Pop Out Many Small Structures from a Very Large Microscopic Image

Elena Bernardis[a], Stella X. Yu[b]

[a]*Department of Computer and Information Science, University of Pennsylvania*
Philadelphia, PA 19104
[b]*Department of Computer Science, Boston College*
Chestnut Hill, MA 02467

Abstract

In medical research, many applications require counting and measuring small regions in a large image. Segmenting these images poses a dilemma in terms of segmentation granularity due to fine structures and segmentation complexity due to large image sizes. We propose a constrained spectral graph partitioning framework to address the former while also reducing the segmentation complexity associated with the latter. The final segmentation is obtained from a set of patch segmentations, independently derived but subject to stitching constraints between neighboring patches. Individual segmentation is based on local pairwise cues designed to pop out all cells simultaneously from their common background, while the constraints are derived from mutual agreement analysis on patch segmentations from a previous round of segmentation. Our results show that we successfully extract many small regions in a variety of images.

Keywords: Segmentation; Spectral Graph Partitioning; Digital Pathology.

1. Introduction

There is often a need in medical research to count, measure, and compare numerous small structures in a large image. As the examples in Fig. 1 illustrate, images can be very different in nature and visual appearance. They could be from a frog's inner ear (left) or from a drosophila's fly brain region from electron microscopy (EM) data (right). On the left, regions of interest could be the larger scale haircell bundles or the smaller scale individual stereocilia that compose them. While cluster intensities vary across the image, cell intensities peak towards the center of each bundle. On the right, salient regions assume a larger
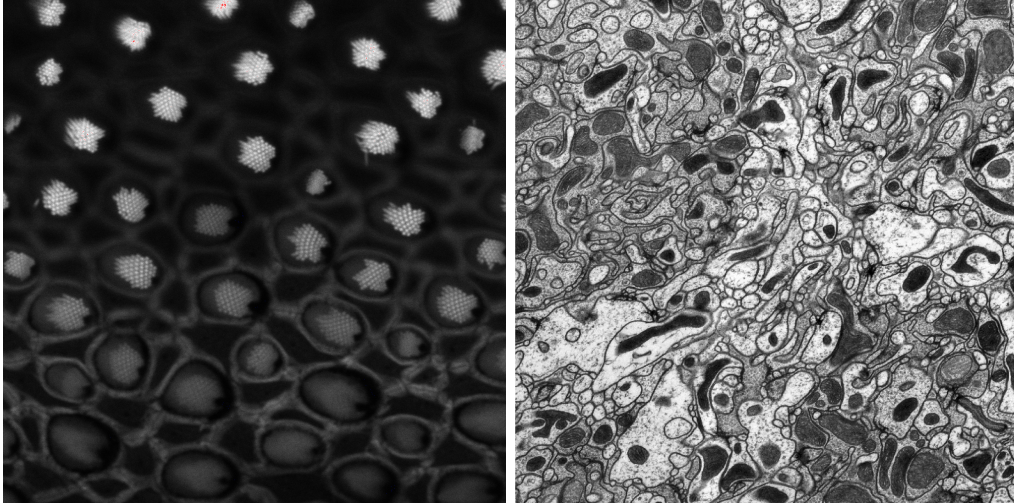
Figure 1: Many small structures in a large image. Left: stereocilia bundles of the frog's inner ear, 1600×1600 pixels (Image Courtesy: Medha Pathak and David Corey at Harvard). Right: electron microscopy (EM) data from the medulla brain region of the drosophila fly, 1800×1800 pixels (Image courtesy: Mitya Chklovskii, C Zhiyuan Lu, Rick Fetter, Shinya Takemura and Ian Meinertzhagen at Janelia Farm Research Institute). Cells are approximately 15 pixels in diameter.

range of shapes and they are densely packed in the image together with new elongated structures. Whether one wants to extract a 15 pixel cilia from the frog's hairbundles or a similar size vesicle in the fly's brain, one has to address complexity issues associate with a 1600×1600 pixels or more image. In both images, the regions of interest are very small when compared to the large image size.

Finding these small structures is a challenging segmentation problem on its own. Figure 2 displays images from two human pathology image datasets in cancer research. The appearance variety of these cells illustrate common problems encountered when dealing with these small regions. Rows 1 and 2 display images of epithelial and embryonic cells, with faint boundaries, large intensity variations and occlusions. Rows 3 and 4 display histopathological images of tumor-like lesions with textured cells and non-homogenous backgrounds.

The challenge of segmenting many small structures in a very large image is therefore two-fold: fine segmentation granularity when dealing with the size of the small segment and segmentation complexity when dealing with the size of the large image.
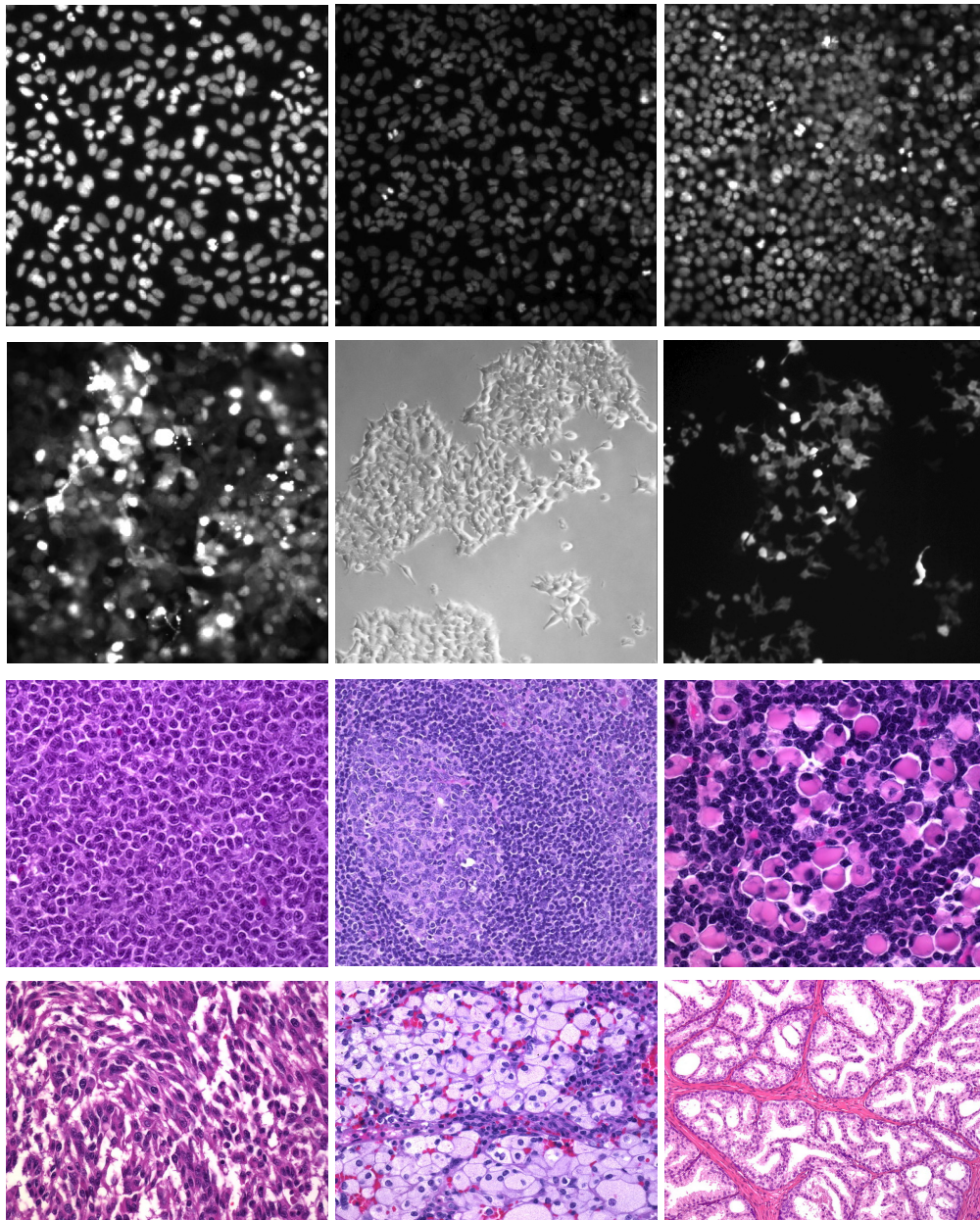
Figure 2: Segmenting cells challenges. **Row 1:** Epithelial A549 and embryonic kidney HEK293T cells featuring faint boundaries, varying dot intensities and occlusions. **Row 2:** More HEK cells of different convex geometries often appearing as cell clusters (Image courtesy: Nisha Sosale at UPenn). **Row 3:** Histopathology images depicting tumor-like lesions: cells of various sizes and textures in spleen cell lymphoma. **Row 4:** spleen tumor cells metastatic melanoma, clusters of glands in ovarian cells and clear cell cribriform hyperplasia from prostate illustrating cells on non-homogenous backgrounds (Image courtesy: www.webpathology.com).

1.1. Challenge 1: Segmenting many small structures

Segmentation of medical images often appears to be governed by global intensity levels, yet imaging noise and local intensity fluctuation present considerable challenges. Faint regions, similar intensities between adjacent regions and conjoined cells make these images challenging even for a human eye.

Many segmentation approaches for these type of image are based on mathematical morphology and energy-driven methods. In mathematical morphology, the watershed transform (Meyer, 1994) is applied to extract an initial set of contours, and markers or seeds are used to refine the contours of interest. The watershed transform is computationally very efficient (Meyer, 2005), but finding seeds automatically is application-driven and can be very challenging. Without proper seeds, oversegmentation results, since watershed is easily disrupted by local intensity fluctuations. Less prone to local noise fluctuations, energy-driven methods involve the minimization of an energy function formulated either on regions (Mumford and Shah, 1989; Geman and Geman, 1990; Zhu and Yuille, 1996) or contours, such as snakes (Xu and Prince, 1998) and level set methods (Malladi and Sethian, 1997). These algorithms though are computationally costly and they depend on initial seed choices. Various techniques have been proposed to combine the benefits of watershed and energy-driven methods, e.g. level sets for watershed (Tai et al., 2007) or watersnakes (Hieu et al., 2003) that allows to inject smoothness priors in the watershed formulation.

Graph cuts methods have also been employed to overcome the limitations of watershed algorithms, e.g. segmenting a single connected component with isoperimetric graph partitioning (Grady, 2006a). In (Coutsy et al., 2009), watersheds are formulated within a graph setup. Their theory together with a larger family of segmentation methods including random walker (Grady, 2006b) are generalized in the theory of power watersheds (Couprie et al., 2009). The latter though also depend on initial seeds and are thus not tailored for images with many small conjoined regions with faint boundaries and no clear minima.

