)$ $\hat{x}_i = \arg\max_{k\in M} m_i^k(t)$;
(6) Compute $(\forall\ell{>}0)(\forall i{\in}V^\ell)$ $\hat{\boldsymbol{r}}_i = \sum_{j\in V^{\ell+1}} \boldsymbol{\mu}_{ij}(t)\xi(ij;t)$;
```

Fig. 3. The SVA inference algorithm [13] adapted to account for fixed single-layer observables in DTSBNs ; t and t_{in} are counters in the outer and inner loops, respectively; N_ε, ε, and ε_μ control the convergence criteria for the two loops.

3.2 *Implementation Issues*

As discussed in [13], for SVA inference, it is necessary to undertake additional computation steps to prevent numerical underflow. Here, the most problematic is computation of Q_{ij}^{kl} in Step (2.2) in Fig. 3. Fortunately, if the λ's are appropriately scaled, then the computation of Q_{ij}^{kl} does not change when the scaled $\tilde{\lambda}$ values are used. Thus, if the λ's are scaled as

$$\tilde{\lambda}_i^k \triangleq \frac{\lambda_i^k}{\sum_{a\in M}\lambda_i^a},\ \forall i{\in}\ V,\ \forall k{\in}M\ , \tag{24}$$

then, it follows that

$$Q_{ij}^{kl} = \frac{P_{ij}^{kl}\lambda_i^k}{\sum_{a\in M}P_{ij}^{al}\lambda_i^a} = \frac{P_{ij}^{kl}\tilde{\lambda}_i^k}{\sum_{a\in M}P_{ij}^{al}\tilde{\lambda}_i^a}. \tag{25}$$

Next, note that ε_μ controls the convergence criterion of the inner loop in Fig. 3, where $\boldsymbol{\mu}_{ij}$ and Ω_{ij} are computed. When $\varepsilon_\mu{=}0.01$, the average number of iteration steps, t_{in}, in the inner loop, ranges from 3 to 5 depending on the image size, where the latter corresponds to $256{\times}256$ images.

Finally, although SVA guarantees a global unique solution to $Q(Z, X, R')$, setting an inappropriate value of ε that controls the convergence criterion of the outer loop in Fig. 3 may lead to sub-optimal solutions. Therefore, the additional convergence-control parameter N_ε needs to be specified as well. In our experiments the two convergence parameters are set as N_ε=10 and ε=0.01.

4 Learning

In order to perform the SVA inference, it is first necessary to learn the parameters of the joint prior, Θ=$\{\gamma(ij), \Sigma_{ij}, P_{ij}^{kl}, \theta\}$, $\forall i, j{\in}V$, $\forall k, l{\in}M$, on a given set of training images. Below, we first explain how to compute $\gamma(ij)$ and θ, and then discuss learning Σ_{ij} and P_{ij}^{kl}.

The connectivity probabilities $\gamma(ij)$ are set to be uniform over i's candidate parents $\forall j{\in}\{0, V^{\ell+1}\}$, where $\gamma(i0)$ is the probability that i is a root. This allows DTSBNs to form arbitrary structures adapted to the given image in inference. In other words, the uniform γ's do not favor any particular component-subcomponent structure of objects in the image by DTSBNs. Next, the parameters of a Gaussian-mixture density θ can be learned by the EM algorithm on a given set of training images [31].

Parameters Σ_{ij} and P_{ij}^{kl}, on the other hand, require a more involved learning procedure, since they characterize nodes at higher levels, where the ground truth is not readily available. These parameters can be learned on training images, by using standard maximum likelihood (ML) optimization. Usually, in ML optimization, it is assumed that for N independently generated training images with observables $\{Y^n\}$, n=$1, ..., N$, corresponding configurations of latent variables – in our case $\{(Z^n, X^n, R'^n)\}$ – are given. However, for multiscale generative models, in general, neither the true image-class labels for nodes at higher levels nor their dynamic connections are given. Therefore, "true" configurations $\{(\hat{Z}^n, \hat{X}^n, \hat{R}'^n)\}$ must be estimated.

This is achieved through an iterative learning procedure, where in step t it is first assume that $\Theta(t)$=$\{\gamma(ij), \Sigma_{ij}(t), P_{ij}^{kl}(t), \theta\}$ is given,[3] and then conduct inference for each training image n=$1, ..., N$

$$(\hat{Z}^n, \hat{X}^n, \hat{R}'^n) = \arg\min_{Z,X,R'} \mathbb{E}\{\mathcal{C}((Z, X, R'), (Z^*, X^*, R'^*))|Y^n, R^0, \Theta(t)\}, \quad (26)$$

as explained in Section 3. Once the estimates $\{(\hat{Z}^n, \hat{X}^n, \hat{R}'^n)\}$, n=$1, ..., N$, are found, ML optimization is applied to compute $\Theta(t{+}1)$. Here, the iterations are run until $|P_{ij}^{kl}(t{+}1) - P_{ij}^{kl}(t)|/P_{ij}^{kl}(t) < 0.01$.

