IRREGULAR-STRUCTURE TREE MODELS FOR IMAGE INTERPRETATION

By

SINISA TODOROVIC

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2005

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Dr. Michael Nechyba for his wise and patient guidance of my research for this dissertation. As my former advisor, Dr. Nechyba has been directing but on no account confining my interests. I especially appreciate his readiness and expertise to help me solve numerous implementation issues. Most importantly, I am thankful for the friendship that we have developed collaborating on this work.

Also, I thank my current advisor Dr. Dapeng Wu for putting extra effort to help me finalize my PhD studies. I am grateful for his invaluable pieces of advise in choosing my future research goals, as well as for practical concrete steps that he undertook to help me find a job.

My thanks also go to Dr. Jian Li, who helped me a lot in the transition period in which I was supposed to change my advisor. Her research group provided for a stimulating environment for me to endeavor investigating areas that are beyond the work presented in this dissertation.

Also, I thank Dr. Antonio Arroyo, whose brilliant lectures on machine intelligence have inspired me to do research in the field of machine learning. As the director of the Machine Intelligence Lab (MIL), Dr. Arroyo has created a warm, friendly, and hard working atmosphere among the “MIL-ers.” Thanks to him, I have decided to join the MIL, which has proved on numerous occasions to be the right decision. I thank all the members of the MIL for their friendship and support.

I thank Dr. Takeo Kanade and Dr. Andrew Kurdila for sharing their research efforts on the micro air vehicle (MAV) project with me. The multidisciplinary environment of this project in which I had a chance to collaborate with various researchers with diverse educational backgrounds was a great experience for me.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

LIST OF FIGURES

KEY TO ABBREVIATIONS

The list shown below gives a description of the frequently used acronyms or abbreviations in this work. For each name, the page number corresponds to the place where the name is first used.

B: blue channel of the RGB color space	43
G: green channel of the RGB color space	43
R: red channel of the RGB color space	43
IQT_V: irregular tree with fixed node positions, and with observables present at all levels	26
IQT_{V^0}: irregular tree with fixed node positions, and with observables present only at the leaf-level	26
IT_{V^0}: irregular tree where observables are present only at the leaf-level	13
IT_V: irregular tree where observables are present at all levels	13
g: normalized green channel	43
r: normalized red channel	43
CWT: Complex Wavelet Transform	40
DRF: Discriminative Random Field	51
DTCWT: Dual Tree Complex Wavelet Transform	40
DWT: Discrete Wavelet Transform	37
EM: Expectation-Maximization algorithm	7
KL: Kullback-Leibler divergence	17
MAP: Maximum A Posteriori	3
MCMC: Markov Chain Monte Carlo method	15
ML: Maximum Likelihood	22
MPM: Maximum Posterior Marginal	69
MRF: Markov Random Field	2
NP: nondeterministic polynomial time	7

RGB: The color space that consists of red, green and blue color values 43

ROC: receiver operating characteristic 52

SVA: structured variational approximation inference algorithm 16

TSBN: tree-structured belief network 5

VA: variational approximation inference algorithm 16

KEY TO SYMBOLS

The list shown below gives a brief description of the major mathematical symbols defined in this work. For each symbol, the page number corresponds to the place where the symbol is first used.

A_{ij}: influence of observables Y on ξ_{ij}	20
B_{ij}: influence of the geometric properties of the network on ξ_{ij}	20
G: number of components in a Gaussian mixture	15
H_i: Shanon's entropy of node i	34
$J(Q,P)$: free energy	64
L: maximum number of levels in the irregular tree	13
M: set of image classes (i.e., object appearances)	13
P_{ij}^{kl}: conditional probability tables	13
Q_{ij}^{kl}: approximate conditional probability tables, given Y and R^0	18
R: positions of all nodes in the irregular tree	13
R': positions of non-leaf nodes in the irregular tree	13
R^0: positions of leaf nodes in the irregular tree	13
V: set of all nodes in the irregular tree	13
V': set of all non-leaf nodes in the irregular tree	13
V^0: set of all leaf nodes in the irregular tree	13
X: random vector of all x_i^k	13
Y: all observables	13
Z: connectivity random matrix	13
$\mathcal{C}$: cost function	20
Ψ: the set of all parameters $\{p_i\}$ in the irregular tree with fixed node positions	28
Σ_{ij} covariance matrix of a relative child-parent displacement $(\boldsymbol{r}_i - \boldsymbol{r}_j)$	13
Θ: set of parameters that characterize an irregular tree	15

$\mathbf{\Omega}_{ij}$: approximate covariance of $\boldsymbol{r}_i$, given that j is the parent of i, and given Y and R^0 18

$\boldsymbol{\mu}_{ij}$: approximate mean of $\boldsymbol{r}_i$, given that j is the parent of i, and given Y and R^0 . . 18

$\boldsymbol{\rho}(i)$: coordinate of an observable random vector in the image plane 13

ℓ: index of levels in the irregular tree 13

γ_{ij}: probability of a node i being the child of j 13

κ: normalization constant 18

θ: set of parameters that characterize a Gaussian mixture 15

ξ_{ij}: approximate probability of i being the child of j, given Y and R^0 18

m_i^k: approximate posterior that node i is labeled as image class k, given Y and R^0 . 19

x_i: image-class of node i 13

x_i^k: image-class indicator if k class is assigned to node i 13

z_{ij}: connectivity indicator random variable between nodes i and j 13

$\mathbf{d}_{ij}$ the mean of relative displacement $\mathbf{r}_i - \mathbf{r}_j$ 13

$\mathbf{r}_i$: position of node i in the image plane 13

$\mathbf{y}_{\boldsymbol{\rho}(i)}$: observable random vector 13

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

IRREGULAR-STRUCTURE TREE MODELS FOR IMAGE INTERPRETATION

By

Sinisa Todorovic

May 2005

Chair: Dapeng Wu
Major Department: Electrical and Computer Engineering

In this dissertation, we seek to accomplish the following related goals: (1) to find a unifying framework to address localization, detection, and recognition of objects, as three sub-tasks of image-interpretation, and (2) to find a computationally efficient and reliable solution to recognition of multiple, partially occluded, alike objects in a given single image. The second problem is to date an open problem in computer vision, eluding a satisfactory solution. For this purpose, we formulate object recognition as Bayesian estimation, whereby class labels with the maximum posterior distribution are assigned to each pixel. To efficiently estimate the posterior distribution of image classes, we propose to model images with graphical models known as *irregular trees*.

The irregular tree specifies probability distributions over both its structure and image classes. This means that, for each image, it is necessary to infer the optimal model structure, as well as the posterior distribution of image classes. We propose several inference algorithms as a solution to this NP-hard problem (nondeterministic polynomial time), which can be viewed as variants of the Expectation-Maximization (EM) algorithm.

After inference, the model represents a forest of subtrees, each of which segments the image. That is, inference of model structure provides a solution to object localization and detection.

With respect to our second goal, we hypothesize that for a successful occluded-object recognition it is critical to explicitly analyze visible object parts. Irregular trees are convenient for such analysis, because the treatment of object parts represents merely a particular interpretation of the tree/subtree structure. We analyze the *significance* of irregular-tree nodes, representing object parts, with respect to recognition of an object as a whole. This information is then exploited toward the ultimate object recognition.

Empirical results demonstrate that irregular trees more accurately model images than their fixed-structure counterparts quad-trees. Also, the experiments reported herein show that our explicit treatment of object parts results in an improved recognition performance, as compared to the strategies in which object components are not explicitly accounted for.

CHAPTER 1
INTRODUCTION

Image interpretation is a difficult challenge that has long been confronting the computer-vision community. A number of factors contribute to the complexity of this problem. The most critical is inherent uncertainty in how the observed visual evidence in images should be attributed to infer object types and their relationships. In addition to video noise, there are various sources of this uncertainty, including variations in camera quality and position, wide-ranging illumination conditions, extreme scene diversity, and the randomness of object appearances, clutter and locations in scenes.

One of the critical hindrances to successful image interpretation is that objects may occlude each other in a complex scene. In the literature, the initial research on the interpretation of scenes with occlusions appeared in early nineties. However, in the last decade relatively small volume of the related literature was published. In fact, a majority of the recently proposed vision systems is not directly aimed at solving the problem of occluded-object recognition; experiments on images with occlusions are reported as a side result only to illustrate the versatility of those systems. This suggests that recognition of partially occluded objects is an open problem in computer vision, which motivates us to seek its solution in this dissertation.

In the initial work, local features (e.g., points, line and curve segments) are used to represent objects, allowing the unoccluded features to be matched with object features, by computing a scalar measure of model fit [1–3]. The unmatched scene features are modeled as spurious features, and the unmatched object features indicate the occluded part of the object. The matching score is either the number of matched object features or the sum of a Gaussian-weighted matching error. The main limitation with these approaches is that they do not account for the spatial correlation among occlusions.

Statistical approaches to occluded-object recognition have also been reported in the literature. For instance, Wells [4], and Ying and Castanon [5] propose probabilistic models to characterize scene features and the correspondence between scene and object features. The authors model both object-feature uncertainty and the probability that the object features are occluded in the scene. They introduce two statistical models for occlusion. One model assumes that each feature can be occluded independently of whether any other features are occluded, whereas the second model accounts for the spatial correlation to represent the extent of occlusion. The spatial correlation is computed using a Markov Random Field (MRF) model with a Gibbs distribution [6]. The main drawback of these systems is a prohibitive computational load; the run-time of these algorithms is exponential in the number of objects to be recognized.

Other related work exploits auxiliary information provided, for example, by image sequences or stereo views of the same scene [5, 7–11], where occlusions are transitory. Since this information in general may not be available, and/or occlusions may remain permanent, in our approach we do not use the strategies of these systems.

A review of the related literature also suggests that the majority of vision systems are designed to deal with only one constrained vision task, such as, for example, image segmentation [5, 10, 11]. However, to conduct image interpretation, as is our goal, it is necessary to perform three related tasks: (1) localization, (2) detection (also called image segmentation), and (3) ultimate recognition of object appearances (also called image classification). Further, in many systems in which the three sub-tasks are addressed, this is not done in a unified manner. Here, as a drawback, the system's architecture comprises a serial connection of separate modules, without any feedback on the accuracy of the ultimate recognition. Moreover, vision systems are typically designed to recognize only a specific instance of object classes appearing in the image (e.g., face), which, in turn, is assumed dissimilar to other objects in the image. However, the assumption of uniqueness of the target class may not be appropriate in many settings. Also, the success of these systems usually depends on *ad hoc* fine-tuning of the feature-extraction methods and system's parameters, optimized for that unique target class. With current demands to design systems capable of classifying thousands of image classes simultaneously, it would be difficult to generalize the outlined approaches.

The small volume of published research addressing occlusions in images suggests that the problem is not fully examined. Also, the drawbacks of the above systems–namely: constrained goals and settings of operation, poor spatial modeling of occlusion, and prohibitive computational load–motivated us to conduct the research reported herein. Our motivation is that most object classes seem to be naturally described by a few characteristic parts or components and their geometrical relation. We hypothesize that it is not the percentage of occlusion that is critical for object recognition, but rather which object parts are occluded. 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 parts. One of the benefits of such analysis is the flexibility to develop various recognition strategies that weigh the information obtained from the detected object parts more judiciously. In the following section, we review some of the reported part-based object-recognition strategies.

1.1 Part-Based Object Recognition

Recently, there has been a flurry of research related to part-based object recognition. For example, Mohan et al. [12] use separate classifiers to detect heads, arms, and legs of people in an image, and a final classifier to decide whether a person is present. However, the approach requires object parts to be manually defined and separated for training the individual part classifiers. To build a system that is easily extensible to deal with different objects, it is important that the part selection procedure be automated. One approach in this direction is developed by Weber et al. [13, 14]. The authors assume that an object is composed of parts and shape, where parts are image patches, which may be detected and characterized by appropriate detectors, and shape describes the geometry of the mutual position of the parts in a way that is invariant with respect to rigid and, possibly, affine transformations. The authors propose a joint probability density over part appearances and shape that models the object class. This framework is appealing in that it naturally allows for parts of different sizes and resolutions. However, due to computational issues, to learn the joint probability density, the authors choose heuristically a small number of parts per each object class, rendering the density unreliable in the case of large variations across images.

Probabilistic detection of object parts has also been reported. For instance, Heisele et al. [15] propose to learn object components from a set of examples based on their discriminative power, and their robustness against pose and illumination changes. For this purpose, they use Support Vector Machines. Also, Felzenszwalb and Huttenlocher [16] represent an object by a collection of parts arranged in a deformable configuration. In their approach, the appearance of each part is modeled separately by Gaussian-mixture distributions, and the deformable configuration is represented by spring-like connections between pairs of parts. The main problem of the mentioned approaches is that they lack the analysis of object parts through scales. It is assumed that parts cannot contain other sub-parts, and that objects are unions of mutually exclusive components, which is hard to justify for more complex object classes.

To address the analysis of object parts through scales Schneiderman and Kanade [17] propose a trainable multi-stage object detector composed of classifiers, each making a decision about whether to cease evaluation, labeling the input as non-object, or to continue further evaluation. The detector orders these stages of evaluation from a low-resolution to a high-resolution search of the image.

The aforementioned approaches are not suitable for recognition of a large number of object classes. As the number of classes increases there is a combinatorial explosion of the number of their parts (i.e., image patches) that need to be evaluated by appropriate detectors.

In this dissertation, we seek a solution to the outlined problems. Our goal it to design a vision system that would analyze multiple object classes through their constituent, ”meaningful” parts at a number of different resolutions. To this end, we resort to a probabilistic framework, as discussed in the following section.

1.2 Probabilistic Framework

We formulate image interpretation as inference of a posterior distribution over pixel random fields for a given image. Once the posterior distribution of image classes is inferred, each pixel can be labeled through Bayesian estimation (e.g., *maximum a posteriori*–MAP). Within this framework, it is necessary to specify the following:

1. The probability distribution of image classes over pixel random fields,
2. The inference algorithms for computing the posterior distribution of image classes,
3. Bayesian estimation for ultimate pixel labeling, that is, object recognition.

Our principal challenge lies in choosing a statistical model for specifying the probability distribution of image classes, since this choice conditions the formulation of inference and Bayesian estimation. A suitable model should be computationally manageable, and sufficiently expressive to represent a wide range of patterns in images. A review of the literature offers four broad classes of models [18]. The descriptive models are constructed based on statistical descriptions of image ensembles with variables only at one level (e.g., [19, 20]). The pseudo-descriptive models reduce the computational cost of descriptive models by imposing partial (or even linear) order among random variables (e.g., [21,22]). The generative models consist of observable and hidden variables, where hidden variables represent a finite number of bases generating an image (e.g., [23, 24]). The discriminative models directly encode posterior distribution of hidden variables given observables (e.g., [25,26]).

The available models differ in structural complexity and difficulty of inference. At one end lie descriptive models, which build statistical descriptions of image ensembles only at the observable (i.e., pixel) level. Other modeling paradigms (i.e., generative, discriminative) impose varying levels of structure through the introduction of hidden variables. However, no principled formulation exists, as of yet, to suggest one approach superior to the others. Therefore, our choice of model is guided by the goal to interpret scenes with partially occluded, alike objects. We seek a model that offers a viable means of recognizing partially occluded objects through recognition of their visible constituent parts. Thus, a prospective model should allow for analysis of object parts towards recognition of objects as a whole.

To alleviate the computational complexity arising from the treatment of multiple object-parts of multiple objects in images, we seek a model that is capable of modeling both whole objects and their sub-parts in a unified manner. That is, a candidate model must be expressive enough to capture component-subcomponent relationships among regions in an image. To accomplish this, it is necessary to analyze pixel neighborhoods of varying size. The literature abounds with reports on successful applications of multiscale statistical models for this purpose [27–32]. Following these trends, we choose the *irregular tree-structured belief network*, or short *irregular tree*. Our choice is directly driven by our image-interpretation strategy and goals, and appears better suited than alternative statistical approaches. Descriptive models lack the necessary structure for component-subcomponent representation we seek to exploit. Discriminative approaches directly model posterior distribution of hidden variables given observables. Consequently, they lose the convenience of assigning physical meaning to the statistical parameters of the model. In contrast, irregular trees can detect objects and their parts simultaneously, as discussed in the following chapters.

Before we continue to present our approach to image interpretation, we give a brief overview of tree-structured generative models in the following section.

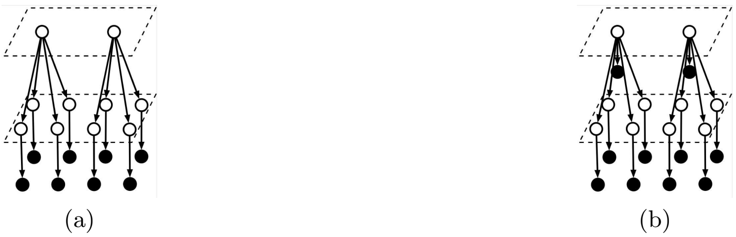
1.3 Tree-Structured Generative Models

Recently, there has been a flurry of research in the field of tree-structured generative models, also known as tree-structured belief networks (TSBNs) [27–33]. The models provide a systematic way to describe random processes/fields and have extremely efficient and statistically optimal inference algorithms. Tree-structured belief networks are characterized by a fixed balanced tree structure of nodes representing hidden (latent) and observable random variables. We focus on TSBNs whose hidden variables take discrete values, though TSBNs can model even continuously valued Gaussian processes [34,35]. The edges of TSBNs represent parent-child (Markovian) dependencies between neighboring layers of hidden variables, while hidden variables, belonging to the same layer, are conditionally independent, as depicted in Figure 1–1. Note that observables depend solely on their corresponding hidden variables. Observables are either present at the finest level only, or could be propagated upward the tree, as dictated by the design choices related to image processing. TSBNs have efficient linear-time inference algorithms, of which, in the graphical-models literature, the best-known is *belief propagation* [36–38]. Cheng and Bouman [29] have used TSBNs for multiscale document segmentation; Kumar and Hebert [39] have employed TSBNs for segmentation of man-made structures in natural scene images; and Schneider et al. [40]

have used TSBNs for simultaneous image denoising and segmentation. All the aforementioned examples demonstrate the powerful expressiveness of TSBNs and the efficiency of their inference algorithms, which is critically important for our purposes.

In spite of these attractive properties, the fixed regular structure of nodes in the TSBN gives rise to "blocky" estimates. The pre-defined tree structure fails to adequately represent the immense variability in size and location of different objects and their subcomponents in images. In the literature, there are several approaches to alleviate this problem. Irving et al. [28] have proposed an overlapping tree model, where distinct nodes correspond to overlapping parts in the image. Li et al. [41] have discussed two-dimensional hierarchical models where nodes are dependent both at any particular layer through a Markov-mesh and across resolutions. In both approaches segmentation results are superior to those when standard TSBNs are used, because the descriptive component of the models is improved at increased computational cost. Ultimately, however, these approaches do not deal with the source of the "blockiness" – namely, the orderly structure of TSBNs.

Not until recently has the research on irregular structures been initiated. Konen et al. [42] have proposed a flexible neural mechanism for invariant pattern recognition based on correlated neuronal activity and the self-organization of dynamic links in neural networks. Also, Montanvert et al. [43], and Bertolino and Montanvert [44] have explored irregular multiscale tessellations that adapt to image content. We join these research efforts building on the work of Adams et al. [45], Adams [46], Storkey [47], and Storkey and Williams [48], by considering the irregular-structured tree belief network.



(a)

(b)

Figure 1–1: Variants of TSBNs: (a) observables (black) at the lowest layer only; (b) observables (black) at all layers; white nodes represent hidden random variables, connected in a balanced quad-tree structure.

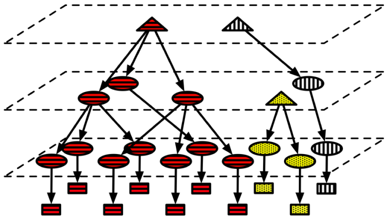


Figure 1–2: An irregular tree consists of a forest of subtrees, each of which segments the image into regions, marked by distinct shading; round- and square-shaped nodes indicate hidden and observable variables, respectively; triangles indicate roots.

In the irregular tree, as in TSBNs, nodes represent random variables, and arcs between them model causal (Markovian) dependence assumptions, as illustrated in Figure 1–2. The irregular tree specifies probability distributions over both its structure and image classes. It is this distribution over tree structures that mitigates the above cited problems with TSBNs.

1.4 Learning Tree Structure from Data is an NP-hard Problem

In order to fully characterize the irregular tree (and any graphical model, for that matter), it is necessary to learn both the graph topology (structure) and the parameters of transition probabilities between connected nodes from training data. Usually, for this purpose, one maximizes the likelihood of the model over training data, while at the same time minimizing the complexity of model structure. Current methods are successful at learning both the structure and parameters from *complete* data. Unfortunately, when the data are *incomplete* (i.e., some random variables are *hidden*), optimizing both the structure and parameters becomes NP-hard (nondeterministic polynomial time) [49, 50].

The principal contribution of this dissertation is that we propose a solution to the NP-hard problem of model-structure estimation. In our approach, we use a variant of the Expectation-Maximization (EM) algorithm [51, 52], to facilitate efficient search over a large number of candidate structures. In particular, the EM procedure iteratively improves its current choice of parameters by using the following two steps. In the Expectation step, current parameters are used for computing the expected value of all the statistics needed to evaluate the current structure. That is, the missing data (hidden variables) are completed by their expected values. In the Maximization step, we replace current parameters with those that maximize the likelihood over the complete data. This second step is essentially equivalent to learning model structure and parameters from complete data, and, hence, can be done efficiently [38, 49, 50].

In the incomplete-data case, a local change in structure of one part of the tree may lead to a structure change in another part of the model. Thus, the available methods for structure estimation evaluate all the neighbors (e.g., networks that differ by a few local changes) of each candidate they visit [53]. The novel idea of our approach is to perform a search for the best structure within EM. In each iteration step, our procedure attempts to find a better network structure, by computing the expected statistics needed for evaluation of alternative structures. In contrast to the available approaches, the EM-based structure search makes a significant progress in each iteration. As we show through experimental validation, our procedure requires relatively few EM iterations to learn non-trivial tree structures.

The outlined image modeling constitutes the core of our approach to image interpretation, which is discussed in the following section.

1.5 Our Approach to Image Interpretation

We seek to accomplish the following related goals: (1) to find a unifying framework to address localization, detection, and recognition of objects, as three sub-tasks of image-interpretation, and (2) to find a computationally efficient and reliable solution to recognition of multiple, partially occluded, alike objects in a given single image. For this purpose, we formulate object recognition as the Bayesian estimation problem, where class labels are assigned to pixels by minimizing the expected value of a suitably specified cost function. This formulation requires efficient estimation of the posterior distribution of image classes (i.e., objects), given an image. To this end, we resort to directed graphical models, known as *irregular trees* [45–48, 54, 55]. As discussed in Section 1.3, the irregular tree specifies probability distributions over both its structure and image classes. This means that, for each image, it is necessary to infer the optimal model structure, as well as the posterior distribution of image classes. By utilizing the Markov property of the irregular tree, we are in a position to reduce computational complexity of the inference algorithm, and, thereby, to efficiently solve our Bayesian estimation problem.

After inference, the model 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. 1–2. These segmented image regions can be interpreted as distinct object appearances in the image. That is, inference of irregular-tree structure provides a solution to localization and detection. Moreover, in inference, we also derive the posterior distribution of image classes over leaf nodes. In order to classify the segmented image regions as a whole, we perform majority voting over the maximum a posteriori (MAP) classes of leaf nodes. In this fashion, we accomplish our first goal.

With respect to our second goal, we hypothesize that the critical factor in a successful occluded-object recognition should be the analysis of visible object parts, which, as discussed before, usually induces prohibitive computational cost. To account explicitly for object parts at various scales, we utilize the Markovian property of irregular trees, which lends itself as a natural solution. Since each root determines a subtree whose leaf nodes form a detected object, we can assign physical meaning to roots as representing whole objects. Also, each descendant of the root down the subtree can be interpreted as the root of another subtree whose leaf nodes cover only a part of the object. Thus, roots' descendants can be viewed as object parts at various scales. Therefore, within the irregular-tree framework, the treatment of object parts represents merely a particular interpretation of the tree/subtree structure.

To reduce the complexity of interpreting all detected object sub-parts, we propose to analyze the *significance* of object components (i.e., irregular-tree nodes) with respect to recognition of objects as a whole. After Bayesian estimation of the irregular-tree structure for a given image, we first find the set of *most significant* irregular-tree nodes. Then, these selected significant nodes are treated as new roots of subtrees. Finally, we conduct MAP classification and majority voting over the selected image regions, descending from the selected *significant* nodes, as illustrated in Fig. 1–3.

1.6 Contributions

Below, we outline the main contributions of this dissertation.