1.2. Challenge 2: Dealing with a large image

Segmentation methods such as watershed and k-means clustering are efficient but unable to deal with large intensity variation (Fig. 3c,d). On the other hand, spectral graph partitioning methods (Shi and Malik, 2000; Yu, 2005) are prized for their ability to grasp the large structural organization of an image from the global integration of local cues. While this property is desired for understanding a real-scene image, it unnecessarily handles a huge number of pixels in a large image, since segmenting cells in one region really should not be influenced by cells far
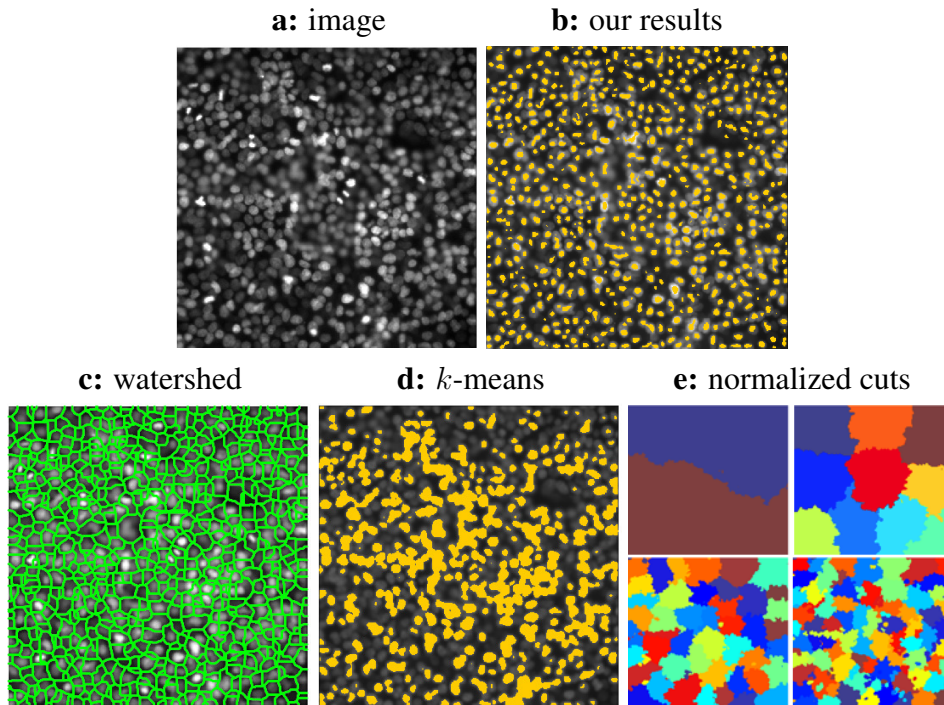
Figure 3: Efficiency versus robustness dilemma for segmenting small regions in a large image. **a,b)** Image of epithelial cells and our resulting cell segmentation (gold). **c,d)** Efficient two-way segmentations by watershed, k-means. Watershed oversegments in the presence of local intensity fluctuations while k-means is unable to distinguish cells of similar intensities apart. **e)** Normalized cuts (N-cuts) for $2, 4, 32, 64$ regions is meant to segment large regions in natural images. While robust in general, it fails for these structures.

from them. It also prevents small structures from being segmented all at once (Fig. 3e), since a larger image size leads to larger regions instead of numerous small ones given a fixed number of segments.

The two main approaches to reduce complexity, coarse-to-fine and multiresolution segmentations (Yu, 2005; Felzenszwalb and Huttenlocher, 2004; Cour et al., 2005; Barbu and Zhu, 2003; Galun et al., 2003; Hofmann et al., 1997; Yu, 2004), are not suitable for this task. The former approach speeds up the segmentation by initializing a finer segmentation with the results of a coarser one, whereas the latter integrates features at multiple scales to yield a better segmentation. Since small structures are not present in either coarser-scale segmentations or coarser-scale features, there is no help to be gained from either approach.

a: input image

b: independent segmentation

c: stitching constraints

d: constrained segmentation
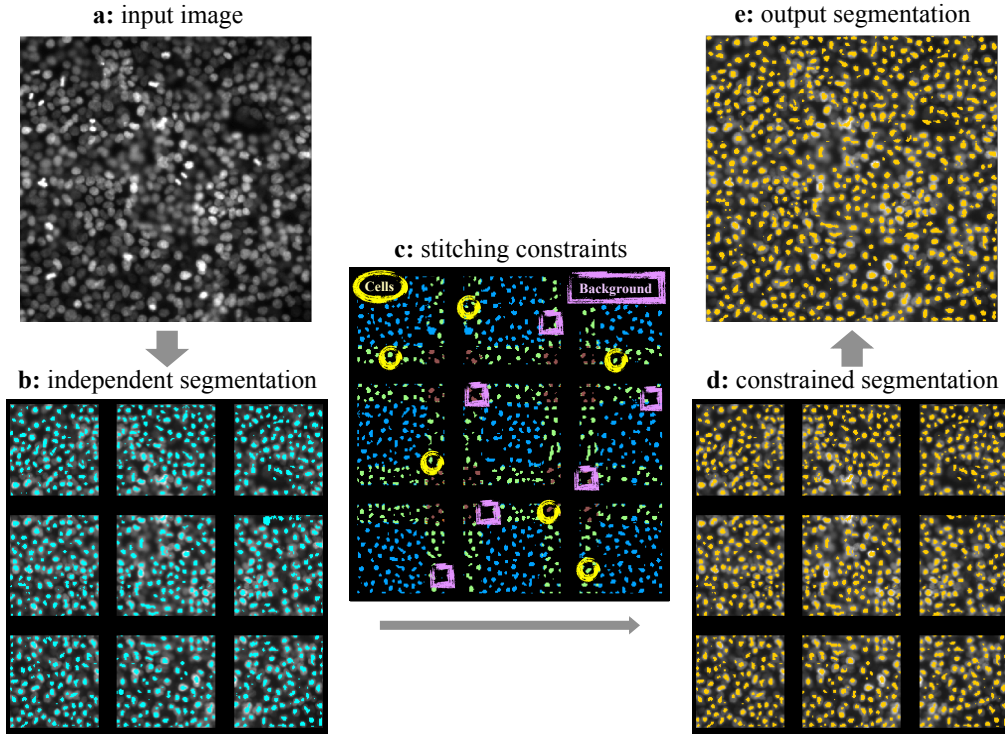
e: output segmentation

Figure 4: Segmentation subject to stitching constraints algorithm overview. Segmenting numerous small structures in a large input image (a) is performed as a series of independent patch segmentation subject to stitching constraints between neighboring patches. The constraints (c) are derived from mutual agreement analysis on adjacent patch segmentations from a previous round. Segmentations between neighboring patches are marked in blue, green, or maroon, if 1, 2, or more than 3 patches agree. Stitching constrains cells (yellow circles) and background (purple squares) to group within each type respectively. The constrained segmentation (d, in gold) improves the initial segmentation (b, in cyan) and can be seamlessly combined to obtain the final output (e).

1.3. Our Solution: Popping out many small structures in a large image

We propose a spectral-graph framework which scales effectively with image size without losing the fine granularity of small segments (as shown in Fig. 3b). Our segmentation algorithm is outlined in Fig. 4.

We focus on segmenting small convex regions (e.g. cells), which we will refer to as *dots*. Each image is divided into patches and each is segmented independently. Pixels in the image become nodes of a weighted graph, and finding dots becomes dissecting the graph based on weighted connections between nodes. Whereas attraction cues are commonly used to encode affinity between pixels, it

6

is the crucial role of repulsion cues that allows popping out all dots simultaneously from a common background. In order to compute these local cues, unlike real-scene image segmentation (Shi and Malik, 2000), we do not use single edge features (e.g. large intensity gradients along region boundaries) to delineate regions. Instead, we use distributive local gradient fields to characterize geometrical distinction between region cores in the foreground and region peripheries in the background. Similar ideas have been used in (Staal et al., 1999) to detect critical points in images with topological numbers, and in the mean shift-generated displacement vectors used in (Tek et al., 2001) to guide active contour models.

The individual segmentations are then used to establish the agreement between the patches, which provide pairwise long-range stitching constraints to be respected by each patch. We run the segmentation again on each patch, but now subject to these pairwise constraints on its pixels. The segmentation can be solved efficiently as a constrained eigenvalue problem (Yu and Shi, 2004). Since these segmentations have mutual agreement in the overlapping areas, their individual solutions can be collapsed into one segmentation on the entire large image. Segmentation subject to stitching is more than simple stitching. Constraints in the overlapping regions propagate through in the optimization process to also improve the interior segmentation.

This paper extends our previous theoretical model on segmentation subject to stitching constraints (Bernardis and Yu, 2010b) and finding dots (Bernardis and Yu, 2010a). In Section 2, we present a constrained spectral-graph partitioning framework that naturally integrates the stitching constraints with the fine segmentation granularity of the individual patches. In section 3, we present a detailed analysis of the parameters used in the algorithm and how they can be adaptively chosen without user interaction. The benefits of stitching are illustrated both on dot structures and on more complex geometries. Finally, we present results and performance comparisons with both general and state-of-the-art domain specific segmentation algorithms. Section 4 concludes the paper.

2. Spectral Graph Partitioning Subject to Stitching Constraints

We formulate the image segmentation problem as a constrained graph partitioning problem over a set of overlapping patches. Each patch is represented by a weighted graph, where nodes denote pixels and weights attached to edges connecting two nodes encode grouping cues between the pixels. Segmenting small structures becomes a two-way node partitioning problem: pixels inside cells form a foreground node set, and those outside form the other background node set.

2.1. Segmentation with Stitching: Constrained Graph Partitioning

We first formulate the constrained spectral graph two-segmentation criterion ε (Yu and Shi, 2001) in terms of pairwise grouping cues encoding short-range (with-group) attraction A and long-range (between-group) repulsion R between background and cells:

$$\max \varepsilon = \frac{\text{within-group attraction}}{\text{total degree of attraction}} + \frac{\text{between-group repulsion}}{\text{total degree of repulsion}} \qquad (1)$$

For each image patch I, we encode the grouping cues in an $n \times n$ weight matrix W, where n is the total number of pixels, to facilitate the foreground-background segmentation. Let:

$$W = A - R + D_R \qquad (2)$$
$$D = D_A + D_R \qquad (3)$$

where $D_M = \text{Diag}(M1_n)$ is the diagonal *degree* matrix for a $n \times n$ matrix M. Note that W could have both positive and negative weights. If we let X denote an $n \times 2$ binary partition matrix, where $X(i, g) = 1$ if pixel i belongs to group g, $g = 1, 2$, the above criterion can be formally written in matrix form as a two-way segmentation using the constrained normalized cuts (Yu and Shi, 2004, 2003):

$$\text{maximize} \qquad \varepsilon(X) = \sum_{g=1}^{2} \frac{X_g^T W X_g}{X_g^T D X_g} \qquad (4)$$

$$\text{subject to} \qquad X \in \{0, 1\}^{n \times 2}, X1_2 = 1_n \qquad (5)$$

$$U^T X = 0 \qquad (6)$$

where 1_n denote an $n \times 1$ vector of 1's.

The near-global optimal solution is given by the eigenvectors of QPQ, where

$$P = D^{-1}W \qquad (7)$$
$$Q = I - D^{-1}U(U^T D^{-1} U)^{-1}U^T. \qquad (8)$$

While the eigensolution of QPQ takes longer than that of P (unconstrained version) to compute at each iteration, it often requires fewer iterations. We follow the eigensolution and its discretization procedures developed in Yu and Shi (2004, 2003) and their code online to obtain a binary segmentation.

We next need to address: **1)** How to compute local grouping cues A and R to achieve the desired segmentation; **2)** How to set up stitching constraints U between adjacent patches.

