

# Segmentation Subject to Stitching Constraints: Finding Many Small Structures in a Large Image

Elena Bernardis<sup>1</sup> and Stella X. Yu<sup>2</sup>

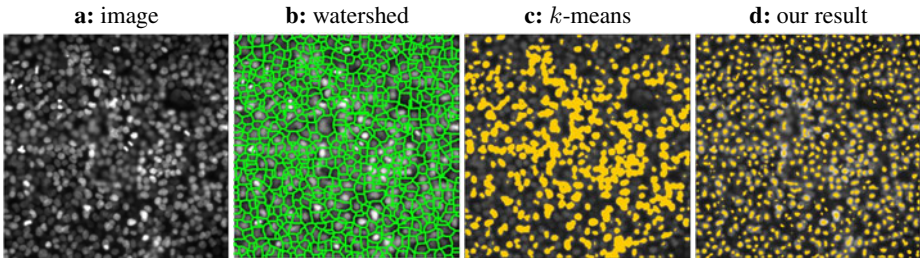
<sup>1</sup> University of Pennsylvania, Philadelphia, PA 19104, USA

<sup>2</sup> Boston College, Chestnut Hill, MA 02467, USA

**Abstract.** Extracting numerous cells in a large microscopic image is often required in medical research. The challenge is to reduce the segmentation complexity on a large image without losing the fine segmentation granularity of small structures. We propose a constrained spectral graph partitioning approach where the segmentation of the entire image is obtained from a set of patch segmentations, independently derived but subject to stitching constraints between neighboring patches. The constraints come from mutual agreement analysis on patch segmentations from a previous round. Our experimental results demonstrate that the constrained segmentation not only stitches solutions seamlessly along overlapping patch borders but also refines the segmentation in the patch interiors.

## 1 Introduction

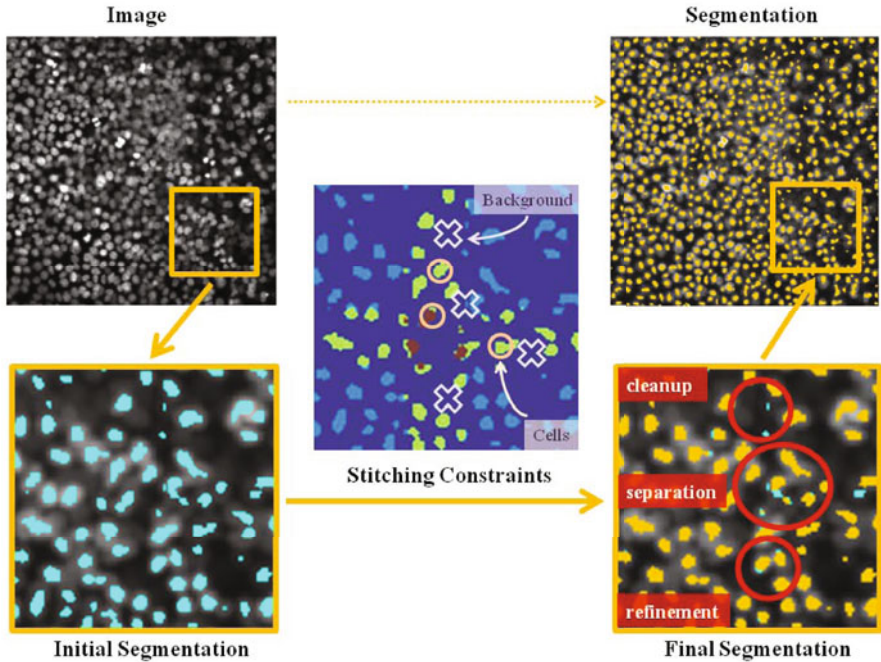
There is often a need in medical research to count, measure, and compare numerous tiny cells in a large image. Segmentation methods such as watershed and  $k$ -means clustering [1,2] are efficient but unable to deal with large intensity variation, whereas spectral graph partitioning methods [3,4] are robust but unable to efficiently find small structures in a large image. Our goal is to make the latter approach scale effectively with the image size yet without losing the fine granularity of small segments (Fig. 1).



