Laplacian Coordinates for Seeded Image Segmentation

Wallace Casaca[1],   Luis Gustavo Nonato[1],   Gabriel Taubin[2]

[1]ICMC, University of São Paulo, São Carlos, Brazil
[2]School of Engineering, Brown University, Providence, United States

{wallace,lgnonato}@icmc.usp.br, taubin@brown.edu

Abstract

Seed-based image segmentation methods have gained much attention lately, mainly due to their good performance in segmenting complex images with little user interaction. Such popularity leveraged the development of many new variations of seed-based image segmentation techniques, which vary greatly regarding mathematical formulation and complexity. Most existing methods in fact rely on complex mathematical formulations that typically do not guarantee unique solution for the segmentation problem while still being prone to be trapped in local minima. In this work we present a novel framework for seed-based image segmentation that is mathematically simple, easy to implement, and guaranteed to produce a unique solution. Moreover, the formulation holds an anisotropic behavior, that is, pixels sharing similar attributes are kept closer to each other while big jumps are naturally imposed on the boundary between image regions, thus ensuring better fitting on object boundaries. We show that the proposed framework outperform state-of-the-art techniques in terms of quantitative quality metrics as well as qualitative visual results.

1. Introduction

Image segmentation is without doubt one of the most studied topics in computer vision and pattern recognition. Prominent applications such as medical imaging, machine vision and object detection have widely inspired the development of a large number of methods for segmenting images. In particular, a growing number of semi-supervised image segmentation methods have been proposed in the last few years, motivated mainly by human capability to recognize and detect patterns. In fact, *seeded/compuer-assisted image segmentation* are now considered among the most relevant image segmentation methods, leveraging the emergence of new mathematical and computational formulations, particularly those based on graph theory. Seeded-based image segmentation methods typically rely on a given set of labeled pixels (the seeds) and on affinity graphs whose nodes correspond to image pixels and edges reflecting the neighborhood structure of the pixels. Edge weights encoding image attributes such as color variation, texture and gradients are used to properly drive the propagation of the seeded labels on the image. Many distinct mathematical approaches and algorithms have been proposed to perform the label propagation [21, 13, 5, 24, 10, 14, 25, 1], most of them making use of energy functional minimization on graphs in order to be effective.

As pointed out by Couprie et al. [8], most seed-based image segmentation methods are variations of a small group of basic techniques such as *Graph Cuts* [5], *Random Walker* [13] and *Watersheds* [10], which differ from each other in terms of their mathematical formulation, pairwise pixel distance and weight computation. Moreover, most existing methods rely on non-quadratic energies, thus demanding the use of sophisticated and computationally costly optimization tools. Ensuring accuracy and smooth solution is also an issue for existing methods.

In this work we present a novel methodology for seed-base image segmentation, called *Laplacian Coordinates*, which relies on the minimization of an novel quadratic energy functional defined from an affinity graph. The notion of Laplacian Coordinates has been initially introduced in [22, 26] to address the problem of surface processing in the field of Geometry Processing. In contrast to most existing algorithms, in particular the three basic ones mentioned above that formally minimize the "distance" between pairwise pixels, the proposed approach minimizes the average of distances while better controlling anisotropic propagation of labels during the segmentation task. As a result, pixels sharing similar attributes are kept closer to each other while jumps are naturally imposed on the boundary between image regions, thus ensuring better fitting on image boundaries as well as a pretty good neighborhood preservation (on average). Moreover, the proposed formulation is guaranteed to have a unique solution, an important trait not always present in seed-based image segmentation methods.

The Laplacian Coordinates pipeline is very simple and

comprises four main steps: *definition of seeds*, *affinity graph building*, *energy functional construction and solution*, and finally *assignment of labels*. An important characteristic of the proposed formulation is that the minimizer of the energy functional is given by the solution of a constrained system of linear equations, making the proposed methodology quite simple to be used and coded. The effectiveness of Laplacian Coordinates is qualitatively and quantitatively tested through experiments against several state-of-the-art approaches using the public "Grabcut" dataset from Microsoft [5].

Contributions In summary, the main contributions of this work are:

1. A novel and easy-to-implement formulation for seed-based image segmentation, which we call *Laplacian Coordinates*.

2. Laplacian Coordinates bears several important properties such as boundary fitting, anisotropy, average neighborhood preservation and unique solution for the minimizer.