[3] Note that parameters $\gamma(ij)$ and θ are fixed to already learned values.

In particular, for the estimated $\hat{z}_{ij}^n$ and $\hat{\boldsymbol{r}}_i^n$ parameters of each training image n, the ML solution to Σ_{ij}, $\forall\ell$, $\forall(i,j) \in V^\ell \times V^{\ell+1}$, is given by

$$\widehat{\Sigma}_{ij} = \frac{1}{N} \sum_{n=1}^{N} \sum_{\left\{\substack{(i,j)\in V^\ell \times V^{\ell+1} \\ \hat{z}_{ij}^n = 1}\right\}} (\hat{\boldsymbol{r}}_i^n - \hat{\boldsymbol{r}}_j^n)(\hat{\boldsymbol{r}}_i^n - \hat{\boldsymbol{r}}_j^n)^T, \tag{27}$$

where the off-diagonal elements are set to zero, since Σ_{ij} is assumed to be a diagonal matrix. In order not to overfit the model, note that the $\widehat{\Sigma}_{ij}$ covariances are equal for all nodes i at the same level.

Further, to learn conditional probability tables P_{ij}^{kl}, the following variational log-likelihood is defined:

$$\begin{aligned} \mathscr{L}(Y|P_{ij}^{kl}) &\triangleq -\int_{R'} dR' \sum_{Z,X} Q(Z,X,R') \log \frac{Q(Z,X,R')}{P(Z,X,R,Y|P_{ij}^{kl})} , \\ &= -KL(Q\|P) + \log P(Y|R^0, P_{ij}^{kl}) + \log P(R^0) , \end{aligned} \tag{28}$$

where $KL(Q\|P)$ is given by Eq. (9). Since for any Q and P distributions $KL(Q\|P) \geq 0$, it follows that the log-likelihood $\log P(Y|R^0, P_{ij}^{kl})$ is lower bounded by $\mathscr{L}(Y|P_{ij}^{kl})$ minus the additive constant $\log P(R^0)$. Consequently, maximizing $\mathscr{L}(Y|P_{ij}^{kl})$ with respect to P_{ij}^{kl} increases the lower bound to $\log P(Y|R^0, P_{ij}^{kl})$. Thus, for a given set of training images $n = 1, ..., N$, optimal $\hat{P}_{ij}^{kl}$ can be computed as

$$\begin{aligned} &\hat{P}_{ij}^{kl} = \arg\max_{P_{ij}^{kl}} \textstyle\sum_{n=1}^{N} \mathscr{L}(Y^n|P_{ij}^{kl}) , \\ &\text{subject to} \quad \textstyle\sum_{k\in M} P_{ij}^{kl} = 1 . \end{aligned} \tag{29}$$

Substituting into Eq. (28) for all the terms, then finding $\partial\mathscr{L}(Y|P_{ij}^{kl})/\partial P_{ij}^{kl}$, and finally accounting for the Lagrange multiplier, yields the solution to Eq. 29, $\forall(i,j) \in V^\ell \times \{0, V^{\ell+1}\}$, $\forall k,l \in M$:

$$\hat{P}_{ij}^{kl} \propto \sum_{n=1}^{N} \sum_{(i,j)\in V^\ell \times \{0,V^{\ell+1}\}} \xi^n(ij) Q_{ij}^{kl;n} m_j^{l;n} , \tag{30}$$

where n in the superscript of $\xi(ij)$, Q_{ij}^{kl}, and m_j^l denotes that these variational parameters are optimized for the n-th image by using the inference algorithm in Fig. 3.

The learning of DTSBN parameters is summarized in Fig. 4.

Learning Algorithm

```
(1) $t=0$; initialize $\Theta(0)=\{\gamma(ij),\Sigma_{ij}(0),P_{ij}^{kl}(0),\theta\}$;
(2) REPEAT
    (2.1) $t=t+1$;
    (2.2) Estimate for $n=1,...,N$:
          $(\hat{Z}^n,\hat{X}^n,\hat{R}'^n)=\arg\min_{Z,X,R'}\mathbb{E}\{\mathcal{C}(\cdot)|Y^n,R^0,\Theta(t-1)\}$,
          by using the inference algorithm in Fig. 3;
    (2.3) Compute $\Sigma_{ij}(t)$ given by Eq. (27);
    (2.4) Compute $P_{ij}^{kl}(t)$ given by Eq. (30);
(3) UNTIL $|P_{ij}^{kl}(t)-P_{ij}^{kl}(t-1)|/P_{ij}^{kl}(t-1)<0.01$
```

Fig. 4. Algorithm for learning the DTSBN parameters.