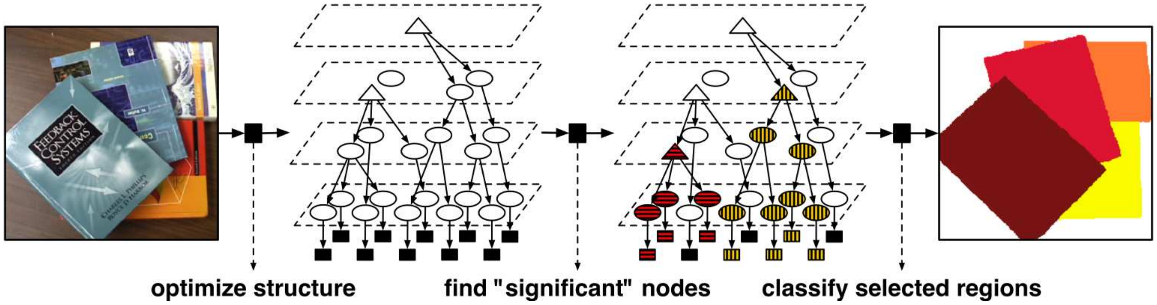


Figure 1–3: Bayesian estimation of the irregular tree along with the analysis of significant tree nodes constitute our approach to recognition of partially occluded, alike objects; shading indicates the two distinct sub-trees under the two "significant" nodes.

We propose an EM-like algorithm for learning a graphical-model, where both model structure and its distributions are learned on a given data simultaneously. The algorithm represents a stage-wise solution to the learning problem known to be NP-hard. While we use the algorithm for learning irregular trees, its generalization to any generative model is straightforward.

A critical part of this learning algorithm is inference of the posterior distribution of image classes on a given data. As is the case for many complex-structure models, exact inference for irregular trees is intractable. To overcome this problem, we resort to variational approximation approach. We assume that there are averaging phenomena in irregular trees that may render a given set of variables in the model approximately independent of the rest of the network. Thereby, we derive the Structured Variational Approximation algorithm that advances existing methods for inference.

In order to avoid variational approximation in inference, we propose two novel architectures and their inference algorithms within the irregular-tree framework. Being simpler, these models allow for exact inference. Moreover, empirically, they exhibit higher accuracy in modeling images than irregular-tree-like models proposed in prior work [45–48].

Along with architectural novelties, we also introduce multi-layered data into the model– an approach that has been extensively investigated in fixed-structure quad-trees [29, 33]. The proposed quad-trees have proved rather successful for various applications including image denoising, classification, and segmentation. Hence, it is important to develop a similar formulation for irregular trees.

We develop a novel approach to object recognition, in which object parts are explicitly analyzed in a computationally efficient manner. As a major theoretical contribution, we define the measure of cognitive significance of object details. The measure provides for a principled algorithm that combines detected object parts toward recognition of an object as a whole.

Finally, we report results of experiments conducted on a wide variety of image datasets, which characterize the proposed models and inference algorithms, and validate our approach to image interpretation.

1.7 Overview

The remainder of the dissertation is organized as follows.

In Chapter 2, we specify two architectures of the irregular-tree model, and derive inference algorithms for them. The architectures differ in the treatment of observable random variables. We also discuss learning of the model parameters. Detailed derivation of the inference algorithm is given in Appendix A.

Next, in Chapter 3, we specify yet another two architectures of the irregular-tree model, for which it is possible to simplify the inference algorithm, as compared to that discussed in Chapter 2. We deliberate the probabilistic inference and learning algorithms for the models.

Further, in Chapter 4, we propose a measure of significance of object parts. This measure ranks object components with respect to the entropy over all image classes (i.e., objects). To incorporate the information of this analysis into the MAP classification, we devise a greedy algorithm, which we refer to as object-part recognition.

The extraction of image features, which we use in our experiments, is thoroughly discussed in Chapter 5. Then, In Chapter 6, we report performance results of different irregular-tree architectures on a large number of challenging images with partially occluded, alike objects.

Finally, in Chapter 7, we summarize the major contributions of the dissertation, and conclude with remarks on the future research.

CHAPTER 2
IRREGULAR TREES WITH RANDOM NODE POSITIONS

2.1 Model Specification

Irregular trees are directed, acyclic graphs with two disjoint sets of nodes representing hidden and observable random vectors. Graphically, we represent all hidden variables as round-shaped nodes, connected via directed edges indicating Markovian dependencies, while observables are denoted as rectangular-shaped nodes, connected only to their corresponding hidden variables, as depicted in Fig. 2–1. Below, we first introduce nodes characterized by hidden variables.

There are V round-shaped nodes, organized in hierarchical levels, V^{ℓ}, $\ell = \{0, 1, ..., L-1\}$, where V^0 denotes the leaf level, and $V' \triangleq V \backslash V^0$. The number of round-shaped nodes is identical to that of the corresponding quad-tree with L levels, such that $|V^{\ell}| = |V^{\ell-1}|/4 = ... = |V^0|/4^{\ell}$. Connections are established under the constraint that a node at level ℓ can become a root, or it can connect only to the nodes at the next $\ell+1$ level. The network connectivity is represented by random matrix Z, where entry z_{ij} is an indicator random variable, such that $z_{ij}=1$ if $i \in V^{\ell}$ and $j \in \{0, V^{\ell+1}\}$ are connected. Z contains an additional zero ("root")

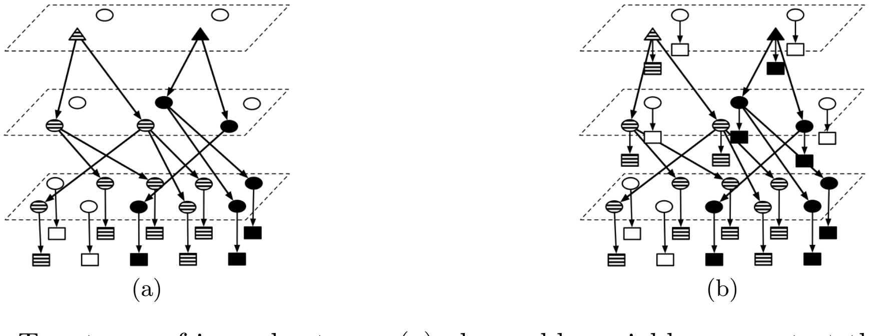


Figure 2–1: Two types of irregular trees: (a) observable variables present at the leaf level only; (b) observable variables present at all levels; round- and square-shaped nodes indicate hidden and observable random variables; triangles indicate roots; unconnected nodes in this example belong to other subtrees; each subtree segments the image into regions marked by distinct shading.

column, where entries z_{i0}=1 if i is a root. Since each node can have only one parent, a realization of Z can have at most one entry equal to 1 in each row. We define the distribution over connectivity as

$$P(Z) \triangleq \textstyle\prod_{\ell=0}^{L-1} \prod_{(i,j)\in V^\ell\times\{0,V^{\ell+1}\}} [\gamma_{ij}]^{z_{ij}} \ , \tag{2.1}$$

where γ_{ij} is the probability of i being the child of j, subject to $\sum_{j\in\{0,V^{\ell+1}\}}\gamma_{ij}$=1.

Further, each round-shaped node i (see Fig. 2–1) is characterized by random position $\boldsymbol{r}_i$ in the image plane. The distribution of $\boldsymbol{r}_i$ is conditioned on the position of its parent $\boldsymbol{r}_j$ as

$$P(\boldsymbol{r}_i|\boldsymbol{r}_j, z_{ij}{=}1) \triangleq \frac{1}{2\pi|\Sigma_{ij}|^{\frac{1}{2}}} \exp(-\frac{1}{2}(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})^T\Sigma_{ij}^{-1}(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij}))\ , \tag{2.2}$$

where Σ_{ij} is a diagonal matrix that represents the order of magnitude of object size, and parameter $\boldsymbol{d}_{ij}$ is the mean of relative displacement $(\boldsymbol{r}_i{-}\boldsymbol{r}_j)$. Storkey and Williams [48] set $\boldsymbol{d}_{ij}$ to zero, which favors undesirable positioning of children and parent nodes at the same locations. From our experiments, this may seriously degrade the image-modeling capabilities of irregular trees, and as such some nonzero relative displacement $\boldsymbol{d}_{ij}$ needs to be accounted for. For roots i, we have $P(\boldsymbol{r}_i|\boldsymbol{r}_0, z_{i0}{=}1){\triangleq}\exp(-\frac{1}{2}(\boldsymbol{r}_i{-}\boldsymbol{d}_i)^T\Sigma_i^{-1}(\boldsymbol{r}_i{-}\boldsymbol{d}_i))/(2\pi|\Sigma_i|^{\frac{1}{2}})$. The joint probability of $R{\triangleq}\{\boldsymbol{r}_i|\forall i{\in}V\}$, is given by

$$P(R|Z) \triangleq \textstyle\prod_{i,j\in V} [\,P(\boldsymbol{r}_i|\boldsymbol{r}_j, z_{ij})\,]^{z_{ij}} \ . \tag{2.3}$$

At the leaf level, V^0, we fix node positions R^0 to the locations of the finest-scale observables, and then use $P(Z, R'|R^0)$ as the prior over positions and connectivity, where $R^0{\triangleq}\{\boldsymbol{r}_i|\forall i{\in}V^0\}$, and $R'{\triangleq}\{\boldsymbol{r}_i|\forall i{\in}V\backslash V^0\}$.

Next, each node i is characterized by an image-class label x_i and an image-class indicator random variable x_i^k, such that x_i^k=1 if x_i=k, where k is a label taking values in the finite set M. Thus, we assume that the set M of unknown image classes is finite. The label k of node i is conditioned on image class l of its parent j and is given by conditional probability tables P_{ij}^{kl}. For roots i, we have $P(x_i^k|x_0^l, z_{i0}{=}1){\triangleq}P(x_i^k)$. Thus, the joint probability of $X{\triangleq}\{x_i^k|i{\in}V, k{\in}M\}$ is given by

$$P(X|Z) = \textstyle\prod_{i,j\in V} \prod_{k,l\in M} \left[P_{ij}^{kl}\right]^{x_i^k x_j^l z_{ij}} \ . \tag{2.4}$$

Finally, we introduce nodes that are characterized by observable random vectors representing image texture and color cues. Here, we make a distinction between two types of irregular trees. The model where observables are present only at the leaf-level is referred to as IT_{V^0}; the model where observables are present at all levels is referred to as IT_V. To clarify the difference between the two types of nodes in irregular trees, we index observables with respect to their locations in the data-structure (e.g., wavelet dyadic squares), while hidden variables are indexed with respect to a node-index in the graph. This generalizes correspondence between hidden and observable random variables of the position-encoding dynamic trees [48]. We define the position of an observable, $\boldsymbol{\rho}(i)$, to be equal to the center of mass of the i-th dyadic square at level ℓ in the corresponding quad-tree with L levels:

$$\boldsymbol{\rho}(i) \triangleq [(n{+}0.5)2^\ell \quad (m{+}0.5)2^\ell]^T \ , \quad \forall i \in V^\ell, \ \ell = \{0, \dots, L{-}1\}, \ n, m = 1, 2, ... \tag{2.5}$$

where n and m denote the row and column in the dyadic square at scale ℓ (e.g., for wavelet coefficients). Clearly, other application-dependent definitions of $\boldsymbol{\rho}(i)$ are possible. Note that while the $\boldsymbol{r}$'s are random vectors, the $\boldsymbol{\rho}$'s are deterministic values fixed at locations where the corresponding observables are recorded in the image. Also, after fixing R^0 to the locations of the finest-scale observables, we have $\forall i{\in}V^0$, $\boldsymbol{r}_i{=}\boldsymbol{\rho}(i)$. The definition, given by Eq. (2.5), holds for IT_{V^0}, as well, for $\ell{=}0$.

For both types of irregular trees, we assume that observables $Y{\triangleq}\{\boldsymbol{y}_{\boldsymbol{\rho}(i)} | \forall i{\in}V\}$ at locations $\boldsymbol{\rho}{\triangleq}\{\boldsymbol{\rho}(i) | \forall i{\in}V\}$ are conditionally independent given the corresponding x_i^k:

$$P(Y|X, \boldsymbol{\rho}) = \textstyle\prod_{i \in V} \prod_{k \in M} \left[P(\boldsymbol{y}_{\boldsymbol{\rho}(i)} | x_i^k, \boldsymbol{\rho}(i))\right]^{x_i^k} \ , \tag{2.6}$$

where for IT_{V^0}, V^0 should be substituted for V. The likelihoods $P(\boldsymbol{y}_{\boldsymbol{\rho}(i)} | x_i^k{=}1, \boldsymbol{\rho}(i))$ are modeled as mixtures of Gaussians: $P(\boldsymbol{y}_{\boldsymbol{\rho}(i)} | x_i^k{=}1, \boldsymbol{\rho}(i)) \triangleq \sum_{g=1}^{G_k} \pi_k(g) \mathcal{N}(\boldsymbol{y}_{\boldsymbol{\rho}(i)}; \boldsymbol{\nu}_k(g), \Xi_k(g))$. For large G_k, a Gaussian-mixture density can approximate any probability density [56]. In order to avoid the risk of overfitting the model, we assume that the parameters of the Gaussian-mixture are equal for all nodes. The Gaussian-mixture parameters can be grouped in the set $\theta \triangleq \{G_k, \{\pi_k(g), \boldsymbol{\nu}_k(g), \Xi_k(g)\}_{g=1}^{G_k} \mid \forall k{\in}M\}$.

Speaking in generative terms, for a given set of V nodes, first $P(Z)$ is defined using Eq. (2.1) and $P(R|Z)$ using Eq. (2.3) to give us $P(Z, R)$. We then impose the condition of fixing the leaf-level node positions to the locations of the finest-scale observables, $\boldsymbol{\rho}^0 \subset \boldsymbol{\rho}$, to obtain $P(Z, R'|R^0{=}\boldsymbol{\rho}^0)$. Combining Eq. (2.4) and Eq. (2.6) with $P(Z, R'|R^0{=}\boldsymbol{\rho}^0)$ results in the joint prior

$$P(Z, X, R', Y|R^0{=}\boldsymbol{\rho}^0) = P(Y|X, \boldsymbol{\rho})P(X|Z)P(Z, R'|R^0{=}\boldsymbol{\rho}^0) \ , \tag{2.7}$$

which fully specifies the irregular tree. All the parameters of the joint prior can be grouped in the set $\Theta \triangleq \{\gamma_{ij}, \boldsymbol{d}_{ij}, \Sigma_{ij}, P^{kl}_{ij}, \theta\}$, $\forall i, j{\in}V$, $\forall k, l{\in}M$.

As depicted in Figure 2–1, a irregular tree is a directed graph. The formalism of the graph-theoretic representation of irregular trees provides general algorithms for computing marginal and conditional probabilities of interest, which is discussed in the following section.

2.2 Probabilistic Inference

Image interpretation, as discussed in Chapter 1, requires computation of posterior probabilities of hidden random variables Z, X, and R', given observables Y and leaf-node positions R^0. However, due to the complexity of irregular trees, the exact probabilistic inference of $P(Z, X, R'|Y, R^0)$ is infeasible. Therefore, we resort to approximate inference methods, which are divided into two broad classes: deterministic approximations and Monte-Carlo methods [57–61].

Markov Chain Monte Carlo (MCMC) methods allow for sampling of the posterior $P(Z, X, R'|Y, R^0)$, and the construction of a Markov chain whose equilibrium distribution is the desired $P(Z, X, R'|Y, R^0)$. Below, we report an experiment for two datasets of 4×4 and 8×8 binary images, samples of which are depicted in Fig. 2–2a, where we learned $P(Z, X, R'|Y, R^0)$ for IT_{V^0} models through Gibbs sampling [62]. Observables y_i were set to binary pixel values; the number of image classes was set to $|M|{=}2$; the number of components in the Gaussian-mixture was set to $G{=}1$; and the maximum number of levels in the model is set to $L{=}3$ and $L{=}4$ for 4×4 and 8×8 images, respectively. The initial irregular-tree structure is a balanced quad-tree (TSBN), where the number of leaf-level nodes is equal to the number of pixels. One iteration of Gibbs sampling consists of sampling each variable, conditioned on the other variables in the irregular tree, until all the variables are sampled. We iterated this procedure until our convergence criterion was met – namely, when $|P_{t+1}(Z, X, R'|Y, R^0){-}P_t(Z, X, R'|Y, R^0)|/P_t(Z, X, R'|Y, R^0){<}\varepsilon$ for $N{=}10$ successive

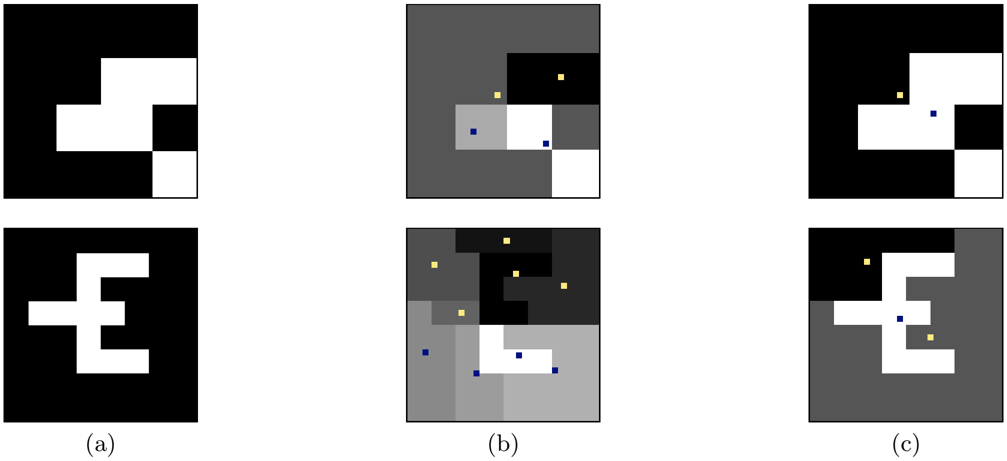


Figure 2–2: Pixel clustering using irregular trees learned by Gibbs sampling: (a) sample 4×4 and 8×8 binary images; (b) clustered leaf-level pixels that have the same parent at level 1; (c) clustered leaf-level pixels that have the same grandparent at level 2; clusters are indicated by different shades of gray; the point in each group marks the position of the parent node.

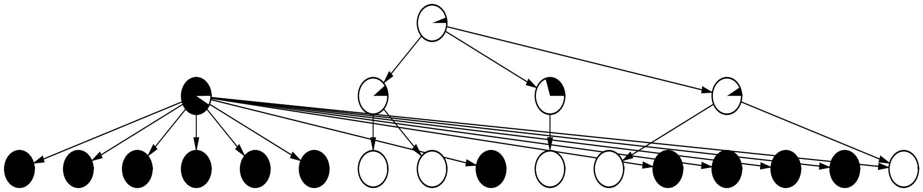


Figure 2–3: Irregular tree learned for the 4×4 image in (a), after 20,032 iterations of Gibbs sampling; nodes are depicted in-line representing 4, 2 and 1 actual rows of the levels 0, 1 and 2, respectively; nodes are drawn as pie-charts representing $P(x_i^k = 1)$, $k \in \{0, 1\}$; note that there are two root nodes for two distinct objects in the image.

iteration steps t, where ε=0.1 and ε=1 for 4×4 and 8×8 images, respectively. For the dataset of 50 binary 4×4 images, on average more than 20,000 iteration steps were required for convergence, while for 50 binary 8×8 binary image, more than 100,000 iterations were required. In Figs. 2–2b-c, we also illustrate the grouping of pixels in the learned irregular trees, while in Fig. 2–3, we depict the irregular tree learned for the 4×4 image in Fig. 2–2a.

From the experimental results, we infer that irregular trees learned through Gibbs sampling are capable of capturing important structural information about image regions at various scales. Generally, however, in MCMC approaches, with increasing model complexity, the choice of proposals in the Markov chain becomes hard, so that the equilibrium distribution is reached very slowly [57, 63]. Hence, in order to achieve faster inference, we resort to variational approximation, a specific type of deterministic approximation [59, 64].

Variational approximation methods have been demonstrated to give good and significantly faster results, when compared to Gibbs sampling [46]. The proposed approaches range from a factorized approximating distribution over hidden variables [45] (a.k.a. mean field variational approximation) to more structured solutions [48], where dependencies among hidden variables are enforced. The underlying assumption in those methods is that there are averaging phenomena in irregular trees that may render a given set of variables approximately independent of the rest of the network. Therefore, the resulting variational optimization of irregular trees provides for principled solutions, while reducing computational complexity. In the following section, we derive a novel Structured Variational Approximation (SVA) algorithm for the irregular tree model defined in Section 2.1.

2.3 Structured Variational Approximation

In variational approximation, the intractable distribution $P(Z, X, R'|Y, R^0)$ is approximated by a simpler distribution $Q(Z, X, R'|Y, R^0)$ closest to $P(Z, X, R'|Y, R^0)$. To simplify notation, below, we omit the conditioning on Y and R^0, and write $Q(Z, X, R')$. The novelty of our approach is that we constrain the variational distribution to the form

$$Q(Z, X, R') \triangleq Q(Z)Q(X|Z)Q(R'|Z) \ , \tag{2.8}$$

which enforces that both class-indicator variables X and position variables R' are 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, our formulation appears to be more appropriate for approximating the true posterior than the mean-field variational approximation $Q(Z, X, R')$=$Q(Z)Q(X)Q(R')$ discussed by Adams et al. [45], and the form $Q(Z, X, R')$=$Q(Z)Q(X|Z)Q(R')$ proposed by Storkey and Williams [48]. We define the approximating distributions as follows:

$$Q(Z) \triangleq \prod_{\ell=0}^{L-1} \prod_{(i,j)\in V^\ell \times \{0, V^{\ell+1}\}} [\xi_{ij}]^{z_{ij}} \ , \tag{2.9}$$

$$Q(X|Z) \triangleq \prod_{i,j\in V} \prod_{k,l\in M} \left[Q_{ij}^{kl}\right]^{x_i^k x_j^l z_{ij}} \ , \tag{2.10}$$

$$Q(R'|Z) \triangleq \prod_{i,j\in V'} [Q(\boldsymbol{r}_i|z_{ij})]^{z_{ij}} = \prod_{i,j\in V'} \left[\frac{\exp\left(-\frac{1}{2}(\boldsymbol{r}_i - \boldsymbol{\mu}_{ij})^T \Omega_{ij}^{-1} (\boldsymbol{r}_i - \boldsymbol{\mu}_{ij})\right)}{2\pi |\Omega_{ij}|^{\frac{1}{2}}} \right]^{z_{ij}} \tag{2.11}$$

where parameters ξ_{ij} correspond to the γ_{ij} connection probabilities, and the Q_{ij}^{kl} are analogous to the P_{ij}^{kl} conditional probability tables. For the parameters of $Q(R'|Z)$, note that covariances Ω_{ij} and mean values $\boldsymbol{\mu}_{ij}$ form the set of Gaussian parameters for a given node $i{\in}V^{\ell}$ over its candidate parents $j{\in}V^{\ell+1}$. Which pair of parameters $(\boldsymbol{\mu}_{ij}, \Omega_{ij})$, is used to generate $\boldsymbol{r}_i$ is conditioned on the given connection between i and j – that is, the current realization of Z. Furthermore, we assume that the Ω's are diagonal matrices, such that node positions along the "x" and "y" image axes are uncorrelated. Also, for roots, suitable forms of Q functions are used, similar to the specifications given in Section 2.1.

To find $Q(Z, X, R')$ closest to $P(Z, X, R'|Y, R^0)$ we resort to a standard optimization method, where Kullback-Leibler (KL) divergence between $Q(Z, X, R')$ and $P(Z, X, R'|Y, R^0)$ is minimized ([65], ch. 2, pp. 12–49, and ch. 16, pp. 482–509). The KL divergence is given by

$$KL(Q\|P) \triangleq \int_{R'} dR' \sum_{Z,X} Q(Z, X, R') \log \frac{Q(Z, X, R')}{P(Z, X, R'|Y, R^0)} . \tag{2.12}$$

It is well known that $KL(Q\|P)$ is non-negative for any two distributions Q and P, and $KL(Q\|P){=}0$ if and only if $Q{=}P$; these properties are a direct corollary of Jensen's inequality ([65], ch. 2, pp. 12–49). As such, $KL(Q\|P)$ guarantees a global minimum – that is, a unique solution to $Q(Z, X, R')$.

By minimizing the KL divergence, we derive the update equations for estimating the parameters of the variational distribution $Q(Z, X, R')$. Below, we summarize the final derivation results. Detailed derivation steps are reported in Appendix A, where we also provide the list of nomenclature. In the following equations, we use κ to denote an arbitrary normalization constant, the definition of which may change from equation to equation. Parameters on the right-hand side of the update equations are assumed known, as learned in the previous iteration step.

2.3.1 Optimization of $Q(X|Z)$

$Q(X|Z)$ is fully characterized by parameters Q_{ij}^{kl}, which are updated as

$$Q_{ij}^{kl} = \kappa P_{ij}^{kl} \lambda_i^k \ , \ \forall i, j \in V \ , \ \forall k, l \in M \ , \tag{2.13}$$

where the auxiliary parameters λ_i^k are computed as

$$\lambda_i^k = \begin{cases} P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k, \boldsymbol{\rho}(i)) & , \ i \in V^0, \\ \prod_{c\in V} \left[\sum_{a\in M} P_{ci}^{ak} \lambda_{ci}^{ak}\right]^{\xi_{ci}} & , \ i \in V', \end{cases} \tag{2.14a}$$

$$\lambda_i^k = P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k, \boldsymbol{\rho}(i)) \prod_{c\in V} \left[\sum_{a\in M} P_{ci}^{ak} \lambda_c^a\right]^{\xi_{ci}} \ , \ \forall i \in V \ , \ \forall k \in M \ , \tag{2.14b}$$

where Eq. (2.14a) is derived for IT_{V^0}, and Eq. (2.14b) for IT_V. Since the ξ_{ci} are non-zero only for child-parent pairs, from Eq. (2.14), we note that λ's are computed for both models by propagating the λ messages of the corresponding children nodes upward. Thus, Q's, given by Eq. (2.13), can be updated by making a single pass up the tree. Also, note that for leaf nodes, $i{\in}V^0$, the ξ_{ci} parameters are equal to 0 by definition, yielding $\lambda_i^k{=}P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k, \boldsymbol{\rho}(i))$ in Eq. (2.14b).