3. The segmentation is reduced to solving a constrained sparse linear system of equations.

4. A comprehensive set of quantitative and qualitative comparisons against state-of-art algorithms that shows the effectiveness of Laplacian Coordinates.

2. Related work

Most seed-based image segmentation methods rely on energy minimization schemes derived from affinity graphs. In particular, effort has been concentrated around three main approaches: *Graph Cuts* [5], *Random Walker* [13] and *Watersheds* [10].

Graph Cuts The *GraphCut framework* (GC) was introduced by Boykov and Jolly [6] to address the problem of interactive N-dimensional image clustering. The rationale behind GC is to consider the image as a graph and finding the minimum cut between seeded regions, minimizing the sum of absolute differences between pairwise pixels. The GC framework uses a maflox/min-cut algorithm to tackle the problem of unique solution and reaches a feasible segmentation. Many extensions of GC have been proposed in the literature, most of them focused on user interfaces such as [21, 17]. A drawback of Graph Cuts is the generation of small segmented regions, which naturally appear due to the underlying mathematical formulation that looks for solutions with minimal boundary length (e.g., see the "car" segmentation in Fig.6).

Random Walker The *Random Walker algorithm* (RW) [13] is a useful and easy-to-implement approach that relies on standard graph Laplacian formulation $Lx = 0$, where L is a matrix built from an edge weight matrix W and a diagonal weighted valency matrix D. In [13], Grady presents an interesting interpretation of the Laplacian formulation by associating to each unseeded pixel the probability of a random walker starting on it to reach a background seed (assuming background seeds are labeled 1). The segmentation is then performed by assigning a background label to a pixel if the probability is greater than 0.5, otherwise the pixel is assigned to a foreground label. In terms of usage, instead of solving $Lx = 0$, the RW algorithm solves $D^{-1}Wx = x$ using the input seeds as constraints, which ensures uniqueness of solution. Moreover, the solution is given as the solution of a linear system. However, it was shown in [27] that the RW methodology does not have an anisotropic behavior, meaning that the method is prone to produce "flatter" solutions. Furthermore, the RW is not so accurate when capturing object boundaries. Our approach shares serval of the good properties of RW while still presenting better performance with respect to boundary fitting and anisotropic label transportation.

Watersheds/Maximum Spanning Forest The idea behind *Watersheds/Maximum Spanning Forest algorithms* (MSF) [10] is to represent image objects as "catchment basins", performing the segmentation by identifying the "basins" and their watershed lines (points with equally likely to assume more than one minimum). In [8] a robust seeded watershed-based framework namely *Power Watershed* (PWS) was proposed, where the outcome relies on the computation of an MSF algorithm. Although watersheds are very popular in the computer vision literature, they are not quite efficient in fitting objects where the gradient is locally irregular (see the first and second image in Fig.7).

Other methods Many other seed image segmentation methods have appeared in the literature. The *Shortest Path/Geodesic-based algorithm* (SP) [4] sets the pixel labeling computing the shorter weighted path from the target pixel to the foreground or background seeds. The method is attractive in terms of speeding but it strongly depends on the position that seeds will be sown. The techniques presented in [2, 7] perform very well with respect to insertion of seeds, requiring just a small amount of user intervention to achieve the segmentation but they rely on a pre-segmentation in order to be effective. The method [18] solves an spectral problem making use of a linear combination of pre-computed eigenvectors constrained to the prior vector to reach a feasible solution. Couprie et al. [8] proved that CG, MSF, PWS, RW and SP methods minimize the same energy functional whose formulation takes into account only first-order pairwise pixels, differing only in terms of an exponent value. In contrast, our formulation relies on the minimization of a weighted average of neighbor pixels, which leads to smoother but accurate solutions.

3. Laplacian Coordinates-based Energy Functional

Let I be a color or grayscale image. For a color image we denote the RGB vector by $I_i = (R_i, G_i, B_i)$, which represents the luminance of red, green and blue channels at the pixel $P_i \in I$. For a grayscale image, I_i is the gray intensity. Let $G = (V, E, W_E)$ be the weighted graph where V is the set of nodes $i \in V$ corresponding to the pixel $P_i \in I$, the edge set E corresponds to pairs of pixels locally connected in an 8-neighborhood, and $w_{ij} \in W_E$ is the weight assigned to edge (P_i, P_j) of the graph. The set $N(i) = \{j : (P_i, P_j) \in E\}$ represents the indices of the pixels P_j that share an edge with pixel P_i and $d_i = \sum_{j \in N(i)} w_{ij}$ is the weighted valency of P_i.