5 Experiments and Discussion

This Section reports the performance of DTSBNs in supervised settings. Since image segmentation is an integral part of our object-recognition systems, we also briefly review the segmentation performance of DTSBNs, extensively discussed in [13].

Experiments are conducted on 360 color images of size 256×256, examples of which are shown in Figs. 5, 7, and 11. This dataset is the augmented version of Dataset IV in [13]. Images in the dataset contain partially occluded objects from a set of 23 classes, where 21 classes are items (e.g., toys, books, cans, etc.) that are similar in appearance, as depicted in Fig. 6, and the remaining 2 classes are two types of background. Here, the image classes are carefully selected to test if DTSBNs are expressive enough to capture very small variations in appearances of some classes (e.g., two different "Fluke" voltage-measuring instruments), challenging even for a human eye, as well as to encode large differences among some other classes (e.g., complexly shaped robots *vs.* books). Moreover, the dataset is carefully designed to contain complex scenes with occlusions, where the most "recognizable" parts of the objects in the scene are hidden. For instance, in Fig. 11, two "Fluke" voltage-measuring instruments, then, two blue books, and two cans, occlude each other, such that a careful analysis of their visible parts is required for successful recognition. Ground truth for each image is determined through hand-labeling of pixels. The dataset is divided into training and test sets by random selection of images, such that 2/3 are used for training (i.e., learning Θ parameters) and 1/3 for testing (i.e., image segmentation and classification).

In our experiments, observables Y include both color and texture cues. Texture is computed as the difference-of-Gaussian function convolved with the image:

$$D(x,y,k,\sigma)\triangleq(G(x,y,k\sigma)-G(x,y,\sigma))*I(x,y), \tag{31}$$

$$G(x, y, \sigma) \triangleq \exp(-(x^2 + y^2)/2\sigma^2)/2\pi\sigma^2, \tag{32}$$

where x and y represent pixel coordinates, and $I(x, y)$ is the intensity image. D provides a close approximation to the scale-normalized Laplacian of Gaussian, $\sigma^2\nabla^2 G$, which has been shown to produce the most stable image features across scales when compared to a range of other possible image functions, such as the gradient or the Hessian [35]. $D(x, y, k, \sigma)$ is computed for three variance scale factors $k{=}\sqrt{2}, 2, \sqrt{8}$ and $\sigma = 2$ pixels. Color is represented by the generalized RGB color space, $r{=}R/(R{+}G{+}B)$, and $g{=}G/(R{+}G{+}B)$, which effectively normalizes variations in brightness. Each r and g color value is normalized over the image to have zero mean and unit variance. Thus, the $\boldsymbol{y}_i$'s are 5-dimensional vectors.

5.1 Image Segmentation

The image-segmentation results presented in Fig. 5, as well as the results in [13], demonstrate that DTSBNs, inferred with SVA, are able to correctly assign one subtree per “object” in an image. Here, 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. Note from Fig. 5 that DTSBNs preserve tree structure for objects across images subject to translation, rotation and scaling. Moreover, note that the marked regions of pixels with the same parent at level 4 for the largest-object scale correspond to the regions of pixels with the same parent at level 3 for the medium-object scale; similarly, the level-4 clustering for the medium-object scale corresponds to the level-3 clustering for the smallest-object scale. In other words, as the object transitions through scales, the tree structure changes by eliminating the lowest-level layer, while the higher-level structure remains intact.

As discussed in [13], the estimated positions of roots in Fig. 5 are very close to the center of mass of whole objects. Moreover, the estimated positions of higher-level nodes (e.g., nodes at levels $\ell{=}3$ and $\ell{=}4$) are very close to the center of mass of object-parts they represent. This can be measured by computing the error of node positions $\boldsymbol{r} = [r^{(\mathrm{x})} \;\; r^{(\mathrm{y})}]$ as a distance from the actual center of mass (CM) of hand-labeled ”meaningful” object parts: $d_{err} = \sqrt{(r^{(\mathrm{x})}{-}r^{(\mathrm{CMx})})^2{+}(r^{(\mathrm{y})}{-}r^{(\mathrm{CMy})})^2}$. For the dataset used in this paper, the averaged error is $d_{err}{=}11.4$, which represents only 4% of the image size. Therefore, our claim that nodes at different levels of DTSBN structure represent object-parts at various scales is supported by experimental evidence that the nodes segment the image into “meaningful” object sub-components and position themselves at the center of mass of these sub-parts.

