



Deposited via The University of Leeds.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/145284/>

Version: Accepted Version

Proceedings Paper:

Fehri, H, Gooya, A, Johnston, SA et al. (2017) Multi-class image segmentation in fluorescence microscopy using polytrees. In: Lecture Notes in Computer Science. International Conference on Information Processing in Medical Imaging, 25-30 Jun 2017, Boone, USA. Springer Verlag, pp. 517-528. ISBN: 9783319590493. ISSN: 0302-9743. EISSN: 1611-3349.

https://doi.org/10.1007/978-3-319-59050-9_41

© Springer International Publishing AG 2017. This is an author produced version of a conference paper published in Lecture Notes in Computer Science. Uploaded in accordance with the publisher's self-archiving policy.

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

Multi-class Image Segmentation in Fluorescence Microscopy Images Using Polytrees

Hamid Fehri^{1,2,3}, Ali Gooya¹, Simon A. Johnston^{2,3}, and Alejandro F. Frangi¹(✉)

¹ Center for Computational Imaging Simulation Technologies in Biomedicine (CISTIB), The University of Sheffield, Sheffield, United Kingdom

² Bateson Centre, Firth Court, University of Sheffield

³ Department of Infection, Immunity and Cardiovascular Disease, Medical School, University of Sheffield

Abstract. Multi-class segmentation is a crucial step in cell image analysis. This process becomes challenging when little information is available for recognising cells from the background, due to their poor discriminative features. To alleviate this, directed acyclic graphs such as trees have been proposed to model top-down statistical dependencies as a prior for improved image segmentation. However, using trees, modelling the relations between labels of multiple classes becomes difficult. To overcome this limitation, we propose a polytree graphical model that captures label proximity relations more naturally compared to tree based approaches. A novel recursive mechanism based on two-pass message passing is developed to efficiently calculate closed form posteriors of graph nodes on the polytree. The algorithm is evaluated using simulated data, synthetic images and real fluorescence microscopy images. Our method achieves Dice scores of 94.5% and 98% on macrophage and seed classes, respectively, outperforming GMM based classifiers.

1 Introduction

Macrophages are cells that play vital roles within the immune system. They recognise invasions to body and combat their agents and are also involved in healing processes. Studying these cells in depth requires quantification of their behaviour in their interactions with other cells and proteins, which needs segmentation of cells in microscopy images. One instance of such studies is the segmentation of nuclear and surface areas of mice macrophages in fluorescence microscopy images [9], which defines a multi-class segmentation problem addressed here. The challenges are more highlighted when limited information is available for recognition of each area, due to the low quality of images and/or complexity of shapes. In these cases, quality of segmentation is subject to improvement by employing inter-object relations.

Segmentation can be a primary step for cell, and in general, object counting. Although not directly considered in this paper, two categories of counting methods exist: first category recognises cells using detection based algorithms [6],

while the other one estimates density based on regression [1]. Former approaches perform based on a prior segmentation or detection of individual cells, while the latter do not rely on individual object detection and rather take a holistic approach towards counting and have been proven to be more efficient. Inspired by the counting application outlined in [1], our segmentation method is capable of extracting the image foreground as the union of subregions having different labels. The segmented foreground can be further used for efficient recognition of individual cells and their nuclei through post processing steps.

Graphical models are tools for modelling associative relations between objects. The key aspect of these models is that the label of each node is determined based on both its own attributes as well as attributes of other nodes connected to it through graph edges. This way, not only all the relevant information is incorporated in inferring the labels, but also label configuration constraints can be effectively projected on the model and be used during the inference. For instance, Chen et al. [5] employed graphical models to incorporate nuclear positions with boundary information for yeast cell segmentation. In a broader context of application, Uzunbas et al. [12] used an interactive graphical-model-based approach which seeks user’s input for nodes with uncertain labels and learns from mislabelled nodes. Another instance is the use of graphical models for combining appearance models with shape priors for retinal segmentation [11].

Two types of graphical models have been used for image segmentation. Markov Random Fields have weighted edges indicating the degree of dependencies between variables. These models require iterative procedures for estimating the hidden random variables and are therefore computationally demanding [4] and only approximate solutions are achievable for them [7]. On the other hand, Bayesian Networks (BNs) have directed edges that show how random variables depend on each other. Laferte et al. [8] proposed an efficient posterior calculation algorithm for nodes on a tree BN using message passing. Tree BNs are models in which there is only one route between any two nodes, and each node (except the root node) has exactly one parent node. Despite the efficiency of their proposed model, it lacks the ability to model complex multi-label configuration constraints. More specifically, the tree model proposed in [8] is capable of modelling two-wise constraints between nodes and thus suitable for binary image segmentation. However, many applications, such as cell/nuclei segmentation involve more complex constraints due to the presence of multiple classes.