3.1. Set up the Graph Weights

There are many different ways to define the weights W_E, such as pixel intensity, gradient, scalability and contour [9, 3, 7, 16]. In order to keep our approach as simple as possible we only consider pixel intensities to define the weights. More precisely, the weight $w_{ij} = w(P_i, P_j)$ assigned to each edge (P_i, P_j) is computed as follows:

$$w_{ij} = \exp\left(-\frac{\beta \|I_i - I_j\|_\infty^2}{\sigma}\right), \quad \sigma = \max_{(P_i, P_j) \in E} \|I_i - I_j\|_\infty \quad (1)$$

where β is a tuning constant. Notice that the weights are positive and symmetric in the sense that $w_{ij} = w_{ji}$. In practice, a small constant $\epsilon = 10^{-6}$ is added into (1) to avoid null weights, as suggested by Grady [14].

3.2. Neighborhood Average Preserving Energy Functional

Given the set of background B and foreground F seeded pixels and their corresponding labels x_B and x_F (without loss of generality, assume that $x_B > x_F$), the following energy functional is minimized with respect to $\mathbf{x}$:

$$E(\mathbf{x}) = k_1 \sum_{i \in B} \|x_i - x_B\|_2^2 + k_2 \sum_{i \in F} \|x_i - x_F\|_2^2 +$$

$$k_3 \sum_{i \in V} \left\| d_i x_i - \sum_{j \in N(i)} w_{ij} x_j \right\|_2^2 \quad (2)$$

where $\mathbf{x} = (x_1, x_2, ..., x_n)$ is the sought solution, that is, the values assigned to the pixels $(P_1, P_2, ..., P_n)$ so as to minimize the functional $E(\mathbf{x})$, n is the number of pixels, w_{ij} is computed as in Equation (1) and k_1, k_2 and k_3 are positive constants. Once the energy (2) is minimized, the segmentation is then obtained by assigning background or foreground labels $y_i \in \{x_B, x_F\}, i \in V$ as follows:
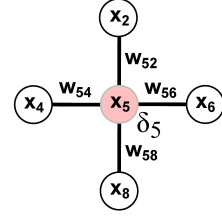

Figure 1. Geometric interpretation of the differential operator δ_i at vertex $i = 5$. It measures the deviation between x_5 and its center of mass $\frac{1}{d_5} \sum_{j \in N(5)} w_{ij} x_j$.

$$y_i = \begin{cases} x_B, & \text{if } x_i \geq \frac{x_B + x_F}{2} \\ x_F, & \text{otherwise} \end{cases} \quad (3)$$

Energy functional (2) is made up of two main components, one accounting for the constraints imposed by the seeds in B and F, called *data term*, and a second component controlling label spread in the neighborhood of each pixel, called *Laplacian Coordinates energy term*. The Laplacian Coordinates energy term can be rewritten in matrix form as follows:

$$\sum_{i \in V} \left\| d_i x_i - \sum_{j \in N(i)} w_{ij} x_j \right\|_2^2 = \|\mathbf{Lx}\|_2^2 \quad (4)$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the *graph Laplacian matrix*, $\mathbf{D}$ is the diagonal matrix where $D_{ii} = d_i$ and $\mathbf{W}$ denotes the weighted adjacency matrix of the graph,

$$W_{ij} = \begin{cases} w_{ij}, & \text{if } (i, j) \in E \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

Notice that each row in $\mathbf{Lx}$ corresponds to the *differential* (or average) *operator* $\delta_i = x_i - \frac{1}{d_i} \sum_{j \in N(i)} w_{ij} x_j$, that is, $(Lx)_i = d_i \delta_i$. In less mathematical terms, δ_i measures how much each node deviates from the weighted average of its neighbors (see Figure 1).