2.2. Local Features and Grouping Cues

To derive the attraction and repulsion cues, we start by extracting local features from the image. The features that make our dot boundaries regions of their own are not statistics which characterize local textural appearance, but patterns which characterize local geometry. Instead of measuring local convexity with curvature numbers, we describe it using a distributed relational representation, i.e., each pixel has a pixel-centric flow field, which is a sink for pixels inside the dots (intensity peaks) and a source for pixels outside the dots. We characterize cells of small convex bright regions as the sinks of local gradient fields as in (Bernardis and Yu, 2010a). Each pixel is associated with a peak direction vector p that indicates where pixels of higher intensity are located in its convex vicinity. Two pixels are attracted to the same region if their pixel-centric local gradient fields are similar, and repelled into different regions if they are of opposite types (e.g. sources and sinks). Computing the cues can therefore be divided into the following three steps.

Step 1: Consider pixel i and its neighborhood $N_d(i)$ of radius r_d (neighborhoods are taken as log-polar) as in Fig. 5a (a simple case of two separate dots, as can be appreciated by viewing the intensity profile in Fig. 5d). If neighbor $a \in N_d(i)$ can be reached in a straight line from i with non-decreasing intensity, a is a higher intensity pixel in the same convex region. Let $p(i)$ be the average direction from its a neighbors, weighted by the total non-decreasing intensity $T(i, a)$ along the straight line from i to a:

$$p(i) \propto \sum_{a \in N_d(i)} T(i, a)(L(a) - L(i)), \; |p(i)| = 1 \tag{9}$$

$$T(i, a) = \sum_{\substack{I(m_1) \leq I(m_t) \leq ... \leq I(m_k) \\ m_1 m_2 ... m_k = \text{line}(i,a)}} I(m_t) \tag{10}$$

where $L(i)$ denotes the 2D location of pixel i in the image, $I(i)$ the intensity of pixel i, and $|\cdot|$ the L_2 norm of a vector. The peak direction vector $p(i)$ thus points from i towards the core of the cell that i belongs to, i.e., the highest intensity of its local convex region. It measures the direction and distance from pixel i to the center of the cell. $T(i, a) = 0$ if no ascending path exists on the specific line (resulting in $p = 0$ at the center of the dots, i.e. where sinks occur).

Step 2: We define the *convexity feature* vector $F(i, a)$ as the inner product of $p(i)$ and $p(a)$ within a *convexity* neighborhood $N_c(i)$, measuring how much a's cell center estimate agrees with i's. The ensemble of $\{F(i, a): a \in N_c(i)\}$ is a
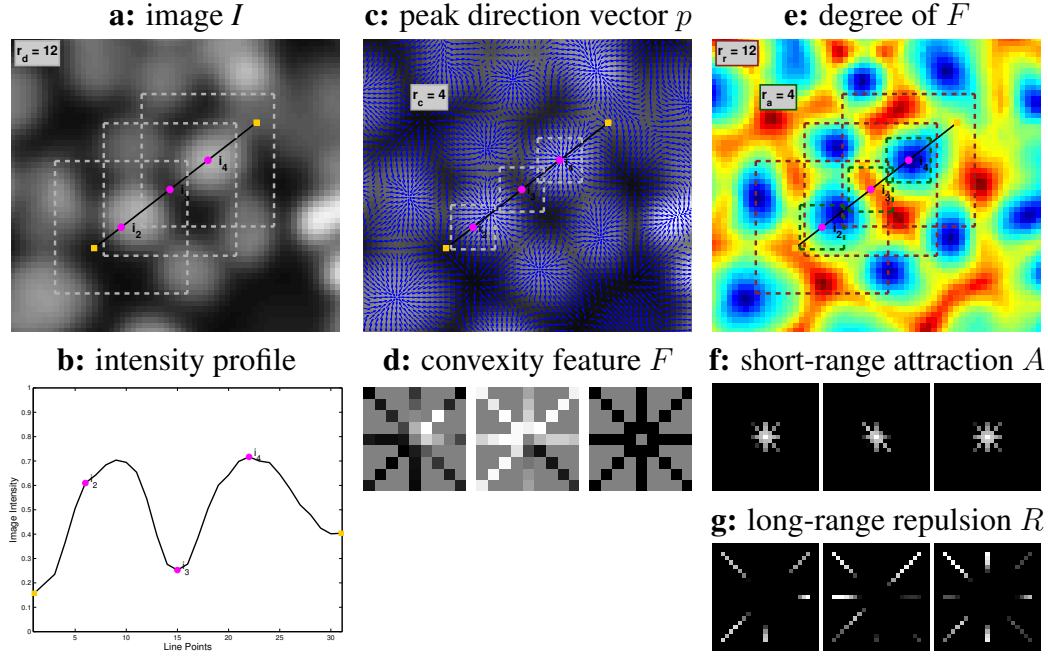
9

Figure 5: Computing pairwise features and grouping cues. **Step 1:** Given **a)** image I, with **b)** a sample intensity profile along the line intersecting a pair of separated dots, an initial radius r_d is used to compute **c)** a *peak direction vector* p at each graph node. This is achieved by taking the average of the direction from i to all its a neighbors, each weighted by the total non-decreasing intensity $T(i, a)$ along the straight line from i to a. **Step 2:** A convexity feature radius r_c is then used to compute **d)** the local *convexity feature vector* F, which characterizes where pixel i lies with respect to closest convex region. F can be visualized by looking at **e)** its degree at each i, i.e. $\sum_{a \in N_c(i)} F(i, a)$. Sinks of the flow (dot centers) result in negative values while sources are positive. **Step 3:** Finally, F is used to compute the pairwise cues: **f)** short-range attraction A and **g)** a long-range repulsion R between graph nodes within radii r_A and r_R of each other.

pixel-centric vector field (i.e. with the absolute direction of $p(i)$ factored out) that characterizes where pixel i is in the shape of a convex region, and we can use the feature similarity S to establish later pairwise pixel attraction and repulsion grouping cues:

$$F(i, a) = <p(i), p(a)>, \qquad\qquad a \in N_c(i) \qquad (11)$$

$$S(i, j) = \frac{<F(i, :), F(j, :)>}{|F(i, :)| \cdot |F(j, :)|}, \qquad\qquad j \in N_A(i), N_R(i) \qquad (12)$$

where N_A and N_R are the attraction and repulsion neighborhoods respectively. $F(i, :)$ shows how much i's neighbors agree with i on the direction the dot lies in,
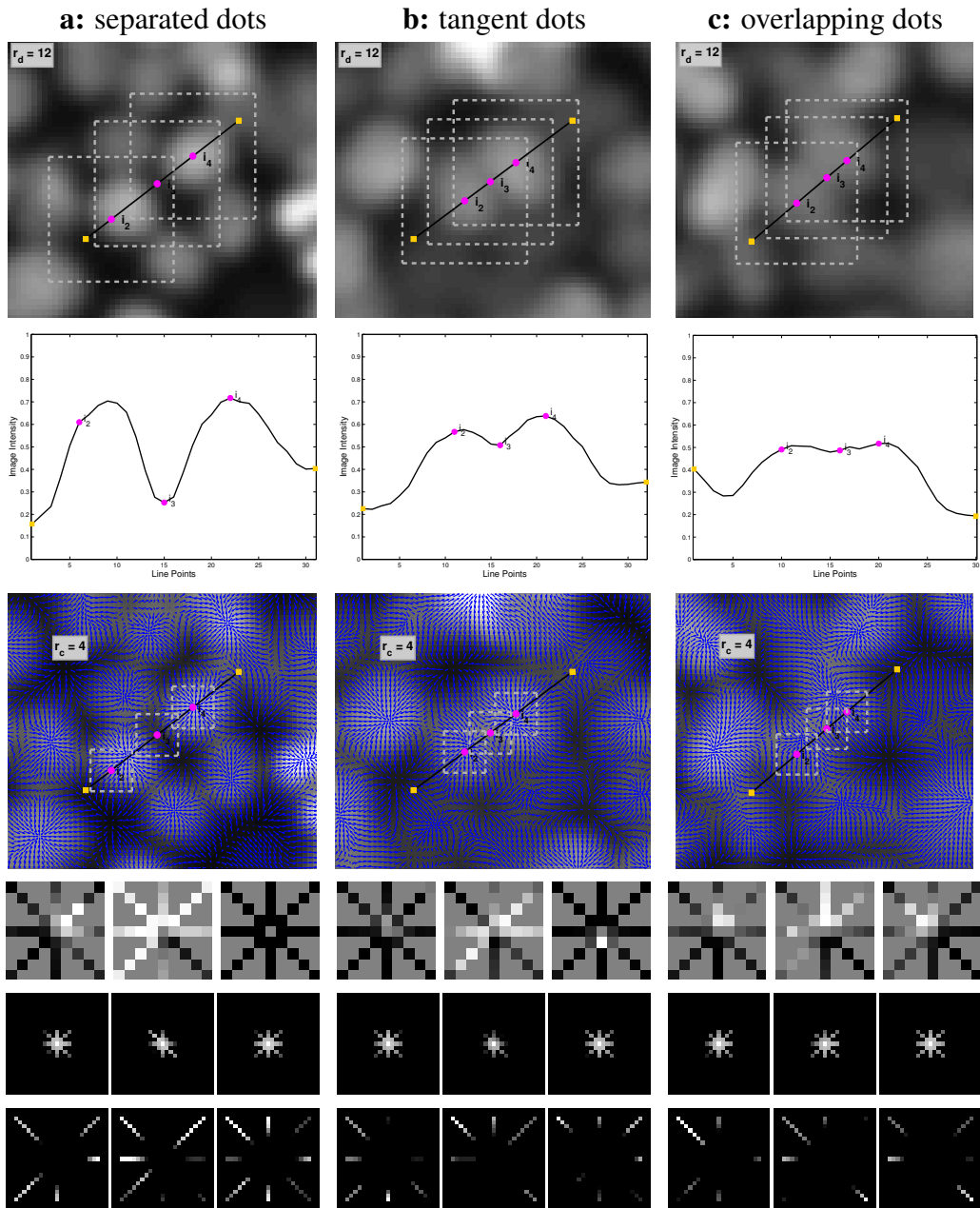
Figure 6: Local pairwise cues based on feature similarity for three typical scenarios. The initial image with intensity profiles (rows 1 and 2) of the line joining the two dot centers is used to compute the peak direction vectors (row 3). Three points (off-centered on the left dot, midpoint between the dots and on-center on the right dot) are selected to show feature vector F (row 4), attraction A (row 5) and repulsion R (row 6).

11

with $p(i)$ itself factored out. Note that $p(i) \in \mathbb{R}^2$ while $F(i,:)$ is a $2r_c \times 2r_c$ vector. $S(i,j)$ is more likely to be positive for nearby pixels inside the same dot, and negative for distant pixels between different dots, giving rise to two kinds of grouping cues. In Fig. 5 we visualize F also by computing its degree at each pixel i, i.e. $\sum_{a \in N_c(i)} F(i,a)$, so that sinks (dot centers) are characterized by negative values and sources by positive ones. We illustrate the sizes of attraction and repulsion neighborhoods N_A and N_R on this last image.

Step 3: Since only convexity cues are used to differentiate dots apart, pairwise grouping cues are based directly on the similarity feature measure S. While the attraction A only operates at short ranges, the repulsion R operates at long ranges pushing apart pixels inside dots from pixels outside and is essential for popping out disconnected regions (Yu and Shi, 2001; Bernardis and Yu, 2010a). The short range attraction A and long-range repulsion R are then defined as:

$$A(i,j) = e^{-\frac{1-S(i,j)}{\sigma}}, \qquad\qquad |L(j) - L(i)| \leq r_A \qquad (13)$$

$$R(i,j) = \frac{1-S(i,j)}{2}, \qquad\qquad |L(j) - L(i)| \leq r_R \qquad (14)$$

Figure 6 illustrates how A and R change as two dots become closer together.