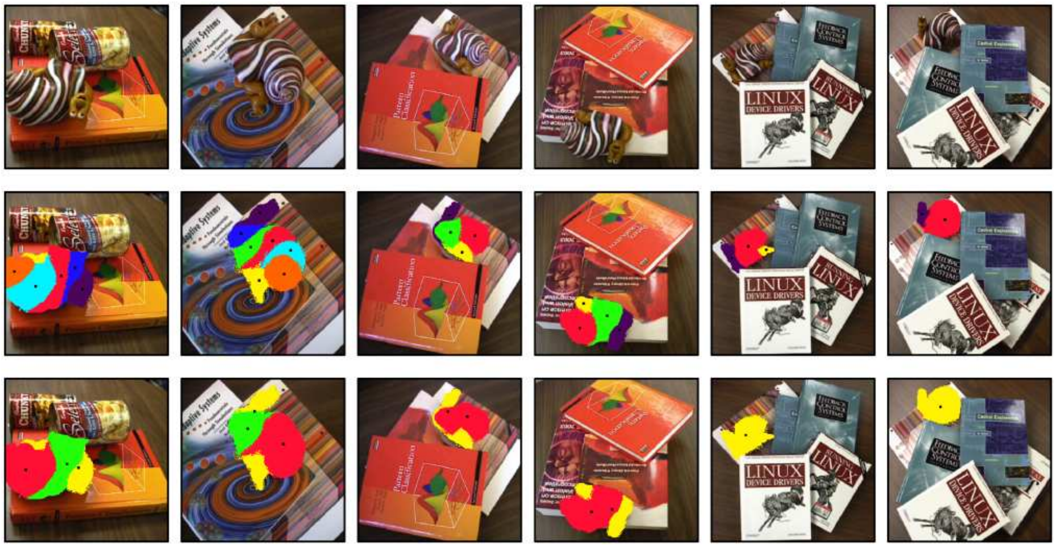


Fig. 5. DTSBN-based image segmentation: invariance across translation, rotation and scaling. (top row) 256×256 images; (middle row) pixel clusters with the same parent at level 3; (bottom row) pixel clusters with the same parent at level 4; points mark the position of parent nodes.

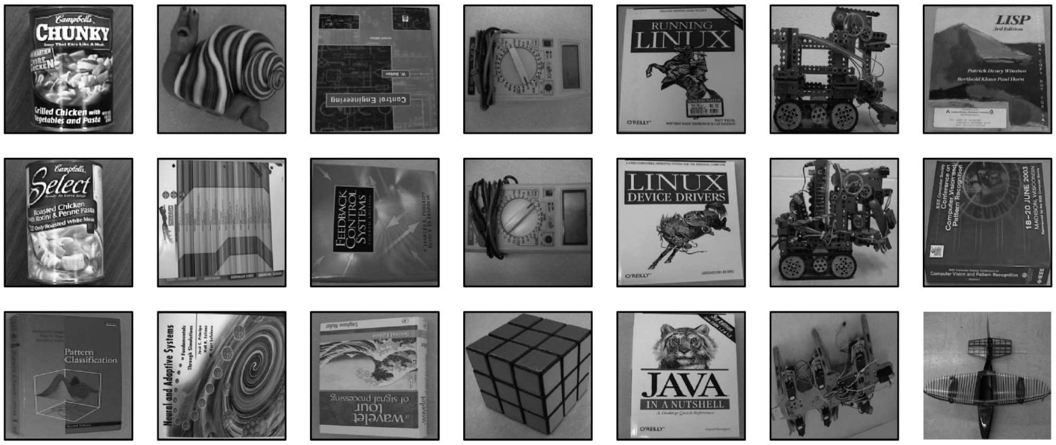


Fig. 6. 21 image classes in our dataset.

5.2 Image Classification

We first compare classification performance of DTSBNs, learned by SVA, with that of the following statistical models: (1) MRFs presented in [17], (2) DRFs proposed in [18], and (3) TSBNs discussed in [21, 22]. These models are representatives of descriptive, discriminative and fixed-structure generative models, respectively.