In this paper, we introduce the application of polytree based BNs to image segmentation. Polytree BNs share features such as existence of a unique route between each two nodes on the graph with their tree relatives, however, each node in a polytree can have more than one parent. In comparison to tree graphical models, polytrees can model more complex label proximity relations on the graph (three-wise and more, based on the number of parent nodes) more naturally to achieve a potentially better image segmentation. This paper presents a tractable and efficient inference algorithm on BN polytrees for the first time. The proposed algorithm calculates closed form posteriors through two successive bottom-up and top-down inference passes on the polytree. The method is

evaluated by ancestral sampling on arbitrary graphs, and applied on synthetic and real microscopy images to segment macrophages and nuclei (seeds) from background. However, we anticipate that more general applications of polytrees using the proposed inference algorithm.

2 Method

Here, the procedure of using our proposed graphical model for image segmentation is presented. First, a polytree is generated for the image by grouping similar pixels and regarding them as nodes of the graph. Next, the parameters of the likelihood functions are trained and labels of the nodes are inferred. Finally, the segmented image is constructed based on the optimal labels on the graph.

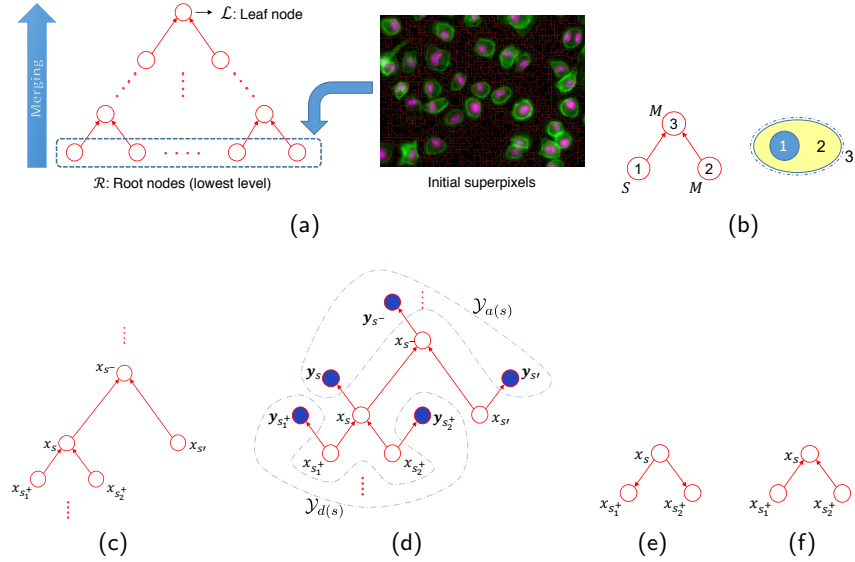


Fig. 1. Graph generation and inference on the polytree. (a) Generation of the polytree for the sample image. (b) Symbolic process of node merging for a synthetic macrophage with a seed. (c) Presenting the notation for nodes connected to an internal node s of the graph. (d) Graphical representation of ascendant, $\mathcal{Y}_{a(s)}$, and descendant, $\mathcal{Y}_{d(s)}$, observation nodes. Edge directions on cliques in conventional directed tree graphical models and the proposed polytree structure are shown in figures (e) and (f), respectively.

2.1 Graph generation

The graph is generated by firstly grouping pixels into locally coherent areas (superpixels), each representing a single root node (Fig. 1a). We use the SEEDS

algorithm [2], which refines the initial grid of identically block shaped superpixels into more coherent ones. The two most similar superpixels are then recursively merged to generate nodes in higher levels of the hierarchical graph, as follows.

Firstly, for each superpixel at the finest level, one (root) node in the lowest level of the graph is created (see Fig. 1a). Each two nodes that achieve the highest score in a similarity metric are then merged to create a new node. The new node corresponds to the union of image regions attached to its two lower level descendant nodes. We define the similarity metric as a superposition of distances using spatial and intensity features of the superpixels. A vector β is introduced to adjust contributions of each of these features in the similarity metric. The choice of β is investigated in section 3.3. After each merging step, the new node and all the other *orphan* nodes, are assessed with the similarity metric to recognise candidate nodes for the subsequent merging step. Region merging is continued until only two orphan nodes remain in the graph, which are eventually merged to create the leaf node that corresponds to the whole image (Fig. 1a). It is worth mentioning that since two nodes are merged at each step of graph evolution, the resulting structure is a binary graph; i.e. each non-root node has exactly two descendent nodes directly connected to it. We call this three-wise structure a *clique* and denote it by *parent1 – child – parent2*.