3.3. Minimizing the Energy Functional

Without loss of generality, let the tuning parameters in the energy functional $E(\mathbf{x})$ (2) assume unitary values, that is, $k_i = 1, i = 1, 2, 3$. $E(\mathbf{x})$ is a quadratic function which can be modeled in a more general matricial form as follows:

$$E(\mathbf{x}) = \mathbf{x}^t (\mathbf{I_S} + \mathbf{L^2}) \mathbf{x} - 2\mathbf{x}^t \mathbf{b} + c, \quad (6)$$

where $\mathbf{I_S}$ is a diagonal matrix such that $I_S(i, i) = 1, i \in S = B \cup F$, and zero, otherwise, $\mathbf{b}$ is the vector where $b(i) = x_B, i \in B$, $b(i) = x_F, i \in F$, and zero, otherwise, and c is a constant. The quadratic form (6) has a unique

minimizer since $\mathbf{I_S} + \mathbf{L^2}$ is symmetric and positive definite. Moreover, its minimizer vector $\mathbf{x}$ is the solution of the following linear system [15]:

$$(\mathbf{I_S} + \mathbf{L^2})\mathbf{x} = \mathbf{b}. \tag{7}$$

Therefore, minimizing $E(\mathbf{x})$ is equivalent to solve the linear system (7), which, in turn, holds quite attractive properties such as symmetry, positive definiteness and sparsity. In fact, equation (7) can be solved efficiently using the supernodal sparse Cholesky factorization algorithm such as the one implemented in *Cholmod* [11] or classical MATLAB solvers.

After solving (7), the segmentation is then performed by trivially assigning a foreground or background label to each pixel of the image according to (3).

3.4. Laplacian Coordinates: Some Properties

Besides being computationally efficient, easy-to-implement and ensuring unique solution, the proposed methodology has additional properties that render it quite attractive to segment images, as discussed below.

Boundary and Constraint Fitting   The main characteristic that differs Laplacian Coordinates with respect to other seed-based approaches is its capability to better propagate the seeds (constraint information). Figure 2 illustrates this fact by comparing Laplacian Coordinates against the Random Walker approach. First row of Figure 2 shows an 1D graph with 500 nodes ordered from left to right. Second row in Figure 2 shows two different distribution of edge weights: on the left, unitary weights are assigned to edges, except for edges in the middle of the graph, where weights have a distribution that decreases and gets close to zero increasing again back to 1. On the right, weights are distributed similarly, but now with two picks isometrically arranged. Constraints (seeds) are imposed in the yellow and purple nodes. As one can easily see on the third row of Figure 2, Laplacian Coordinates spread the constraint information in a smoother way, taking longer to diffuse the constraint information when compared with Random Walker approach. For the sake of illustration, last row in Figure 2 presents the result of applying Laplacian Coordinates and Random Walker when all edge weights are set equal 1. The better preservation of labels can also be observed in Figure 3. Figure 3(b) shows that the Random Walker approach was not able to properly capture the objet contained in the image while Laplacian Coordinates has accurately identified the objects, as depicted in Figure 3(c). The reason for the better performance of Laplacian Coordinates is that labels tend to be preserved in homogeneous regions while Random Walker "diffuse" labels quickly according to the distance from the seeds, as shown in Figure 3(d) and (e).
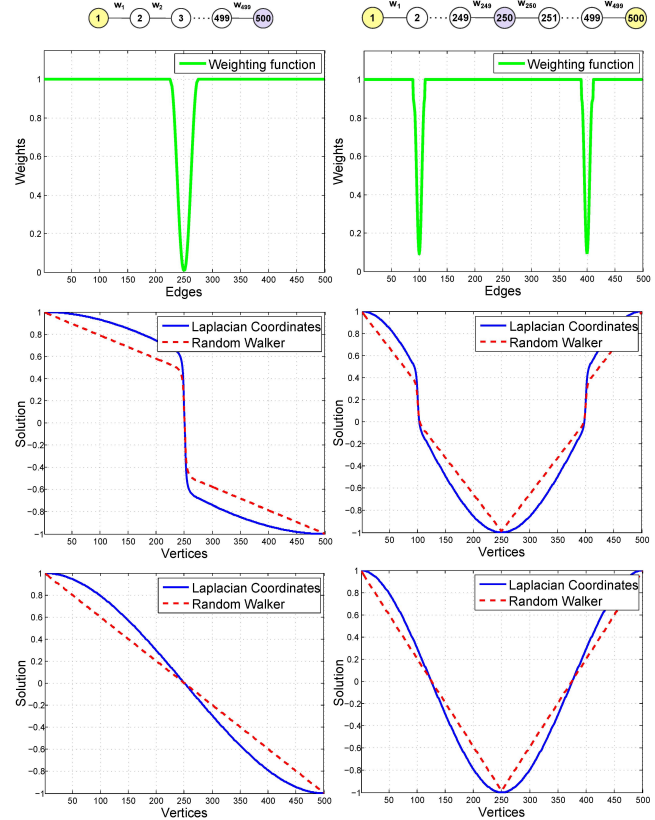