2.3. Stitching Constraints U

A two-way node partitioning can be described by a $n \times 2$ binary partition matrix X, where n is the total number of pixels, $X(i,1)$ and $X(i,2)$ indicating whether pixel i belongs to the inside or outside of a cell.

Our stitching constraints are imposed on the partition indicator X that is to be solved in the optimization. If pixels a and b are known to belong in the same region, we have the constraint $X(a,:) = X(b,:)$, or $X(a,:) - X(b,:) = 0$. All these equations can be described in a linear constraint $U^T X = 0$, where $U(a,k) = 1$, $U(b,k) = -1$ is the k-th constraint that a and b belong to the same region.

The initial first-round patch segmentation does not require any constraints U, although simple intensity thresholding or initial seeds can be introduced. In the second-round patch segmentation, where each patch has been segmented, U comes from a mutual agreement analysis of X in the overlapping regions between neighboring patches: pixels for which two adjacent patches agree on the segmentation become either foreground or background pixels. Only a sparse set of pairwise constraints are needed to ensure that two neighboring patches will have consistent segmentations in their overlapping areas.
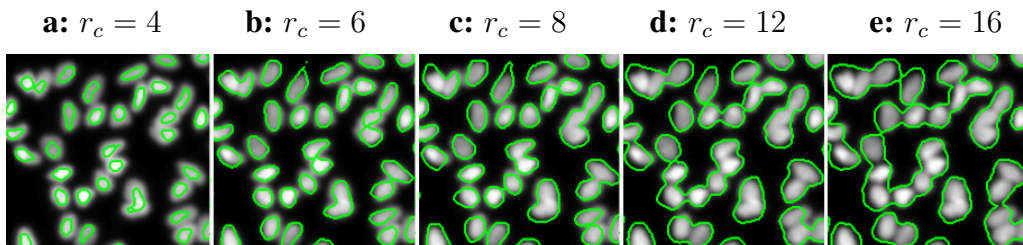
| **a:** $r_c = 4$ | **b:** $r_c = 6$ | **c:** $r_c = 8$ | **d:** $r_c = 12$ | **e:** $r_c = 16$ |

Figure 7: Finding larger dots by increasing the convexity feature radius r_c. As the attraction and repulsion radii remain fixed, we display the change of segmentation boundaries as a function of increasing r_c. This shows how choosing a smaller template allows detection of all dots regardless of their exact sizes with the same set of parameters. As the radius increases, the boundary expands to include the entire dot shape after which neighboring regions start merging together.

Compared with traditional normalized cuts, the increase in time complexity is negligible if the number of constraints is small (Yu and Shi, 2004). Additionally, the space complexity is reduced using patch segmentation with stitching constraints, as the image is broken down into smaller patches and finding numerous small regions becomes possible in a single two-way segmentation.

3. Experiments

We implement our algorithm in MATLAB and present results on different datasets of dot-like structures encountered in human pathology studies. The experimental section starts with a detailed analysis of parameter selection followed by details on the selection and effects of stitching constraints. We extend our model to thin and elongated structures to illustrate the benefits of the segmentation subject to stitching constraints beyond the simpler dot structures. We benchmark our method for the epithelial and embryonic cells against other commonly used segmentation algorithms. Finally, we also compare our dot segmentation method with state of the art domain-specific segmentation algorithms presented at the 2010 International Conference for Pattern Recognition (ICPR) as part of the Pattern Recognition in Histopathological Images contest for counting lymphocytes on histopathological images.

3.1. Segmentation Parameters

The setup of pairwise cues for convex region detection requires four radii, first to compute image features and then to compute pairwise cues:
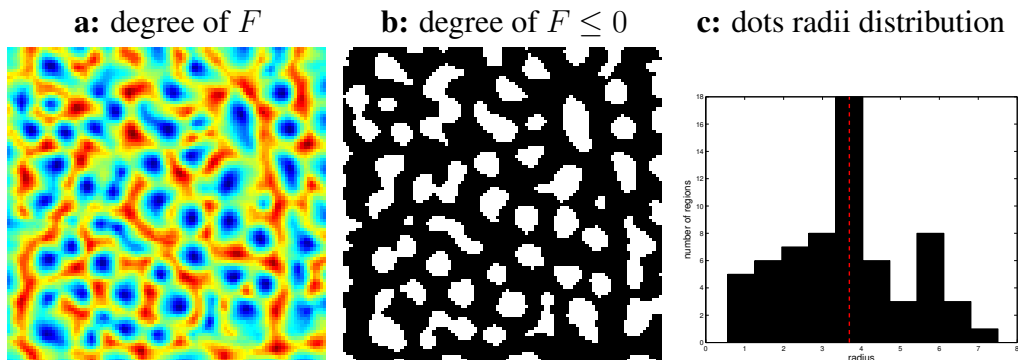
a: degree of F **b:** degree of $F \leq 0$ **c:** dots radii distribution

Figure 8: Automatic core radius estimation for parameter selection. Starting with **a)** the 2D visualization of the degree of the feature vector F, where negative values (blue) represent the sinks (dots) while positive ones (red) are background. We obtain **b)** disconnected components (i.e. possible dots candidates) by simply thresholding the negative values. After computing a radius estimate for each component, we look at **c)** the radii distributions for each patch (shown in the histogram) and we take the peak as possible core radius estimate (red dotted line).

r_d : radius of the neighborhood from which the peak direction vector p is computed. Beyond a certain radius, r_d gives a constant p, so it can be chosen independently of the dot scale.

r_c : radius of the neighborhood used to compute the feature vector F, it encodes the local convexity of the neighborhood. Increasing r_c, while keeping all other parameters fixed, allows to look for larger shapes (Fig. 7). As r_c increases, the boundary expands to first include the entire dot and then to merge adjacent dots together.

r_A : radius used to compute short-range attraction cues. Increasing r_A brings nearby pixels together, so it should be at least comparable to the dot size.

r_R : radius that determines the extent of long-range repulsion cues. The absence of repulsion yields a segmentation similar to the traditional normalized cuts result shown in (Fig. 3e).

Intuitively, r_c has to entirely contain the dot to set a proper 'template' size for the feature vector F computation, while r_A and r_R have to be large enough to capture shape information and surrounding information respectively. r_c, r_A and r_R can be therefore defined in terms of a core radius r_0 that represents the size of the average dot. To compute the core radius, we use the heuristics depicted in Fig. 8. Starting with degree of the convexity feature F (illustrated in Fig. 5), we threshold

14

a: segmenting individual cilia: $r_c = 4, r_R = 20, r_A = 4$
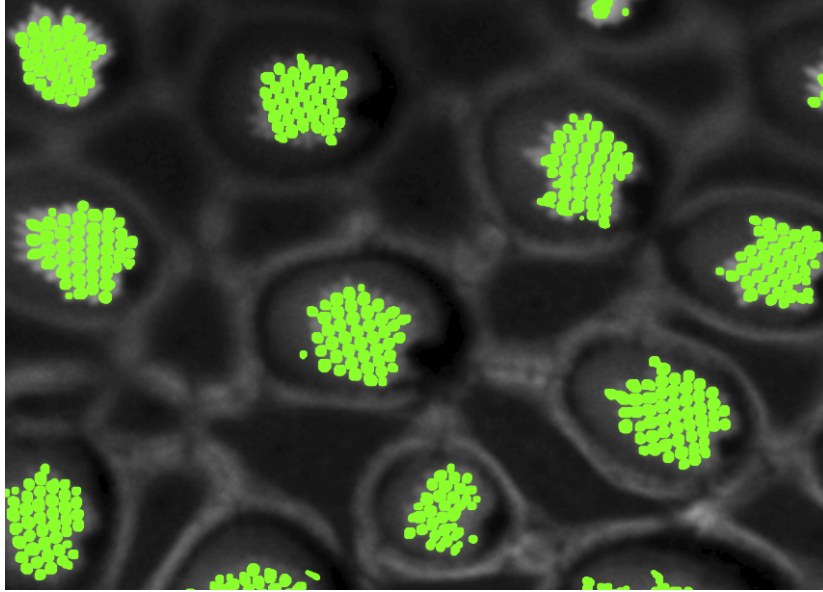
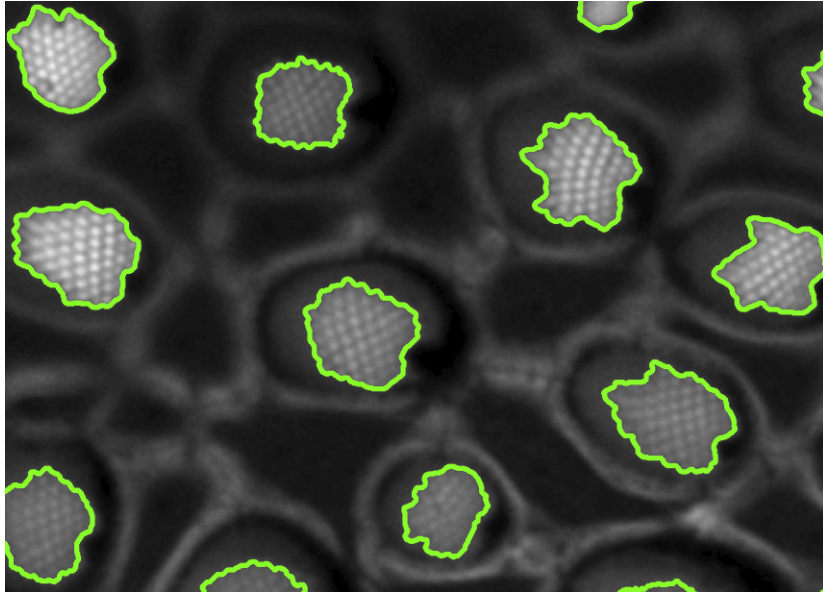**b:** segmenting haircell bundles: $r_c = 30, r_R = 40, r_A = 10$

Figure 9: Hierarchical dot structures and radii parameters. Changing the template size r_c is not sufficient when looking for significantly larger objects. In order to add proper repulsion between the regions, the overall radius r_R must also be increased. In **a)** we have results with the standard set of parameters and in **b)** with increased repulsion and template radii.
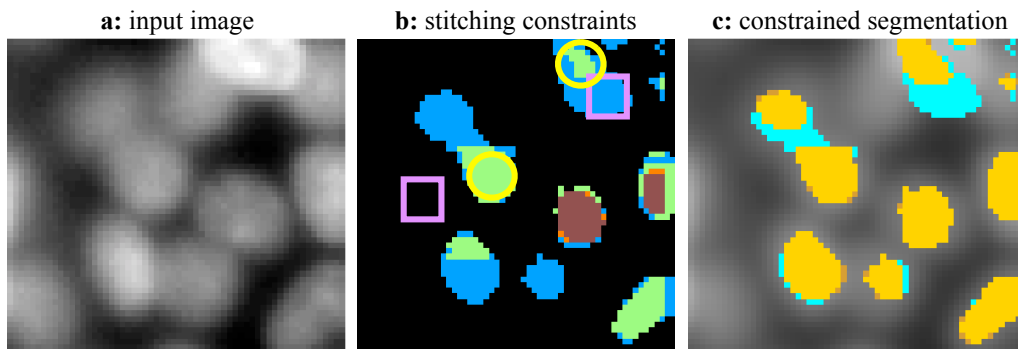
15

Figure 10: Beyond simple stitching. **b)** Constraints are derived from mutual agreement analysis on adjacent patch segmentations from a previous round (blue, green, or orange/maroon indicate 1, 2 or more patches agreement). Stitching constrains cells (circles) and background (squares) to group together respectively. **c)** Constrained segmentation (gold) improves the initial segmentation (cyan) by cleaning up spurious small regions, separating conjoined cells, and refining their boundaries.

the negative values to obtain disconnected components that represent possible dot regions and estimate the radius of each component. We take the radius associated with peak of the radii distribution as core radius estimate, restricting the radius to be at least 4 pixels. The rest of the parameters can then be set accordingly: $r_c = \max(4, 1.5r_0), r_a = \max(4, r_0)$ and $r_R = \max(20, 3r_0)$. In special case scenarios, such as the hierarchical dot phenomenon illustrated in Fig. 9, parameters have to be manually tuned to extract the desired structures.