Once the graph is generated for the image, a set of observed and latent variables are assigned to nodes to define the inference problem, which is presented in the next section. Fig. 1b shows a symbolic process of merging for a cell with a seed. Nodes 1 and 2 align with blue and yellow areas in the synthetic cell. If these two nodes are chosen to be merged based on their value in the similarity metric, node 3 is generated which corresponds to the union of blue and yellow areas and is annotated by the dashed ellipse. This clique is represented by $1 - 3 - 2$.

2.2 Label inference

Let $\mathbb{X} = \{x_s\}$ and $\mathbb{Y} = \{y_s\}$ denote sets of labels (latent variables) and the corresponding observed features at nodes, respectively, and \mathcal{G} denote the set of nodes and edges. We assume that each node can be labelled as background (B), macrophage (M) or seed (S). Equivalently, $x_s \in \Lambda$, where Λ is the set of all possible labels, $\Lambda = \{B, M, S\}$. Following the notation of [8], for an internal node (neither in the lowest level nor the leaf node) s in the graph, s^- , s^+ and s' denote nodes in higher, lower and same layers, respectively (Fig. 1c).

Fig. 1e shows the structure of a clique in conventional trees [8], where graph edges are oriented accordingly from the root node s to the leaves s_1^+ , s_2^+ . For this configuration, the joint probability factorises into probabilities of 1-1 child-parent constraints, as $p(\mathbb{X}) = p(x_{s_1^+}|x_s)p(x_{s_2^+}|x_s)p(x_s)$. The limitation of this structure is that constraints can only be enforced on individual child-parent connections. This factorisation implies that label of each node is dependant on the label of its parent node on the hierarchy. However, many applications such as multi-class cell segmentation require to enforce three-wise prior constraints to ensure the expected clique configurations occur in the model. For instance, in our case of three possible classes of B , M and S , we expect to see an M node

having one M and one S parent nodes, as depicted in Fig. 1b. In the conventional structures, this prior knowledge is projected on the model through setting the probabilities $p(x_{s_i^+} = S|x_s = M)$ and $p(x_{s_i^+} = M|x_s = M)$ to non-zero values. However, enforcing the former constraint also makes $S - M - S$ cliques feasible, even though they are not acceptable based on the nature of the problem. This example shows the shortcoming of conventional directed tree models.

To overcome this limitation, we propose the use of another type of hierarchical graphical models with reverse edge orientations, as depicted in Fig. 1f, called *polytrees* [3]. In a polytree, each node at the lowest graph level (finest image resolution) is a root (in contrast to the single root node in conventional tree graphs) and there is only one leaf node (see Fig. 1a). This subtle yet *effective* change replaces 1-1 child-parent constraints with three-wise constraints on the cliques. The joint probability in this case is factorised as $p(\mathbb{X}) = p(x_s|x_{s_1^+}, x_{s_2^+})p(x_{s_1^+})p(x_{s_2^+})$, in which the term $p(x_s|x_{s_1^+}, x_{s_2^+})$ enforces the constraint on the clique. Using this configuration, the example structure of Fig. 1b requires setting the constraint $p(x_s = M|x_{s_1^+} = S, x_{s_2^+} = M)$ to be nonzero. However, in contrast to the previous structure, enforcing this constraint does not give rise to the emergence of physically infeasible clique structures.

We now derive equations governing the posterior probabilities of graph nodes. Given the observed data \mathbb{Y} , finding the best segmentation is equivalent to inferring the best configuration of labels \mathbb{X} for the graph. Bayesian inference associates the most probable label from the set of possible labels \mathcal{A} , given all observations:

$$\forall s \in \mathcal{G}, \hat{x}_s = \arg \max_{x_s \in \mathcal{A}} p(x_s|\mathbb{Y}) \quad (1)$$

As the dependencies of nodes' labels have been revised in the proposed graph structure, a new set of equations are derived to calculate the closed-form posterior probabilities at each node. The inference algorithm on the polytree calculates the posteriors in two passes. These two consist of a pass from the leaf to the roots, (*top-down* pass), and from the roots to the leaf (*bottom-up* pass).