Further, from Eqs. (2.9) and (2.10), we derive the update equation for the approximate posterior probability m_i^k that node i is assigned to image class k, given Y and R^0, as

$$m_i^k = \int_{R'} dR' \sum_{Z,X} x_i^k Q(Z, X, R') = \sum_{j\in V'} \xi_{ij} \sum_{l\in M} Q_{ij}^{kl} m_j^l \ , \ \forall i \in V, \forall k \in M. \tag{2.15}$$

Note that the m_i^k can be computed by propagating image-class probabilities in a single pass downward. This upward-downward propagation, specified by Eqs. (2.14) and (2.15), is very reminiscent of belief propagation for TSBNs [31, 36]. For the special case when $\xi_{ij}{=}1$ only for one parent j, we obtain the standard λ-π rules of Pearl's message passing scheme for TSBNs.

2.3.2 Optimization of $Q(R'|Z)$

$Q(R'|Z)$ is fully characterized by parameters $\boldsymbol{\mu}_{ij}$ and Ω_{ij}. The update equation for $\boldsymbol{\mu}_{ij}$, $\forall (i,j){\in}V^\ell{\times}\{0, V^{\ell+1}\}$, $\ell{>}0$, is given by

$$\boldsymbol{\mu}_{ij}{=} \left[\sum_{p\in V'} \xi_{jp} \Sigma_{ij}^{-1} {+} \sum_{c\in V'} \xi_{ci} \Sigma_{ci}^{-1}\right]^{-1} \left[\sum_{p\in V'} \xi_{jp} \Sigma_{ij}^{-1} (\boldsymbol{\mu}_{jp}{+}\boldsymbol{d}_{jp}) + \sum_{c\in V'} \xi_{ci} \Sigma_{ci}^{-1} (\boldsymbol{\mu}_{ci}{-}\boldsymbol{d}_{ij})\right] , \tag{2.16}$$

where c and p denote children and grandparents of node i, respectively. Further, for all node pairs $\forall(i,j){\in}V^{\ell}{\times}\{0,V^{\ell+1}\}$, $\ell{>}0$, where $\xi_{ij}{\neq}0$, Ω_{ij} is updated as

$$\begin{aligned}\mathrm{Tr}\{\Omega_{ij}^{-1}\} = \quad & \mathrm{Tr}\{\Sigma_{ij}^{-1}\}\left(1{+}\sum_{p\in V'}\xi_{jp}\left[\frac{\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}\}}{\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}}\right]^{\frac{1}{2}}\right) + \\ & + \sum_{c\in V'}\xi_{ci}\mathrm{Tr}\{\Sigma_{ci}^{-1}\}\left(1{+}\left[\frac{\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ci}\}}{\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ij}\}}\right]^{\frac{1}{2}}\right),\end{aligned} \tag{2.17}$$

where, once again, c and p denote children and grandparents of node i, respectively. Since the Ω's and Σ's are assumed diagonal, it is straightforward to derive the expressions for the diagonal elements of the Ω's from Eq. (2.17). Note that both $\boldsymbol{\mu}_{ij}$ and Ω_{ij} are updated summing over children and grandparents of i, and, therefore, must be iterated until convergence.

2.3.3 Optimization of $Q(Z)$

$Q(Z)$ is fully characterized by connectivity probabilities ξ_{ij}, which are computed as

$$\xi_{ij} = \kappa\gamma_{ij}\ \exp(A_{ij} - B_{ij})\ ,\ \forall\ell,\ \forall(i,j){\in}V^{\ell}{\times}\{0,V^{\ell+1}\}\ , \tag{2.18}$$

where A_{ij} represents the influence of observables Y, while B_{ij} represents the contribution of the geometric properties of the network to the connectivity distribution. These are defined in Appendix A.

2.4 Inference Algorithm and Bayesian Estimation

For the given set of parameters Θ characterizing the joint prior, observables Y, and leaf-level node positions R^0, the standard Bayesian estimation of optimal $\hat{Z}$, $\hat{X}$, and $\hat{R}'$ requires minimizing the expectation of a cost function $\mathcal{C}$:

$$(\hat{Z},\hat{X},\hat{R}'){=}\arg\min_{Z,X,R'}\mathbb{E}\{\mathcal{C}((Z,X,R'),(Z^*,X^*,R'^*))|Y,R^0,\Theta\}, \tag{2.19}$$

where $\mathcal{C}(\cdot)$ penalizes the discrepancy between the estimated configuration (Z,X,R') and the true one (Z^*,X^*,R'^*). We propose the following cost function:

$$\mathcal{C}((Z,X,R'),(Z^*,X^*,R'^*)){\triangleq}\sum_{i,j\in V}[1{-}\delta(z_{ij}{-}z_{ij}^*)]{+}\sum_{i\in V}\sum_{k\in M}[1{-}\delta(x_i^k{-}x_i^{k*})]{+}\sum_{i\in V'}[1{-}\delta(\boldsymbol{r}_i{-}\boldsymbol{r}_i^*)], \tag{2.20}$$

where * indicates true values, and $\delta(\cdot)$ is the Kronecker delta function. Using the variational approximation $P(Z,X,R'|Y,R^0)\approx Q(Z)Q(X|Z)Q(R'|Z)$, from Eqs. (2.19) and (2.20), we derive:

$$\hat{Z}= \arg\min_Z \textstyle\sum_Z Q(Z) \sum_{\ell=0}^{L-1} \sum_{(i,j)\in V^\ell\times\{0,V^{\ell+1}\}}[1-\delta(z_{ij}-z^*_{ij})], \tag{2.21}$$

$$\hat{X}= \arg\min_X \textstyle\sum_{Z,X} Q(Z)Q(X|Z) \sum_{i\in V} \sum_{k\in M}[1-\delta(x^k_i-x^{k*}_i))], \tag{2.22}$$

$$\hat{R}'= \arg\min_{R'} \textstyle\int_{R'} dR' \sum_Z Q(Z)Q(R'|Z) \sum_{i\in V'}[1-\delta(\boldsymbol{r}_i-\boldsymbol{r}^*_i)]. \tag{2.23}$$

Given the constraints on connections, discussed in Section 2.1, minimization in Eq. (2.21) is equivalent to finding parents:

$$(\forall\ell)(\forall i\in V^\ell)(Z_{\cdot i}\neq 0)\ \ \hat{\jmath}= \arg\max_{j\in\{0,V^{\ell+1}\}} \xi_{ij}\ ,\ \ \text{for IT}_{V^0}\ , \tag{2.24a}$$

$$(\forall\ell)(\forall i\in V^\ell)\ \ \hat{\jmath}= \arg\max_{j\in\{0,V^{\ell+1}\}} \xi_{ij}\ ,\ \ \text{for IT}_{V}\ , \tag{2.24b}$$

where ξ_{ij} is given by Eq. (2.18); $Z_{\cdot i}$ denotes the i-th column of Z, and $Z_{\cdot i}\neq 0$ indicates that there is at least one non-zero element in column $Z_{\cdot i}$; that is, i has children, and thereby is included in the tree structure. Note that due to the distribution over connections, after estimation of Z, for a given image, some nodes may remain without children. To preserve the generative property in IT_{V^0}, we impose an additional constraint on Z that nodes above the leaf level must have children in order to be able to connect to upper levels. On the other hand, in IT_V, due to multi-layered observables, all nodes V must be included in the tree structure, even if they do not have children. The global solution to Eq. (2.24a) is an open problem in many research areas. Therefore, for IT_{V^0}, we propose a stage-wise optimization, where, as we move upwards, starting from the leaf level $\ell=\{0,1,...,L-1\}$, we include in the tree structure optimal parents at $V^{\ell+1}$ according to

$$(\forall i\in V^\ell)(\hat{Z}_{\cdot i}\neq 0)\ \ \hat{\jmath}= \arg\max_{j\in\{0,V^{\ell+1}\}} \xi_{ij}, \tag{2.25}$$

where $\hat{Z}_{\cdot i}$ denotes i-th column of the estimated $\hat{Z}$, and $\hat{Z}_{\cdot i}\neq 0$ indicates that i has already been included in the tree structure when optimizing the previous level V^ℓ.

Next, from Eq. (2.22), the resulting Bayesian estimator of image-class labels, denoted as $\hat{x}_i$, is

$$(\forall i \in V)\ \hat{x}_i = \arg\max_{k \in M} m_i^k\ , \tag{2.26}$$

where the approximate posterior probability m_i^k that image class k is assigned to node i is given by Eq. (2.15).

Finally, from Eq. (2.23), optimal node positions are estimated as

$$(\forall \ell > 0)(\forall i \in V^\ell)\ \hat{\boldsymbol{r}}_i = \arg\max_{\boldsymbol{r}_i} \textstyle\sum_Z Q(\boldsymbol{r}_i|Z)Q(Z) = \sum_{j \in \{0, V^{\ell+1}\}} \boldsymbol{\mu}_{ij} \xi_{ij}, \tag{2.27}$$

where $\boldsymbol{\mu}_{ij}$ and ξ_{ij} are given by Eqs. (2.16) and (2.18), respectively.

The inference algorithm for irregular trees is summarized in Fig. 2–4. The specified ordering of parameter updates for $Q(Z)$, $Q(X|Z)$, and $Q(R'|Z)$ in Fig. 2–4, steps (4)–(10), is arbitrary; theoretically, other orderings are equally valid.

2.5 Learning Parameters of the Irregular Tree with Random Node Positions

Variational inference presumes that model parameters: $\Theta = \{\gamma_{ij}, \boldsymbol{d}_{ij}, \Sigma_{ij}, P_{ij}^{kl}, \theta\}$, $\forall i, j \in V$, $\forall k, l \in M$, and V, L, M, are available. These parameters can be learned off-line through standard *Maximum Likelihood* (ML) optimization. Usually, for the ML optimization, it is assumed that N, independently generated, training images, with observables $\{Y^n\}_{n=1}^N$ and latent variables $\{(Z^n, X^n, R'^n)\}_{n=1}^N$, 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, configurations $\{(\hat{Z}^n, \hat{X}^n, \hat{R'}^n)\}$ must be estimated from the training images.

To this end, we propose an iterative learning procedure. In initialization, we first set $L = \log_2(|V^0|)$, where $|V^0|$ is equal to the size of a given image. The number of image classes $|M|$ is also assumed known. Next, due to a huge diversity of possible configurations of objects in images, for each node $i \in V^\ell$, we initialize γ_{ij} to be uniform over i's candidate parents $\forall j \in \{0, V^{\ell+1}\}$. Then, for all pairs $(i, j) \in V^\ell \times V^{\ell+1}$ at level ℓ, we set $\boldsymbol{d}_{ij} = \boldsymbol{\rho}(i) - \boldsymbol{\rho}(j)$; namely, the $\boldsymbol{d}_{ij}$ are initialized to the relative displacement of the centers of mass of the i-th and j-th dyadic squares in the corresponding quad-tree with L levels, specified in Eq. (2.5). For roots i, we have $\boldsymbol{d}_i = \boldsymbol{\rho}(i)$. Also, we set diagonal elements of $\boldsymbol{\Sigma}_{ij}$ to the diagonal elements

```
Inference Algorithm
Assume that $V$, $L$, $M$, $\Theta$, $N_\varepsilon$, $\varepsilon$, and $\varepsilon_\mu$ are given.
(1) Initialization: $t$=0; $t_{\text{in}}$=0; $(\forall i,j{\in}V)$ $(\forall k,l{\in}M)$ $\xi_{ij}(0){=}\gamma_{ij}$; $Q_{ij}^{kl}(0){=}P_{ij}^{kl}$; $\boldsymbol{\mu}_{ij}(0)$ ="node
locations in the corresponding quad-tree"; diagonal elements of $\boldsymbol{\Omega}_{ij}(0)$ are set to the area
of dyadic squares in the corresponding quad-tree;
(2) repeat Outer Loop
    (3) $t = t + 1$;
    (4) Compute in bottom-up pass for $\ell$=0, 1, ..., $L{-}1$, $\forall i{\in}V^\ell$, $\forall k{\in}M$,
          $\lambda_i^k(t)$ given by Eq. (2.14); $Q_{ij}^{kl}(t)$ given by Eq. (2.13);
    (5) Compute in top-down pass for $\ell$=$L{-}1, L{-}2, ..., 0$, $\forall i{\in}V^\ell$, $\forall k{\in}M$,
          $m_i^k(t)$ given by Eq. (2.15);
    (6) repeat Inner Loop
        (7) $t_{\text{in}} = t_{\text{in}} + 1$;
        (8) Compute $\forall i,j{\in}V'$,
              $\boldsymbol{\mu}_{ij}(t_{\text{in}})$ given by Eq. (2.16); $\Omega_{ij}(t_{\text{in}})$ given by Eq. (2.17);
    (9) until $|\boldsymbol{\mu}_{ij}(t_{\text{in}}){-}\boldsymbol{\mu}_{ij}(t_{\text{in}}{-}1)|/\boldsymbol{\mu}_{ij}(t_{\text{in}}{-}1) < \varepsilon_\mu$;
    (10) Compute $\forall i,j{\in}V'$,
          $\xi_{ij}(t)$ given by Eq. (2.18);
(11) 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 ;
(12) Estimation of $\hat{Z}$: compute in bottom-up pass for $\ell$=0, 1, ..., $L{-}1$,
        for $\text{IT}_{V^0}$:  $(\forall i{\in}V^\ell)(\hat{Z}_{\cdot i}{\neq}0)$  $\hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}}\xi_{ij}(t)$,
        for $\text{IT}_V$:  $(\forall i{\in}V^\ell)$ $\hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}}\xi_{ij}(t)$;
(13) Estimation of $\hat{X}$: compute $(\forall i{\in}V)$ $\hat{x}_i = \arg\max_{k\in M} m_i^k(t)$;
(14) Estimation of $\hat{R}'$: compute $(\forall \ell{>}0)(\forall i{\in}V^\ell)$ $\hat{\boldsymbol{r}}_i = \sum_{j\in\{0,V^{\ell+1}\}}\boldsymbol{\mu}_{ij}(t)\xi_{ij}(t)$;
```

Figure 2–4: Inference of the irregular tree given Y, R^0, and Θ; 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.

of a matrix $\boldsymbol{d}_{ij}\boldsymbol{d}_{ij}^T$. The number of components G_k in a Gaussian mixture for each class k is set to G_k=3, which is empirically validated to be appropriate. Other parameters of the Gaussian mixture, θ, are estimated by using the EM algorithm [52,56] on the hand-labeled training images. Finally, conditional probability tables, P_{ij}^{kl}, are initialized to be uniform over possible image classes.

After initialization of Θ, we run an iterative learning procedure, where in step t we conduct SVA inference of the irregular tree on the training images, as explained in the previous section. After inference of the posterior probability that class k is assigned to node i, m_i^k, given by Eq. (2.15), and posterior connectivity probability, ξ_{ij}, given by Eq. (2.18), on all training images, $n = 1, ..., N$, we update only P_{ij}^{kl} and γ_{ij} as

$$P_{ij}^{kl}(t{+}1) = \frac{1}{N}\sum_{n=1}^{N} m_i^{k;n}(t) \ , \tag{2.28}$$

$$\gamma_{ij}(t{+}1) = \frac{1}{N}\sum_{n=1}^{N} \xi_{ij}^{n}(t) \ . \tag{2.29}$$

Other parameters in $\Theta(t{+}1){=}\{\gamma_{ij}(t{+}1), \boldsymbol{d}_{ij}, \Sigma_{ij}, P_{ij}^{kl}(t{+}1), \theta\}$, are fixed to their initial values. In the next iteration step, we use $\Theta(t{+}1)$ for SVA inference of the irregular tree on the training images. We assume that the learning algorithm converged when

$$\frac{|P_{ij}^{kl}(t{+}1) - P_{ij}^{kl}(t)|}{P_{ij}^{kl}(t)} < \varepsilon \ ,$$

where $\varepsilon > 0$ is a pre-specified parameter.

2.6 Implementation Issues

In this section, we list algorithm-related details that are necessary for the experimental results, presented in Chapter 6, to be reproducible.

First, direct implementation of Eq. (2.13) would result in numerical underflow. Therefore, we introduce the following scaling procedure:

$$\tilde{\lambda}_i^k \triangleq \frac{\lambda_i^k}{S_i}, \quad \forall i{\in}V, \quad \forall k{\in}M \ , \tag{2.30}$$

$$S_i \triangleq \sum_{k\in M} \lambda_i^k \ . \tag{2.31}$$

Substituting the scaled $\tilde{\lambda}$'s into Eq. (2.13), we obtain

$$Q_{ij}^{kl} = \frac{P_{ij}^{kl}\lambda_i^k}{\sum_{a\in M} P_{ij}^{al}\lambda_i^a} = \frac{P_{ij}^{kl}\tilde{\lambda}_i^k}{\sum_{a\in M} P_{ij}^{al}\tilde{\lambda}_i^a} \ . \tag{2.32}$$

In other words, computation of Q_{ij}^{kl} does not change when the scaled $\tilde{\lambda}'s$ are used.

Second, to reduce computational complexity, we consider, for each node i, only the 7×7 box encompassing parent nodes j that neighbor the parent of the corresponding quad-tree. Consequently, the number of possible children nodes c of i is also limited. Our experiments show that the omitted nodes, either children or parents, contribute negligibly to the update equations. Thus, we limit overall computational cost as the number of nodes increases.

Finally, the convergence criterion of the inner loop, where $\boldsymbol{\mu}_{ij}$ and Ω_{ij} are computed, is controlled by parameter ε_{μ}. When ε_{μ}=0.01, the average number of iteration steps, t_{in}, in the inner loop, is from 3 to 5, depending on the image size, where the latter is obtained for 128×128 images. The convergence criterion of the outer loop is controlled by parameters N_{ε} and ε. Simplifications that we use in practice may lead to sub-optimal solutions of SVA. From our experience, though, the algorithm recovers from unstable stationary points for sufficiently large N. In our experiments, we set N_{ε}=10 and ε=0.01.

After the inference algorithm (Fig. 2–4) converged, we then estimate the values of hidden variables $(\hat{Z}, \hat{X}, \hat{R}')$ for a given image, thereby conducting image interpretation.

CHAPTER 3
IRREGULAR TREES WITH FIXED NODE POSITIONS

In the previous chapter, two architectures of the irregular tree are presented, which are fully characterized by the following joint prior:

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

As discussed in Section 2.2, the inference of the posterior distribution $P(Z,X,R'|Y,R^0)$ is intractable, due to the complexity of the model. The node-position variables, R', are the main culprit for conducting approximate inference. On the other hand, the R' are very useful, because they constrain possible network configurations. In order to avoid approximate inference, in this chapter, we introduce yet another architecture of the irregular tree, where the R' are eliminated, and where the constraints on the tree structure are directly modeled in the distribution of connectivity Z.

3.1 Model Specification

Similar to the model specification in the previous chapter, we introduce two architectures: one with observables only at the leaf level, and the other with observables propagated to higher levels. The main difference from the architectures IT_V and IT_{V^0} is that node positions are identical to those of the quad-tree. Therefore, we refer to the architectures presented in this chapter as irregular quad trees IQT_V and IQT_{V^0}.

The irregular quad tree is a directed acyclic graph with nodes in set V, organized in hierarchical levels, V^ℓ, $\ell{=}\{0,1,...,L\}$, where V^0 denotes the leaf level. The layout of nodes is identical to that of the quad-tree, modeling for example the dyadic pyramid of wavelet coefficients, such that the number of nodes at level ℓ can be computed as $|V^\ell|{=}|V^{\ell-1}|/4{=}...{=}|V^0|/4^\ell$. Unlike for position-encoding dynamic trees [48], we assume that nodes are fixed at locations of the corresponding quad-tree. Consequently, irregular model structure is achieved only through establishing arbitrary connections between nodes. Connections are established under the constraint that a node at level ℓ can become a root or it can connect only to the nodes at the next $\ell{+}1$ level. The network connectivity is represented by a random matrix, Z, where entry z_{ij} is an indicator random variable, such that $z_{ij}{=}1$ if $i{\in}V^{\ell}$ and $j{\in}V^{\ell+1}$ are connected. Z contains an additional zero ("root") column, where entries $z_{i0}{=}1$ if i is a root node. Each node can have only one parent, or can be a root. Note that due to the distribution over connections, after estimation of Z, for a given image, in IQT$_V$, some nodes may remain without children.

Each node i is characterized by an image-class random variable, x_i, which can take values in a finite class set C. Given Z, the label x_i of node i is conditioned on x_j of its parent j as $P(x_i|x_j, z_{ij}{=}1)$. The joint probability of image-class variables $X{=}\{x_i\}$, $\forall i{\in}V$, is given by

$$P(X|Z){=}\textstyle\prod_{\ell=0}^{L}\prod_{i\in V^{\ell}} P(x_i|x_j, z_{ij}{=}1), \tag{3.1}$$

where for roots we use priors $P(x_i)$. We assume that the conditional probability tables $P(x_i|x_j, z_{ij}{=}1)$ are equal for all the nodes at all levels, as in [33]. Such a unique conditional probability table is denoted as Φ.

Next, we assume that observables $\boldsymbol{y}_i$ are conditionally independent given the corresponding x_i:

$$P(Y|X) \;=\; \textstyle\prod_{i\in V} P(\boldsymbol{y}_i|x_i)\;, \tag{3.2}$$

$$P(\boldsymbol{y}_i|x_i{=}k) \;=\; \textstyle\sum_{g=1}^{G}\pi_k(g)N(\boldsymbol{y}_i;\boldsymbol{\nu}_k(g),\Xi_k(g))\;, \tag{3.3}$$

where for IQT$_{V^0}$ instead of V we write V^0 in Eq. (3.2). $P(\boldsymbol{y}_i|x_i{=}k)$, $k{\in}M$, is modeled as a mixture of Gaussians. The Gaussian-mixture parameters can be grouped in $\theta{=}\{\pi_k(g), \boldsymbol{\nu}_k(g), \Xi_k(g), G_k\}$, $\forall k{\in}M$.

Finally, we specify the connectivity distribution. In the previous chapter, it is defIQTined as the prior $P(Z){=}\prod_{i,j\in V} P(z_{ij}{=}1)$, and then the constraint on possible tree structures is imposed through introducing an additional set of random variables – namely, random node positions R. The main purpose of the R's is to provide for the mechanism that the connections between close nodes are favored. That approach has two major disadvantages. First, the additional R variables render the exact inference of the dynamic tree intractable, enforcing the use of approximate inference methods (variational approximation). Second, the decision if nodes i and j should be connected is not informed on the actual values of x_i and x_j. To improve upon the model formulation of the previous chapter, we seek to eliminate the R's, and to incorporate the information on image-class labels and node positions in the connectivity distribution. We reason that connections between parents and children, whose relative distance is small, should be favored over those that are far apart. At the same time, we seek to establish a mechanism that groups nodes belonging to the same image class, and separates those assigned to different classes.

Let us first examine relative distances between nodes. Due to symmetry of the node layout (equal to that of the quad-tree), we divide the set of all candidate parents j into classes of equidistance from child i, as depicted in Fig. 3–1. We specify that relative distances can take integer values $d_{ij}{=}\{0,1,2,...,d_i^{\max}\}$, where if i is a root $n_{i0}{\triangleq}0$. Note that $d_i^{\max}$ values vary for different positions of i at one level, as well as for different levels to which i belongs.

Given X, we specify the conditional connectivity distribution as

$$P(Z|X) = \prod_{\ell=0}^{L} \prod_{(i,j)\in V^\ell\times\{0,V^{\ell+1}\}} P(z_{ij}{=}1|x_i,x_j), \tag{3.4}$$

$$P(z_{ij}{=}1|x_i,x_j) = \kappa \begin{cases} p_i & , \ i \text{ is a root,} \\ p_i(1-p_i)^{d_{ij}} & , \text{ if } x_i{=}x_j, \\ p_i(1-p_i)^{d_i^{\max}-d_{ij}} & , \text{ if } x_i{\neq}x_j, \end{cases} \tag{3.5}$$

$$\text{subject to} \quad \sum_{j\in\{0,V^{\ell+1}\}} P(z_{ij}{=}1|x_i,x_j) = 1, \tag{3.6}$$

where κ is a normalizing constant, and p_i is the parameter of the geometric distribution. From Eq. (3.5), we observe that when $x_i{=}x_j$, $P(z_{ij}{=}1|x_i,x_j)$ decreases as d_{ij} becomes larger, while when $x_i{\neq}x_j$, $P(z_{ij}{=}1|x_i,x_j)$ increases for greater distances d_{ij}. Hence, the form of $P(z_{ij}{=}1|x_i,x_j)$, given by Eq. (3.5), satisfies the aforementioned desirable properties. To avoid overfitting, we assume that p_i is equal for all nodes i at the same level. The parameters of $P(Z|X)$ can be grouped in the parameter set $\Psi{=}\{p_i\}$, $\forall i{\in}V$.