For MRFs, it is assumed that the label field $P(X)$ is a homogeneous and isotropic MRF, given by the generalized Ising model with only pairwise nonzero potentials [17]. The likelihoods $P(\boldsymbol{y}_i|x_i)$ are assumed conditionally independent given the labels. Thus, the posterior energy function is given by

$$U(X|Y) = \sum_{i \in V^0} \log P(\boldsymbol{y}_i|x_i) + \sum_{i \in V^0} \sum_{j \in \mathcal{N}_i} V_2(x_i, x_j), \tag{33}$$

$$V_2(x_i, x_j) = \begin{cases} \beta_{MRF} & , \text{ if } x_i = x_j \ , \\ -\beta_{MRF} & , \text{ if } x_i \neq x_j \ . \end{cases} \tag{34}$$

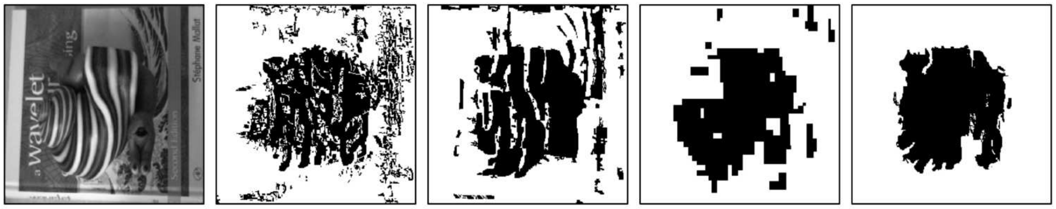
where $\mathcal{N}_i$ denotes the neighborhood of i, $P(\boldsymbol{y}_i|x_i)$ is a G-component mixture of Gaussians given by Eq. (5), and V_2 is the interaction parameter. Details on learning the model parameters as well as on inference for a given image can be found in [17]. Next, the posterior energy function of the DRF is given by

$$U(X|Y) = \sum_{i \in V^0} A_i(x_i, Y) + \sum_{i \in V^0} \sum_{j \in \mathcal{N}_i} I_{ij}(x_i, x_j, Y) \tag{35}$$

where $A_i = \log \sigma(x_i W^T \boldsymbol{y}_i)$ and $I_{ij} = \beta_{DRF}(K x_i x_j + (1-K)(2\sigma(x_i x_j V^T \boldsymbol{y}_i) - 1))$ 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 a given object, that object is treated as a positive example, while all other objects are treated as negative examples (“one against all” strategy). For details on how to learn the model parameters, and how to conduct inference for a given image, see [18]. Finally, TSBNs or quad-trees are defined to have the same number of nodes V and levels L as DTSBNs. In our experiments, learning of TSBN parameters and inference are performed with the algorithms discussed in depth in [22].

After inference of MRF, DRF, TSBN, and DTSBN on a given image, for each model, pixel labeling is conducted through MAP classification. Fig. 7 illustrates an example of pixel labeling for one image in our dataset. Since the ground truth for each test image is available, it is possible to estimate both pixel-labeling error and object-recognition error. Here, a hand-labeled image region is said to be correctly recognized as an object if the majority of MAP-classified pixel labels in the region are equal to the true labeling of that object. For estimating the object-recognition error, the following instances are counted as error: (1) merging two distinct objects into one, and (2) swapping the identity of objects. The object-recognition error over all objects in 120 test images is summarized in Fig. 8. The bars in Fig. 8 represent the overall recognition error, while the black portion of each bar indicates the ratio of swapped-identity errors. For instance, for DTSBNs the overall recognition error is 9.6%, of which 37% of instances were caused by swapped-identity errors. Fig. 9 shows average pixel-labeling error.

For the two-class recognition problem, ROC (*receiver operating characteristic*) curves are another method of visualizing performance. A typical two-class example is shown in Fig. 7, where pixels labeled as “toy-snail” are considered true positives, while pixels labeled as “book” are considered true negatives. Fig. 10 plots ROC curves for MRF, DRF, TSBN and DTSBN based decision boundaries. From Fig. 10, note that the DTSBN-based image classification is the most accurate, since its ROC curve is the closest to the left-hand and top borders of the ROC space, as compared



(a) 256x256 (b) MRF (c) DRF (d) TSBN (e) DTSBN

Fig. 7. MAP pixel labeling using different statistical models.

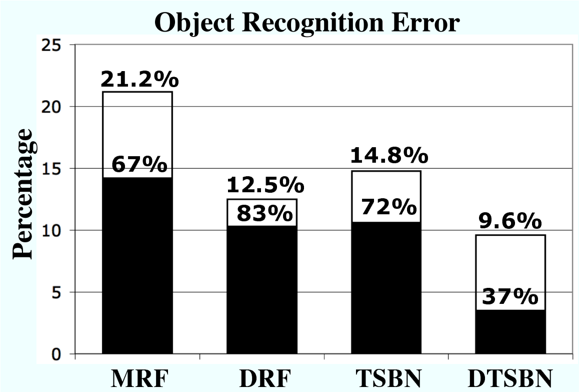


Fig. 8. Object recognition error.