**Fig. 1.** Segmenting numerous small regions in a large image. **a)** Epithelial cells, which stained with Hoechst fluoresce blue normally and red when transformed by virus, are small structures of varying intensity in this microscopic image. They must be segmented, counted, and measured. **b,c,d)** Two-way segmentations by watershed,  $k$ -means, and our method. With large intensity variation for the cells, watershed fails with oversegmentation and  $k$ -means fails with conjoined cells of similar intensities, whereas our method correctly pops out all the individual cells.

Spectral graph partitioning methods [3,4] 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 not only 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 from them), but also prevents small structures from being segmented all at once (since a larger image size leads to larger regions instead of numerous small ones given a fixed number of segments). Therefore, finding many small regions in a large image faces two challenges: segmentation complexity from dealing with the large and segmentation granularity from dealing with the small.

The two main approaches to reduce complexity, coarse-to-fine and multiresolution segmentations [4,5,6,7,8,9,10], are not suitable for this task. A coarse-to-fine approach speeds up the segmentation by initializing a finer segmentation with the results of a coarser one, whereas a multiresolution approach 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.



**Fig. 2.** Algorithm overview. Segmenting numerous small structures in a large image can be performed as a series of independent patch segmentation subject to stitching constraints between neighbouring patches. The constraints are derived from mutual agreement analysis on adjacent patch segmentations from a previous round. Segmentations between neighbouring patches are marked in blue, green, or maroon, if 1, 2, or more than 3 patches agree. Constrained segmentation (gold) improves the initial segmentation (cyan) in three different ways: clean-up of spurious small regions, separation of conjoined cells, and refinement of cell boundaries.

We propose performing segmentation on smaller patches while subject to stitching constraints in their overlapping areas (Fig. 2). We first break down the image into patches and segment each patch independently, based on local pairwise cues that distinguish cells from their background [11]. Their 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 independent 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 [12]. 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 constraints does more than stitching solutions together at patch borders. The constraints in the overlapping regions are propagated through local cues in the optimization process to improve the interior segmentation of a patch. Constrained segmentation in individual patches achieves reduced complexity without losing global consistency, refining segmentations both inside and between patches. We detail our model and experiments in Sections 2 and 3 respectively.

## 2 Spectral Graph Partitioning Subject to Stitching Constraints

We formulate our cell segmentation task as a constrained graph partitioning problem on 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 in the image becomes a two-way node partitioning problem: pixels inside cells form a foreground node set, and those outside form the other background node set.

We need to address: **1)** What features and grouping cues to use to facilitate this foreground-background segregation [11]; **2)** How to set up constraints between neighboring patches; **3)** How to integrate these constraints into the segmentation [12,13].

### 2.1 Features $F$ and Grouping Cues $W$

We characterize cells of small convex bright regions as the sinks of local gradient fields. 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  $F$ 's are similar, and repelled into different regions if their  $F$ 's are of opposite types (e.g. sources and sinks).

Consider pixel  $i$  and its neighbourhood  $N(i)$ . If neighbour  $a \in N(i)$  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  $a$  neighbours, weighted by the total non-decreasing intensity  $T(i, a)$  along the straight line from  $i$  to  $a$ :

$$p(i) \propto \sum_{a \in N(i)} T(i, a)(L(a) - L(i)), |p(i)| = 1 \quad (1)$$

$$T(i, a) = \sum_{\substack{I(m_1) \leq I(m_t) \leq \dots \leq I(m_k) \\ m_1 m_2 \dots m_k = \text{line}(i, a)}} I(m_t) \quad (2)$$

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. 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.

We define  $F(i, a)$  as the inner product of  $p(i)$  and  $p(a)$ , measuring how much  $a$ 's cell center estimate agrees with  $i$ 's. The ensemble of  $\{F(i, a): a \in N(i)\}$  is a 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 pairwise pixel grouping cues:

$$F(i, a) = \langle p(i), p(a) \rangle, \quad a \in N(i) \quad (3)$$

$$S(i, j) = \frac{\langle F(i, :), F(j, :)\rangle}{|F(i, :)| \cdot |F(j, :)|}, \quad j \in N(i) \quad (4)$$

$S(i, j)$  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 [11]: The short-range attraction  $A$  is proportional to similarity  $S$  and the long-range repulsion  $R$  is proportional to dissimilarity  $1 - S$ . The total effective weight  $W$  is  $A - R$ .