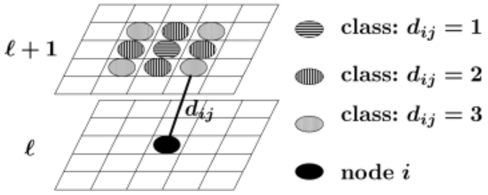


Figure 3–1: Classes of candidate parents j that are characterized by a unique relative distance d_{ij} from child i.

The introduced parameters of the model can be grouped in the parameter set $\Theta=\{\Phi,\theta,\Psi\}$. In the next section we explain how to infer the “best” configuration of Z and X from the observed image data Y, provided that Θ is known.

3.2 Inference of the Irregular Tree with Fixed Node Positions

The standard Bayesian formulation of the inference problem consists in minimizing the expectation of some cost function $\mathcal{C}$, given the data

$$(\hat{Z},\hat{X})= \arg\min\nolimits_{Z,X} \mathbb{E}\{\mathcal{C}((Z,X),(Z',X'))|Y,\Theta\} \ , \tag{3.7}$$

where $\mathcal{C}$ penalizes the discrepancy between the estimated configuration (Z,X) and the true one (Z',X'). We propose the following cost function:

$$\begin{aligned}\mathcal{C}((Z,X),(Z',X')) &= \mathcal{C}(X,X')+\mathcal{C}(Z,Z') \ , & (3.8)\\ &= \sum_{\ell=0}^{L-1}\sum_{i\in V^\ell}[1-\delta(x_i-x'_i)] + \sum_{\ell=0}^{L-1}\sum_{(i,j)\in V^\ell\times\{0,V^{\ell+1}\}}[1-\delta(z_{ij}-z'_{ij})], & (3.9)\end{aligned}$$

where $'$ stands for true values, and $\delta(\cdot)$ is the Kronecker delta function. From Eq. (3.9), the resulting Bayesian estimator of X is

$$\forall i\in V, \quad \hat{x}_i= \arg\max\nolimits_{x_i\in C} P(x_i|Z,Y). \tag{3.10}$$

Next, given the constraints on connections in the irregular tree, we derive that minimizing $\mathbb{E}\{\mathcal{C}(Z,Z')|Y,\Theta\}$ is equivalent to finding a set of optimal parents $\hat{j}$ such that

$$(\forall\ell)(\forall i\in V^\ell)(Z_{\cdot i}\neq 0) \ \ \hat{j}= \arg\max\nolimits_{j\in\{0,V^{\ell+1}\}} P(z_{ij}|x_i,x_j) \ , \ \text{ for IQT}_{V^0} \ , \tag{3.11a}$$

$$(\forall\ell)(\forall i\in V^\ell) \ \ \hat{j}= \arg\max\nolimits_{j\in\{0,V^{\ell+1}\}} P(z_{ij}|x_i,x_j) \ , \ \text{ for IQT}_{V} \ , \tag{3.11b}$$

where $Z_{\cdot i}$ is the i-th column of Z, and $Z_{\cdot i}{\neq}0$ represents the event "node i has children", that is, "node i is included in the irregular-tree structure." The global solution to Eq. (3.11a) is an open problem in many research areas. We propose a stage-wise optimization, where, as we move upwards, starting from the leaf level $\ell{=}\{0,1,...,L\}$, we include in the tree structure optimal parents at $V^{\ell+1}$ according to

$$(\forall i{\in}V^{\ell})(\hat{Z}_{\cdot i}{\neq}0)\ \hat{j}{=}\arg\max\nolimits_{j\in\{0,V^{\ell+1}\}} P(z_{ij}{=}1|x_i,x_j), \tag{3.12}$$

where $\hat{Z}_{\cdot i}{\neq}0$ denotes an estimate that i has already been included in the tree structure when optimizing the previous level V^{ℓ}.

By using the results in Eqs. (3.10) and (3.12), we specify the inference algorithm for the irregular quad tree, which is summarized in Fig. 3–2. In a recursive step t, we first assume that estimate $Z(t{-}1)$ of the previous step $t{-}1$ is known and then derive estimate $X(t)$ using Eq. (3.10); then, substituting $X(t)$ in Eq. (3.12) we derive estimate $Z(t)$. We consider the algorithm converged if $P(Y,X|Z)$ does not vary more than some threshold ε for N_{ε} consecutive iteration steps t. In our experiments, we set $\varepsilon = 0.01$ and $N_{\varepsilon} = 10$.

Steps 2 and 6 in the algorithm can be interpreted as inference of $\hat{X}$ given Y for a fixed-structure tree. In particular, for Step 2, where the initial structure is the quad-tree, we can use the standard inference on quad-trees, where, essentially, belief messages are propagated in only two sweeps up and down the tree [29,31,33]. For Step 6, the irregular tree represents a forest of subtrees, which also have fixed, though irregular, structure; therefore, we can use the very same tree-inference algorithm for each of the subtrees. For completeness, in Appendix B, we present the two-pass maximum posterior marginal estimation of X proposed by Laferte et al. [33].

3.3 Learning Parameters of the Irregular Tree with Fixed Node Positions

Analogous to the learning algorithm discussed in the previous chapter, the parameters of the irregular tree with fixed node positions can be learned by using the standard ML optimization. Here, we assume that N, independently generated, training images, with observables $\{Y^n\}$, $n{=}1,...,N$, are given. As explained before, configurations of latent variables $\{(Z^n,X^n)\}$ must be estimated.

```
Inference Algorithm
(1) $t = 0$; initialize irregular-tree structure $Z(0)$ to quad-tree;
(2) Compute $\forall i{\in}V,\ x_i(0){=}\arg\max_{x_i\in C} P(x_i|Z(0),Y)$;
(3) repeat
    (4) $t = t + 1$;
    (5) Compute in bottom-up pass for $\ell{=}0,1,...,L$
        for IQT$_{V^0}$: $(\forall i{\in}V^\ell)(\hat{Z}_{\cdot i}{\neq}0)$ $\hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}} P(z_{ij}{=}1|x_i,x_j)$;
        for IQT$_V$: $(\forall i{\in}V^\ell)$ $\hat{j}{=}\arg\max_{j\in\{0,V^{\ell+1}\}} P(z_{ij}{=}1|x_i,x_j)$;
    (6) Compute $\forall i{\in}V,\ x_i(t){=}\arg\max_{x_i\in C} P(x_i|Z(t),Y)$;
    (7) $\hat{X}{=}X(t)$; $\hat{Z}{=}Z(t)$;
(8) until $|\frac{P(Y,\hat{X}|\hat{Z})-P(Y,X(t-1)|Z(t-1))}{P(Y,X(t-1)|Z(t-1))}|{<}\varepsilon$ for $N_\varepsilon$ consecutive iteration steps.
```

Figure 3–2: Inference of the irregular tree with fixed node positions, given observables Y and the model parameters Θ.

To this end, we propose an iterative learning procedure, where in step t we first assume that $\Theta(t){=}\{\Phi(t),\theta(t),\Psi(t)\}$ is given and then conduct inference for each training image, $n{=}1,...,N$,

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

as explained in Section 3.2. Once the estimates $\{(\hat{Z}^n,\hat{X}^n)\}$ are found, we apply standard ML optimization to compute $\Theta(t{+}1)$.

More specifically, suppose, in the learning step t, realizations of random variables $(Y^n,\hat{X}^n,\hat{Z}^n)$ are given for $n{=}1,...,N$. Then the parameters of Gaussian-mixture distributions, in step $t+1$, can be computed using the standard EM algorithm [56]:

$$\begin{aligned}
P(\omega_c(g)|y_i,x_i{=}c) &= \frac{P(y_i|x_i{=}c)\pi_c(g)}{\sum_{g=1}^{G_c}P(y_i|x_i{=}c)\pi_c(g)}, && (3.13)\\
\hat{\pi}_c(g) &= \frac{1}{n_c}\sum_{i=1}^{n_c}P(\omega_c(g)|y_i,\hat{x}_i{=}c), && (3.14)\\
\hat{\nu}_c(g) &= \frac{\sum_{i=1}^{n_c}y_iP(\omega_c(g)|y_i,\hat{x}_i{=}c)}{\sum_{i=1}^{n_c}P(\omega_c(g)|y_i,\hat{x}_i{=}c)}, && (3.15)\\
\hat{\Xi}_c(g) &= \frac{\sum_{i=1}^{n_c}(y_i-\hat{\nu}_c(g))(y_i-\hat{\nu}_c(g))^{\mathrm{T}}P(\omega_c(g)|y_i,\hat{x}_i{=}c)}{\sum_{i=1}^{n_c}P(\omega_c(g)|y_i,\hat{x}_i{=}c)}, && (3.16)
\end{aligned}$$

where n_c is the total number of all the nodes over N training images that are classified as class c. To compute $P(\omega_c(g)|y_i,x_i{=}c)$ in Eq. (3.13), we use Gaussian-mixture parameters from the previous learning step t. For all classes we set $G_c{=}3$.

Next, we explain how to learn the parameters of the connectivity distribution, $\Psi(t{+}1) = \{p_i(t{+}1)\}_{i\in V}$, by using the ML principle:

$$\Psi(t{+}1) = \arg\max_{\Psi} \prod_{n=1}^{N} P(\hat{Z}^n|\hat{X}^n, \Psi(t{-}1)). \tag{3.17}$$

Here, we consider two cases for IQT$_V$ and IQT$_{V^0}$ models. Recall that parameters p_i are equal for all nodes i at the same level ℓ. Given the estimates $\{(\hat{Z}^n, \hat{X}^n)\}$, for each training image $n{=}1, ..., N$, from Eqs. (3.5) and (3.17), we derive for IQT$_V$:

$$\hat{p}(\ell){=}\frac{N|V^\ell|}{\displaystyle\sum_{n=1}^{N}\sum_{i\in V^\ell}\left[1 + \mathrm{I}(\hat{x}_i^n{=}\hat{x}_j^n)d_{ij}^n + \mathrm{I}(\hat{x}_i^n{\neq}\hat{x}_j^n)(d_i^{\max}{-}d_{ij}^n)\right]}, \tag{3.18}$$

where I$(\cdot)$ is an indicator function, j is an estimated parent of node i, d_{ij}^n denotes the relative distance assigned to the estimated connection $\hat{z}_{ij}^n{=}1$.

For IQT$_{V^0}$, given the estimates $\{(\hat{Z}^n, \hat{X}^n)\}$, for each training image $n{=}1, ..., N$, we analyze the set of nodes $i{\in}V^\ell$ included in the corresponding irregular tree, i.e., $\hat{Z}_{\cdot i}^n{\neq}0$. Thus, from Eqs. (3.5), and (3.17), we derive:

$$\hat{p}_i(\ell){=}\frac{\displaystyle\sum_{n=1}^{N}\sum_{i\in V^\ell}\mathrm{I}(\hat{Z}_{\cdot i}^n \neq 0)}{\displaystyle\sum_{n=1}^{N}\sum_{i\in V^\ell}\mathrm{I}(\hat{Z}_{\cdot i}^n \neq 0)\left[1 + \mathrm{I}(\hat{x}_i^n{=}\hat{x}_j^n)d_{ij}^n + \mathrm{I}(\hat{x}_i^n{\neq}\hat{x}_j^n)(d_i^{\max}{-}d_{ij}^n)\right]}, \tag{3.19}$$

where I$(\cdot)$ is an indicator function, j is an estimated parent of node i, d_{ij}^n denotes the relative distance assigned to the estimated connection $\hat{z}_{ij}^n{=}1$.

Finally, to learn the conditional probability table Φ, we use the standard EM algorithm on fixed-structure trees, thoroughly discussed in [33]. Note that to obtain the estimates $\{(\hat{Z}^n, \hat{X}^n)\}$, for each training image $n{=}1, ..., N$, in the learning step t, we in fact have to conduct the MPM estimation, given in in Appendix B in Fig. B. By using already available $P(x_i, x_j|Y_{d(i)}^n, \hat{z}_{ij}^n{=}1)$ and $P(x_i|Y_{d(i)}^n)$, obtained for each image n as in Fig B, we derive

$$\hat{\Phi} = \frac{1}{N}\sum_{n=1}^{N}\frac{\sum_{i\in V} P(x_i, x_j|Y_{d(i)}^n, \hat{z}_{ij}^n{=}1)}{\sum_{i\in V} P(x_j|Y_{d(i)}^n)}\ . \tag{3.20}$$

The overall learning procedure is summarized in Fig. 3–3.

Learning Algorithm

```
(1) $t = 0$; initialize $\Theta(0)=\{\Phi(0), \theta(0), \Psi(0)\}$;
(2) Estimate for $n = 1, ..., N$
    $(\hat{Z}^n, \hat{X}^n) = \arg\min_{Z,X} \mathbb{E}\{\mathcal{C}((Z, X), (Z', X'))|Y^n, \Theta(0)\}$;
(3) repeat
    (4) $t = t + 1$;
    (5) Compute:
            $\theta(t)$ as in Eqs. (3.13)–(3.16);
            $p(\ell; t)$, for IQT$_V$ as in Eq. (3.18); for IQT$_{V^0}$ as in Eq. (3.19);
            $\Phi(t)$, as in Eq. (3.20);
    (6) Estimate for $n = 1, ..., N$
            $(\hat{Z}^n, \hat{X}^n) = \arg\min_{Z,X} \mathbb{E}\{\mathcal{C}((Z, X), (Z', X'))|Y^n, \Theta(t)\}$,
        using the inference algorithm in Fig. 3–2;
    (7) $\Theta^* = \Theta(t)$;
(8) until $(\forall n)$ $|\frac{P(Y^n, \hat{X}^n|\hat{Z}^n, \Theta^*) - P(Y^n, \hat{X}^n|\hat{Z}^n, \Theta(t-1))}{P(Y^n, \hat{X}^n|\hat{Z}^n, \Theta(t-1))}| < \varepsilon$ for $N_\varepsilon$ consecutive iteration steps.
```

Figure 3–3: Algorithm for learning the parameters of the irregular tree; for notational simplicity, in Step (8) we do not indicate the different estimates of $(\hat{Z}^n, \hat{X}^n)$ for Θ^* and $\Theta(t-1)$.

Once Θ^* is learned, we can localize, detect and recognize objects in the image, by conducting the inference algorithm, presented in Fig. 3–2.

CHAPTER 4
COGNITIVE ANALYSIS OF OBJECT PARTS

Inference of hidden variables $(\hat{Z}, \hat{X})$, can be viewed as building a forest of subtrees, each segmenting an image into arbitrary (not necessarily contiguous) regions, which we interpret as objects. Since, each root determines a subtree, whose leaf nodes form a detected object, we assign physical meaning to roots by assuming they represent whole objects. Moreover, each descendant of the root can be viewed as the root of another subtree, whose leaf nodes cover only a part of the object. Hence, we say that roots' descendants represent object parts at various scales.

Strategies for recognizing detected objects naturally arise from a particular interpretation of the tree/sub-tree structure. Below, we make a distinction between two such strategies. The analysis of image regions under the roots leads to the *whole-object recognition strategy*, while the analysis of image regions determined by roots' descendants constitutes the *object-part recognition strategy.* For both approaches, final recognition is conducted by majority voting over MAP labels, $\hat{x}_i$, of leaf nodes.[1]

The reason for analyzing smaller image regions than those under the roots stems from our hypothesis that the information of fine-scale object details may prove critical for the recognition of an object as a whole in scenes with occlusions. To reduce the complexity of interpreting all detected object sub-parts, we propose to analyze the *significance* of object components (i.e., irregular-tree nodes) with respect to recognition of objects as a whole.

[1] The literature offers various strategies that outperform majority-voting classification (e.g., multiscale Bayesian classification [29], and multiscale Viterbi classification [32]); however, they do not account explicitly for occlusions, and, as such, do not significantly outperform majority voting for scenes with occluded objects.

4.1 Measuring Significance of Object Parts

We hypothesize that the significance of object parts with respect to object recognition depends on both local, innate object properties and global scene properties. While innate properties represent characteristic object features, which differentiate one object from another, global scene properties describe interdependencies of object parts in the overall image composition. It is necessary to account for both local and global cues, as the most conspicuous object component need not necessarily be the most significant for that object's recognition in the presence of alike objects.

The analysis of innate object properties is handled through inference of the irregular tree, where, for a given image, we compute $P(x_i|\hat{Z},Y)$, $\forall i{\in}V$, as explained in Chapters 2 and 3. To account for the influence of global scene properties, for each node i, we compute Shanon's entropy over the set of image classes, M, as

$$(\forall i{\in}\, V)(\hat{z}_i{\neq}0)\ \ H_i{=}-\sum_{x_i\in M} P(x_i|\hat{Z},Y)\log P(x_i|\hat{Z},Y)\ . \tag{4.1}$$

Since node i represents an object part, we define H_i as a measure of significance of that object part. Note that a node with small entropy is characterized by a "peaky" distribution $P(x_i|\hat{Z},Y)$ with the maximum, say, at $x_i = k \in M$. This indicates that the error of classification will be small when i is labeled as class c. Recall that during inference, the belief message of i is propagated down the subtree in belief propagation [33], which is likely to render i's descendants with small entropies, as well. Thus, the classification error of the whole region of leaf nodes under i is likely to be small, when compared to some other image region under, say, node j such that $H_j{>}H_i$. Consequently, i is more "significant" for recognition of class k than node j. In brief, the most significant object part has the smallest entropy over all nodes in a given sub-tree $\mathcal{T}$:

$$i^* = \max_{i\in\mathcal{T}} H_i\ . \tag{4.2}$$

In Figs. 4–1 and 4–2, we illustrate the most significant object part under each root, where entropy is computed over seven and six image classes, shown in Figs. 4–1(top) and 4–2(top), respectively. The experiment is conducted as explained in Chapter 2, using the

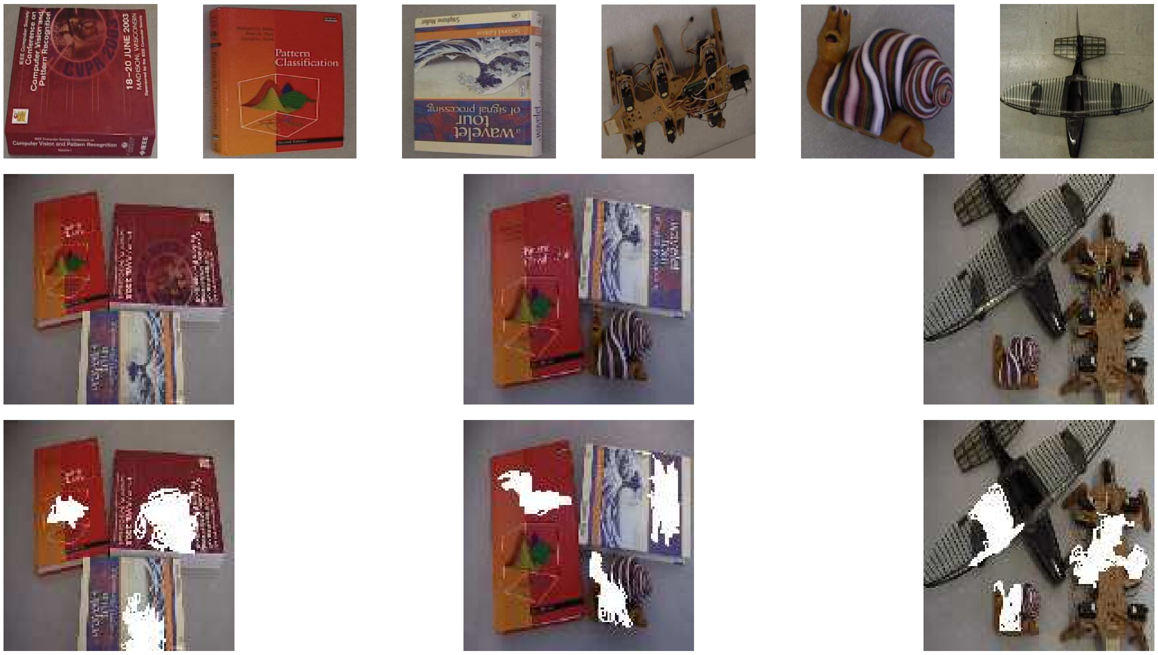


Figure 4–1: For each subtree of IT_V, representing an object in the 128×128 image, a node i^* is found with the highest entropy for $|M| = 6 + 1 = 7$ possible image classes (top row). Bright pixels are descendants of i^* at the leaf level and indicate the object part represented by i^*.

irregular tree with random node positions, and observables at all levels (IT_V). Details on computing observables Y in this experiment are explained in Chapter 5. Note that for different scenes different object parts are established as the most significant with respect to the entropy measure.

4.2 Combining Object-Part Recognition Results

Once nodes are ranked with respect to the entropy measure, we are in a position to devise a criterion to optimally combine this information toward ultimate object recognition. Herewith, we propose a simple greedy algorithm, which, nonetheless, shows remarkable improvements in performance over the whole-object recognition approach.

Under each root, we first select the descendant node with the smallest entropy. Each selected node determines a subtree, whose leaf nodes form an object part. Then, we conduct majority voting over these selected image regions. In the second round, we select under each root the descendant node with the smallest entropy, such that it does not belong to any of the subtrees selected in the first round. Now, these nodes determine new subtrees, whose leaf nodes form object parts that do not overlap with the selected image regions in

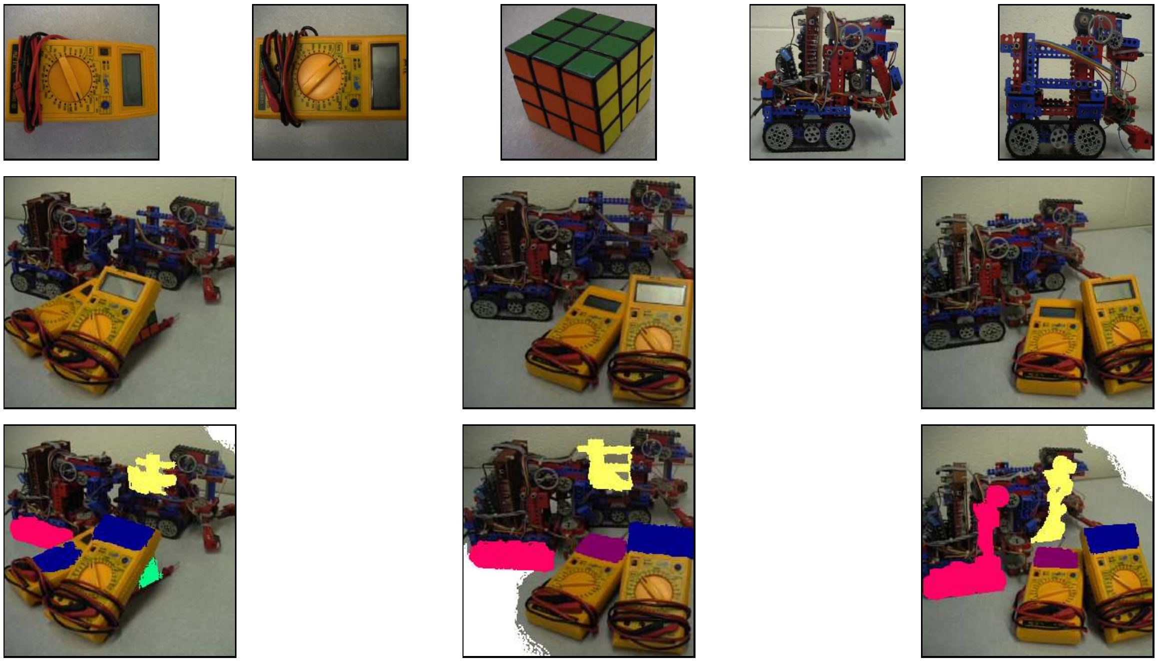


Figure 4–2: For each subtree of IT_V, representing an object in the 256×256 image, a node i^* is found with the highest entropy for $|M| = 5 + 1 = 6$ possible image classes (top row). Bright pixels are descendants of i^* at the leaf level and indicate the object part represented by i^*; the images represent the same scene viewed from three different angles; the most significant object parts differ over various scenes.

the first round. Then, we conduct majority voting over the newly selected image regions. This procedure is repeated until we exhaustively cover all the pixels in the image. This stage-wise majority voting over non-overlapping image regions constitutes the final step in the object-part recognition strategy (see Fig. 1–3).

CHAPTER 5
FEATURE EXTRACTION

In Chapters 2 and 3, we have introduced four architectures of the irregular tree, referred to as IT_V, IT_{V^0}, IQT_V, and IQT_{V^0}. To compute the observable (feature) random vectors Y's for these models, we account for both color and texture cues.

5.1 Texture

For the choice of texture-based features, we have considered several filtering, model-based and statistical methods for texture feature extraction. Our conclusion complies with the comparative study of Randen and Husoy [66] that for problems with many textures with subtle spectral differences, as in the case of our complex classes, it is reasonable to assume that the spectral decomposition by a filter bank yields consistently superior results over other texture analysis methods. Our experimental results also suggest that it is crucial to analyze both local as well as regional properties of texture. As such, we employ the wavelet transform, due to its inherent representation of texture at different scales and locations.