Figure 2. Comparison between the solution obtained from Laplacian Coordinates and the classical Random Walker algorithm under the same initial conditions. Line graphs are shown in the top row with seeded vertices in yellow and purple while the corresponding edge weights are shown in the second row. The solution with and without the mentioned weights are given in the third and fourth rows.

Solution in Terms of Extended Neighborhood   An interesting interpretation of the solution of Laplacian Coordinates is that each pixel x_i is written not only in terms of the first-order neighbors but taking distant neighborhoods, instead. In mathematical terms, in an unconstrained pixel P_i we have that $(Lx)_i = \frac{1}{d_i}\sum_{j \in N(i)} w_{ij}(Lx)_j$. Therefore, the solution x_i takes into account an extended neighborhood, mathematically expressed by (see Fig.4 for an illustration) the equation:

$$x_i = \frac{1}{d_i} \sum_{j \in N(i)} w_{ij}\left(x_j + \frac{\delta_j}{d_i}\right)$$

$$= \frac{1}{d_i} \sum_{j \in N(i)} w_{ij}x_j + \frac{1}{d_i^2} \sum_{j \in N(i)} w_{ij}\left(\sum_{p \in N(j)} w_{jp}(x_j - x_p)\right)$$

Therefore, information coming from the constraints takes longer to be diffused by the Laplacian Coordinates approach.
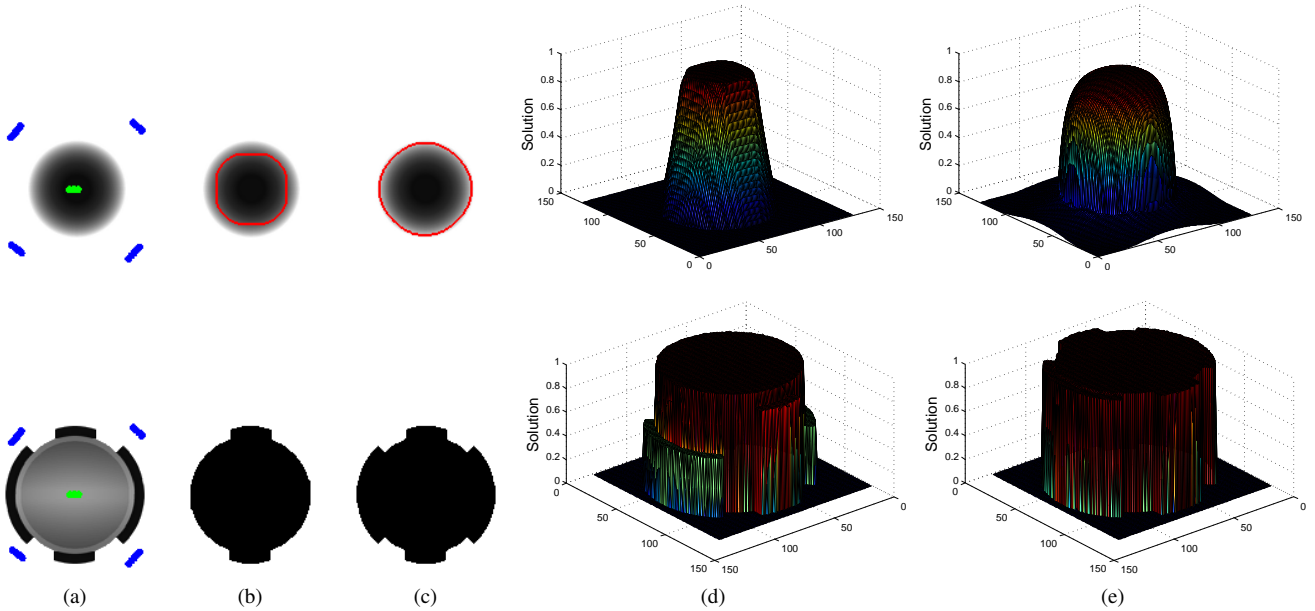
Figure 3. Random Walker and Laplacian Coordinates image boundary fitting capability. (a) Seeded images, (b)-(c) the segmentation results obtained from Random Walker and Laplacian Coordinates, and (d)-(e) graphs of the solution associated to (b) and (c), respectively.
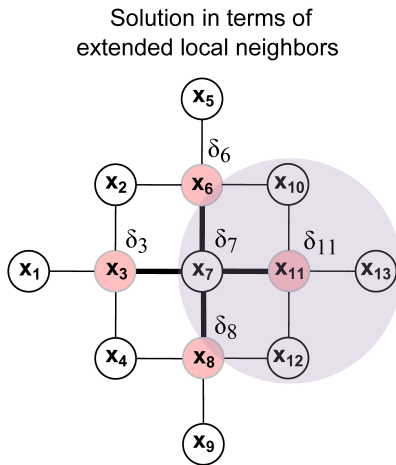
Solution in terms of
extended local neighbors