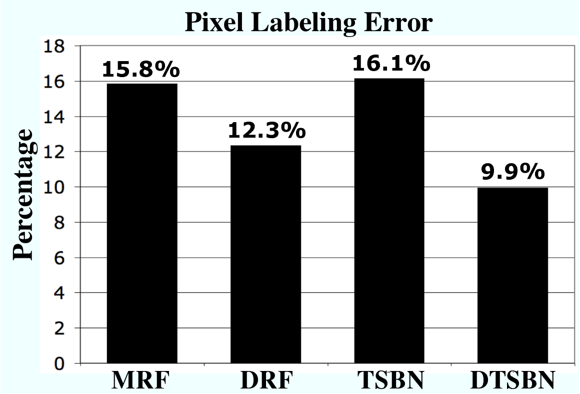


Fig. 9. Pixel labeling error.

to the ROC curves of the other models.

From the results reported in Figs. 8 and 9, as well as form Fig. 10, note that DTSBNs outperform the other three models. However, recognition performance of all the models suffers substantially when an image contains occlusions. While for some applications the literature reports vision systems with impressively small classification errors (e.g., 2.5% hand-written digit recognition error [36]), in the case of complex scenes this error is much higher [1–5]. To some extent, our results could have been improved with more discriminative image features and/or more sophisticated classification algorithms than majority rule. However, none of these will alleviate the fundamental problem of "traditional" recognition approaches: the

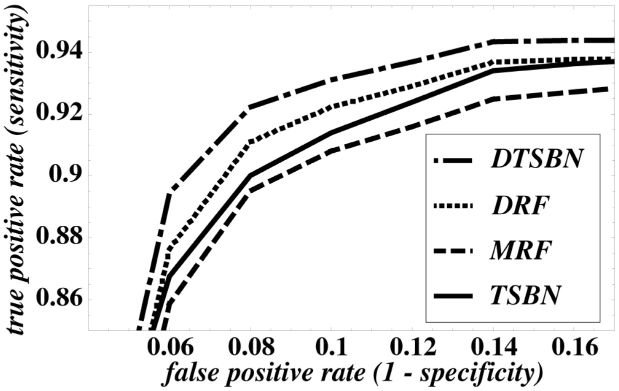


Fig. 10. ROC curves for the image in Fig. 7a with DTSBNs, TSBNs, DRFs and MRFs.

lack of explicit analysis of visible object parts. Thus, the poor classification performance of MRF, DRF, and TSBN, reported in Figs. 8 and 9, can be interpreted as follows. Accounting for only pairwise potentials between adjacent nodes in MRF and DRF is not sufficient to analyze complex configurations of objects in the scene. Also, the analysis of fixed-size pixel neighborhoods at various scales in TSBN leads to “blocky” estimates, and consequently to poor classification performance. Therefore, we hypothesize that the main reason why DTSBNs outperform the other models is their capability to represent object details at various scales, which in turn provides for explicit analysis of visible object parts. In other words, *recognition of object parts* is critical and should condition recognition of the object as a whole, in the face of the occlusion problem. Thus, instead of applying more sophisticated image-feature-extraction tools and better classification procedures than majority vote, a more radical change to our recognition strategy is introduced below.

5.3 Object-part Recognition Strategy

Recall from Section 5.1 that DTSBNs are capable of capturing 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. This convenient property of DTSBNs leads us to an object-part recognition strategy, where, in contrast to the whole-object recognition strategy, presented in the previous section, recognition is conducted in two stages. Thus, after inference of DTSBN structure for a given image, recognition now begins by treating children nodes of roots as new roots, each of which segments the image into smaller regions corresponding to object parts. Then, labels are assigned to all pixels that are descendants of these new roots, through MAP classification. Majority voting follows to identify the selected image regions under the new roots. Note that the treatment of subtrees under children nodes, here, is exactly the same as subtrees under the roots in the whole-object recognition strategy. Hence, the pixel majority vote identifies the selected

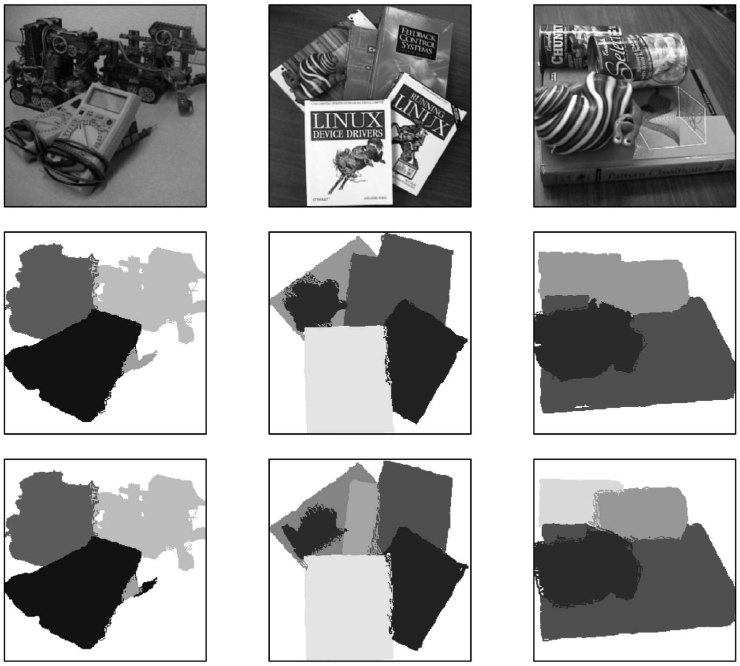


Fig. 11. Comparison of two recognition strategies: (top) challenging images of size 256×256 containing objects that are very similar in appearance; (middle) classification using the whole-object recognition strategy; (bottom) classification using the part-object recognition strategy; each recognized object in the image is marked with a different color.