Probability of a node's label x_s , given all data \mathbb{Y} , is computed by marginalising the probability of the clique over two parent nodes s_1^+ and s_2^+ given \mathbb{Y} and is called *joint posterior*.

$$p(x_s|\mathbb{Y}) = \sum_{x_{s_1^+}, x_{s_2^+}} p(x_s, x_{s_1^+}, x_{s_2^+}|\mathbb{Y}) \quad (2)$$

Here, three-wise constraints on cliques appear in the posterior calculation. Using the D-separation rule [10], the joint posterior is expanded as follows:

$$\begin{aligned} p(x_s, x_{s_1^+}, x_{s_2^+}|\mathbb{Y}) &= p(x_s|x_{s_1^+}, x_{s_2^+}, \mathbb{Y})p(x_{s_1^+}, x_{s_2^+}|\mathbb{Y}) \\ &= p(x_s|x_{s_1^+}, x_{s_2^+}, \mathcal{Y}_{a(s)})p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}, \mathcal{Y}_{d(s_1^+, s_2^+)}) \end{aligned} \quad (3)$$

where $\mathcal{Y}_{a(\cdot)}$ and $\mathcal{Y}_{d(\cdot)}$ refer to the set of observation nodes of the ascendant and descendant nodes, respectively. For each node s (or a set of nodes \mathcal{S}) ascendant

nodes refer to the set of all nodes that are connected to s (\mathcal{S}) through edges with inward directions. Similarly, descendant nodes include the nodes that are connected to node s (\mathcal{S}) through outward oriented graph edges. The union of ascendant and descendant observation nodes constructs the set of all observations (see Fig. 1d for a graphical explanation).

We first elaborate the term $p(x_s|x_{s_1^+}, x_{s_2^+}, \mathcal{Y}_{a(s)})$ on the right-hand side of Eq. 3. This term enforces posteriors of infeasible configurations to zero, as it is a product of the joint probability of a child node and its two parent nodes.

$$p(x_s|x_{s_1^+}, x_{s_2^+}, \mathcal{Y}_{a(s)}) = \frac{p(x_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)})}{\sum_{x'_s} p(x'_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)})} \quad (4)$$

Using D-separation rule, the nominator becomes:

$$\begin{aligned} p(x_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)}) &= p(x_{s_1^+}, x_{s_2^+}|x_s)p(x_s|\mathcal{Y}_{a(s)}) \\ &= \frac{p(x_s, x_{s_1^+}, x_{s_2^+})}{p(x_s)}p(x_s|\mathcal{Y}_{a(s)}). \end{aligned} \quad (5)$$

Term $p(x_s, x_{s_1^+}, x_{s_2^+})$ in Eq. 5 controls the occurrence of feasible and infeasible configurations on the graph, by being set to zero and nonzero values for them, respectively. Term $p(x_s|\mathcal{Y}_{a(s)})$ in Eq. 5 is the posterior of node s having given observations of all its ascendant nodes as well as its own observation. This *top-down posterior* is expanded as:

$$p(x_s|\mathcal{Y}_{a(s)}) \propto \sum_{x_{s^-}, x_{s'}} p(\mathbf{y}_s|x_s)p(\mathbf{y}_{s'}|x_{s'})p(x_{s'}|\mathcal{Y}_{d(s')}) \frac{p(x_s, x_{s'}, x_{s^-})}{p(x_{s^-})p(x_{s'})} p(x_{s^-}|\mathcal{Y}_{a(s^-)}) \quad (6)$$

Eq. 6 indicates that having calculated the likelihood probabilities $p(\mathbf{y}_s|x_s)$ and $p(\mathbf{y}_{s'}|x_{s'})$ and the posterior $p(x_{s'}|\mathcal{Y}_{d(s')})$, the top-down posterior of node s is calculated based on top-down posterior of node s^- . This implies that a top-down recursion calculates the top-down posteriors for all nodes.

On the other hand, term $p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}, \mathcal{Y}_{d(s_1^+, s_2^+)})$ on the right-hand side of Eq. 3 is factorised by several usages of D-separation rule. This factorisation separates parts that are calculated from ascendant and descendant nodes.

$$\begin{aligned} p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}, \mathcal{Y}_{d(s_1^+, s_2^+)}) &\propto p(\mathcal{Y}_{a(s_1^+, s_2^+)}, \mathcal{Y}_{d(s_1^+, s_2^+)}|x_{s_1^+}, x_{s_2^+})p(x_{s_1^+}, x_{s_2^+}) \\ &= p(\mathcal{Y}_{a(s_1^+, s_2^+)}|x_{s_1^+}, x_{s_2^+})p(\mathcal{Y}_{d(s_1^+, s_2^+)}|x_{s_1^+}, x_{s_2^+})p(x_{s_1^+}, x_{s_2^+}) \\ &= p(\mathcal{Y}_{a(s_1^+, s_2^+)}|x_{s_1^+}, x_{s_2^+})p(\mathcal{Y}_{d(s_1^+)}|x_{s_1^+})p(\mathcal{Y}_{d(s_2^+)}|x_{s_2^+})p(x_{s_1^+}, x_{s_2^+}) \\ &\propto p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}) \frac{p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)})}{p(x_{s_1^+})} \frac{p(x_{s_2^+}|\mathcal{Y}_{d(s_2^+)})}{p(x_{s_2^+})} \end{aligned} \quad (7)$$