3.2. Constraint Propagation and Elongated Structures

Segmentation subject to stitching constraints is not simple stitching. The constraints set on the overlapping regions propagate to the interior regions and are able to correct improper initial segmentation, as shown in cyan in Fig. 10. A few constraints are sufficient to correct mistakes, such as two cells erroneously segmented as one, spurious segment cleanup and boundary refinement. We choose an overlap size that is able to contain at least one entire dot in order to be informative. For all results, we use 20 pixels of overlap and patches of approximately 256×256 pixels. Clearly, larger overlap would allow more constraints between cells but one must take into account the tradeoff of having to compute more patches and the complexity increase due to increasing the number of constraints.

Recalling the EM data from the medulla brain region of the drosophila fly from Fig. 1, there is often also a need to segment small and thin structures. The strength of our method is that it can be extended to a larger set of geometries. The
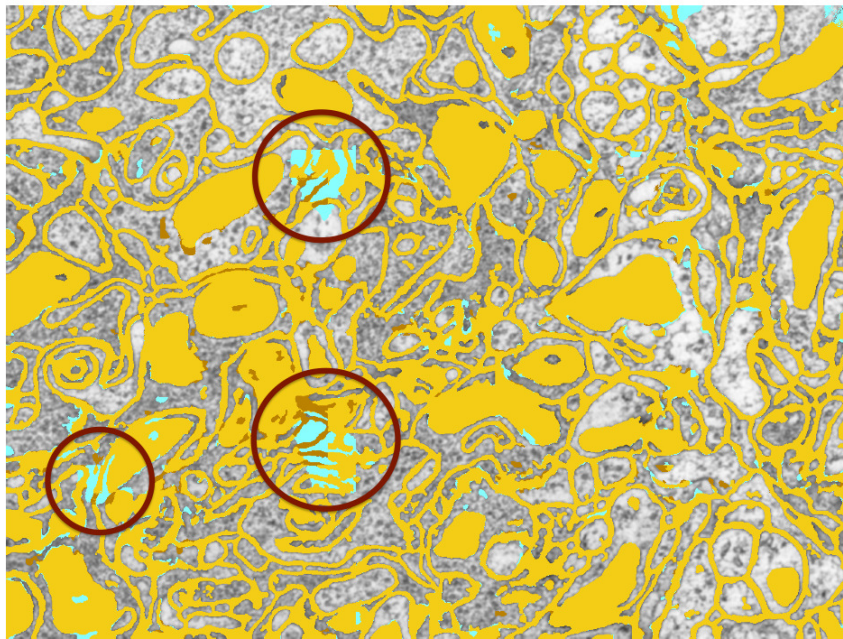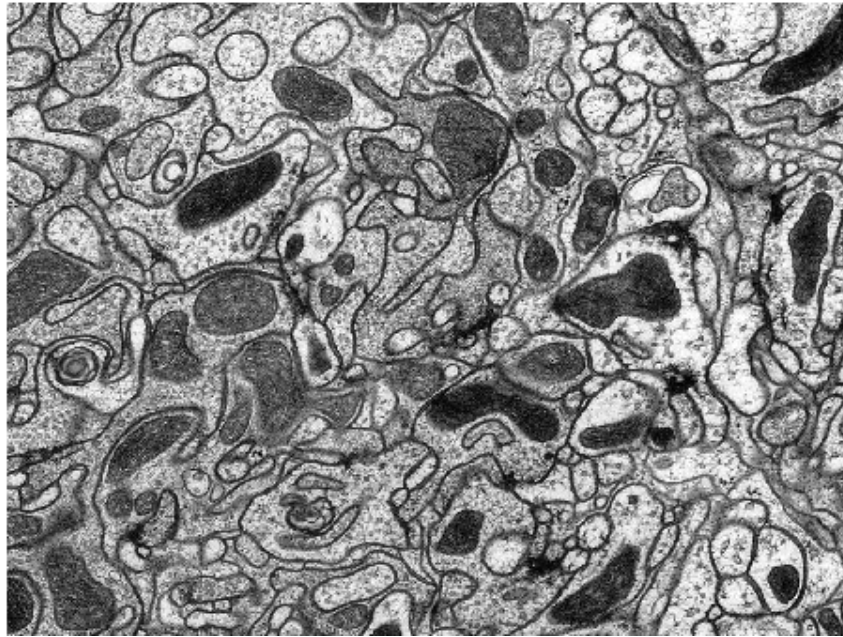
16

Figure 11: Thin structures and constraint propagation.The initial segmentation (cyan) is corrected by the constrained segmentation (brown). The segments where the two segmentations agree are depicted in gold. Constrained segmentation is more than simple stitching. Its effectiveness can be better seen for membrane structures that have a geometry that inherently propagates throughout the image to interior regions which are farther away and not located within the overlapping regions.

same visual popout applies, since salient regions have a common repelling background, but in order to extract the more complicated geometries, new attraction and repulsion cues have to be defined.

To avoid breaking each line into many small convex regions, we add an additional pairwise component $\tilde{S}(i,j) = \exp\left(-\frac{1}{\sigma}(\Phi(i) - \Phi(j))^2\right)$ based on the degree $\Phi(i) = \sum_{a \in N_c(i)} F(i,a)$ of the feature vector F, introduced in Section 2.2. The pairwise attraction and repulsion cues are now a function of:

$$T(i,j) = \beta S(i,j) + (1-\beta)\tilde{S}(i,j) \tag{15}$$

where β a constant parameter and $S(i,j)$ is the similarity feature measure from Eqn. 12. The pairwise attraction and repulsion cues are then defined as:

$$R(i,j) = |\alpha - T(i,j)|, \qquad |L(j) - L(i)| \le r_A,\ T < \alpha \tag{16}$$
$$A(i,j) = |T(i,j) - \alpha|, \qquad |L(j) - L(i)| \le r_R,\ T \ge \alpha. \tag{17}$$

For the images shown, we empirically chose parameters $\sigma = 0.25, \beta = 0.35$ and $\alpha = 0.35$. As before, attraction A only operates at short ranges and repulsion R operates at long ranges, to give a total effective weight W is $A - R + D_R$.

The benefits of the stitching are highlighted in Fig. 11. The initial (cyan) segmentation is improved after the constraints are enforced to obtain a final (gold) segmentation that corrects segmentation mistakes also outside the overlapping regions. The yellow segmentation denotes where the segmentations, before and after the constraints, agree. The elongated structures then allow the stitching benefits to be more pronounced as the constraints propagate within the regions to the interior of the image. The final results for the original 1800×1800 pixel image, obtained using a 8×8 stitching grid, are shown in Fig. 18. A post-processing step, dividing regions with larger diameters, is added to discriminate between membranes (gold) and all the other structures (blue).

3.3. Benchmark with General Segmentation Methods

To compare our results with other commonly used segmentation algorithms, we run our method on images (a selection was given in Fig. 2) from two datasets of dot-like structures encountered in human pathology: 1) 512×512 pixels bright field and fluorescent images of epithelial and embryonic cells; and 2) 600×900 pixels histopathology images from various tumor-related lesions. Figures 14, 15, 16 and 17 illustrate sample results of our segmentation method.

We benchmark our results against human labeled dot centers. Given m ground-truth dot centers and n segment centers for an image, let D_{ij} be the Euclidean

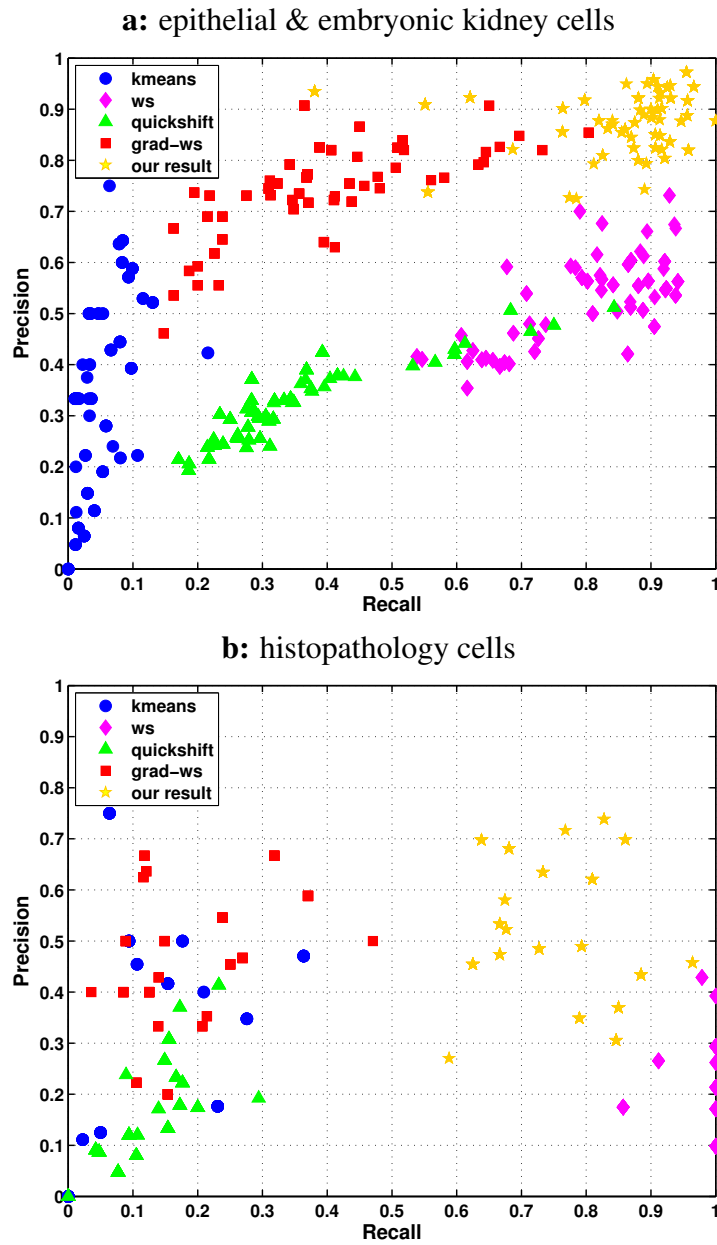**a:** epithelial & embryonic kidney cells

b: histopathology cells

Figure 12: Precision-recall statistics for **a)** epithelial cells images and **b)** histopathology images. Our method (yellow square dots, upper right corner) has better precision and recall overall when compared with quickshift, k-means, gradient-based and standard watershed.

distance between dot i and segment j. If it is less than a certain radius threshold ρ, we consider (i, j) a matched detection. We then define:

$$\text{precision} = \frac{\#\{j : \min_{i=1}^{m} D_{ij} \leq \rho\}}{n} = \frac{\text{\# nearest dots within radius } \rho}{\text{\# segments}}$$

$$\text{recall} = \frac{\#\{i : \min_{j=1}^{n} D_{ij} \leq \rho\}}{m} = \frac{\text{\# nearest segments within radius } \rho}{\text{\# dots}}$$

The precision measures the proportion of true dots among all the segments, and the recall measures the proportion of segments among all the true dots. We compare our method with the following segmentation methods:

k-means: We use MATLAB's built-in function. It clusters pixels based on their intensity values, thus has trouble separating conjoined like-intensity cells and increasing k only leads to clustering instability.