image regions as object parts. Finally, in the second stage of our recognition strategy, another round of majority voting is conducted at the original roots over the labels of identified object parts that descend from a unique root. Therefore, in the second stage, object-part majority voting leads to ultimate recognition of an object as a whole. The block-diagram of the object-part recognition strategy is shown in Fig. 2.

Fig. 11 presents classification results using the whole-object and object-part recognition strategies on three images from our dataset containing objects that are very similar in appearance. In the leftmost example, both strategies fail to make a distinction between the two different “Fluke” voltage-measuring instruments (see Fig. 6), since the part that differentiates one object most from another is occluded, making it a difficult case for recognition even for a human interpreter. In the other two images, note that the object-part recognition strategy is more successful than the whole-object approach. The recognition error averaged over all objects in 120 test images is only 5.8%, an improvement of nearly 40% over the reported error of 9.6% in the previous section.

These results support our hypothesis that for successful recognition of partially occluded objects it is critical to analyze visible object details at various scales.

6 Conclusions

This paper addresses the problem of detecting and recognizing partially occluded objects in complex scenes. We have shown that a careful analysis of visible fine-scale object details can be critical for recognition accuracy in such scenes, leading to the development of two object-recognition strategies. DTSBNs facilitate the analysis of multiple sub-parts of multiple objects in an image, and, as such, offer an intuitively appealing framework for recognition in occluded scenes.

The proposed generative model DTSBN can be viewed as a special case of the DT model introduced in [13]. Unlike the DT, the DTSBN is applicable in supervised settings, since its single-layer observables are not allowed to change in inference along with dynamic changes of model structure. The difference in observable information between DTs and DTSBNs renders a direct comparison between DTSBNs and DTs beyond the scope of this paper. However, it is worth noting that image segmentation performance of the DT with observables present only at the lowest level is surprisingly just slightly worse than the performance of the DT with multi-layer observables present at all model levels, as reported in [13].

For inference of DTSBNs we have used our SVA algorithm, which relaxes poorly justified independence assumptions of Storkey and Williams [12], and converges to larger likelihoods an order of magnitude faster than competing algorithms [13]. For learning the parameters of the joint prior distribution of the DTSBN, we have derived the training algorithm based on standard maximum likelihood (ML) optimization.

Experiments within the proposed framework have illustrated the capability of DTSBNs to capture important component-subcomponent structures in images. For both DTSBN-based recognition strategies (whole-object and object-part), our results demonstrate better performance of the DTSBN generative framework compared with representatives of descriptive, discriminative, and fixed-structure statistical models. Furthermore, the object-part recognition strategy, which explicitly represents object components at various scales, decreases recognition error an additional 40% over the same dataset, when compared to the "traditional" whole-object approach.

The results presented in this paper support our hypothesis that for successful recognition of partially occluded objects it is critical to analyze visible object details at various scales. Ultimately, what allows us to overcome the computational complexity issues for such an approach to recognition is the proposed generative-model framework. Both the computationally efficient SVA inference algorithm and the object-part recognition strategy arise from the causal, Markov property of DTSBNs. Consequently, we anticipate our future research efforts to improve upon available recognition approaches by utilizing the causality of the generative-model paradigm.

References

[1] B. J. Frey, N. Jojic, 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, D. Castanon, Partially occluded object recognition using statistical models, Int'l J. Computer Vision 49 (1) (2002) 57–78.

[3] G. J. III, B. Bhanu, Recognition of articulated and occluded objects, IEEE Trans. Pattern Anal. Machine Intell. 21 (7) (1999) 603–613.

[4] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: Proc. IEEE Comp. Soc. Conf. Computer Vision and Pattern Rec., Vol. 2, 2003, pp. 264–271.

[5] W. M. Wells, Statistical approaches to feature-based object recognition, Int'l J. Computer Vision 21 (1) (1997) 63–98.