Similar to Eq. 6, term $p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)})$ on the right-hand side of Eq. 7 is calculated through a top-down recursion as below.

$$p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}) \propto \sum_{x_s} p(\mathbf{y}_{s_1^+}|x_{s_1^+})p(\mathbf{y}_{s_2^+}|x_{s_2^+})p(x_{s_1^+}, x_{s_2^+}|x_s)p(x_s|\mathcal{Y}_{a(s)}) \quad (8)$$

Using a definition similar to that of top-down posterior, terms $p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)})$ and $p(x_{s_2^+}|\mathcal{Y}_{d(s_2^+)})$ in Eq. 7 are called *bottom-up* posteriors as they are calculated based on posteriors of their descendant nodes. For each node s in the graph, the bottom-up posterior is written as

$$p(x_s|\mathcal{Y}_{d(s)}) \propto \sum_{x_{s_1^+}, x_{s_2^+}} p(\mathbf{y}_{s_1^+}|x_{s_1^+})p(\mathbf{y}_{s_2^+}|x_{s_2^+}) \\ p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)})p(x_{s_2^+}|\mathcal{Y}_{d(s_2^+)})p(x_s|x_{s_1^+}, x_{s_2^+}) \quad (9)$$

Derivations of Eq. 6, 8 and 9 are not included due to the shortage of space.

Making use of Eq. 3, 4, 5 and 7 the node's posterior in Eq. 2 given all the observations is written as follows.

$$p(x_s|\mathbb{Y}) \propto \sum_{x_{s_1^+}, x_{s_2^+}} \frac{p(x_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)})}{\sum_{x'_s} p(x'_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)})} \\ p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}) \frac{p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)})}{p(x_{s_1^+})} \frac{p(x_{s_2^+}|\mathcal{Y}_{d(s_2^+)})}{p(x_{s_2^+})} \quad (10)$$

Eq. 10 calculates the posterior at each node s using three marginal posteriors $p(x_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)})$, $p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)})$ and $p(x_s|\mathcal{Y}_{d(s)})$, in Eq. 5, 8 and 9. Each of these terms are calculated through either top-down or bottom-up recursions. The inference is summarised in Algorithm 1. Note that \mathcal{R} and \mathcal{L} denote the set of root nodes and the leaf node in the graph, respectively.

3 Results

3.1 Validation of the inference algorithm in a two-class labelling problem

To assess the performance of the core inference algorithm, regardless of the problem it is applied to, ancestral sampling was used [3]. This technique, which is applied to generative models, mimics the creation of the observed data by generating sets of *fantasy* data. In this case, the model is a perfect representation of the observed data. Using the generated fantasy data as the observed data and comparing the inferred labels with the sampled values gives a measure for the method performance.

Consider we aim to draw a sample $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N$ from the joint distribution $p(\mathbb{X}, \mathbb{Y})$. The graph consists of N nodes and x_N denotes the only leaf node in the graph. To draw this sample, we start from the root nodes by taking samples \hat{x}_s from the probability distribution $p(x_s)|_{s \in \mathcal{R}}$ for all root nodes. Visiting the internal nodes in an upward recursion, we draw a sample \hat{x}_s from the conditional distribution $p(x_s|x_{s_1^+}, x_{s_2^+})$, for which the parent labels $x_{s_1^+}$ and $x_{s_2^+}$ have been sampled in the previous stages. Once we have sampled from the leaf node of the graph, x_N , we will have obtained a sample from the joint distribution $p(\mathbb{X}, \mathbb{Y})$.

Algorithm 1 Label inference on polytree directed graphical model

□ **Preliminary pass.** This initial upward recursion computes prior marginals for each node. $p(x_s|x_{s_1^+}, x_{s_2^+})$ are parameters set based on the nature of problem the model is representing.

for all $s \in \mathcal{R}$ **do**

$$p(x_s) = \frac{1}{|\Lambda|}$$

end for

for all $s \notin \mathcal{R}$ **do**

$$p(x_s) = \sum_{x_{s_1^+}, x_{s_2^+}} p(x_s|x_{s_1^+}, x_{s_2^+})p(x_{s_1^+})p(x_{s_2^+})$$

$$p(x_{s_1^+}, x_{s_2^+}|x_s) = \frac{p(x_s|x_{s_1^+}, x_{s_2^+})p(x_{s_1^+})p(x_{s_2^+})}{p(x_s)}$$

end for

△ **Bottom-up pass.** Upward recursion for calculating bottom-up posteriors of nodes.