5.1.1 Wavelet Transform

Wavelet atom functions, being well localized both in space and frequency, retrieve texture information quite successfully [67]. The conventional discrete wavelet transform (DWT) may be regarded as equivalent to filtering the input signal with a bank of bandpass filters, whose impulse responses are all given by scaled versions of a mother wavelet. The scaling factor between adjacent filters is 2:1, leading to octave bandwidths and center frequencies that are one octave apart. The octave-band DWT is most efficiently implemented by the dyadic wavelet decomposition tree of Mallat [68], where wavelet coefficients of an image are obtained convolving every row and column with impulse responses of lowpass and highpass filters, as shown in Figure 5–1. Practically, coefficients of one scale are obtained convolving every second row and column from the previous finer scale. Thus, the filter output is a wavelet subimage that has four times less coefficients than the one at the

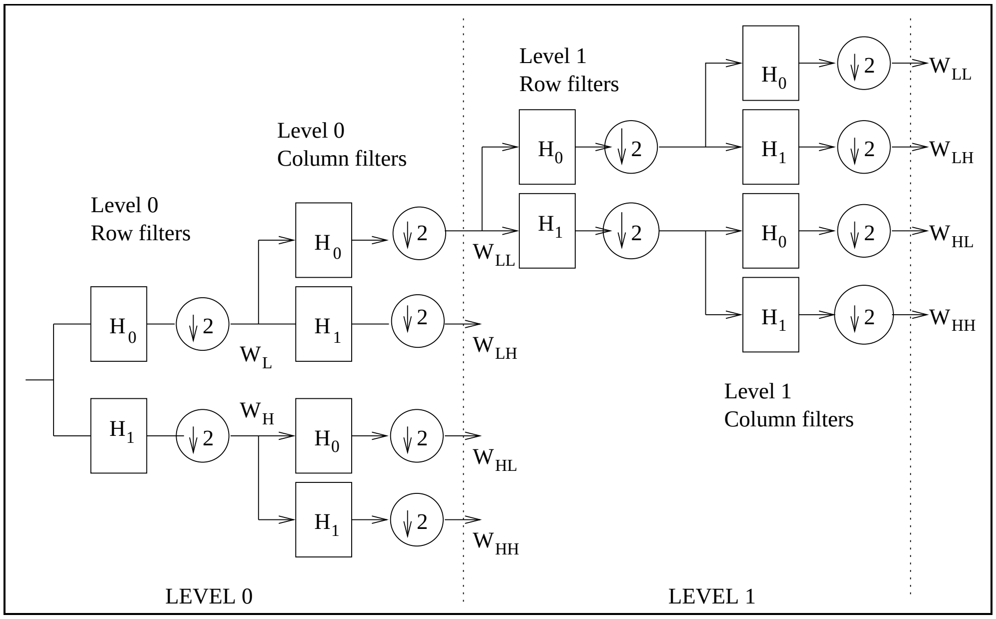


Figure 5–1: Two levels of the DWT of a two-dimensional signal.

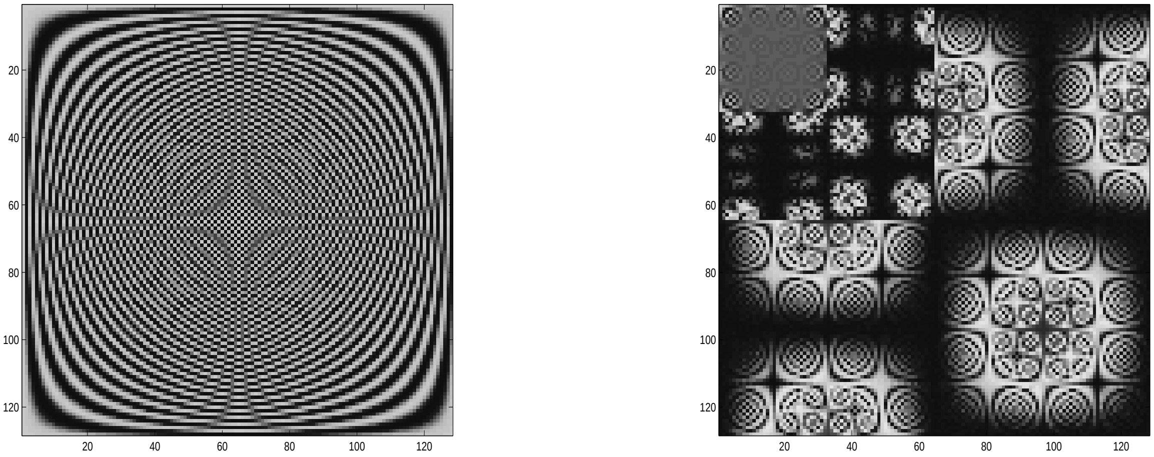


Figure 5–2: The original image (left) and its two-scale dyadic DWT (right).

previous scale. The lowpass filter is denoted with H_0 and the highpass filter with H_1. The wavelet coefficients W have in index L denoting lowpass output and H for highpass output.

Separable filtering of rows and columns produces four subimages at each level, which can be arranged as shown in Figure 5–2. The same figure also illustrates well the directional selectivity of the DWT, because W_{LH}, W_{HL} , and W_{HH}, bandpass subimages can select horizontal, vertical and diagonal edges, respectively.

5.1.2 Wavelet Properties

The following properties of the DWT have made wavelet-based image processing very attractive in recent years [30, 67, 69]:

1. locality: each wavelet coefficient represents local image content in space and frequency, because wavelets are well localized simultaneously in space and frequency
2. multi-resolution: DWT represents an image at different scales of resolution in space domain (i.e., in frequency domain); regions of analysis at one scale are divided up into four smaller regions at the next finer scale (Fig. 5–2)
3. edge detector: edges of an image are represented by large wavelet coefficients at the corresponding locations
4. energy compression: wavelet coefficients are large only if edges are present within the support of the wavelet, which means that the majority of wavelet coefficients have small values
5. decorrelation: wavelet coefficients are approximately decorrelated, since the scaled and shifted wavelets form orthonormal basis; dependencies among wavelet coefficients are predominantly local
6. clustering: if a particular wavelet coefficient is large/small, then the adjacent coefficients are very likely to also be large/small
7. persistence: large/small values of wavelet coefficients tend to propagate through scales
8. non-Gaussian marginal: wavelet coefficients have peaky and long-tailed marginal distributions; due to the energy compression property only a few wavelet coefficients have large values, therefore Gaussian distribution for an individual coefficient is a poor statistical model

It is also important to introduce shortcomings of the DWT. Discrete wavelet decompositions suffer from two main problems, which hamper their use for many applications, as follows [70]:

1. lack of shift invariance: small shifts in the input signal can cause major variations in the energy distribution of wavelet coefficients
2. poor directional selectivity: for some applications horizontal, vertical and diagonal selectivity is insufficient

When we analyze the Fourier spectrum of a signal, we expect the energy in each frequency bin to be invariant to any shifts of the input. Unfortunately, the DWT has a significant drawback that the energy distribution between various wavelet scales depends critically on the position of key features of the input signal, whereas ideally dependence

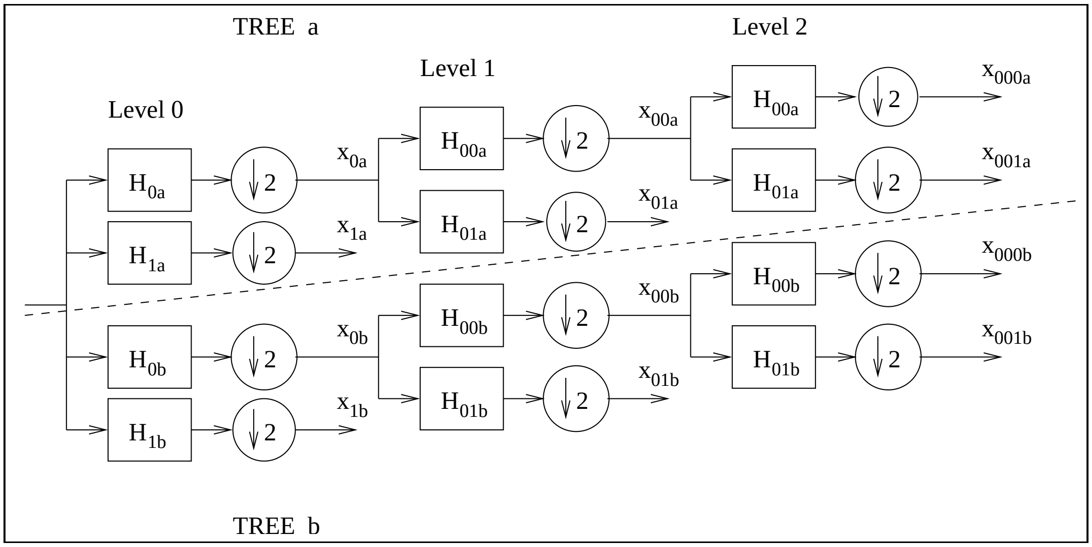


Figure 5–3: The Q-shift Dual-Tree CWT.

is on just the features themselves. Therefore, the real DWT is unlikely to give consistent results when used in texture analysis.

In literature, there are several approaches proposed to overcome this problem (e.g., Discrete Wavelet Frames [67, 71]), all increasing computational load with inevitable redundancy in the wavelet domain. In our opinion, the Complex Wavelet Transform (CWT) offers the best solution providing additional advantages, described in the following subsection.

5.1.3 Complex Wavelet Transform

The structure of the CWT is the same as in Figure 5–1, except that the CWT filters have complex coefficients and generate complex output. The output sampling rates are unchanged from the DWT, but each wavelet coefficient contains a real and imaginary part, thus a redundancy of 2:1 for one-dimensional signals is introduced. In our case, for two-dimensional signals, the redundancy becomes 4:1, because two adjacent quadrants of the spectrum are required to represent fully a real two-dimensional signal, adding an extra 2:1 factor. This is achieved by additional filtering with complex conjugates of either the row or column filters [70].

Despite its higher computational cost, we prefer the CWT over the DWT because of the CWT's following attractive properties. The CWT is shown to posses almost shift and rotational invariance, given suitably designed biorthogonal or orthogonal wavelet filters. We

Table 5–1: Coefficients of the filters used in the Q-shift DTCWT.

H_{13} (symmetric)	H_{19} (symmetric)	H_6
-0.0017581	-0.0000706	0.03616384
0	0	0
0.0222656	0.0013419	-0.08832942
-0.0468750	-0.0018834	0.23389032
-0.0482422	-0.0071568	0.76027237
0.2968750	0.0238560	0.58751830
0.5554688	0.0556431	0
0.2968750	-0.0516881	-0.11430184
-0.0482422	-0.2997576	0
$\vdots$	0.5594308	0
	-0.2997576	
	$\vdots$	

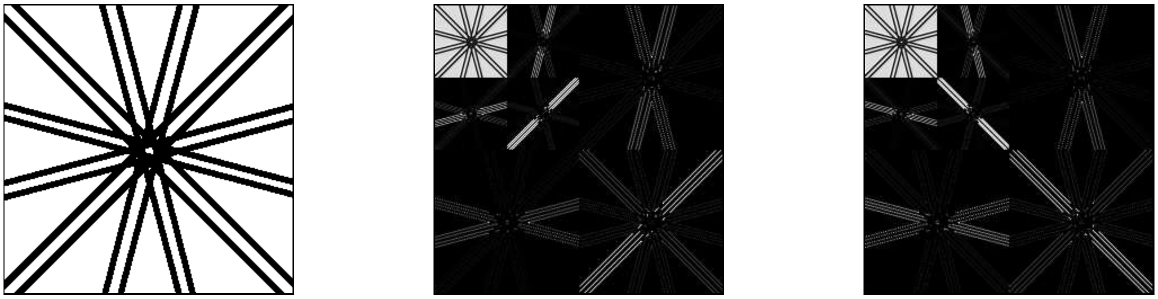


Figure 5–4: The CWT is strongly oriented at angles ±15°, ±45°, ±75°.

implement the Q-shift Dual-Tree CWT scheme, proposed by Kingsbury [72], as depicted in Figure 5–3. The figure shows the CWT of only one-dimensional signal x, for clarity. The output of the trees a and b can be viewed as real and imaginary parts of complex wavelet coefficients, respectively. Thus, to compute the CWT, we implement two real DWT's (see Fig. 5–1), obtaining a wavelet frame with redundancy two. As for the DWT, here, lowpass and highpass filters are denoted with 0 and 1 in index, respectively. The level 0 comprises odd-length filters $H_{0a}(z) = H_{0b}(z) = H_{13}(z)$ (13 taps) and $H_{1a}(z) = H_{1b}(z) = H_{19}(z)$ (19 taps). Levels above the level 0 consist of even-length filters $H_{00a}(z) = z^{-1}H_6(z^{-1})$, $H_{01a}(z) = H_6(-z)$, $H_{00b}(z) = H_6(z)$, $H_{01b}(z) = z^{-1}H_6(-z^{-1})$, where the impulse response of the filters H_{13}, H_{19} and H_6 is given in the table 5–1.

Aside from being shift invariant, the CWT is superior to the DWT in terms of directional selectivity, too. A two-dimensional CWT produces six bandpass subimages (analogous to the three subimages in the DWT) of complex coefficients at each level, which are strongly oriented at angles of $\pm 15°, \pm 45°, \pm 75°$, as illustrated in Figure 5–4.

Another advantageous property of the CWT exerts in the presence of noise. The phase and magnitude of the complex wavelet coefficients collaborate in a non trivial way to describe data [70]. The phase encodes the coherent (in space and scale) structure of an image, which is resilient to noise, and the magnitude captures the strength of local information that could be very susceptible to noise corruption. Hence, the phase of complex wavelet coefficients might be used as a principal clue for image denoising. However, our experimental results have shown that phase is not a good feature choice for sky/ground modeling. Therefore, we consider only magnitudes.

In summary, for texture analysis in IT_V and IQT_V, we choose the complex wavelet transform (CWT) applied to the intensity (gray-scale) image, due to its shift-invariant representation of texture at different scales, orientations and locations.

5.1.4 Difference-of-Gaussian Texture Extraction

In IT_{V^0} and IQT_{V^0}, observables are present only at the leaf level. Therefore, for these models, multiscale texture extraction is superfluous. Here, we compute the difference-of-Gaussian function convolved with the image as

$$D(x,y,k,\sigma){=}(G(x,y,k\sigma){-}G(x,y,\sigma)){*}I(x,y), \tag{5.1}$$

where x and y represent pixel coordinates, $G(x,y,\sigma)\triangleq\exp(-(x^2+y^2)/2\sigma^2)/2\pi\sigma^2$, and $I(x,y)$ is the intensity image. In addition to reduced computational complexity, as compared to the CWT, the function 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 and the Hessian [73,74]. We compute $D(x,y,k,\sigma)$ for three scales $k{=}\sqrt{2}, 2, \sqrt{8}$ and $\sigma = 2$.

5.2 Color

The color information in a video signal is usually encoded in the RGB color space. For color features, in all models, 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 IT_V and IQT_V, the Y's of higher-level nodes are computed as the mean of the r's and g's of their children nodes of the initial quad-tree structure. Each color observable is normalized to have zero mean and unit variance over the dataset.

In summary, the $\boldsymbol{y}$'s are 8 dimensional vectors for IT_V and IQT_V, and 5 dimensional vectors for IT_{V^0} and IQT_{V^0}.

CHAPTER 6
EXPERIMENTS AND DISCUSSION

We report experiments on image segmentation and classification for six sets of images. Dataset I comprises fifty, 64×64, simple-scene images with object appearances of 20 distinct objects shown in Fig. 6–1. Samples of dataset I are given in Figs. 6–2, 6–3, and 6–4. Dataset II contains 120, 128×128, complex-scene images with partially occluded object appearances of the same 20 distinct objects as for dataset I images. Examples of dataset II are shown in Figs. 6–11, 6–12, 6–15. Note that objects appearing in datasets I and II are carefully chosen to test if irregular trees are expressive enough to capture very small variations in appearances of some classes (e.g., two different types of cans in Fig. 6–1), as well as to encode large differences among some other classes (e.g., wiry-featured robot and books in Fig. 6–1).

Next, dataset III contains fifty, 128×128, natural-scene images, samples of which are shown in Figs. 6–5 and 6–6.

For dataset IV we choose sixty, 128 × 128 images from a database that is publicly available at the Computer Vision Home Page. Dataset IV contains a video sequence of two people approaching each other, who wear alike shirts, but different pants, as illustrated in Fig. 6–16. The sequence is interesting, because the most significant "object" parts for differentiating between the two persons (i.e., pants) get occluded. Moreover, the images represent scenes with clutter, where recognition of partially occluded, similar-in-appearance people becomes harder. Together with the two persons, there are 12 possible image classes appearing in dataset II, as depicted in Fig. 6–16a. Here, each image is treated separately, without making use of the fact that the background scene does not change in the video sequence.

Further, dataset V consists of sixty, 256 × 256 images, typical samples of which are shown in Figs. 6–17b. The images in dataset V represent the video sequence of a complex scene, which is observed from different view points by moving a camera horizontally clockwise. Together with the background, there are 6 possible image classes, as depicted in Figs. 6–17a.

Finally, dataset VI consists of sixty, 256×256 natural-scene images, samples of which are shown in Figs. 6–18. The images in dataset VI represent the video sequence of a row of houses, which is observed from different view points. The houses are very similar in appearance, so that the recognition task becomes very difficult, when details differentiating one house from another are occluded. There are 8 possible image classes: 4 different houses, *sky*, *road*, *grass*, and *tree*, as marked with different colors in Figs. 6–18.

All datasets are divided into training and test sets by random selection of images, such that 2/3 are used for training and 1/3 for testing. Ground truth for each image is determined through hand-labeling of pixels.

6.1 Unsupervised Image Segmentation Tests

We first report experiments on unsupervised image segmentation using IT_{V^0} and IT_V. Irregular-tree based image segmentation is tested on datasets I and III, and conducted by the algorithm given in Fig. 2–4. Since in unsupervised settings the parameters of the model are not known, we initialize them as discussed in the initialization step of the learning algorithm in Section 2.5. After Bayesian estimation of the irregular tree, each node defines one image region composed of those leaf nodes (pixels) that are that node's descendants. Results presented in Figs. 6–2, 6–3, 6–4, 6–5, and 6–6 suggest that irregular trees are able to parse images into "meaningful" parts by assigning one subtree per "object" in the image. Moreover, from Figs. 6–2 and 6–3, we also observe that irregular trees, inferred through SVA, preserve structure for objects across images subject to translation, rotation and scaling. In Fig. 6–2, note that the level-4 clustering for the larger-object scale in Fig. 6–2(top-right) corresponds to the level-3 clustering for the smaller-object scale in Fig. 6–2(bottom-center). In other words, as the object transitions through scales, the tree structure changes by eliminating the lowest-level layer, while the higher-order structure remains intact.

We also note that the estimated positions of higher-level hidden variables in IT_{V^0} and IT_V are very close to the center of mass of object parts, as well as of whole objects. We compute the error of estimated root-node positions $\hat{\boldsymbol{r}}$ as the distance from the actual center of mass $\boldsymbol{r}_{CM}$ of hand-labeled objects, $d_{err}{=}||\hat{\boldsymbol{r}}{-}\boldsymbol{r}_{CM}||$. Also, we compare our SVA inference

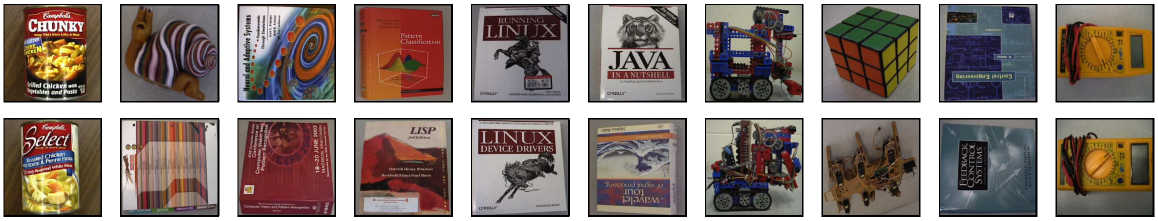


Figure 6–1: 20 image classes in type I and II datasets.

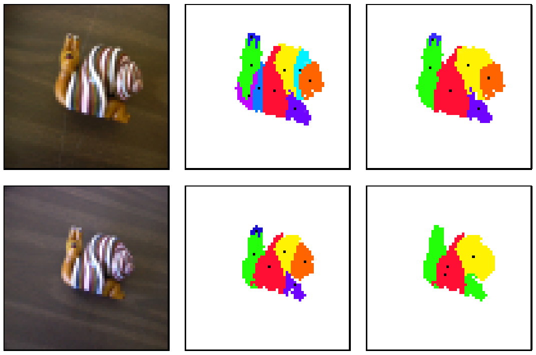


Figure 6–2: Image segmentation using IT_{V^0}: (left) dataset I images; (center) pixel clusters with the same parent at level ℓ=3; (right) pixel clusters with the same parent at level ℓ=4; points mark the position of parent nodes. Irregular-tree structure is preserved through scales.

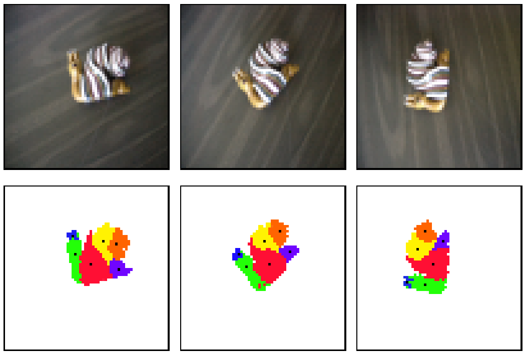


Figure 6–3: Image segmentation using IT$_{V^0}$: (top) dataset I images; (bottom) pixel clusters with the same parent at level 3. Irregular-tree structure is preserved over rotations.

algorithm with variational approximation (VA)[1] proposed by Storkey and Williams [48]. The averaged error values over the given test images for VA and SVA are reported in Table 6–1. We observe that the error significantly decreases as the image size increases, because in summing node positions over parent and children nodes, as in Eq. (2.16) and Eq. (2.17), more statistically significant information contributes to the position estimates. For example, d_{err}^{III}=6.18 for SVA is only 4.8% of the dataset-III image size, whereas d_{err}^{I}=4.23 for SVA is 6.6% of the dataset-I image size.

In Table 6–2, we report the percentage of erroneously grouped pixels, and, in Table 6–3, we report the object detection error, when compared to ground truth, averaged over each dataset. For estimating the object detection error, the following instances are counted

[1] Although the algorithm proposed by Storkey and Williams [48] is also structured variational approximation, to differentiate that method from ours, we slightly abuse the notation.

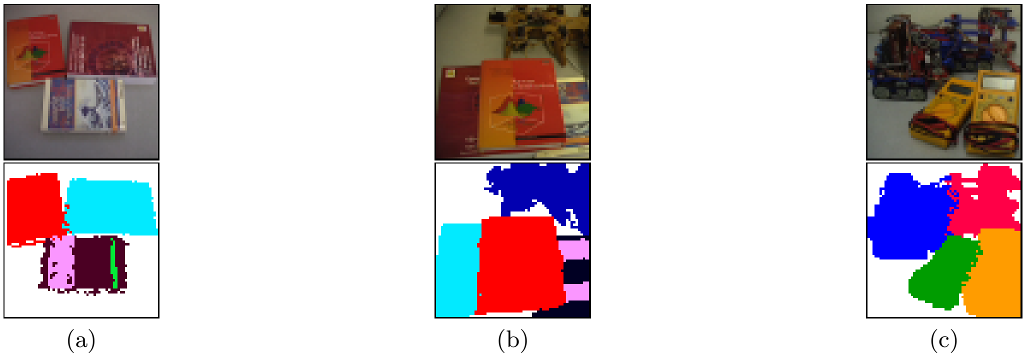

(a) (b) (c)

Figure 6–4: Image segmentation by irregular trees learned using SVA: (a)-(c) IT_{V^0} for dataset I images; all pixels labeled with the same color are descendants of a unique root.

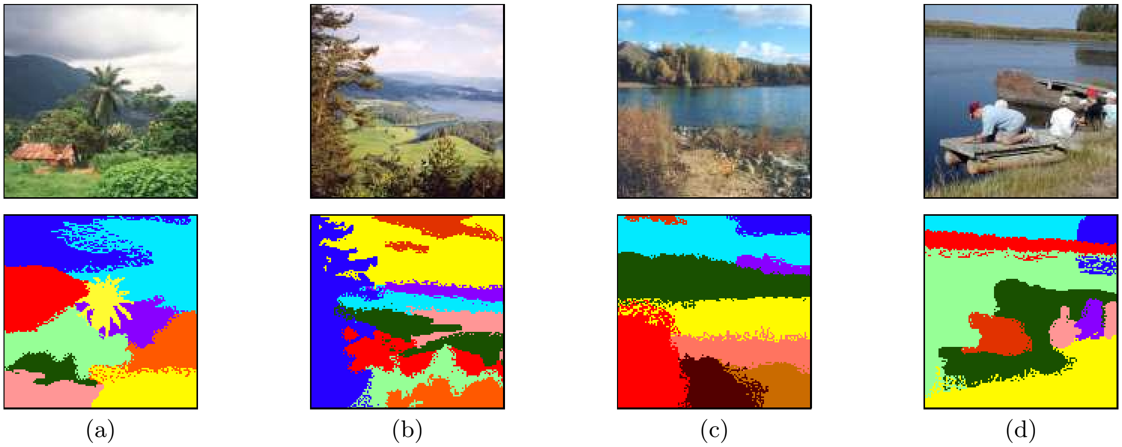

(a) (b) (c) (d)

Figure 6–5: Image segmentation by irregular trees learned using SVA: (a) IT_{V^0} for a dataset III image; (b)-(d) IT_V for dataset III images; all pixels labeled with the same color are descendants of a unique root.