[6] F. Dell'Acqua, R. Fisher, Reconstruction of planar surfaces behind occlusions in range images, IEEE Trans. Pattern Anal. Machine Intell. 24 (4) (2002) 569–575.

[7] M. H. Lin, C. Tomasi, Surfaces with occlusions from layered stereo, IEEE Trans. Pattern Anal. Machine Intell. 26 (8) (2004) 1073–1078.

[8] A. Mittal, L. S. Davis, M2tracker: a multi-view approach to segmenting and tracking people in a cluttered scene, Int'l J. Computer Vision 51 (3) (2003) 189–203.

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

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

[11] A. J. Storkey, Dynamic trees: a structured variational method giving efficient propagation rules, in: Proc. 16th Conf. on Uncertainty in Artificial Intelligence, Stanford, California, 2000, pp. 566–573.

[12] A. J. Storkey, C. K. I. Williams, Image modeling with position-encoding dynamic trees, IEEE Trans. Pattern Anal. Machine Intell. 25 (7) (2003) 859–871.

[13] S. Todorovic, M. C. Nechyba, Dynamic trees for unsupervised segmentation and matching of image regions, IEEE Trans. Pattern Anal. Machine Intell. 27 (11) (2005) 1762–1777.

[14] C. K. I. Williams, M. K. Titsias, Greedy learning of multiple objects in images using robust statistics and factorial learning, Neural Computation 16 (5) (2004) 1039–1062.

[15] S. C. Zhu, Statistical modeling and conceptualization of visual patterns, IEEE Trans. Pattern Anal. Machine Intell. 25 (6) (2003) 691–712.

[16] A. E. C. Pece, Editorial: Generative model based vision, Computer Vision and Image Understanding, this issue (2005).

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

[18] S. Kumar, 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.

[19] X. Feng, C. K. I. Williams, S. N. Felderhof, Combining belief networks and neural networks for scene segmentation, IEEE Trans. Pattern Anal. Machine Intell. 24 (4) (2002) 467–483.

[20] C. A. Bouman, M. Shapiro, A multiscale random field model for Bayesian image segmentation, IEEE Trans. Image Processing 3 (2) (1994) 162–177.

[21] H. Cheng, C. A. Bouman, Multiscale Bayesian segmentation using a trainable context model, IEEE Trans. Image Processing 10 (4) (2001) 511–525.

[22] J.-M. Laferté, P. Pérez, F. Heitz, Discrete Markov image modeling and inference on the quadtree, IEEE Trans. Image Processing 9 (3) (2000) 390–404.

[23] H. Choi, R. G. Baraniuk, Multiscale image segmentation using wavelet-domain hidden Markov models, IEEE Trans. Image Processing 10 (9) (2001) 1309–1321.

[24] M. K. Schneider, P. W. Fieguth, W. C. Karl, A. S. Willsky, Multiscale methods for the segmentation and reconstruction of signals and images, IEEE Trans. Image Processing 9 (3).

[25] P. Sajda, C. Spence, L. Parra, A multi-scale probabilistic network model for detection, synthesis, and compression in mammographic image analysis, Medical Image Analysis 7 (2) (2003) 187–204.

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

[27] W. W. Irving, P. W. Fieguth, A. S. Willsky, An overlapping tree approach to multiscale stochastic modeling and estimation, IEEE Trans. Image Processing 6 (11) (1997) 1517–1529.

[28] M. Wainwright, E. Simoncelli, A. Willsky, Random cascades on wavelet trees and their use in analyzing and modeling natural images, Applied and Computational Harmonic Analysis 11 (1) (2001) 89–123.

[29] J. Li, R. M. Gray, R. A. Olshen, Multiresolution image classification by hierarchical modeling with two-dimensional Hidden Markov Models, IEEE Trans. Inform. Theory 46 (5) (2000) 1826–1841.

[30] A. Montanvert, P. Meer, A. Rosenfield, Hierarchical image analysis using irregular tessellations, IEEE Trans. Pattern Anal. Machine Intell. 13 (4) (1991) 307–316.

[31] M. Aitkin, D. B. Rubin, Estimation and hypothesis testing in finite mixture models, J. Royal Stat. Soc. B-47 (1) (1985) 67–75.

[32] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, L. K. Saul, An introduction to variational methods for graphical models, Machine Learning 37 (2) (1999) 183–233.

[33] T. M. Cover, J. A. Thomas, Elements of information theory, Wiley Interscience Press, New York, 1991.

[34] D. J. C. MacKay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, UK, 2003, Ch. 29, pp. 357–386.

[35] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int'l J. Computer Vision 60 (2) (2004) 91–110.

[36] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Machine Intell. 24 (4) (2002) 509–522.