for all $s \in \mathcal{R}$ **do**

$$p(x_s|\mathcal{Y}_{d(s)}) = p(x_s)$$

end for

for all $s \notin \mathcal{R}$ **do**

$$p(x_s|\mathcal{Y}_{d(s)}) \propto \sum_{x_{s_1^+}, x_{s_2^+}} p(\mathbf{y}_{s_1^+}|x_{s_1^+})p(\mathbf{y}_{s_2^+}|x_{s_2^+})p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)})p(x_{s_2^+}|\mathcal{Y}_{d(s_2^+)})p(x_s|x_{s_1^+}, x_{s_2^+})$$

end for

▽ **Top-down pass.** Downward recursion for calculating top-down posteriors and calculation of complete posteriors from marginal posteriors.

if $s = \mathcal{L}$ **then**

$$p(x_s|\mathcal{Y}_{a(s)}) = p(x_s|\mathbf{y}_s) \propto p(\mathbf{y}_s|x_s)p(x_s)$$

end if

for all $s \neq \mathcal{L}$ **do**

$$p(x_s|\mathcal{Y}_{a(s)}) \propto \sum_{x_{s^-}, x_{s_1^+}} p(\mathbf{y}_s|x_s)p(\mathbf{y}_{s_1^+}|x_{s_1^+})p(x_{s_1^+}|\mathcal{Y}_{d(s_1^+)}) \frac{p(x_s, x_{s_1^+}|x_{s^-})}{p(x_{s_1^+})} p(x_{s^-}|\mathcal{Y}_{a(s^-)})$$

$$p(x_s, x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s)}) = p(x_{s_1^+}, x_{s_2^+}|x_s)p(x_s|\mathcal{Y}_{a(s)})$$

$$p(x_{s_1^+}, x_{s_2^+}|\mathcal{Y}_{a(s_1^+, s_2^+)}) \propto \sum_{x_s} p(\mathbf{y}_{s_1^+}|x_{s_1^+})p(\mathbf{y}_{s_2^+}|x_{s_2^+})p(x_{s_1^+}, x_{s_2^+}|x_s)p(x_s|\mathcal{Y}_{a(s)})$$

end for

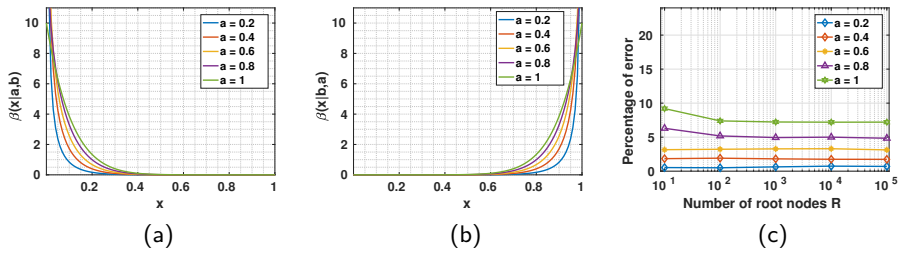


Fig. 2. Likelihood distribution functions and their effect on ancestral sampling inference accuracy. (a) and (b) show β distributions chosen as class-conditional likelihood functions in the ancestral sampling. b value was fixed and curves correspond to values of a from 0.2 to 1. (c) Percentage of wrongly inferred labels using ancestral sampling.

In this section only, we considered two classes for latent variables x_s and selecting y_s from the continuous range $[0, 1]$ and beta distributions were used as class conditional likelihood functions, $p(y_s|x_s)$. Labels done for different numbers of root nodes ranging from 10 to 100000 (leading to 19 to 199999 nodes in total as the graph is binary) and for different selectivities of the likelihood functions, controlled by parameters of β distributions. Fig. 2a and 2b show beta distributions for different values of parameter a . Fig. 2c depicts percentage of nodes with wrongly inferred labels for different numbers of nodes in the graph and for different beta distributions. This experiment verifies the correctness of the developed derivations and also indicates that more selective likelihood distribution functions result in higher inference accuracies. The best accuracy was achieved for $a = 0.2$ with the average error of 0.30%.

3.2 Validation of the algorithm on a synthetic image

The method was evaluated on a synthetic image containing two types of objects. A multi-scale first and second order derivative based invariant features have been utilized here. The segmentation results were compared to those of using Gaussian Mixture Models (GMMs). Yellow and blue areas in the synthetic image represent macrophages and seeds, respectively (chosen to be similar to the real image dataset). At each node, median of intensity across the superpixel was utilised as feature. Random noise was added to these features. Random noise was added to feature values of each node, which were median of node’s intensity.

Segmentation was done for different noise levels added to the synthetic image. Fig. 3 shows the synthetic image and its segmentation results for four cases of 0%, 30%, 60% and 65% of noise amplitudes, in each row, respectively. First column depicts the input image and second and third columns show results of segmentation for GMM and the proposed method, respectively. Dice scores of segmentations for GMM and the proposed method are shown in Fig. 4a. It can be seen that the proposed algorithm is more robust to intensive additive noise compared to GMM.