Watersheds: Watershed is directly applied to either the intensity image (MATLAB's built-in function) or the gradient magnitudes (with radius 5) of the image. While the standard watershed results tend to be over-fragmented in the presence of local intensity fluctuation, the gradient-based watershed results tend to miss many dots of weak contrast but improves on the precision.

Quickshift: We use the online code by Vedaldi and Fulkerson (2010). Analogous to meanshift, quickshift can be used to partition an image into a set of superpixels. It enhances intensity differences, but it is sensitive to scale choices and cannot break up dots based on convexity.

The precision-recall statistics in Fig. 12 shows that our method works better than others at segmenting small regions in terms of both precision and recall. Our stitching constraints can be appreciated by comparing the quality of segmentation without and with constraints: While there is no significant improvement in the recall, there is an average improvement of 0.04 in the precision.

3.4. Benchmark with Domain Specific Methods

We also compare our dot segmentation method with state of the art domain-specific segmentation algorithms presented at the 2010 International Conference for Pattern Recognition (ICPR) as part of the Pattern Recognition in Histopathological Images contest for counting lymphocytes on histopathological images. For this benchmark we use an additional set of 100×100 pixels histopathology images, very kindly provided by the contest organizers. Given the 100×100 pixels dimensions, we did not divide the images and simply ran the segmentation once.
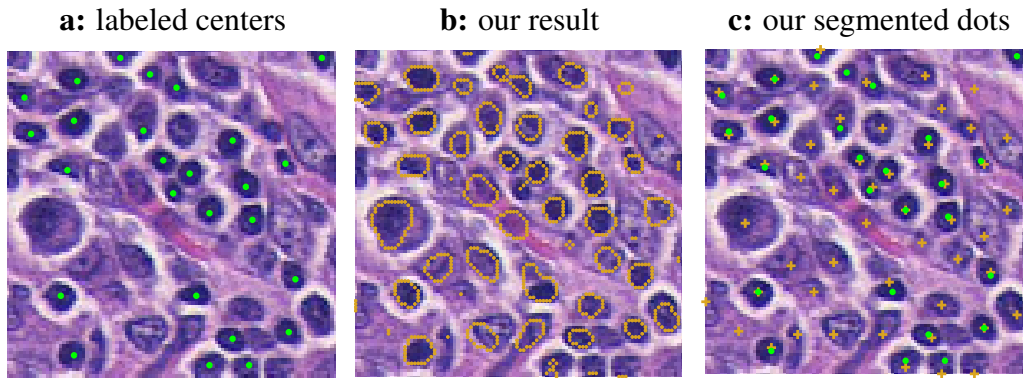
a: labeled centers   **b:** our result   **c:** our segmented dots

Figure 13: Sample results from the 'Counting Lymphocytes on Histopathology Images' contest dataset (Fatakdawala et al., 2010; Basavanhally et al., 2010), courtesy of Anant Madabhushi at Rutgers. **a)** Labeled ground-truth was provided for comparison. **b)** Our results: boundaries of our computed segments and **c)** the centers (gold pluses) extracted from them together with the ground truth (green dots). Performance details are given in Table 1. We display our results on the inverse images, for clearer display. As in the contest, the goodness of the boundaries was not measured as specific boundaries cannot be identified to begin with. Our algorithm is not domain specific, which explains why we pick up many more dots that are not identified by experts as lymphocytes. A post-processing step, for example based on color, can be applied to refine the results.

Each image took approximately 2 seconds to segment. Sample results are shown in Fig. 13. Finally, we compare performances in Table 1 with the algorithms presented at the contest:

Method 1: (Kuse et al., 2010) based on extracellular matrix segmentation, i.e. mean shift clustering for color approximation followed by hsv space thresholding. Texture features are extracted from the cells and then used to train a SVM classifier to find the lymphocytes.

Method 2: (Graf et al., 2010) based on connected components. A first processing step involves thresholding and morphological operators to improve the quality of the images for recognition; a second recognition step then extracts the lymphocytes with a template matching method.

Method 3: (Cheng et al., 2010) starts with a segmentation based on multi-phase level sets, followed by morphological operations to clean-up the image of small spurious regions. Features are then used to identify the target cells.

Method 4: (Panagiotakis et al., 2010) based on the estimation of a mixture of Gaussians for determining the probability distribution of the principal image component. Lymphocyte are detected after post-processing to eliminate small regions, using a transferable belief model for knowledge modeling.

21

Ranking	Group	m_d	s_d	m_N	s_N
1	Kuse et al. (2010)	3.04	3.40	14.01	4.4
2	Panagiotakis et al. (2010)	2.87	3.80	14.23	6.3
3	Graf et al. (2010)	7.60	6.30	24.50	16.2
4	Cheng et al. (2010)	8.10	6.98	26.67	12.5
our results ($\rho = 5$)		3.22	3.92	5.40	3.68
our results ($\rho = 4$)		2.84	2.89	8.20	4.75
our results ($\rho = 3$)		2.20	1.80	12.90	5.38
our results ($\rho = 2$)		1.12	0.71	16.75	7.47

Table 1: Our results compared with the finalists of the PR in HIMA (ICPR 2010) contest performance ranking provided online, for the 'Counting Lymphocytes on Histopathology Images' dataset. The ranking of performance, where m and s denote the mean and standard deviation, respectively. For all numbers, smaller numbers represent a better result. The criteria are (a) the Euclidean distance d between the ground truth and the result provided by the participants; (b) the absolute difference between the true number of cells and N the number of cells found. We show our results for tolerances $\rho = 2, 3, 4, 5$ pixels from the center. Our results are very intuitive. Decreasing the tolerance allows to find less true dots (higher m_N) but with higher precision (smaller d). For each choice of radii, we perform well when compared with the other proposed methods.

The criteria for evaluation are (a) the Euclidean distance between the ground truth and the segmented lymphocytes; (b) the absolute difference between the true number of cells and the number of cells found, denoted by N. m and s denote the mean and standard deviation, respectively. As before, we consider (i, j) a matched detection if the segment center is within a ρ threshold from the ground-truth dot (we use tolerance radii $\rho = 2, 3, 4, 5$ pixels). For all numbers, smaller numbers represent a better result. We are not able to distinguish lymphocytes from the other dot structures present because our method is not domain-specific. For fairness we stress that we did not have different training and testing sets. We used one image to test our automatic core radius estimator and then ran the algorithm on the entire image set. For each tolerance radius, we perform well when compared with the other proposed methods. As intuitively expected, decreasing the tolerance allows to find less true dots (higher m_N) but with higher precision (smaller d) and vice-versa, so the threshold can be fixed according to desired balance.

4. Summary

Segmenting small structures in a large image presents a scale dilemma between the image size and the segment size. Our approach resolves this by decou-

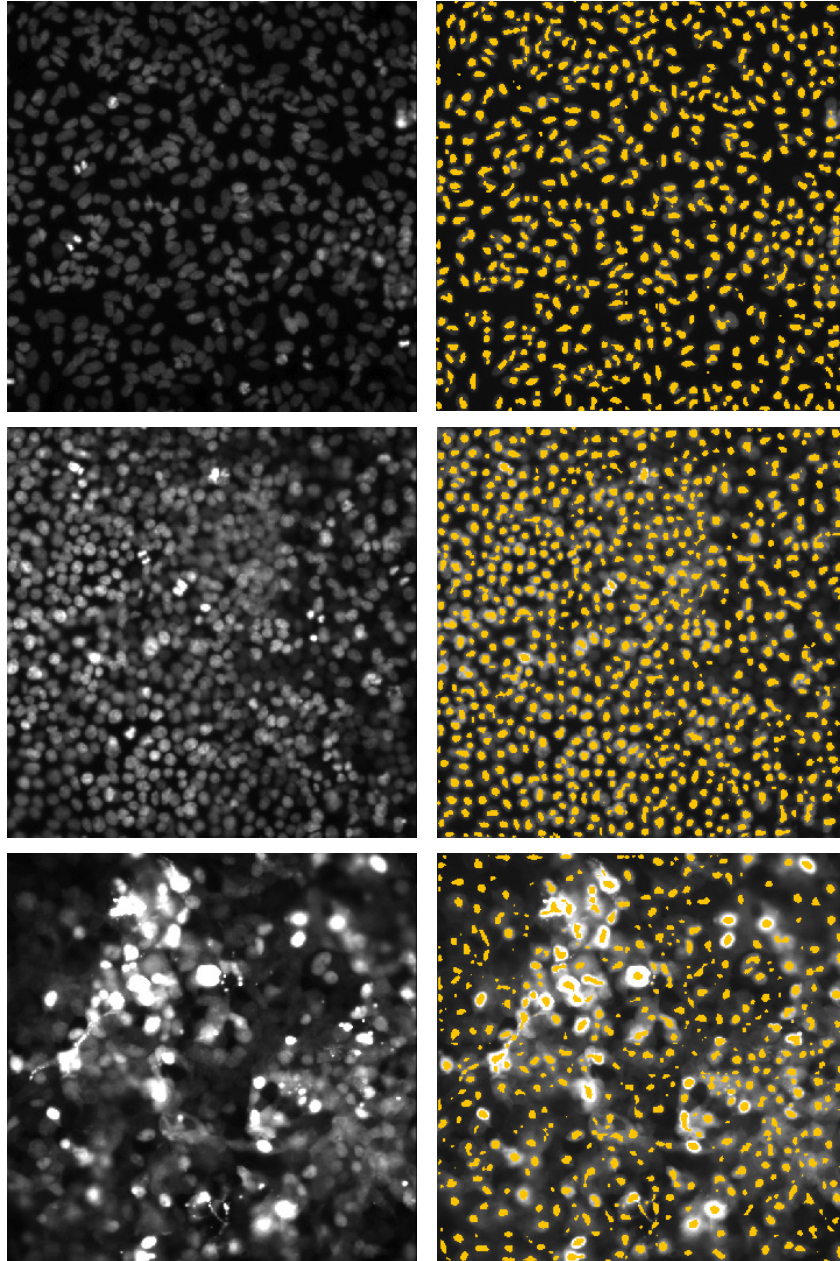**a:** A549 & HEK293T cells　　　　**b:** our results