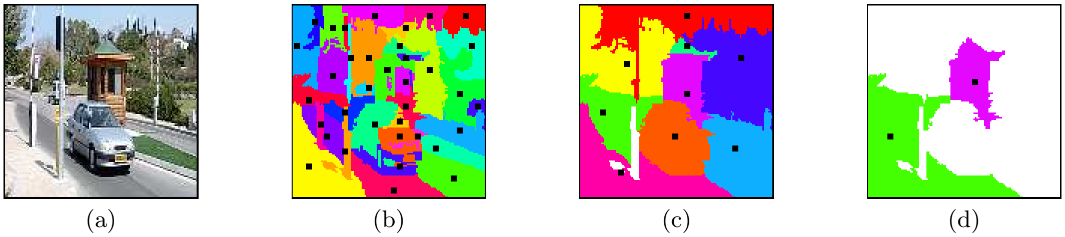

(a) (b) (c) (d)

Figure 6–6: Image segmentation using IT_V: (a) a dataset III image; (b)-(d) pixel clusters with the same parent at levels ℓ=3, 4, 5, respectively; white regions represent pixels already grouped by roots at the previous scale; points mark the position of parent nodes.

Table 6–1: Root-node distance error

	IT_{V^0}		IT_V	
dataset	VA	SVA	VA	SVA
I	6.32	4.61	6.14	4.23
III	9.15	6.87	8.99	6.18

Table 6–2: Pixel segmentation error

		datasets	
		I	III
IT$_{V^0}$	VA	7%	10%
	SVA	4%	9%
IT$_V$	VA	7%	11%
	SVA	4%	7%

Table 6–3: Object detection error

		datasets	
		I	III
IT$_{V^0}$	VA	4%	13%
	SVA	3%	10%
IT$_V$	VA	4%	10%
	SVA	2%	6%

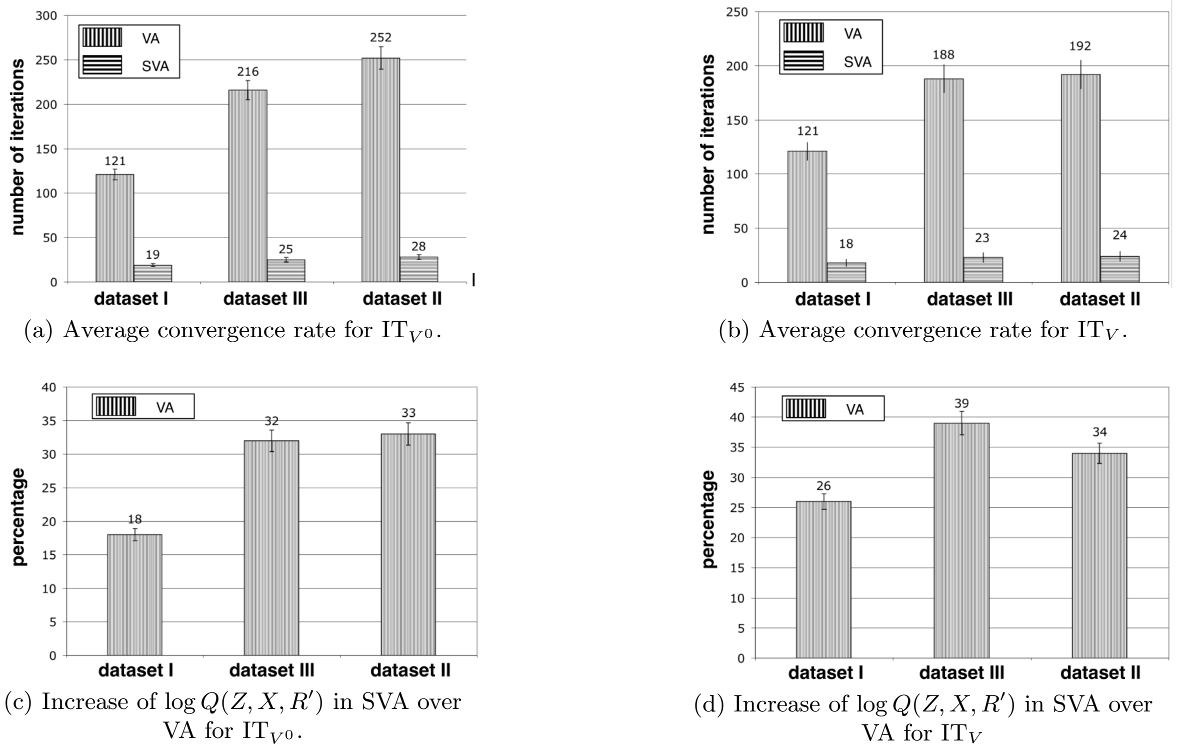
as error: (1) merging two distinct objects into one (i.e., failure to detect an object), and (2) segmenting an object into sub-regions that are not actual object parts. On the other hand, if an object is segmented into several "meaningful" sub-regions, verified by visual inspection, this type of error is not included. Overall, we observe that SVA outperforms VA for image segmentation using IT$_{V^0}$ and IT$_V$. Interestingly, the segmentation results for IT$_V$ models are only slightly better than for IT$_{V^0}$ models.

It should be emphasized that our experiments are carried out in an *unsupervised* setting, and, as such, cannot not be equitably evaluated against *supervised* object recognition results reported in the literature. Take, for instance, the segmentation in Fig. 6–5d, where two boys dressed in white clothes (i.e., two similar-looking objects) are merged into one subtree. Given the absence of prior knowledge, the ground-truth segmentation for this image is arbitrary, and the resulting segmentation ambiguous; nevertheless, we still count it towards the object-detection error percentages in Table 6–3.

Our claim that nodes at different levels of irregular trees 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.

6.2 Tests of Convergence

In this section, we report on the convergence properties of the inference algorithms for IT$_{V^0}$, IT$_V$, IQT$_{V^0}$, and IQT$_V$. First, we compare our SVA inference algorithm with variational approximation (VA) [48]. In Fig. 6–7a-b, we illustrate the convergence rate of computing $P(Z, X, R'|Y, R^0) \approx Q(Z, X, R')$ for SVA and VA, averaged over the given datasets. Numbers above bars represent the mean number of iteration steps it takes for the algorithm to converge. We consider the algorithm converged when $|Q(Z, X, R'; t +$



(a) Average convergence rate for IT_{V^0}.

(b) Average convergence rate for IT_{V}.

(c) Increase of $\log Q(Z,X,R')$ in SVA over VA for IT_{V^0}.

(d) Increase of $\log Q(Z,X,R')$ in SVA over VA for IT_{V}

Figure 6–7: Comparison of inference algorithms: (a)-(b) convergence rate averaged over the given datasets; (c)-(d) percentage increase in $\log Q(Z,X,R')$ computed in SVA over $\log Q(Z,X,R')$ computed in VA.

$1) - Q(Z,X,R';t)|/Q(Z,X,R';t)<\varepsilon$ for N_ε consecutive iteration steps t, where N_ε=10 and ε=0.01 (see Fig. 2–4, Step (11)). Overall, SVA converges in the fewest number of iterations. For example, the average number of iterations for SVA on dataset III is 25 and 23 for IT_{V^0} and IT_{V}, respectively, which takes approximately 6s and 5s on a Dual 2 GHz PowerPC G5. Here, the processing time also includes image-feature extraction.

For the same experiments, in Fig. 6–7c-d, we report the percentage increase in $\log Q(Z,X,R')$ computed using our SVA over $\log Q(Z,X,R')$ obtained by VA. We note that SVA results in larger approximate posteriors than VA. The larger $\log Q(Z,X,R')$ means that the assumed form of the approximate posterior distribution $Q(Z,X,R')$=$Q(Z)Q(X|Z)Q(R'|Z)$ more accurately represents underlying stochastic processes in the image than VA.

Now, we compare the convergence of the inference algorithm for IQT_{V^0} with SVA and VA for IT_{V^0}. For simplicity, we refer to the inference algorithm for the model IQT_{V^0}, also, as IQT_{V^0}, slightly abusing the notation. The parameters that control the convergence

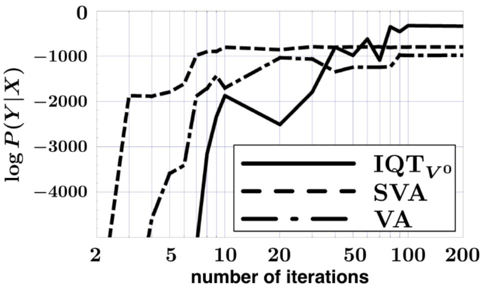


Figure 6–8: Typical convergence rate of the inference algorithm for IT$_{V^0}$ on the 128 × 128 dataset IV image in Fig. 6–16b; SVA and VA inference algorithms are conducted for IT$_{V^0}$ model.

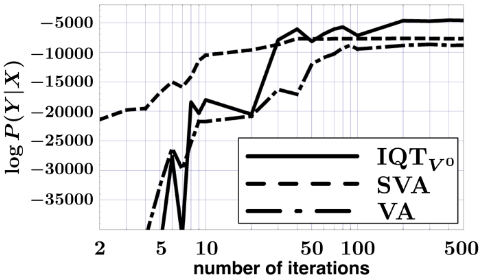


Figure 6–9: Typical convergence rate of the inference algorithm for IT$_{V^0}$ on the 256 × 256 dataset V image in Fig. 6–17b; SVA and VA inference algorithms are conducted for IT$_{V^0}$ model.

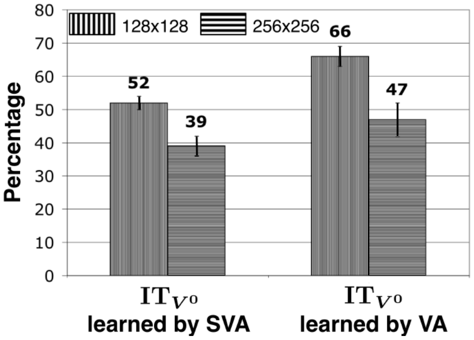


Figure 6–10: Percentage increase in log-likelihood $\log P(Y|X)$ of IQT$_{V^0}$ over $\log P(Y|X)$ of IT$_{V^0}$, after 500 and 200 iteration steps for datasets IV and V, respectively.

criterion for the inference algorithms of the three models are N=10, and ε=0.01. Figs. 6–8 and 6–9 illustrate typical examples of the convergence rate. We observe that the inference algorithm for IQT$_{V^0}$ converges slightly slower than SVA and VA for IT$_{V^0}$. The average number of iteration steps for IQT$_{V^0}$ is approximately 160 and 230, which takes 6s and 17s on a Dual 2 GHz PowerPC G5, for datasets IV and V, respectively.

The bar-chart in Fig. 6–10 shows the percentage $\frac{\log P_1 - \log P_2}{|\log P_1|}$, where $P_1{=}P(Y|X)$ is the likelihood of IT_{V^0}, and $P_2{=}P(Y|X)$ of IQT_{V^0}. We observe that $P(Y|X)$ of IQT_{V^0}, after the algorithm converged, is larger than $P(Y|X)$ of IT_{V^0}. The larger likelihood means that the model structure and inferred distributions more accurately represent underlying stochastic processes in the image.

6.3 Image Classification Tests

We compare classification performance of IT_{V^0} with that of the following statistical models: (1) Markov Random Field (MRF) [6], (2) Discriminative Random Field (DRF) [25], and (3) Tree-Structured Belief Network (TSBN) [29, 33]. These models are representatives of descriptive, discriminative and fixed-structure generative models, respectively. Below, we briefly explain the models.

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 [6]. The likelihoods $P(\boldsymbol{y}_i|x_i)$ are assumed conditionally independent given the labels. Thus, the posterior energy function is given by

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

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

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. (2.6), 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 Stan Li's book [6].

Next, the posterior energy function of the DRF is given by

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

where $A_i{=}\log\sigma(x_iW^T\boldsymbol{y}_i)$ and $I_{ij}{=}\beta_{DRF}(Kx_ix_j + (1{-}K)(2\sigma(x_ix_jV^T\boldsymbol{y}_i){-}1))$ 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 an object, we treat that object as a positive example, and all other objects 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, we refer the reader to the paper of Kumar and Hebert [25].

Further, TSBNs or quad-trees are defined to have the same number of nodes V and levels L as irregular trees. For both IT_{V^0} and TSBNs, we use the same image features. When we operate on wavelets, which is a multiscale image feature, we in fact propagate observables to higher levels. In this case, we refer to the counterpart of IT_V as TSBN↑. To learn the parameters of TSBN or TSBN↑, and to perform inference on a given image, we use the algorithms thoroughly discussed by Laferte et al. [33].

Finally, irregular-tree based image classification is conducted by employing the inference algorithms in Fig. 2–4 for IT_{V^0} and IT_V, and the inference algorithms in Fig. 3–2 for IQT_{V^0} and IQT_V. Since image classification represents a supervised machine learning problem, it is necessary to first learn model parameters on training images. For this purpose, we employ the learning algorithms discussed in Section 2.5 for IT_{V^0} and IT_V, and the learning algorithms discussed in Section 3.3 for IQT_{V^0} and IQT_V.

After inference of MRF, DRF, TSBN, and the irregular tree, on a given image, for each model, we conduct pixel labeling by using the MAP classifier. In Fig. 6–11, we illustrate an example of pixel labeling for a dataset-II image. Here, we say that an image region is correctly recognized as an object if the majority of MAP-classified pixel labels in that region are equal to the true labeling of the 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 40 test images in dataset II is summarized in Table 6–4. In each cell of Table 6–4, the first number indicates the overall recognition error, while the number in parentheses indicates the ratio of swapped-identity errors. For instance, for IT_{V^0} the overall recognition error is 9.6%, of which 37% of instances were caused by swapped-identity errors. Moreover, Table 6–5 shows average pixel-labeling error.

Next, we examine the *receiver operating characteristic* (ROC) of MRF, DRF, TSBN and IT_{V^0} for a two-class recognition problem. From the set of image classes given in Fig. 6–1, we choose "toy-snail" and "wavelets-book" as the two possible classes in the following

Table 6–4: Object recognition error

image type	MRF	DRF	TSBN	IT_{V^0}
dataset II	21.2% (67%)	12.5% (83%)	14.8% (72%)	9.6% (37%)

Table 6–5: Pixel labeling error

image type	MRF	DRF	TSBN	IT_{V^0}
dataset II	15.8%	12.3%	16.1%	9.9%

set of experiments. The task is to label two-class-problem images containing “toy-snail” and ”wavelets-book” objects, a typical example of which is shown in Fig. 6–12. Here, pixels labeled as “toy-snail” are considered true positives, while pixels labeled as “book” are considered true negatives. In Fig. 6–13, we plot ROC curves for the two-class problem, where we compare the performance of IT_{V^0} with those of MRF, DRF and TSBN . From Fig. 6–13, we observe that image classification with IT_{V^0} is the most accurate, since its ROC curve is the closest to the left-hand and top borders of the ROC space, as compared to the ROC curves of the other models. Further, in Fig. 6–14, we plot ROC curves for the same two-class problem, where we compare the performance of IT_V, with those of IT_{V^0}, TSBN, and TSBN↑. From Fig. 6–14, we observe that image classification with IT_V is the most accurate, and that both IT_{V^0} and IT_V outperform their fixed-structure counterparts TSBN and TSBN↑.

From the results reported in Tables 6–4 and 6–5, as well as form Figs. 6–13 and 6–14, we note that irregular trees 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 [75]), in the case of

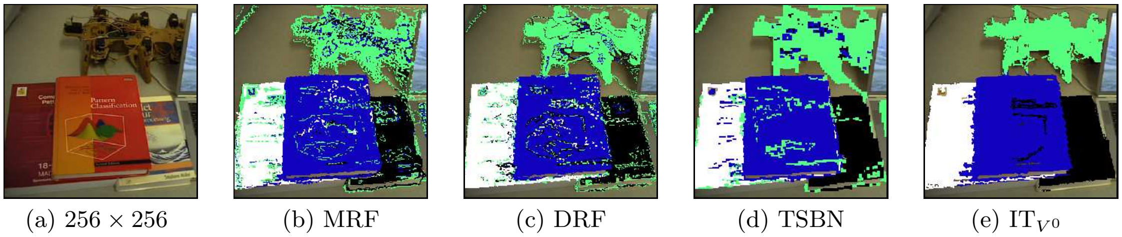
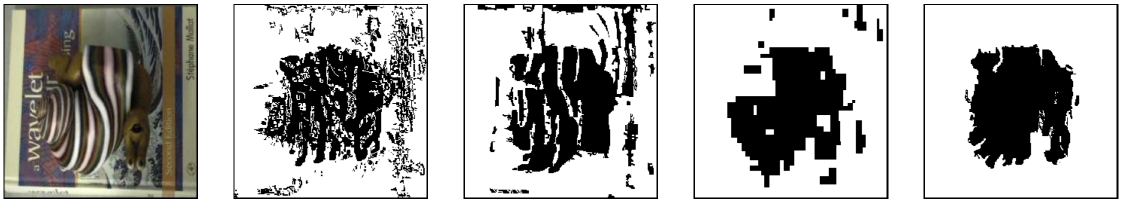

(a) 256 × 256 (b) MRF (c) DRF (d) TSBN (e) IT_{V^0}

Figure 6–11: 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.



(a) 256 × 256 (b) MRF (c) DRF (d) TSBN (e) IT$_{V^0}$

Figure 6–12: MAP pixel labeling using different statistical models.

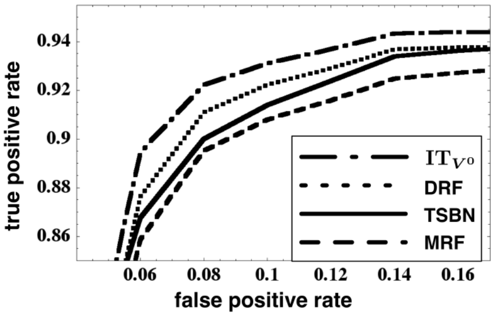


Figure 6–13: ROC curves for the image in Fig. 6–12a with IT$_{V^0}$, TSBN, DRF and MRF.

complex scenes this error is much higher [4, 5, 11, 76, 77]. To some extent, our results could have been improved had we employed 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 lack of explicit analysis of visible object parts. Thus, the poor classification performance of MRF, DRF, and TSBN, reported in Tables 6–4 and 6–5, 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

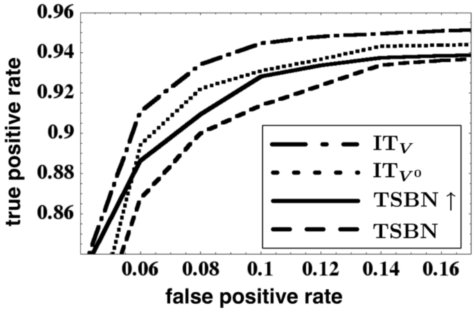


Figure 6–14: ROC curves for the image in Fig. 6–12a with IT$_V$, IT$_{V^0}$, TSBN, and TSBN↑.

to poor classification performance. Therefore, we hypothesize that the main reason why irregular trees 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, 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.

To support our hypothesis, instead of applying more sophisticated image-feature-extraction tools and better classification procedures than majority vote, we introduce a more radical change to our recognition strategy.

6.4 Object-Part Recognition Strategy

Recall from Section 6.1 that irregular trees are capable of capturing component sub-component structures at various scales, such that root nodes represent the center of mass of distinct objects, while children nodes down the subtrees represent object parts. As such, irregular trees provide a natural and seamless framework for identifying candidate image regions as object parts, requiring no additional training for such identification. To utilize this convenient property, we conduct the object-part recognition strategy presented in Section 4.2.

We compare the performance of the whole-object and part-object recognition strategies. The whole-object approach can be viewed as a benchmark strategy, in the sense that a majority of existing vision systems does not explicitly analyze visible object parts at various scales. In these systems, once the object is detected, the whole image region is identified through MAP classification, as is done in the previous section.

In Fig. 6–15, we present classification results for IT$_{V^0}$, using the whole-object and object-part recognition strategies on dataset-II images. In Fig. 6–15a, both strategies succeed in recognizing two different "Fluke" voltage-measuring instruments (see Fig. 6–1). However, in Fig. 6–15b, the whole-object recognition strategy fails to make a distinction between the objects, since the part that differentiates most one object from another is occluded, making it a difficult case for recognition even for a human interpreter. In the other two images, we observe that the object-part recognition strategy is more successful than the whole-object approach.

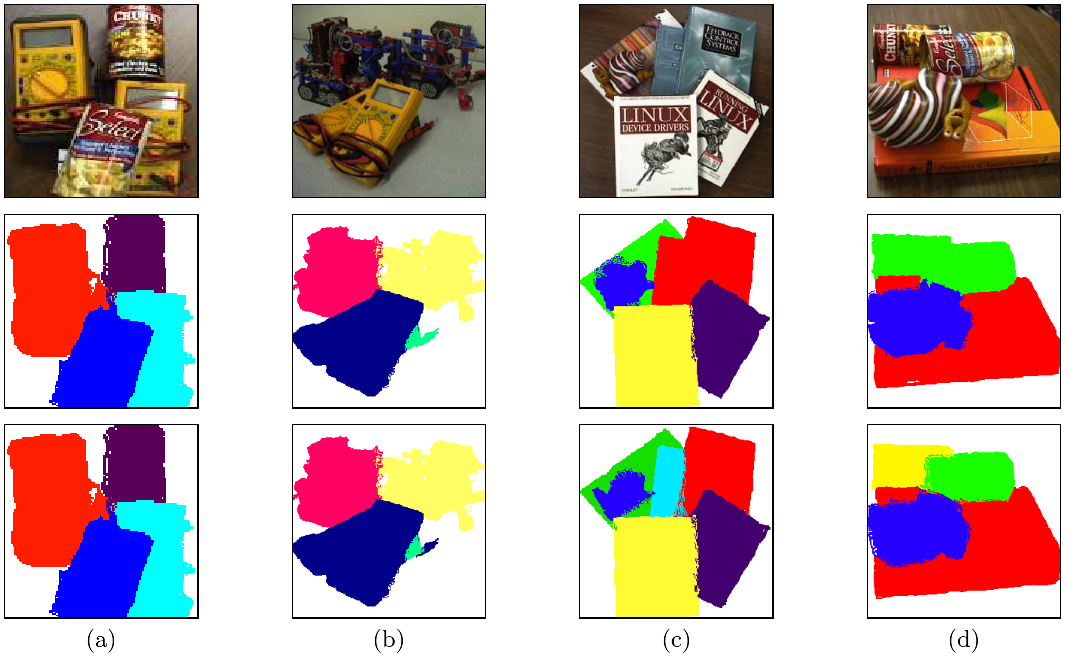


Figure 6–15: Comparison of two recognition strategies on dataset II for IT$_{V0}$: (top) 128 × 128 challenging images containing objects that are very similar in appearance; (middle) classification using the whole-object recognition strategy; (bottom) classification using the part-object recognition strategy; each recognized object in the image is marked with a different color.

For estimating the object-recognition error of IT$_{V0}$ on dataset-II images, the following instances are counted as error: (1) merging two distinct objects into one (i.e., object not detected), and (2) swapping the identity of objects (i.e., object correctly detected but misclassified as one of the objects in the class of known objects). The recognition error averaged over all objects in 40 test images in dataset II is only 5.8%, an improvement of nearly 40% over the reported error of 9.6% in the previous section.