3.3 Applying the method to multi-class cell segmentation

We applied the proposed method to the image set BBBC020 from the Broad Bioimage Benchmark Collection [9] including 20 two-channel fluorescent microscopy images with ground truth annotations and compared the results with GMM. Multi-scale first and second derivatives of the image were calculated and a 10-dimensional feature vector \mathbf{y}_s containing median of feature values over the node’s region was created for each node s . Class-conditional distribution functions were considered to be Gaussian and their parameters were trained by feeding the algorithm with a few samples of manually segmented cells. Fig 5 shows three samples from the image dataset (first column) that were segmented using GMM classifier (second column) and the proposed method (third column). Dice scores of using the two methods on the dataset of real images are depicted in Fig. 4b. Segmentation results and the Dice scores indicate that the proposed method

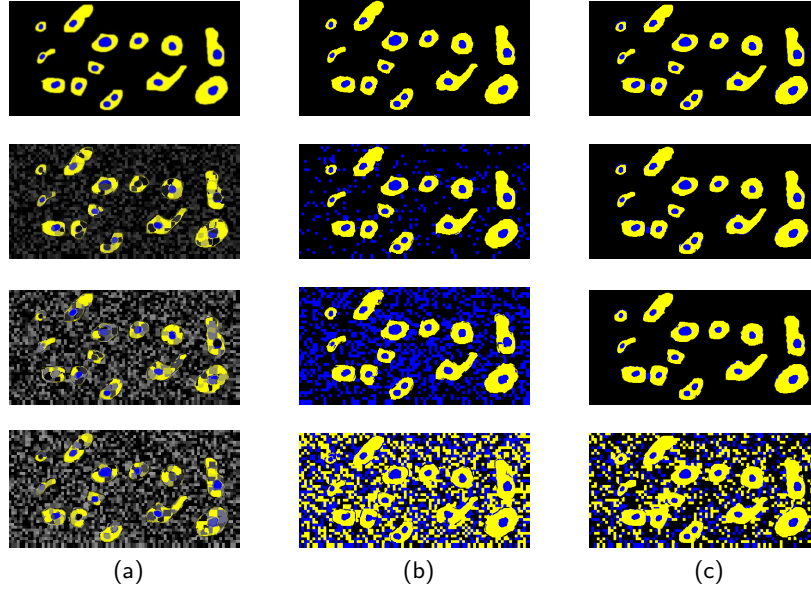


Fig. 3. Segmentation of the synthetic image for different random noise levels using GMM and the proposed method. Rows of column (a) correspond to images, with 0%, 30%, 60% and 65% (of maximum image intensity levels), respectively. (b) and (c) show results of segmentation using GMM and the proposed method for the input images of the first column, respectively. Last row illustrates the case where the proposed method cannot handle the added noise.

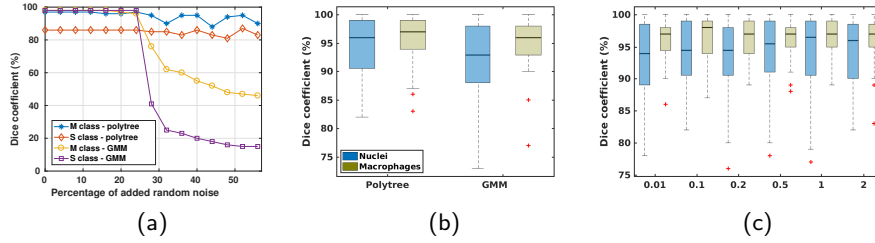


Fig. 4. Dice accuracies of segmentation for the proposed method and Gaussian Mixture Model (GMM). (a) Dice score for segmentation of two classes of objects, M and S, for the proposed method and GMM for different amounts of random noise added to the synthetic image. The proposed method shows more robustness to noise. (b) Accuracy of segmenting bone-marrow macrophages dataset for macrophage surface and the seeds for the proposed method and GMM. (c) Effect of β vector used in graph generation on segmentation accuracy. Horizontal axis shows the ratio of weights of intensity features to those of spatial features.

segments the images more accurately, which is due to considering semantically meaningful relationships between labels of nodes.

In graph generation, the β vector, used for similarity metric, consists of uniform weights (set as one) over intensity and β weighted spatial features (barycenters of the superpixels). To select a suitable value for β , we run a 5-fold crossvalidation experiment where we learn the Gaussian likelihoods from training sets and measure the Dice score on the test set. Fig. 4c shows how the choice of this parameter affects segmentation results. Based on this, we chose $\beta = 0.1$.