Figure 14: Results (b) on human alveolar basal epithelial A549 cells and embryonic kidney HEK293T cells (a). A quantitative measure of our segmentation is given in Fig. 12.
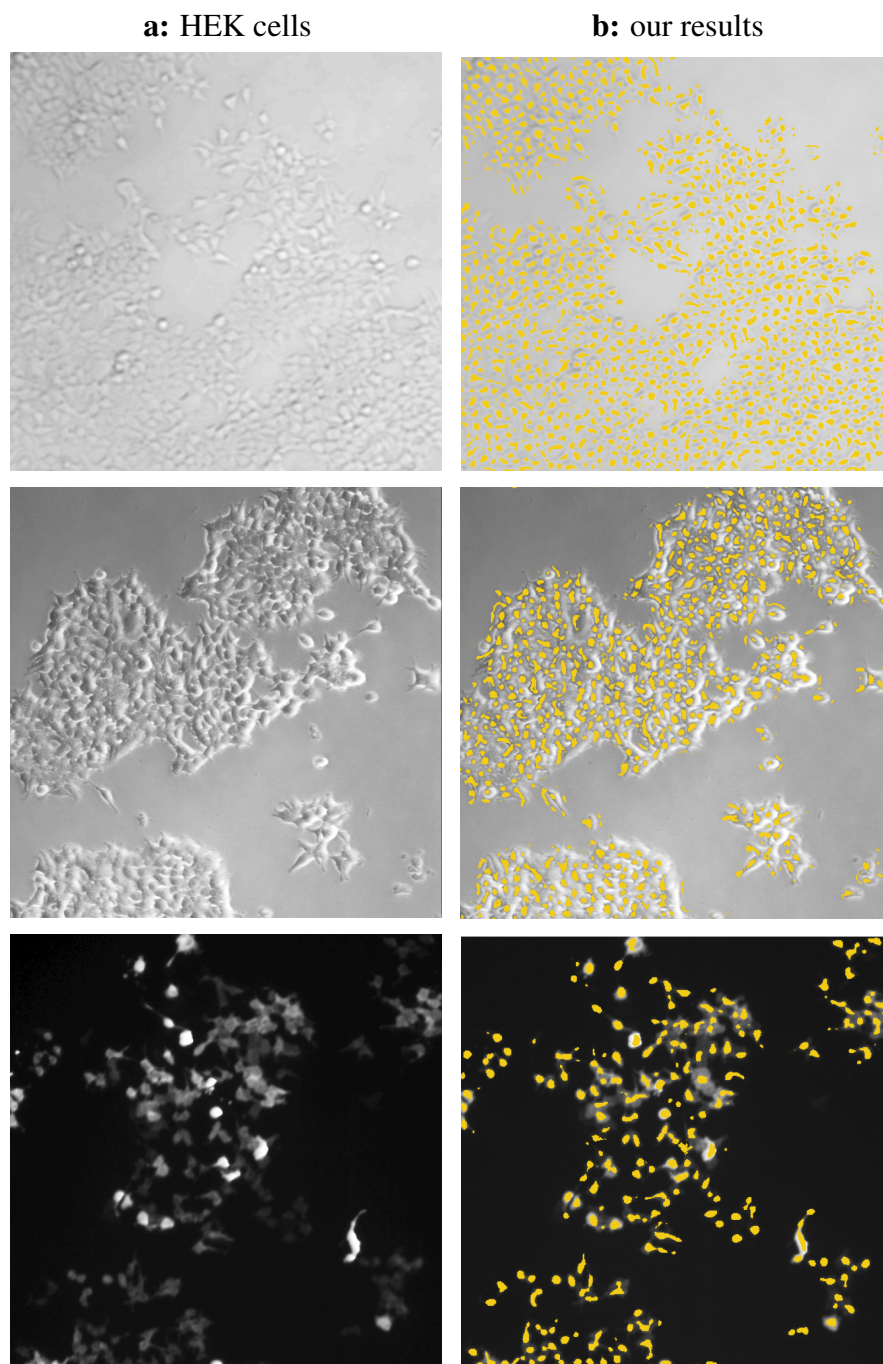
a: HEK cells   **b:** our results

Figure 15: Results (b) on HEK cells (a) with a variety of 'convex shapes'. Our method pops out all the cells in these images with the same parameters. Minimal post-processing with morphological operations allows to mask out the 'flat' background regions.

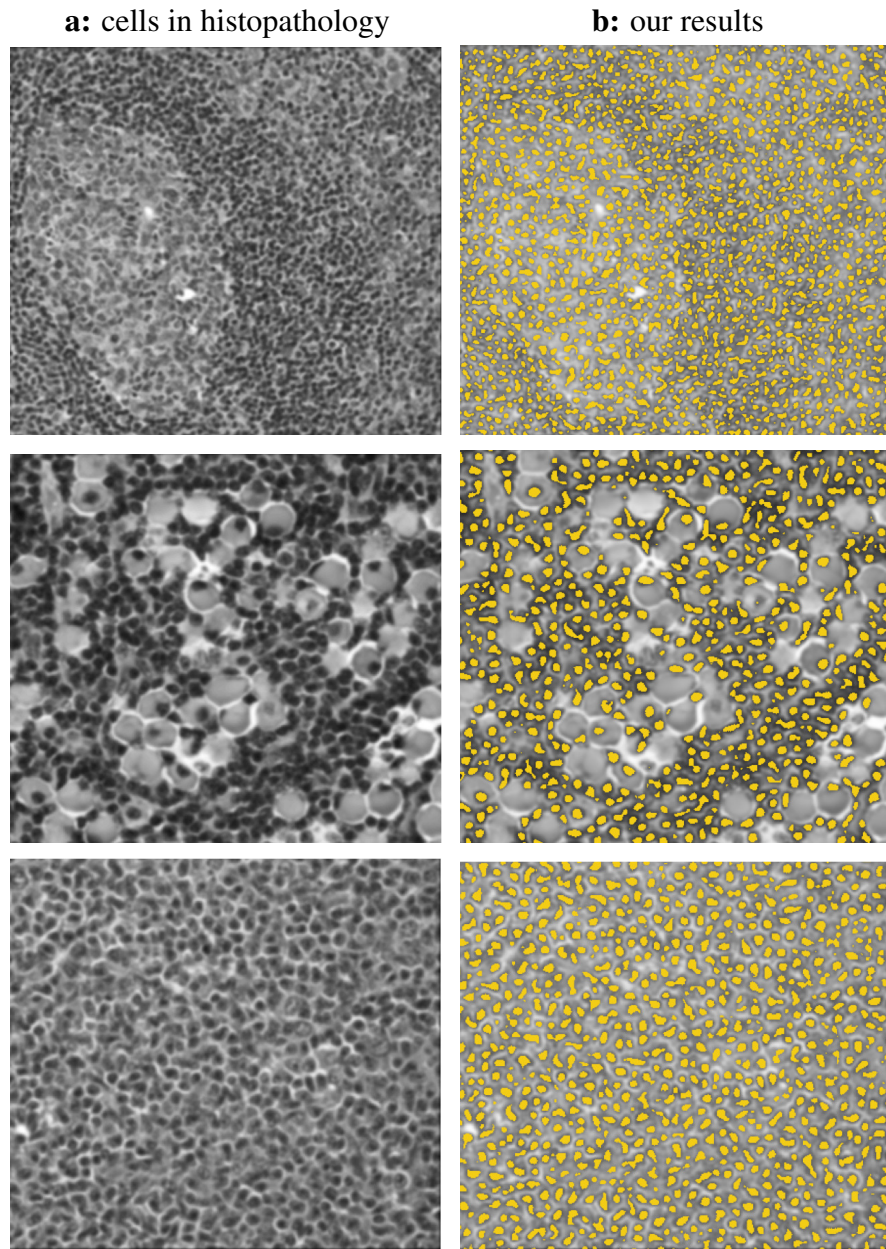**a:** cells in histopathology   **b:** our results

Figure 16: Results (b) on histopathology images (a) depicting tumor-like lesions in the spleen (top,bottom) and in mediastinum (center) containing 'dots' of various sizes and textures. Our method pops out all the cells in these images with the same parameters and no post-processing.

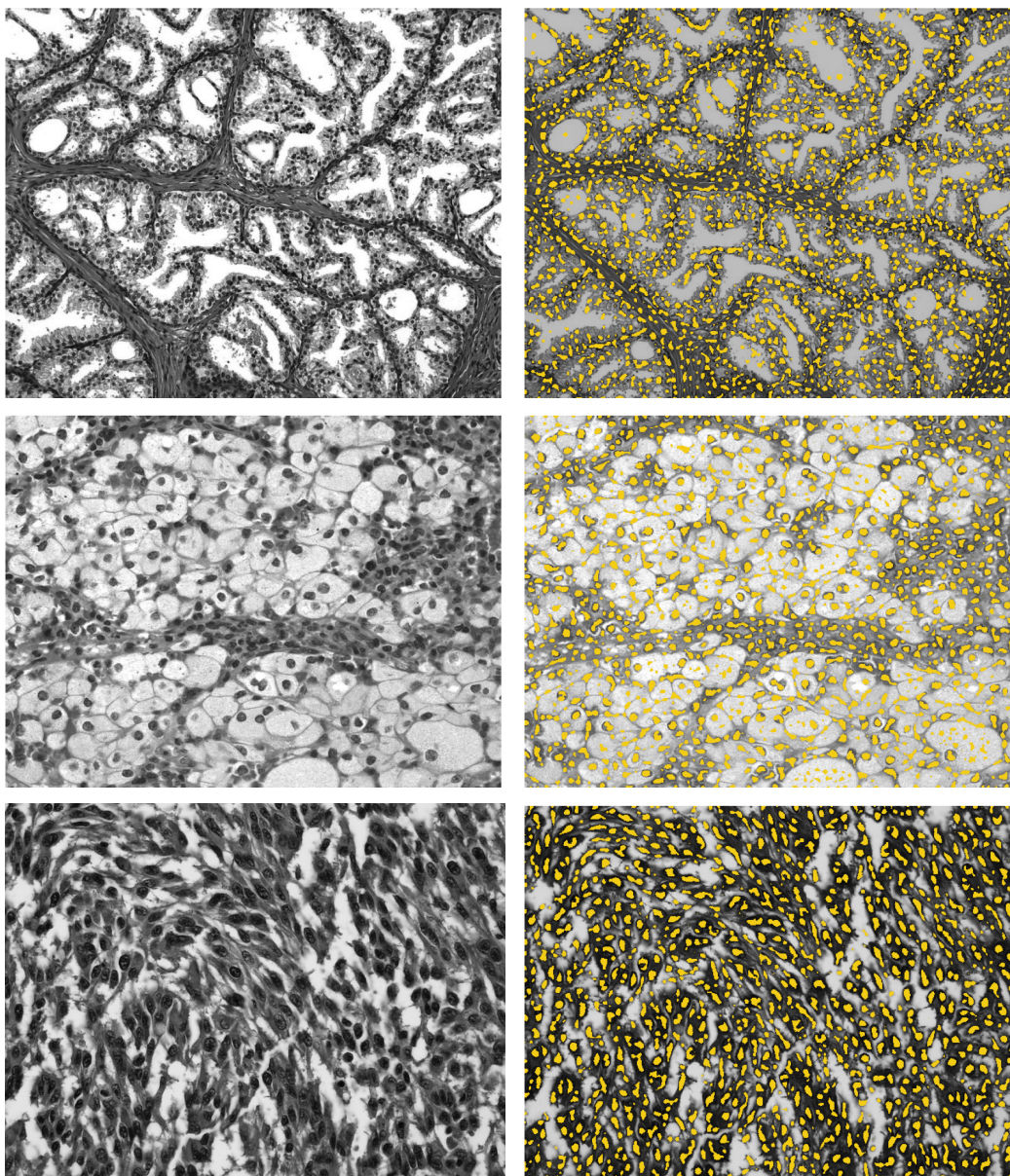**a:** cells in histopathology   **b:** our results