We also recorded the object-recognition error of IQT$_{V0}$ over all objects in 20 test images of datasets IV, V, and VI, respectively. The results are summarized in Table 6–6. In each cell of Table 6–6, the first number indicates the overall recognition error, while the number in parentheses indicates the ratio of merged-object errors. For instance, for dataset V and the whole-object strategy, the overall recognition error is 21.2%, of which slightly more than half (56%) were caused by merged-object errors. The results in Table 6–6 clearly demonstrate significantly improved recognition performance, as well as reduction in false

Table 6–6: Object recognition error for IQT$_{V^0}$

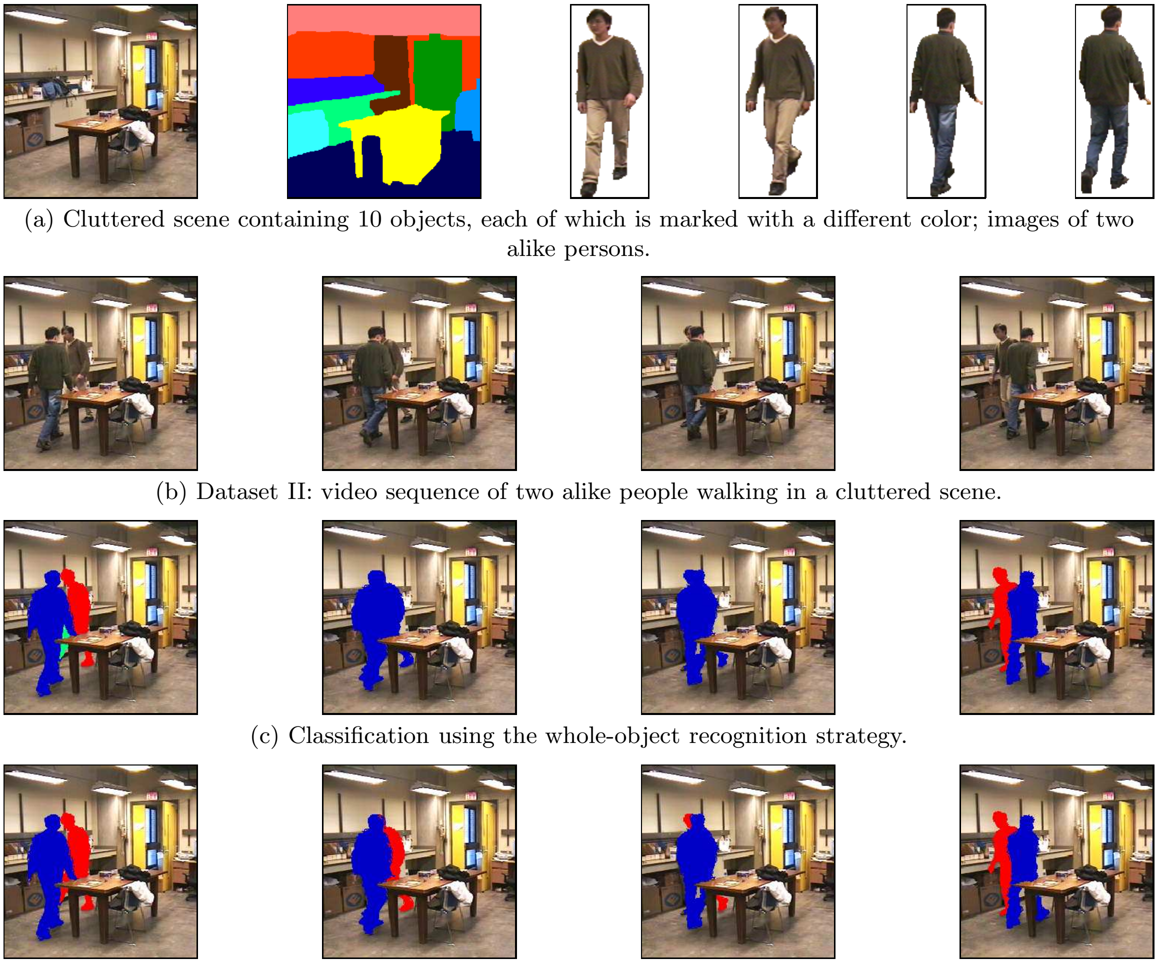
	datasets		
strategy	IV	V	VI
whole-object	11.6% (85%)	21.2% (56%)	26.3% (44%)
object-part	3.3% (100%)	8.7% (92%)	12.5% (81%)

Table 6–7: Pixel labeling error for IQT$_{V^0}$

	datasets		
strategy	IV	V	V
whole-object	9.6%	17.9%	16.3%
object-part	4.3%	6.7%	8.3%

alarm and swapped-identity types of error for the object-part, as compared with the whole-object approach. Also, Table 6–7 shows that the object-part strategy reduces pixel-labeling error.

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



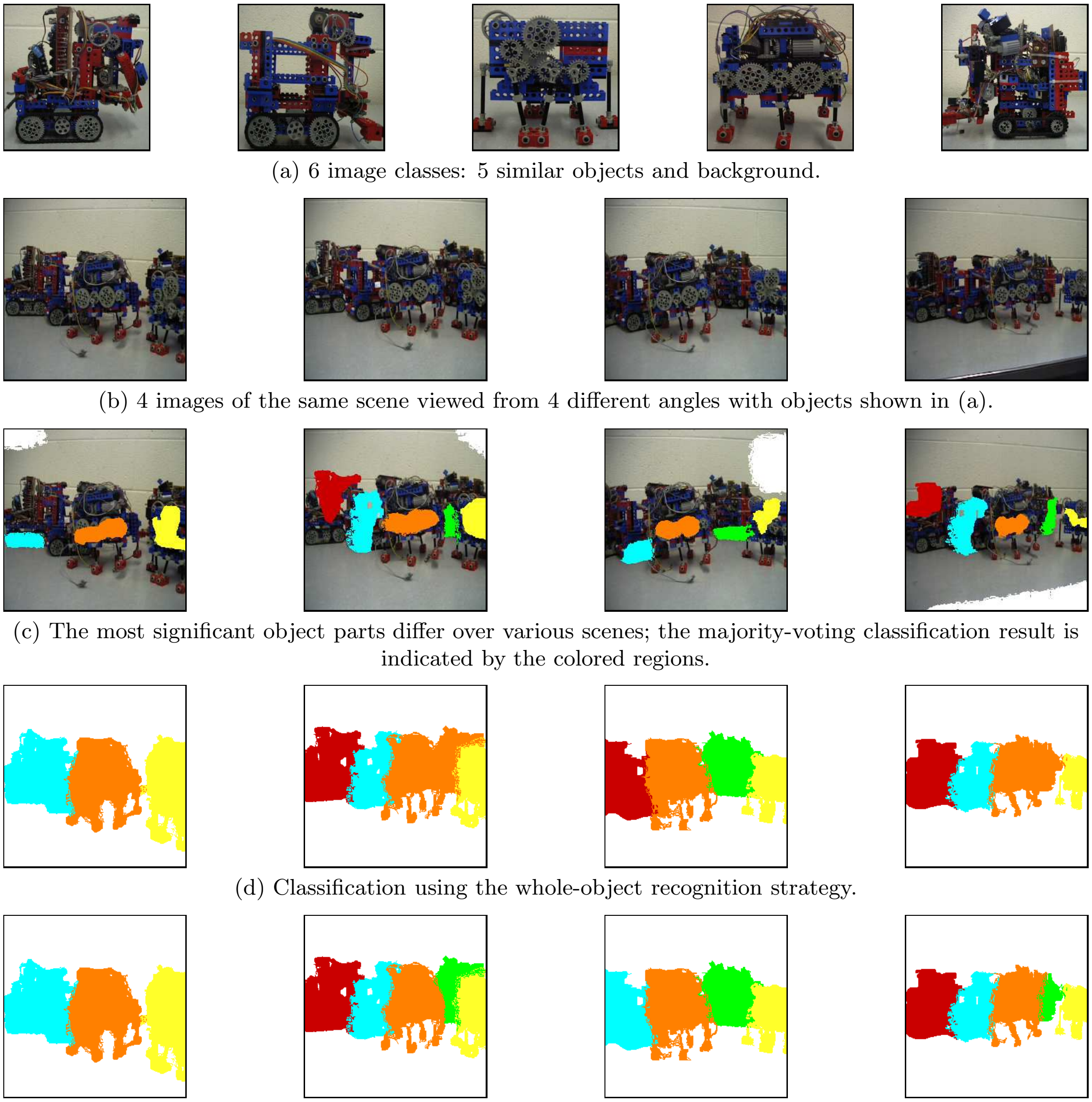
(a) Cluttered scene containing 10 objects, each of which is marked with a different color; images of two alike persons.

(b) Dataset II: video sequence of two alike people walking in a cluttered scene.

(c) Classification using the whole-object recognition strategy.

(d) Classification using the part-object recognition strategy.

Figure 6–16: Recognition results over dataset IV for IQT_{V0}.



(a) 6 image classes: 5 similar objects and background.

(b) 4 images of the same scene viewed from 4 different angles with objects shown in (a).

(c) The most significant object parts differ over various scenes; the majority-voting classification result is indicated by the colored regions.

(d) Classification using the whole-object recognition strategy.

(e) Classification using the object-part recognition strategy.

Figure 6–17: Recognition results over dataset V for IQT_{V^0}.

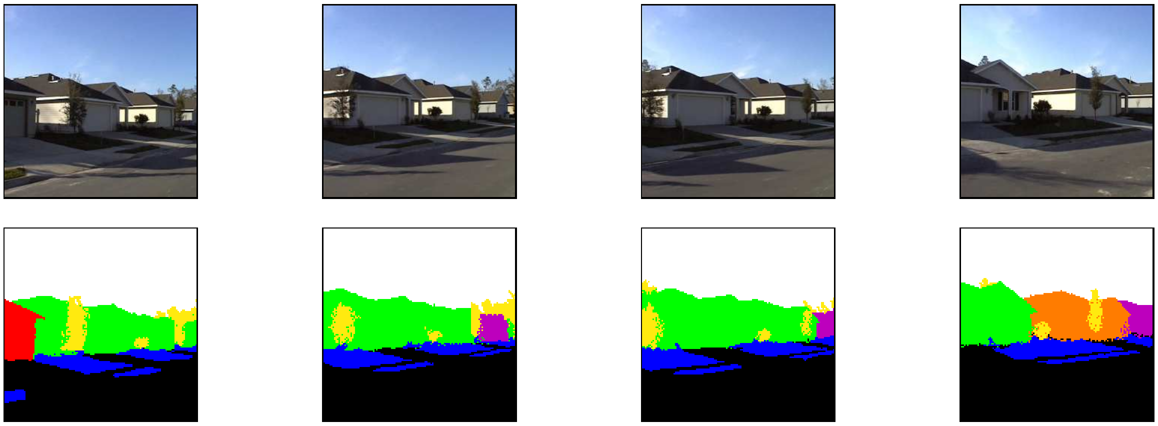


Figure 6–18: Classification using the part-object recognition strategy; Recognition results for dataset VI.

CHAPTER 7
CONCLUSION

7.1 Summary of Contributions

In this dissertation, we have addressed detection and recognition of partially occluded, alike objects in complex scenes – the problem that has eluded, as of yet, a satisfactory solution. The experiments reported herein show that "traditional" approaches to object recognition, where objects are first detected and then identified as a whole, yield poor performance in complex settings. Therefore, we speculate that a careful analysis of visible, fine-scale object details may prove critical for recognition. However, in general, the analysis of multiple sub-parts of multiple objects gives rise to prohibitive computational complexity. To overcome this problem, we have proposed to model images with irregular trees, which provide a suitable framework for developing novel object-recognition strategies – in particular, object-part recognition. Here, object details at various scales are first detected through tree-structure estimation; then, these object parts are analyzed as to which component of an object is the most significant for recognition of that object; finally, information on cognitive significance of each object part is combined toward the ultimate image classification. Empirical evidence demonstrates that this explicit treatment of object parts results in an improved recognition performance, as compared to the strategies where object components are not explicitly accounted for.

In Chapter 2, we have proposed two architectures within the irregular-tree framework, referred to as IT_{V^0} and IT_V. For each architecture, we have developed an inference algorithm. Gibbs sampling has been shown to be successful at finding trees that have high posterior probability; however, at a great computational price, which renders the algorithm impractical. Therefore, we have proposed Structured Variational Approximation (SVA) for inference of IT_{V^0} and IT_V, which relaxes poorly justified independence assumptions in prior work. We have shown that SVA converges to larger posterior distributions, an order of magnitude faster than competing algorithms. We have also demonstrated that IT_{V^0} and

IT_V overcome the blocky segmentation problem of TSBNs, and that they possess certain invariance to translation, rotation, and scaling transformations.

In Chapter 3, we have proposed another two architectures, referred to as IQT_{V^0} and IQT_V. In these models, we have constrained the node positions to be fixed, such that only connections can control irregular tree structure. At the same time, we have made the distribution of connections dependent on image classes. This formulation has allowed us to avoid variational-approximation inference, and to develop the exact inference algorithm for IQT_{V^0} and IQT_V. We have shown that it converges slower than SVA; however, it yields larger likelihoods, which in general means that IQT_{V^0} represents underlying stochastic processes in the image more accurately than IT_{V^0}.

In experiments on unsupervised image segmentation, we have shown the capability of irregular trees to capture important component-subcomponent structures in images. Empirical evidence demonstrates that root nodes represent the center of mass of distinct objects, while children nodes down the subtrees represent object parts. As such, irregular trees provide a natural and seamless framework for identifying candidate image regions as object parts, requiring no additional training for such identification. In Chapter 4, we have proposed to explicitly analyze the significance of object parts (i.e., tree nodes) with respect to recognition of an object as a whole. We have defined entropy as a measure of such cognitive significance. To avoid the costly approach of analyzing every detected object part, we have devised a greedy algorithm, referred to as object-part recognition. The comparison of whole-object and part-object approaches indicates that the latter method generates significantly better recognition performance and reduced pixel-labeling error.

Ultimately, what allows us to overcome obstacles in analyzing scenes with occlusions in a computationally efficient and intuitively appealing manner is the generative-model framework we have proposed. 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, alike objects.

7.2 Opportunities for Future Work

The analysis in the previous chapters suggests the following opportunities for future work. One promising thrust of research would be to investigate relationships among descriptive, generative and discriminative statistical models. We anticipate that these studies will lead to a greater integration of the modeling paradigms, yielding richer and more advanced classes of models. Here, the most critical issue is that of computationally manageable inference. With recent advances in the area of belief propagation (e.g., Generalized Belief Propagation [78]), the new algorithms may make it possible to solve real-world problems that were previously computationally intractable.

Within the irregular-tree framework, it is possible to continue further investigation toward replacing the current discrete-valued node variables with real-valued ones. Thereby, a real-valued version of the irregular tree can be specified. Gaussians could be used as a probability distribution to govern continuous random variables, represented by nodes, due to their tractable properties. Such a model could then operate directly on real-valued pixel data, improving the state-of-the-art techniques for solving various image-processing problems, including super resolution, image enhancement, and compression.

Further, with respect to the measure of significance of irregular-tree nodes, one can pursue investigation of more complex information-theoretic concepts than Shanon's entropy. For example, we anticipate that joint entropy and mutual information may yield a more efficient cognitive analysis, which in turn could eliminate the need for the greedy algorithm discussed in Section 4.2.

The analysis of object parts can be interpreted as integration of information from multiple complementary and/or competitive sensors, each of which has only limited accuracy. As such, further research could be conducted on formulating the optimal strategy for combining the pieces of information of object parts toward ultimate object recognition. We anticipate that algorithms such as the adaptive boosting (AdaBoost) [79] and Support Vector Machine [80] may prove useful for this purpose.

Another promising research topic is to incorporate available prior knowledge into the proposed Bayesian estimation framework, where we have assumed that all classification errors are equally costly. However, in many applications, some errors are more serious than others. Cost-sensitive learning methods are needed to address this problem [81].

On a broader scale, the research reported in this dissertation can be viewed as solving a more general machine learning problem, with experimental validation on images as data. This problem concerns supervised learning from examples, where the goal is to learn a function $X = f(Y)$ from N training examples of the form $\{(Y_n, f(Y_n))\}_{n=1}^{N}$. Here, X_n and Y_n contain sub-components, the meaning of which differs for various applications. For example, in computer vision, each Y_n might be a vector of image pixel values, and each X_n might be a partition of that image into segments and an assignment of labels to each segment. Most importantly, the components of Y_n form a sequence (e.g., a sequence on the 2D image lattice). Therefore, learning a classifier function $X = f(Y)$ represents the *sequential supervised learning* problem [82]. Thus, in this dissertation, we have addressed sequential supervised learning, the solutions of which can be readily applied to a wide range of problems beyond computer vision, such as, for example, speech processing, where the components of Y form a sequence in time.

APPENDIX A
DERIVATION OF VARIATIONAL APPROXIMATION

Preliminaries. Computation of $KL(Q\|P)$, given by Eq. (2.12), is intractable, because it depends on $P(Z,X,R'|Y,R^0)$. Note, though, that $Q(Z,X,R')$ does not depend on $P(Y|R^0)$ and $P(R^0)$. Consequently, by subtracting $\log P(Y|R^0)$ and $\log P(R^0)$ from $KL(Q\|P)$, we obtain a tractable criterion $J(Q,P)$, whose minimization with respect to $Q(Z,X,R')$ yields the same solution as minimization of $KL(Q\|P)$:

$$J(Q,P)\triangleq KL(Q\|P)-\log P(Y|R^0)-\log P(R^0)=\int_{R'} dR' \sum_{Z,X} Q(Z,X,R')\log\frac{Q(Z,X,R')}{P(Z,X,R,Y)} . \tag{A.1}$$

$J(Q,P)$ is known alternatively as Helmholtz free energy, Gibbs free energy, or free energy [59]. By minimizing $J(Q,P)$, we seek to compute parameters of approximate distributions $Q(Z)$, $Q(X|Z)$ and $Q(R'|Z)$. It is convenient, first, to reformulate Eq. (A.1) as $J(Q,P) = L_Z + L_X + L_R$. We define auxiliary L_Z, L_X, and L_R as $L_Z \triangleq \sum_Z Q(Z)\log\frac{Q(Z)}{P(Z)}$, $L_X \triangleq \sum_{Z,X} Q(Z)Q(X|Z)\log\frac{Q(X|Z)}{P(X|Z)P(Y|X,\boldsymbol{\rho})}$, and $L_R \triangleq \int_{R'} dR' \sum_Z Q(Z)Q(R'|Z)\log\frac{Q(R'|Z)}{P(R|Z)}$. To derive expressions for L_Z, L_X, L_R, we first observe:

$$\langle z_{ij}\rangle=\xi_{ij},\ \left\langle x_i^k\right\rangle=m_i^k,\ \left\langle x_i^k x_j^l\right\rangle=Q_{ij}^{kl}m_j^l\ \Rightarrow\ m_i^k={\textstyle\sum_{j\in V}}\xi_{ij}{\textstyle\sum_{l\in M}}Q_{ij}^{kl}m_j^l,\ \forall i\in V,\forall k\in M, \tag{A.2}$$

where $\langle\cdot\rangle$ denotes expectation with respect to $Q(Z,X,R')$. Consequently, from Eqs. (2.1), (2.9) and (A.2), we have

$$L_Z = {\textstyle\sum_{ij\in V}}\xi_{ij}\log[\xi_{ij}/\gamma_{ij}]\ . \tag{A.3}$$

Next, from Eqs. (2.4), (2.10) and (A.2), we derive

$$L_X = {\textstyle\sum_{i,j\in V}\sum_{k,l\in M}}\xi_{ij}Q_{ij}^{kl}m_j^l\log[Q_{ij}^{kl}/P_{ij}^{kl}] - {\textstyle\sum_{i\in V}\sum_{k\in M}} m_i^k\log P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k,\boldsymbol{\rho}(i))\ . \tag{A.4}$$

Note that for DT_{V^0}, V in the second term is substituted with V^0. Finally, from Eqs. (2.3), (2.11) and (A.2), we get

$$L_R = \frac{1}{2}\textstyle\sum_{i,j\in V'}\xi_{ij}\left(\log\frac{|\Sigma_{ij}|}{|\Omega_{ij}|} - \mathrm{Tr}\left\{\Omega_{ij}^{-1}\Omega_{ij}\right\} + \mathrm{Tr}\left\{\Sigma_{ij}^{-1}\langle(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})^T\rangle\right\}\right). \tag{A.5}$$

Let us now consider the expectation in the last term:

$$\begin{aligned}
&\langle(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})(\boldsymbol{r}_i{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})^T\rangle = \langle(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij}{+}\boldsymbol{\mu}_{ij}{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij}{+}\boldsymbol{\mu}_{ij}{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij})^T\rangle = \\
&= \Omega_{ij} + 2\langle(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})(\boldsymbol{\mu}_{jp}{-}\boldsymbol{r}_j{-}\boldsymbol{d}_{ij}{-}\boldsymbol{\mu}_{jp}{+}\boldsymbol{\mu}_{ij})^T\rangle + \\
&\quad + \langle(\boldsymbol{r}_j{-}\boldsymbol{\mu}_{jp}{+}\boldsymbol{d}_{ij}{+}\boldsymbol{\mu}_{jp}{-}\boldsymbol{\mu}_{ij})(\boldsymbol{r}_j{-}\boldsymbol{\mu}_{jp}{+}\boldsymbol{d}_{ij}{+}\boldsymbol{\mu}_{jp}{-}\boldsymbol{\mu}_{ij}))\rangle = \\
&= \Omega_{ij}{+}2\langle(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})(\boldsymbol{\mu}_{jp}{-}\boldsymbol{r}_j)^T\rangle{+}\langle(\boldsymbol{r}_j{-}\boldsymbol{\mu}_{jp})(\boldsymbol{r}_j{-}\boldsymbol{\mu}_{jp})^T{+}(\boldsymbol{\mu}_{ij}{-}\boldsymbol{\mu}_{jp}{-}\boldsymbol{d}_{ij})(\boldsymbol{\mu}_{ij}{-}\boldsymbol{\mu}_{jp}{-}\boldsymbol{d}_{ij})^T\rangle{=} \\
&= \Omega_{ij} + \textstyle\sum_{p\in V'}\xi_{jp}\left(2\Psi_{ijp} + \Omega_{jp} + \mathcal{M}_{ijp}\right),
\end{aligned} \tag{A.6}$$

where the definitions of auxiliary matrices Ψ_{ijp} and $\mathcal{M}_{ijp}$ are given in the second to the last derivation step above, and i-j-p is a child-parent-grandparent triad. It follows from Eqs. (A.5) and (A.6) that

$$L_R = \frac{1}{2}\sum_{i,j\in V'}\xi_{ij}\left(\log\frac{|\Sigma_{ij}|}{|\Omega_{ij}|} - 2 + \mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\} + \sum_{p\in V'}\xi_{jp}\mathrm{Tr}\{\Sigma_{ij}^{-1}(2\Psi_{ijp}{+}\Omega_{jp}{+}\mathcal{M}_{ijp})\}\right). \tag{A.7}$$

In Eq. (A.7), the last expression left to compute is $\mathrm{Tr}\{\Sigma_{ij}^{-1}\Psi_{ijp}\}$. For this purpose, we apply the Cauchy-Schwartz inequality as follows:

$$\begin{aligned}
\mathrm{Tr}\{\Sigma_{ij}^{-1}\Psi_{ijp}\} &= \mathrm{Tr}\{\Sigma_{ij}^{-\frac{1}{2}}\Sigma_{ij}^{-\frac{1}{2}}\langle(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})(\boldsymbol{\mu}_{jp}{-}\boldsymbol{r}_j)^T\rangle\} = \mathrm{Tr}\{\langle\Sigma_{ij}^{-\frac{1}{2}}(\boldsymbol{r}_i{-}\boldsymbol{\mu}_{ij})(\boldsymbol{\mu}_{jp}{-}\boldsymbol{r}_j)^T\Sigma_{ij}^{-\frac{1}{2}}\rangle\}, \\
&\le \mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}\}^{\frac{1}{2}},
\end{aligned} \tag{A.8}$$

where we used the fact that the Σ's and Ω's are diagonal matrices. Although the Cauchy-Schwartz inequality in general does not yield a tight upper bound, in our case it appears reasonable to assume that variables $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ (i.e., positions of object parts at different scales) are uncorrelated. Substituting Eq. (A.8) into Eq. (A.7), we finally derive the upper bound for L_R as

$$L_R \le \frac{1}{2}\sum_{i,j\in V'}\xi_{ij}\left(\log\frac{|\Sigma_{ij}|}{|\Omega_{ij}|} - 2 + \mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\} + \sum_{p\in V'}\xi_{jp}\mathrm{Tr}\{\Sigma_{ij}^{-1}\left(\Omega_{jp}+\mathcal{M}_{ijp}\right)\}+\right.$$
$$\left. + 2\sum_{p\in V'}\xi_{jp}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}\}^{\frac{1}{2}}\right) . \tag{A.9}$$

Optimization of $Q(X|Z)$. $Q(X|Z)$ is fully characterized by parameters Q_{ij}^{kl}. From the definition of L_X, we have $\partial J(Q,P)/\partial Q_{ij}^{kl}{=}\partial L_X/\partial Q_{ij}^{kl}$. Due to parent-child dependencies in Eq. (A.2), it is necessary to iteratively differentiate L_X with respect to Q_{ij}^{kl} down the subtree of node i. For this purpose, we introduce three auxiliary terms F_{ij}, G_i, and λ_i^k, which facilitate computation, as shown below:

$$\begin{aligned} &F_{ij} \triangleq \sum_{k,l\in M}\xi_{ij}Q_{ij}^{kl}m_j^l\log[Q_{ij}^{kl}/P_{ij}^{kl}]\ , \\ &G_i \triangleq \sum_{d,c\in d(i)}F_{dc} - \left\{\sum_{k\in M}m_i^k\log P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k,\boldsymbol{\rho}(i))\right\}_{V^0}\ , \ \Rightarrow \frac{\partial L_X}{\partial Q_{ij}^{kl}} = \frac{\partial F_{ij}}{\partial Q_{ij}^{kl}} + \frac{\partial G_i}{\partial m_i^k}\frac{\partial m_i^k}{\partial Q_{ij}^{kl}}, \\ &\lambda_i^k \triangleq \exp(-\partial G_i/\partial m_i^k), \end{aligned} \tag{A.10}$$

where $\{\cdot\}_{V^0}$ denotes that the term is included in the expression for G_i if i is a leaf node for DT_{V^0}. For DT_V, the term in braces $\{\cdot\}$ is always included. This allows us to derive update equations for both models simultaneously. After finding the derivatives $\partial F_{ij}/\partial Q_{ij}^{kl} = \xi_{ij}m_j^l(\log[Q_{ij}^{kl}/P_{ij}^{kl}]{+}1)$ and $\partial m_i^k/\partial Q_{ij}^{kl}{=}\xi_{ij}m_j^l$, and substituting these expressions in Eq. (A.10), we arrive at

$$\partial L_X/\partial Q_{ij}^{kl} = \xi_{ij}m_j^l(\log[Q_{ij}^{kl}/P_{ij}^{kl}] + 1 - \log\lambda_i^k)\ . \tag{A.11}$$

Finally, optimizing Eq. (A.11) with the Lagrange multiplier that accounts for the constraint $\sum_{k\in M}Q_{ij}^{kl}{=}1$ yields the desired update equation: $Q_{ij}^{kl} = \kappa P_{ij}^{kl}\lambda_i^k$, introduced in Eq. (2.13).