Figure 4. Geometric sketch showing the solution x_i ($i = 7$) in terms of its neighborhood pixels. In the example, the purple circle illustrates the points used for calculating the differential coordinate δ_i at pixel $i = 11$.

4. Results, Comparisons and Evaluation

In this section we provide comparisons against various existing state-of-the-art techniques. Input seeds are set as $x_B = 1$ and $x_F = -1$ in Eq.(3) while $\sigma = 0.1$ in Eq.(1).

In [21], a popular seeded image segmentation database called "Grabcut" dataset was introduced containing 50 images, their ground-truth, and seeded maps marking foreground and background regions of the images. The bench-mark data set is available on the Microsoft Cambridge website and also includes 20 images from the *Berkeley Image Segmentation Benchmark Database* [3]. We use this benchmark dataset to compare the proposed Laplacian Coordinates approach (LC) against the five classical seed-based segmentation techniques described in Sec.II: Graph Cuts (GC)[1] [21], Power Watershed (PWS)[2] [8], Maximum Spanning Forest with Kruskal's (MSFK) and Prim's (MSFP) algorithms[2] [10, 8] and Random Walker (RW)[3] [13]. Quantitative evaluations are performed comparing the quality in terms of segmentation region refinement as well as the accuracy in preserving ground-truth boundaries.

Region Quality   We make use of three distinct region quality metrics to gauge the quality of Laplacian Coordinates, namely,

- **Rand Index** (RI): measures the closeness between the output segmentation and the ground-truth by counting the number of pixel pairs that have the same label [23]. The higher the value the better.

- **Global Consistency Error** (GCE): computes how much a segmentation can be viewed as a refinement of other [19]. Lower values are better.

- **Variation of Information** (VoI): quantifies the distance between ground-truth and segmentation in terms of their relative entropies [20]. Values close to 0 are better.

---

[1] available at http://grabcut.weebly.com/code.html
[2] available at http://powerwatershed.sourceforge.net
[3] available at www.cns.bu.edu/~lgrady/

Method	RI ($\uparrow$)	GCE ($\downarrow$)	VoI ($\downarrow$)
GC	0.9714	0.0268	0.1877
MSFK	0.9690	0.0292	0.2013
MSFP	0.9689	0.0293	0.2018
PWS	0.9704	0.0278	0.1931
RW	0.9700	0.0280	0.1934
LC	**0.9715**	**0.0262**	**0.1836**

Table 1. Comparison of six seed-based segmentation methods regarding to region quality metrics. The proposed Laplacian Coordinates framework outperforms all other five evaluated techniques.

Boundary Quality The harmonic average score *F-score* summarizes the *Recall* and *Precision* image segmentation benchmarks [12, 3], measuring how much the segmentation matches the ground-truth boundaries. Recall can be viewed as the proportion of boundary pixels in the segmentation for which it is possible to find a matching boundary pixel in the ground-truth image. Precision holds the opposite situation. The matching is established in terms of the boundary pixel proximity for different values of radius R, as proposed in [12].