Unlike real-scene image segmentation [3,4], we do not use single edge features (e.g. large intensity gradients along region boundaries) to delineate regions. We use distributive local gradient fields to characterize geometrical distinction between region cores in the foreground and region peripheries in the background. Similar ideas about such features can be found in [14] on detecting critical points in images with topological numbers. While the individual pairwise grouping cues have poor precisions for localizing region boundaries, they taken together in global integration result in segmentations that are sensitive to geometrical variation yet robust to intensity variations.

## 2.2 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 neighbouring patches: those pixels that two patches agree on the segmentation become either foreground or background pixels. Only a sparse set of pairwise constraints (usually between distant pixels) are needed to ensure that two neighbouring patches will have consistent segmentations in their overlapping areas.

### 2.3 Segmentation with Stitching: Constrained Graph Partitioning $X$

For each image patch  $I$ , after having computed its pairwise grouping cues  $W$  and stitching constraints  $U$ , we obtain a two-way segmentation using the constrained normalized cuts criterion [12]. Formally, this criterion can be written in the following matrix form:

$$\text{maximize} \quad \varepsilon(X) = \sum_{g=1}^2 \frac{X_g^T W X_g}{X_g^T D X_g} \quad (5)$$

$$\text{subject to} \quad X \in \{0, 1\}^{n \times 2}, X 1_2 = 1_n \quad (6)$$

$$U^T X = 0 \quad (7)$$

where  $1_n$  denote an  $n \times 1$  vector of 1's and  $D$  is the diagonal *degree* matrix for a  $n \times n$  weight matrix  $W$ . Note that  $W$  could have both positive and negative weights, and the negative ones are essential for popping out disconnected regions [15,11].

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

$$P = D^{-1}W \quad (8)$$

$$Q = I - D^{-1}U(U^T D^{-1}U)^{-1}U^T \quad (9)$$

While the eigensolution of  $QPQ$  takes a longer time than that of  $P$  (unconstrained version) to compute at each iteration, it often requires fewer iterations and could be in fact faster. We follow the eigensolution and its discretization procedures developed in [12,13] and their code online to obtain a binary segmentation.

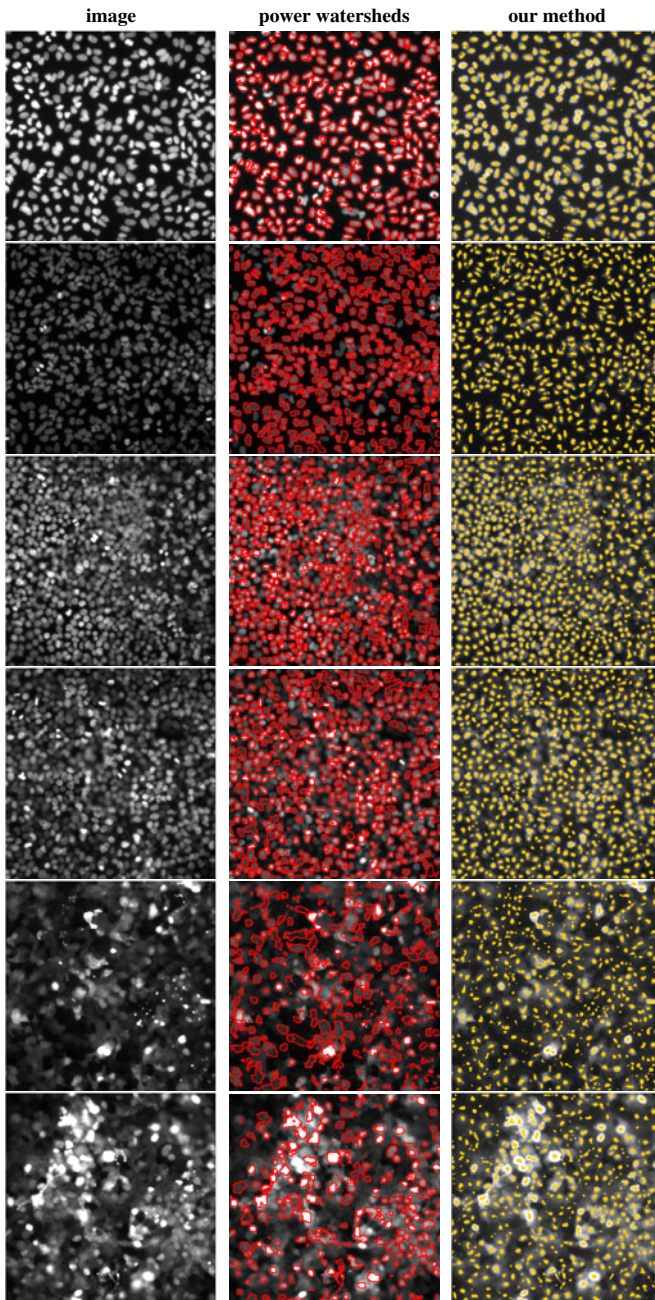
The space and time complexity is much reduced using patch segmentation with stitching constraints, as the image is broken down to smaller patches and finding numerous small regions becomes possible in a single two-way segmentation.