To compute λ_i^k, we first find

$$\begin{aligned} \partial G_i/\partial m_i^k &= \sum_{c\in c(i)}\left(\partial F_{ci}/\partial m_i^k + \sum_{a\in M}(\partial G_c/\partial m_c^a)(\partial m_c^a/\partial m_i^k)\right) - \{\log P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k,\boldsymbol{\rho}(i))\}_{V^0}\ , \\ &= \sum_{c\in c(i)}\sum_{a\in M}\xi_{ci}Q_{ci}^{ak}\left(\log[Q_{ci}^{ak}/P_{ci}^{ak}] + \partial G_c/\partial m_c^a\right) - \{\log P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k,\boldsymbol{\rho}(i))\}_{V^0} \end{aligned} \tag{A.12}$$

and then substitute Q_{ij}^{kl}, given by Eq. (2.13), into Eq. (A.12), which results in $\lambda_i^k = \{P(\boldsymbol{y}_{\boldsymbol{\rho}(i)}|x_i^k,\boldsymbol{\rho}(i))\}_{V^0}\prod_{c\in V}\left[\sum_{a\in M}P_{ci}^{ak}\lambda_c^a\right]^{\xi_{ci}}$, as introduced in Eq. (2.14).

Optimization of $Q(R'|Z)$. $Q(R'|Z)$ is fully characterized by parameters $\boldsymbol{\mu}_{ij}$ and Ω_{ij}. From the definition of L_R, we observe that $\partial J(Q)/\partial\Omega_{ij}{=}\partial L_R/\partial\Omega_{ij}$ and $\partial J(Q)/\partial\boldsymbol{\mu}_{ij}{=}\partial L_R/\partial\boldsymbol{\mu}_i$. Since the Ω's are positive definite, from Eq. (A.9), it follows that

$$\begin{aligned}\partial L_R/\partial\Omega_{ij} = 0.5\ \xi_{ij} \Big(&-\mathrm{Tr}\{\Omega_{ij}^{-1}\}+\mathrm{Tr}\{\Sigma_{ij}^{-1}\}+\textstyle\sum_{c\in V'}\xi_{ci}\mathrm{Tr}\{\Sigma_{ci}^{-1}\}+\\ &+\textstyle\sum_{p\in V'}\xi_{jp}\mathrm{Tr}\{\Sigma_{ij}^{-1}\}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{-\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{jp}\}^{\frac{1}{2}}+\\ &+\textstyle\sum_{c\in V'}\xi_{ci}\mathrm{Tr}\{\Sigma_{ci}^{-1}\}\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ij}\}^{-\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ci}\}^{\frac{1}{2}}\Big)\ . \qquad \text{(A.13)}\end{aligned}$$

From $\partial L_R/\partial\Omega_{ij}{=}0$, it is straightforward to derive the update equation for Ω_{ij} given by Eq. (2.17).

Next, to optimize the $\boldsymbol{\mu}_{ij}$ parameters, from (A.9), we compute

$$\begin{aligned}\frac{\partial L_R}{\partial\boldsymbol{\mu}_{ij}} &= \frac{\partial}{\partial\boldsymbol{\mu}_{ij}}\left[\frac{1}{2}\textstyle\sum_{i,j,p\in V'}\ \xi_{ij}\xi_{jp}(\boldsymbol{\mu}_{ij}-\boldsymbol{\mu}_{jp}-\boldsymbol{d}_{jp})^T\Sigma_{ij}^{-1}(\boldsymbol{\mu}_{ij}-\boldsymbol{\mu}_{jp}-\boldsymbol{d}_{jp})\right]\ ,\\ &= \textstyle\sum_{c,p\in V'}\Big(\xi_{ij}\xi_{jp}\Sigma_{ij}^{-1}(\boldsymbol{\mu}_{ij}{-}\boldsymbol{\mu}_{jp}{-}\boldsymbol{d}_{jp}) - \xi_{ci}\xi_{ij}\Sigma_{ci}^{-1}(\boldsymbol{\mu}_{ci}{-}\boldsymbol{\mu}_{ij}{-}\boldsymbol{d}_{ij})\Big)\ . \qquad \text{(A.14)}\end{aligned}$$

Then, from $\partial L_R/\partial\boldsymbol{\mu}_{ij}{=}0$, it is straightforward to compute the update equation for $\boldsymbol{\mu}_{ij}$ given by Eq. (2.16).

Optimization of $Q(Z)$. $Q(Z)$ is fully characterized by the parameters ξ_{ij}. From the definitions of L_Z, L_X, and L_R we see that $\partial J(Q)/\partial\xi_{ij} = \partial(L_X{+}L_R{+}L_Z)/\partial\xi_{ij}$. Similar to the optimization of Q_{ij}^{kl}, we need to iteratively differentiate L_X as follows:

$$\partial L_X/\partial\xi_{ij} = \partial F_{ij}/\partial\xi_{ij} + \textstyle\sum_{k\in M}(\partial G_i/\partial m_i^k)(\partial m_i^k/\partial\xi_{ij}) \qquad \text{(A.15)}$$

where F_{ij} and G_i are defined as in Eq. (A.10). Substituting the derivatives $\partial G_i/\partial m_i^k{=}{-}\log\lambda_i^k$, and $\partial F_{ij}/\partial\xi_{ij}{=}\sum_{k,l\in M}Q_{ij}^{kl}m_j^l\log[Q_{ij}^{kl}/P_{ij}^{kl}]$, and $\partial m_i^k/\partial\xi_{ij}{=}\sum_{l\in M}Q_{ij}^{kl}m_j^l$ into Eq. (A.15) we obtain

$$\frac{\partial L_X}{\partial\xi_{ij}}{=}\textstyle\sum_{k,l\in M}Q_{ij}^{kl}m_j^l\left(\log\dfrac{Q_{ij}^{kl}}{P_{ij}^{kl}}-\log\lambda_i^k\right) = -\textstyle\sum_{k,l\in M}Q_{ij}^{kl}m_j^l\log\left(\sum_{a\in M}P_{ij}^{al}\lambda_i^a\right) = -\,A_{ij}\ , \qquad \text{(A.16)}$$

Next, we differentiate L_R, given by Eq. (A.9), with respect to ξ_{ij} as

$$\begin{aligned}\partial L_R/\partial\xi_{ij} \ &= \ \frac{1}{2}\log|\Sigma_{ij}|/|\Omega_{ij}| - 1 + \frac{1}{2}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\} +\\ &\quad +\frac{1}{2}\textstyle\sum_{p\in V'}\xi_{jp}\left(\mathrm{Tr}\{\Sigma_{ij}^{-1}(\Omega_{jp}{+}\mathcal{M}_{ijp})\}{+}2\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{ij}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ij}^{-1}\Omega_{tu}\}^{\frac{1}{2}}\right) +\end{aligned}$$

$$+\frac{1}{2}\sum_{c\in V'}\xi_{ci}\left(\mathrm{Tr}\{\Sigma_{ci}^{-1}\,(\Omega_{ij}+\mathcal{M}_{cij})\}+2\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ci}\}^{\frac{1}{2}}\mathrm{Tr}\{\Sigma_{ci}^{-1}\Omega_{ij}\}^{\frac{1}{2}}\right), \tag{A.17}$$

$$= B_{ij} - 1\ , \tag{A.18}$$

where indexes c, j and p denote children, parents and grandparents of node i, respectively. Further, from Eq. (A.3), we get

$$\partial L_Z/\partial\xi_{ij} = 1 + \log\xi_{ij}/\gamma_{ij}\ . \tag{A.19}$$

Finally, substituting Eqs. (A.16), (A.18) and (A.19) into $\partial J(Q)/\partial\xi_{ij}$=0 and adding the Lagrange multiplier to account for the constraint $\sum_{j\in V'}\xi_{ij}$=1, we solve for the update equation of ξ_{ij} given by Eq. (2.18).

APPENDIX B
INFERENCE ON THE FIXED-STRUCTURE TREE

The inference algorithm for *Maximum Posterior Marginal* (MPM) estimation on the quad-tree is known to alleviate implementation issues related to underflow numerical error [33]. The whole procedure is summarized in Fig. B–1. The algorithm assumes that the tree structure is fixed and known. Therefore, in Fig. B–1, we simplify notation as $P(x_i|Z,Y)\rightarrow P(x_i|Y)$ and $P(x_i|x_j,Z)\rightarrow P(x_i|x_j)$. Also, we denote with $c(i)$ children of i, and with $d(i)$ the set of all the descendants down the tree of node i including i itself. Thus, $Y_{d(i)}$ denotes a set of all observables down the subtree whose root is i. Also, for computing $P(x_i|Y_{d(i)})$, in the bottom-up pass, $\propto$ means that equality holds up to a multiplicative constant that does not depend on x_i.

Two-pass MPM estimation on the tree

```
↓ Preliminary downward pass: $\forall i \in V^{L-1}, V^{L-2}, ..., V^{0}$,
• $P(x_i) = \sum_{x_j} P(x_i|x_j)P(x_j)$,
↑ Bottom-up pass:
    ■ Initialize leaf nodes: $\forall i \in V^{0}$,
        • $P(x_i|y_i) \propto P(y_i|x_i)P(x_i)$,
        • $P(x_i, x_j|y_i) = P(x_i|x_j)P(x_j)P(x_i|y_i)/P(x_i)$,
    ▲ compute upward $\forall i \in V^{1}, V^{2}..., V^{L}$,
        • $P(x_i|Y_{d(i)}) \propto P(x_i) \prod_{c \in c(i)} \sum_{x_c} \dfrac{P(x_c|Y_{d(c)})P(x_c|x_i)}{P(x_c)}$,
        • $P(x_i, x_j|Y_{d(i)}) = P(x_i|x_j)P(x_j)P(x_i|Y_{d(i)})/P(x_i)$,
↓ Top-down pass:
    ■ Initialize root: $i \in V^{L}$,
        • $P(x_i|Y) = P(x_i|Y_{d(i)})$,
        • $\hat{x}_i = \arg\max_{x_i} P(x_i|Y)$,
    ▼ compute downward $\forall i \in V^{L-1}, V^{L-2}..., V^{0}$,
        • $P(x_i|Y) = \sum_{x_j} \dfrac{P(x_i, x_j|Y_{d(i)})}{\sum_{x_i} P(x_i, x_j|Y_{d(i)})} P(x_j|Y)$,
        • $\hat{x}_i = \arg\max_{x_i} P(x_i|Y)$
```

Figure B–1: Steps 2 and 5 in Fig. 3–2: MPM estimation on the fixed-structure tree. Distributions $P(y_i|x_i)$ and $P(x_i|x_j)$ are assumed known.

REFERENCES

[1] W. E. L. Grimson and T. Lozano-Perez, "Localizing overlapping parts by searching the interpretation tree," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 9, no. 4, pp. 469–482, 1987.

[2] S. Z. Der and R. Chellappa, "Probe-based automatic target recognition in infrared imagery," *IEEE Trans. Image Processing*, vol. 6, no. 1, pp. 92–102, 1997.

[3] P. C. Chung, E. L. Chen, and J. B. Wu, "A spatiotemporal neural network for recognizing partially occluded objects," *IEEE Trans. Signal Processing*, vol. 46, no. 7, pp. 1991–2000, 1998.

[4] W. M. Wells, "Statistical approaches to feature-based object recognition," *Intl. J. Computer Vision*, vol. 21, no. 1, pp. 63–98, 1997.

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

[6] S. Z. Li, *Markov Random Field modeling in image analysis*, Springer-Verlag, Tokyo, Japan, 2nd edition, 2001.

[7] M. H. Lin and C. Tomasi, "Surfaces with occlusions from layered stereo," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 8, pp. 1073–1078, 2004.

[8] A. Mittal and L. S. Davis, "M2tracker: a multi-view approach to segmenting and tracking people in a cluttered scene," *Intl. J. Computer Vision*, vol. 51, no. 3, pp. 189–203, 2003.

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

[10] F. Dell'Acqua and R. Fisher, "Reconstruction of planar surfaces behind occlusions in range images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 4, pp. 569–575, 2002.

[11] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proc. IEEE Conf. Computer Vision Pattern Rec.*, Madison, WI, 2003, vol. 2, pp. 264–271, IEEE, Inc.

[12] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 23, no. 4, pp. 349–361, 2001.

[13] M. Weber, M Welling, and P. Perona, "Towards automatic discovery of object categories," in *Proc. IEEE Conf. Comp. Vision Pattern Rec.*, Hilton Head Island, SC, 2000, vol. 2, pp. 101–109, IEEE, Inc.

[14] M. Weber, M Welling, and P. Perona, "Unsupervised learning of models for recognition," in *Proc. 6th European Conf. Comp. Vision*, Dublin, Ireland, 2000, vol. 1, pp. 18–32, Springer.

[15] B. Heisele, T. Serre, M. Pontil, T. Vetter, and T. Poggio, "Categorization by learning and combining object parts," in *Advances in Neural Information Processing Systems, 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds., vol. 2, pp. 1239–1245. MIT Press, Cambridge, MA, 2002.

[16] P. F. Felzenszwalb and Daniel P. Huttenlocher, "Pictorial structures for object recognition," *Intl. J. of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.

[17] H. Schneiderman and T. Kanade, "Object detection using the statistics of parts," *Intl. J. Computer Vision*, vol. 56, no. 3, pp. 151–177, 2004.

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

[19] S. C. Zhu, Y. N. Wu, and D. B. Mumford, "Minimax entropy principle and its applications to texture modeling," *Neural Computation*, vol. 9, no. 8, pp. 1627–1660, 1997.

[20] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 6, no. 6, pp. 721–741, 1984.

[21] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling," in *Proc. Intl. Conf. Computer Vision*, Kerkyra, Greece, 1999, vol. 2, pp. 1033–1038, IEEE, Inc.

[22] J. S. De Bonet and P. Viola, "Texture recognition using a non-parametric multi-scale statistical model," in *Proc. IEEE Conf. Computer Vision Pattern Rec.*, Santa Barbara, CA, 1998, pp. 641–647, IEEE, Inc.

[23] M. J. Beal, N. Jojic, and H. Attias, "A graphical model for audiovisual object tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 7, pp. 828–836, 2003.

[24] J. Coughlan and A. Yuille, "Algorithms from statistical physics for generative models of images," *Image and Vision Computing*, vol. 21, no. 1, pp. 29–36, 2003.

[25] S. Kumar and M. Hebert, "Discriminative random fields: A discriminative framework for contextual interaction in classification," in *Proc. IEEE Intl. Conf. Comp. Vision*, Nice, France, 2003, vol. 2, pp. 1150–1157, IEEE, Inc.

[26] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data," in *Proc. Intl. Conf. Machine Learning*, Williams College, MA, 2001, pp. 282–289.

[27] 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.

[28] 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.

[29] H. Cheng and C. A. Bouman, "Multiscale Bayesian segmentation using a trainable context model," *IEEE Trans. Image Processing*, vol. 10, no. 4, pp. 511–525, 2001.

[30] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using Hidden Markov Models," *IEEE Trans. Signal Processing*, vol. 46, no. 4, pp. 886–902, 1998.

[31] 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.

[32] S. Todorovic and M. C. Nechyba, "Towards intellignet mission profiles of Micro Air Vehicles: multiscale Viterbi classification," in *Proc. 8th European Conf. Computer Vision*, Prague, Czech Republic, 2004, vol. 2, pp. 178–189, Springer.

[33] J.-M. Laferté, P. Pérez, and F. Heitz, "Discrete Markov image modeling and inference on the quadtree," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 390–404, 2000.

[34] M. R. Luettgen and A. S. Willsky, "Likelihood calculation for a class of multiscale stochastic models, with application to texture discrimination," *IEEE Trans. Image Processing*, vol. 4, no. 2, pp. 194–207, 1995.

[35] P. L. Ainsleigh, N. Kehtarnavaz, and R. L. Streit, "Hidden Gauss-Markov models for signal classification," *IEEE Trans. Signal Processing*, vol. 50, no. 6, pp. 1355–1367, 2002.

[36] J. Pearl, *Probabilistic reasoning in intelligent systems : networks of plausible inference*, Morgan Kaufamnn, San Mateo, CA, 1988.

[37] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, "Tree-based reparameterization framework for analysis of sum-product and related algorithms," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1120–1146, 2003.

[38] Brendan J. Frey, *Graphical Models for Machine Learning and Digital Communication*, The MIT Press, Cambridge, MA, 1998.

[39] S. Kumar and M. Hebert, "Man-made structure detection in natural images using a causal multiscale random field," in *Proc. IEEE Conf. Computer Vision Pattern Rec.*, Madison, WI, 2003, vol. 1, pp. 119–126, IEEE, Inc.

[40] M. K. Schneider, P. W. Fieguth, W. C. Karl, and A. S. Willsky, "Multiscale methods for the segmentation and reconstruction of signals and images," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 456–468, 2000.

[41] J. Li, R. M. Gray, and R. A. Olshen, "Multiresolution image classification by hierarchical modeling with two-dimensional Hidden Markov Models," *IEEE Trans. Inform. Theory*, vol. 46, no. 5, pp. 1826–1841, 2000.

[42] W. K. Konen, T. Maurer, and C. von der Malsburg, "A fast dynamic link matching algorithm for invariant pattern recognition," *Neural Networks*, vol. 7, no. 6-7, pp. 1019–1030, 1994.

[43] A. Montanvert, P. Meer, and A. Rosenfield, "Hierarchical image analysis using irregular tessellations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 4, pp. 307–316, 1991.

[44] P. Bertolino and A. Montanvert, "Multiresolution segmentation using the irregular pyramid," in *Proc. Intl. Conf Image Processing*, Lausanne, Switzerland, 1996, vol. 1, pp. 257–260, IEEE, Inc.

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

[46] N. J. Adams, *Dynamic trees: a hierarchical probabilistic approach to image modeling*, Ph.D. dissertation, Division of Informatics, Univ. of Edinburgh, Edinburgh, UK, 2001.

[47] A. J. Storkey, "Dynamic trees: a structured variational method giving efficient propagation rules," in *Uncertainty in Artificial Intelligence*, C. Boutilier and M. Goldszmidt, Eds., pp. 566–573. Morgan Kauffamnn, San Francisco, CA, 2000.

[48] 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.

[49] M. I. Jordan, Ed., *Learning in Graphical Models (Adaptive Computation and Machine Learning)*, MIT press, Cambridge, MA, 1999.

[50] M. I. Jordan, "Graphical models," *Statistical Science (spec. issue on Bayesian statistics)*, vol. 19, pp. 140–155, 2004.

[51] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, pp. 1–39, 1977.

[52] G. J. McLachlan and K. T. Thriyambakam, *The EM algorithm and extensions*, John Wiley & Sons, New York, NY, 1996.

[53] D. M. Chickering and D. Heckerman, "Efficient approximations for the marginal likelihood of incomplete data given a Bayesian network," in *Proc. 12th Conf. Uncertainty Artificial Intelligence*, Portland, OR, 1996, pp. 158–168, Assoc. Uncertainty Artificial Intelligence.

[54] S. Todorovic and M. C. Nechyba, "Interpretation of complex scenes using generative dynamic-structured models," in *CD-ROM Proc. IEEE CVPR 2004, Workshop on Generative-Model Based Vision (GMBV)*, Washington, DC, 2004, IEEE, Inc.

[55] S. Todorovic and M. C. Nechyba, "Detection of artificial structures in natural-scene images using dynamic trees," in *Proc. 17th Intl. Conf. Pattern Rec.*, Cambridge, UK, 2004, pp. 35–39, Intl. Assoc. Pattern Rec.

[56] M. Aitkin and D. B. Rubin, "Estimation and hypothesis testing in finite mixture models," *J. Royal Stat. Soc.*, vol. B-47, no. 1, pp. 67–75, 1985.

[57] R. M. Neal, "Probabilistic inference using Markov Chain Monte Carlo methods," Tech. Rep. CRG-TR-93-1, Connectionist Research Group, Univ. of Toronto, 1993.

[58] D. A. Forsyth, J. Haddon, and S. Ioffe, "The joy of sampling," *Intl. J. Computer Vision*, vol. 41, no. 1-2, pp. 109–134, 2001.

[59] 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.

[60] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge Univ. Press, Cambridge, UK, 2003.

[61] D. Barber and P. van de Laar, "Variational cumulant expantions for intractable distributions," *J. Artificial Intell. Research*, vol. 10, pp. 435–455, 1999.

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

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

[64] T. S. Jaakkola, "Tutorial on variational approximation methods," in *Adv. Mean Field Methods*, M. Opper and D. Saad, Eds., pp. 129–161. MIT press, Cambridge, MA, 2000.

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

[66] Trygve Randen and Hakon Husoy, "Filtering for texture classification: A comparative study," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 4, pp. 291–310, 1999.

[67] Stephane Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 2nd edition, 2001.

[68] Stephane G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 7, pp. 674–693, 1989.

[69] Jerome M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, 1993.

[70] N. G. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *J. Applied Comp. Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, 2001.

[71] Michael Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. on Image Processing*, vol. 4, no. 11, pp. 1549–1560, 1995.

[72] Nick Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *Journal of Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, 2001.

[73] T. Lindeberg, "Scale-space theory: a basic tool for analysing structures at different scales," *J. Applied Statistics*, vol. 21, no. 2, pp. 224–270, 1994.

[74] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. J. Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[75] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 4, pp. 509–522, 2002.

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

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

[78] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Generalized Belief Propagation," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds., pp. 689–695. MIT Press, Cambridge, MA, 2001.

[79] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Computer System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[80] V. N. Vapnik, *Statistical learning theory*, John Wiley & Sons, Inc., New York, NY, 1998.

[81] P. Domingos, "Metacost: A general method for making classifiers cost-sensitive," in *Proc. 15th Intl. Conf. Knowledge Discovery Data Mining*, San Diego, CA, 1999, pp. 155–164, ACM Press.

[82] T. G. Dietterich, "Machine learning for sequential data: A review," in *Lecture Notes in Computer Science*, T. Caelli, Ed., vol. 2396, pp. 15–30. Springer-Verlag, Heidelberg, Germany, 2002.

BIOGRAPHICAL SKETCH

Sinisa Todorovic was born in Belgrade, Serbia, in 1968. He graduated from Mathematical High School–Belgrade in 1987. He received his B.S. degree in electrical and computer engineering at the University of Belgrade, Serbia, in 1994. From 1994 until 2001, he worked as a software engineer in the communications industry. In fall 2001, Sinisa Todorovic enrolled in the master's degree program at the Department of Electrical and Computer Engineering, University of Florida, Gainesville. He became a member of the Center for Micro Air Vehicle Research, where he conducted research in statistical image modeling and multi-resolution signal processing. Sinisa Todorovic earned his master's degree (M.S. thesis option) in December, 2002, after which he continued his studies toward a Ph.D. degree in the same Department. He received two certificates for outstanding academic accomplishment in 2002 and 2003. He expects to graduate in May, 2005.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Dapeng Wu, Chair
Assistant Professor of Electrical and
Computer Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Antonio A. Arroyo
Associate Professor of Electrical and
Computer Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Jian Li
Professor of Electrical and Computer
Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Andrew J. Kurdila
Professor of Mechanical and Aerospace
Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Michael C. Nechyba
VP of R&D Department,
Pittsburgh Pattern Recognition, Inc.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Takeo Kanade
U.A. Helen Whitaker University Professor of
Robotics, Carnegie Mellon University

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

May 2005

Pramod P. Khargonekar
Dean, College of Engineering

Winfred M. Phillips
Dean, Graduate School

IRREGULAR-STRUCTURE TREE MODELS FOR IMAGE INTERPRETATION

Sinisa Todorovic
(352) 846–3993
sinisha@ufl.edu
Department of Electrical and Computer Engineering
Chair: Dapeng Wu
Degree: Doctor of Philosophy
Graduation Date: May 2005

Recognition of partially occluded, alike objects in a cluttered scene is to date an open problem in computer vision. In this dissertation, we propose to explicitly analyze detected, visible object parts, and then to incorporate this information toward recognition of an object as a whole. Since analysis of object components, in general, is computationally expensive, we propose to model images with multiscale graphical models known as irregular trees. The irregular tree can be viewed as an image representation, where nodes represent image details at various scales. We propose several architectures of irregular-tree models, as well as inference algorithms for estimating their structure (topology) and probability distributions. By analyzing the significance of each node in the model with respect to object recognition, we notably improve recognition performance, as compared to the strategies where object components are not explicitly accounted for. The techniques developed in this dissertation advance currently available methods for machine video analysis, which are widely used in security systems for surveillance and tracking people/objects of interest.