Figure 17: Results (b) on histopathology images (a) depicting tumor-like lesions in the ovaries (top), spleen (center) and prostate (bottom) containing 'dots' within a variety of background structures. Our method pops out all the cells in these images with the same parameters and no post-processing. Images were darkened for final result display only.
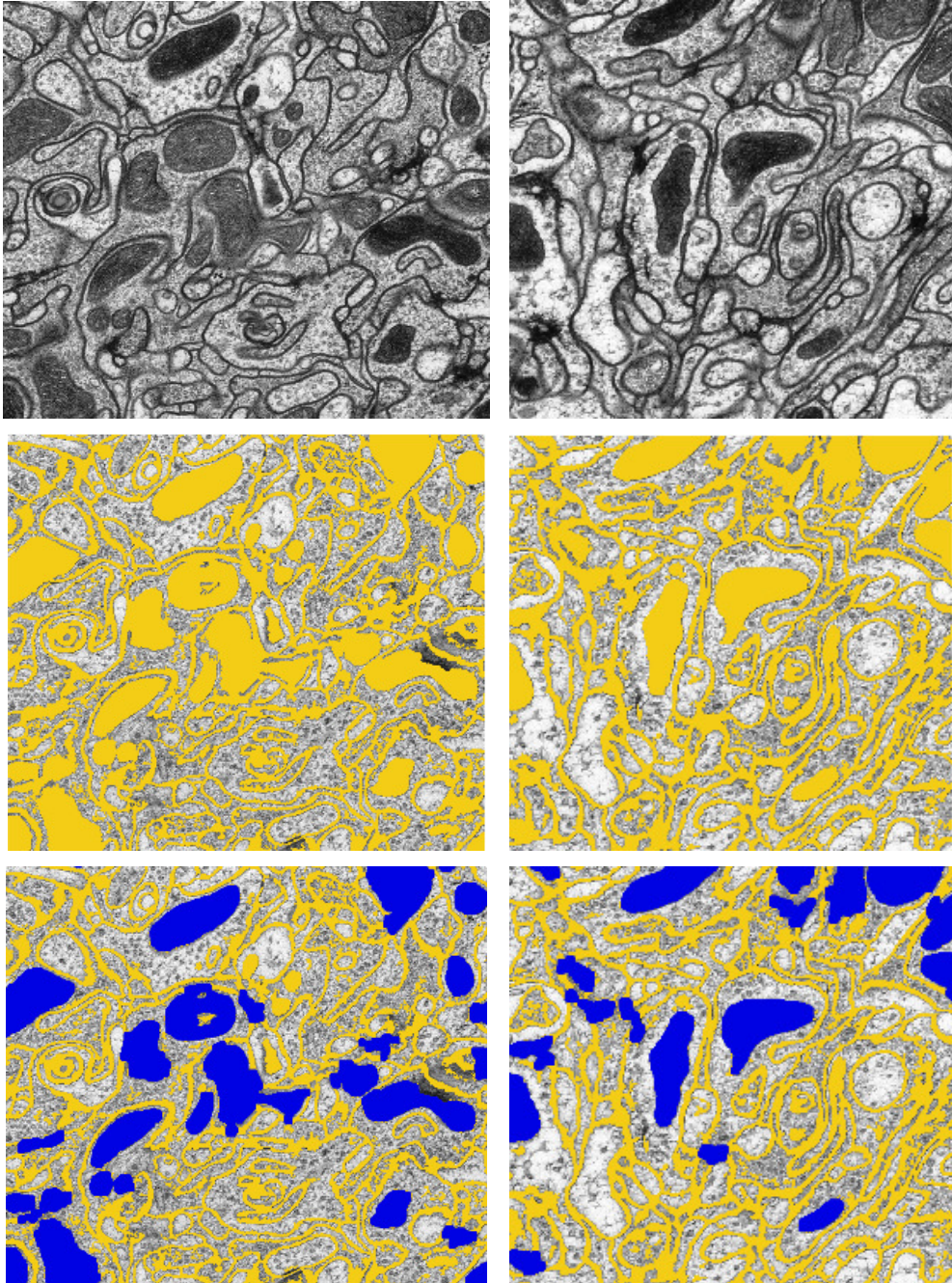
Figure 18: Results on EM brain sections of drosophila fly. Extending the weights beyond the simple 'dots' structures shows the potential of our method for a large variety of microscopic images. Images are approximately 900x900 cutouts (for display purpose) of the original 1800x1800 pixel image. We differentiate large regions from thin membranes with a simple sequence of morphological operations to obtain (bottom row) thin membranes in gold and mitochondria in blue.

pling the two sizes in constrained patch segmentations. Although segmentation subject to stitching constraints could work with any patch segmenter, a spectral graph partitioning formulation naturally integrates stitching constraints together with the foreground/background patch segmentation. The experimental section features a detailed parameter analysis to segment dots of different shapes and sizes, and illustrations of the different cases that arise in segmenting dots. We benchmark our results with general segmentation algorithms and we also compare our method with domain-specific algorithms presented at the 'Counting Lymphocytes in Histopathology Images' contest at ICPR 2010. Our method outperforms state-of-the-art results in both precision and recall by identifying all faint and conjoined cells simultaneously in a two-way segmentation, without need for post-processing.

References

Barash, D., Comaniciu, D., 2004. A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. Image Vision Comput. 22, 73–81.

Barbu, A., Zhu, S.C., 2003. Graph partition by swendsen-wang cuts. In: Proceedings of the Ninth IEEE Computer Society Conference on International Conference of Computer Vision- Volume 2, ICCV 2003, 320–.

Basavanhally, A., Ganesan, S., Agner, S., Monaco, J.P., Feldman, M.D., Tomaszewski, J.E., Bhanot, G. and Madabhushi, A., 2010. Computerized image-based detection and grading of lymphocytic infiltration in HER2+ breast cancer histopathology. IEEE Transactions Biomedical Engineering, 57, 3, 642–53.

Bernardis, E., Yu, S.X., 2010a. Finding dots: Segmentation as popping out regions from boundaries. In: Proceedings of the Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, IEEE Computer Society, 199–206.

Bernardis, E., Yu, S.X., 2010b. Segmentation subject to stitching constraints: Finding many small structures in a large image. In: Proceedings of the 13th International Conference on Medical image computing and computer-assisted intervention: Part I, MICCAI 2010, Springer-Verlag, Berlin, Heidelberg, 119–126.

Cheng, J, Veronika, M, Rajapakse, J, 2010. Identifying Cells in Histopathological Images. In: Proceedings of the ICPR 2010 Contests, Springer.

Couprie, C., Grady, L., Najman, L., Talbot, H., 2009. Power watersheds: a new image segmentation framework extending graph cuts, random walker and optimal spanning forest. In: Proceedings of the 12th International Conference of Computer Vision, ICCV 2009, 731–738.

Cour, T., Benezit, F., Shi, J., 2005. Spectral segmentation with multiscale graph decomposition. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, IEEE Computer Society, Washington, DC, USA. 1124–1131.

Cousty, J., Bertrand, G., Najman, L. and Couprie, M., 2009. Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31, 1362–1374.

Fatakdawala, H., Basavanhally, A., Xu, J., Bhanot, G., Ganesan, S., Feldman, M., Tomaszewski, J. and Madabhushi, A., 2010. Expectation-maximization-driven geodesic active contour with overlap resolution (EMaGACOR): application to lymphocyte segmentation on breast cancer histopathology. IEEE Transactions on Biomedical Engineering , 57, 7, 1676–89.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient graph-based image segmentation. International Journal of Computer Vision, 59, 167–81.

Galun, M., Sharon, E., Basri, R., Brandt, A., 2003. Texture segmentation by multiscale aggregation of filter responses and shape elements. In: Proceedings of the Ninth IEEE International Conference of Computer Vision- Volume 2, ICCV 2003, 716–.

Geman, S. and Geman, D., 1990. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. Readings in uncertain reasoning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 452–72.

Grady, L., 2006. Random Walks for Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28, 11, IEEE Computer Society, Los Alamitos, CA, USA, 1768–1783.

Grady, L., 2006. Fast, Quality, Segmentation of Large Volumes - Isoperimetric Distance Trees. In: Proceedings of the 9th European Conference on Computer Vision, ECCV 2006, 449–62.

Graf, F., Grzegorzek, M. and Paulus, D. 2010. Counting Lymphocytes in Histopathology Images Using Connected Component. In: Proceedings of the 20th International Conference of Pattern Recognition (ICPR) 2010 Contests, Springer.

Hofmann, T., Puzicha, J., Buhmann, J.M., 1997. An optimization approach to unsupervised hierarchical texture segmentation. In: Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3-Volume Set - Volume 3, 213–.

Kuse, M., Sharma, T. and Gupta, S., 2010. A Classification Scheme for Lymphocyte Segmentation in H & E Stained Histology Images. In: Proceedings of the 20th International Conference of Pattern Recognition (ICPR) 2010 Contests, Springer, 235–243.

Malladi, R. and Sethian, J.A., 1997. Level Set Methods for Curvature Flow, Image Enhancement, and Shape Recovery in Medical Images. In: Proceedings of the Conference on Visualization and Mathematics, Springer-Verlag, 329–45.

Meyer, F., 1994. Topographic distance and watershed lines. Signal Process. 38, 113–125.

Meyer, F., 2005. Morphological segmentation revisited. Space, Structure and Randomness, Springer, Springer, Heidelberg, 315-347.

Mumford, D. and Shah, J., 1989. Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics, 42, 5, 577–684.

Nguyen, H.T., Worring, M. and van den Boomgaard, R., 2003. Watersnakes: Energy-Driven Watershed Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25, 330–342.

Panagiotakis, C, Ramasso, E, Tziritas, G, 2010. Lymphocyte Segmentation using the Transferable Belief Model. Proceedings of the 20th International Conference of Pattern Recognition (ICPR) 2010 Contests, Springer.

Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22, 888–905.

Staal, J., Kalitzin, S., Romeny, B.T.H., Viergever, M., 1999. Detection of critical structures in scale space. Scale-Space Theories in Computer Vision. 105–116.

Tai, X.C. and Hodneland, E. and Weickert, J. and Bukoreshtliev, N.V. and Lundervold, A. and Gerdes, H.H., 2007. Level Set Methods for Watershed Image Segmentation, In: Proceedings of the of the 1st International Conference in Scale Space and Variational Methods in Computer Vision, SSVM 2007, 178–90.

Tek, H.D., Comaniciu, D. and Williams, J.P., 2001. Vessel Detection by Mean Shift Based Ray Propagation. In: Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, 228–235.

Vedaldi, A and Fulkerson, B., 2010. VLFeat - An open and portable library of computer vision algorithms. Proceedings of the 18th annual ACM international conference on Multimedia, MM 2010, 1469–1472.

Xu, C. and Prince, J.L., 1998. Snakes, shapes, and gradient vector flow. IEEE Transactions on Image Processing, 7, 3, 359–369.

Yu, S.X., 2004. Segmentation using multiscale cues. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, 247–254.

Yu, S.X., 2005. Segmentation induced by scale invariance. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 444–451.

Yu, S.X., Shi, J., 2001. Understanding popout through repulsion. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, 752–757.

Yu, S.X., Shi, J., 2003. Multiclass spectral clustering. In: Proceedings of the Ninth International Conference of Computer Vision, ICCV 2003, 313–319.

Yu, S.X., Shi, J., 2004. Segmentation given partial grouping constraints. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26, 173–83.

Zhu, S.C. and Yuille, A., 1996. Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multi-band Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 18, 884–900.