## 3 Experiments

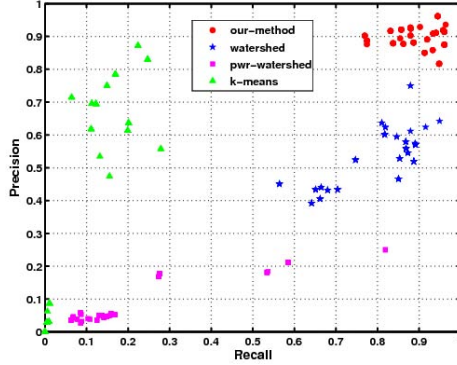
We implement our algorithm in MATLAB and apply it to sets of  $512 \times 512$  microscopic images. The cells in these images are 15 pixels in diameter on average. We use the same set of parameters as in [11] for deriving the weights  $W$  on each patch of size  $256 \times 256$ . The pixel neighbourhood radius is 12 pixels, and the overlap radius between patches is 20 pixels. Since the computational complexity mainly depends on the patch size, the entire image can be arbitrarily large.

Fig. 3 shows our results on 3 appearance types of medical images, each representative of a large class of images in its own domain. The cells have a large range of intensity, and fainter ones could be darker than some background pixels elsewhere in the image. Worse still, cells are not always isolated, but rather packed closely next to each other, making the separation even hard for the human eye. In the last rows of Fig. 3, fainter cells are so overwhelmed by those brightest cells that they can only be seen with a close inspection of what appears to be smudged backgrounds.

These images are very challenging to segment and most existing approaches fail. Our method, however, is capable of finding these cells, including faint ones and conjoined ones, all at once in a two-way segmentation, without any need for post-processing.



**Fig. 3.** Results by Power watersheds and our method on human alveolar basal epithelial A549 cells (rows 1-4) and embryonic kidney HEK293T cells (rows 5-6). While the quality of segmentation degrades for power watersheds when the cells have a larger intensity variation, our method pops out all the cells in these images with the same parameters and no post-processing.



**Fig. 4.** Precision-recall statistics for  $k$ -means, watershed, power watersheds, and our method. Our method (red round dots, upper right corner) has better precision and recall overall.

We benchmark our method against human labeled dot centers, in comparison with 3 other approaches:  $k$ -means, standard watershed (MATLAB built-in implementation), and power watersheds [2] provided by its authors.

Given  $m$  ground-truth dot centers and  $n$  segment centers for an image, let  $D_{ij}$  be the Euclidean distance between dot  $i$  and segment  $j$ . If it is less than a certain radius threshold  $\rho$ , we consider  $(i, j)$  a matched detection.

$$\text{precision} = \frac{\#\{j : \min_{i=1}^m D_{ij} \leq \rho\}}{n} = \frac{\# \text{ nearest dots within radius } \rho}{\# \text{ segments}} \quad (10)$$

$$\text{recall} = \frac{\#\{i : \min_{j=1}^n D_{ij} \leq \rho\}}{m} = \frac{\# \text{ nearest segments within radius } \rho}{\# \text{ dots}} \quad (11)$$

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.