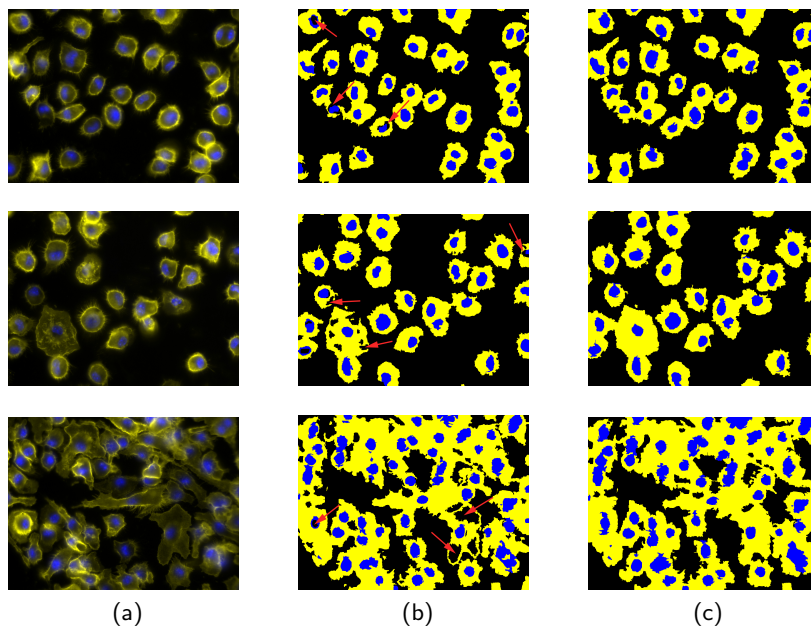


Fig. 5. Segmentation results for 3 samples from the image dataset. Column (a) shows real image samples. Columns (b) and (c) show segmentation results for GMM and the proposed method, respectively. Red arrows point to areas of the cell or the seeds that were not recognised by GMM, while the proposed algorithm segmented them correctly.

4 Conclusion

We proposed polytree based directed graphical models for multi-class cell segmentation. A non-iterative algorithm was derived for inferring labels of nodes on polytree that calculates closed form posteriors through two successive bottom-up and top-down recursions. The derived recursive inference algorithm is novel and contributes to the state-of-the-art from methodological perspective. The method was evaluated by ancestral sampling on the graph level, in addition to

segmentation of cell surfaces and seeds in synthetic and real microscopy images. This study was focused on the application of the proposed method to image segmentation and three-wise constraints. However, the designed formulation can be straightforwardly used for more complex applications where more complicated constraints are required to be projected on the model.

Acknowledgments. This work was supported by MRC fellowship (MR/J009156/1) and the Krebs Institute fellowship.

References

1. Arteta, C., Lempitsky, V., Noble, J.A., Zisserman, A.: Interactive object counting. In: European Conference on Computer Vision. pp. 504–518. Springer (2014)
2. Van den Bergh, M., Boix, X., Roig, G., Van Gool, L.: Seeds: Superpixels extracted via energy-driven sampling. *International Journal of Computer Vision* 111(3), 298–314 (2015)
3. Bishop, C.: *Pattern Recognition and Machine Learning*. Springer-Verlag New York (2006)
4. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence* 23(11), 1222–1239 (2001)
5. Chen, S.C., Zhao, T., Gordon, G.J., Murphy, R.F.: A novel graphical model approach to segmenting cell images. In: *Computational Intelligence and Bioinformatics and Computational Biology, 2006. CIBCB'06. 2006 IEEE Symposium on*. pp. 1–8. IEEE (2006)
6. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 580–587 (2014)
7. Komodakis, N., Paragios, N., Tziritas, G.: Mrf energy minimization and beyond via dual decomposition. *IEEE transactions on pattern analysis and machine intelligence* 33(3), 531–552 (2011)
8. Laferté, J.M., Pérez, P., Heitz, F.: Discrete markov image modeling and inference on the quadtree. *IEEE Transactions on image processing* 9(3), 390–404 (2000)
9. Ljosa, V., Sokolnicki, K.L., Carpenter, A.E.: Annotated high-throughput microscopy image sets for validation. *Nat Methods* 9(7), 637 (2012)
10. Pearl, J.: *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann (2014)
11. Rathke, F., Schmidt, S., Schnörr, C.: Probabilistic intra-retinal layer segmentation in 3-d oct images using global shape regularization. *Medical image analysis* 18(5), 781–794 (2014)
12. Uzunbas, M.G., Chen, C., Metaxas, D.: An efficient conditional random field approach for automatic and interactive neuron segmentation. *Medical image analysis* 27, 31–44 (2016)