Fig. 4 shows that our method performs much better than these other methods in terms of both precision and recall.  $k$ -means, clustering pixels based on their intensity values, particularly has trouble separating conjoined like-intensity cells, while increasing  $k$  only leads to clustering instability. While power watersheds has lower precision, it does noticeably improve boundary shapes of *segmented cells* over standard watershed, which is not properly measured in the precision-recall statistics. However, it tends to miss faint cells and the segmentation degrades with larger intensity variation (Fig. 3).

The quality of our segmentations depends on our patch segmenter and stitching constraints. The precision-recall statistics in Fig. 4 shows that our spectral graph partitioning approach works better than others at popping out small regions. 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 ( $p = 0.46$ ,  $t$  test), there is an average improvement of 0.04 in the precision ( $p = 0.007$ ,  $t$  test).

To summarize, we present an efficient method for segmenting many small regions in a large image by constructing a set of patch segmentations which are independently derived but subject to stitching constraints between them. The quality of entire

segmentation depends on how to segment individual patches and how to stitch them together. Although our method works with any patch segmenter, it naturally integrates the stitching problem into each patch's spectral graph partitioning formulation. Our results demonstrate that stitching as partitioning constraints not only reduces segmentation complexity, but also corrects segmentation incongruences and imperfections.

Segmenting small structures in a large image faces a scale dilemma between the image size and the segment size, and our approach resolves the dilemma by decoupling the two sizes in constrained patch segmentations. We can expand it to a broader variety of shapes, e.g. thin structures such as blood vessels, by tailoring the weights to allow directional extension in the evaluation of pairwise pixel similarity.

**Acknowledgements.** This research is funded by NSF CAREER IIS-0644204 and a Clare Boothe Luce Professorship to Stella X. Yu. We would like to thank Nisha Sosale at the University of Pennsylvania for providing images and Camille Couprie at Université Paris Est and ESIEE for providing power watersheds results.

## References

1. Meyer, F.: Topographic distance and watershed lines. *Signal Process* 38(1), 113–125 (1994)
2. Couprie, C., Grady, L., Najman, L., Talbot, H.: Power watersheds: a new image segmentation framework extending graph cuts, random walker and optimal spanning forest. In: *ICCV* (2009)
3. Shi, J., Malik, J.: Normalized cuts and image segmentation. *PAMI* 22(8), 888–905 (2000)
4. Yu, S.X.: Segmentation induced by scale invariance. In: *CVPR* (2005)
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. *IJCV* 59(2), 167–181 (2004)
6. Cour, T., Benezit, F., Shi, J.: Spectral segmentation with multiscale graph decomposition. In: *CVPR 2005*, pp. 1124–1131. IEEE Computer Society, Washington (2005)
7. Barbu, A., Zhu, S.C.: Graph partition by swendsen-wang cuts. In: *ICCV* (2003)
8. Galun, M., Sharon, E., Basri, R., Brandt, A.: Texture segmentation by multiscale aggregation of filter responses and shape elements. In: *ICCV* (2003)
9. Hofmann, T., Puzicha, J., Buhmann, J.M.: An optimization approach to unsupervised hierarchical texture segmentation. In: *ICIP* (1997)
10. Yu, S.X.: Segmentation using multiscale cues. In: *CVPR* (2004)
11. Bernardis, E., Yu, S.X.: Finding dots: Segmentation as popping out regions from boundaries. In: *CVPR* (2010)
12. Yu, S.X., Shi, J.: Segmentation given partial grouping constraints. *PAMI* 26(2) (2004)
13. Yu, S.X., Shi, J.: Multiclass spectral clustering. In: *ICCV*, October 11-17 (2003)
14. Staal, J., Kalitzin, S., Romeny, B.T.H., Viergever, M.: Detection of critical structures in scale space. In: Nielsen, M., Johansen, P., Fogh Olsen, O., Weickert, J. (eds.) *Scale-Space 1999*. LNCS, vol. 1682, p. 105. Springer, Heidelberg (1999)
15. Yu, S.X., Shi, J.: Understanding popout through repulsion. In: *CVPR* (2001)