Table 1 shows the scores obtained by each method taking into account the three region quality metrics for the Microsoft "Grabcut" dataset. Laplacian Coordinates clearly outperforms the other five methods in all quality metrics. Regarding *F-score*, the Laplacian Coordinates also presents very good performance, specially when the parameter R increases. As one can see from Figure 5, the proposed approach shows a better *F-score* than other techniques, outperforming all for R equal or bigger than 7. These quantitative results show the effectiveness of Laplacian Coordinates as a seeded image segmentation method.
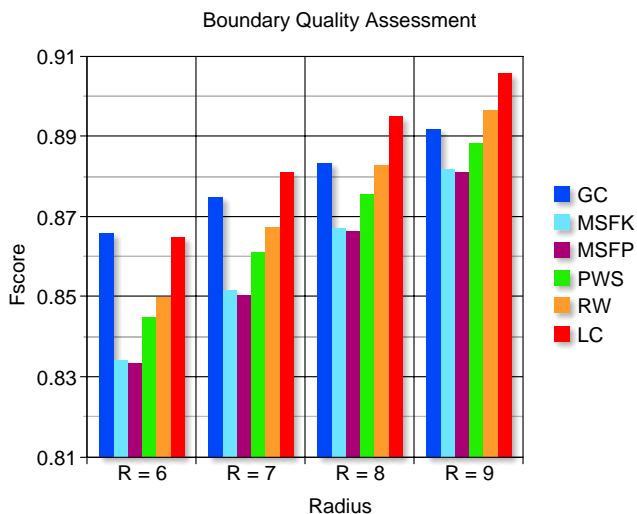

Figure 5. *F-score* quality metric. Laplacian Coordinates is considerably better than other methods when parameter R increases.
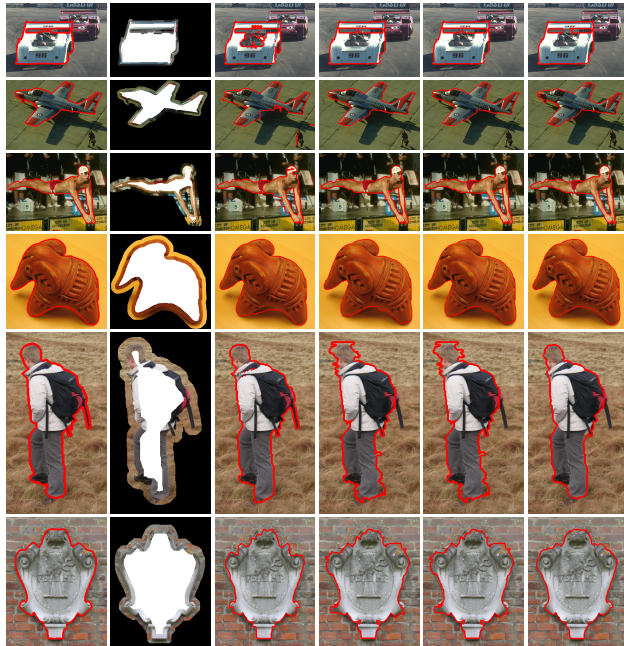

Figure 6. From left to right: Ground-truth, the tri-map images (seeds and the unknown region) provided by the Grabcut dataset, the segmentations resulting from GC, MSFK, MSKP and our approach.

Figures 6 and 7 present qualitative results comparing GC, MSFK, MSFP, PWS and RW against Laplacian Coordinates. One can see that, besides accurately capturing boundaries, Laplacian Coordinates tends to simultaneously generate smoother and fitter boundary curves, a characteristic not present in the other approaches, which are less accurate while still producing more jagged boundary curves.

Seeding Flexibility and Adaptability Figure 8 shows the robustness of Laplacian Coordinates in producing different segmentations by just selecting new targets in the image. Notice from the two initial configurations (left and middle columns) of Figure 8 that both objects (the boys) are accurately segmented, attesting the accuracy of the proposed approach. In fact, an even more general solution can be obtained by simultaneously seeding the two targets of the image, as depicted in the last column of Figure 8.

Multiple-Region Segmentation We conclude this section showing that Laplacian Coordinates can easily be extended to segment an image in several parts. This extension is carried out by simply solving N system of linear equations similar to Equation (7):

$$(\mathbf{I_S} + \mathbf{L^2})\mathbf{x}^{(\mathbf{j})} = \mathbf{b}^{(\mathbf{j})}, \qquad (8)$$

but setting $I_S(i, i) = 1$ for all seeded pixels in the image and specifying different $b^{(j)}$ for each one of the given labels
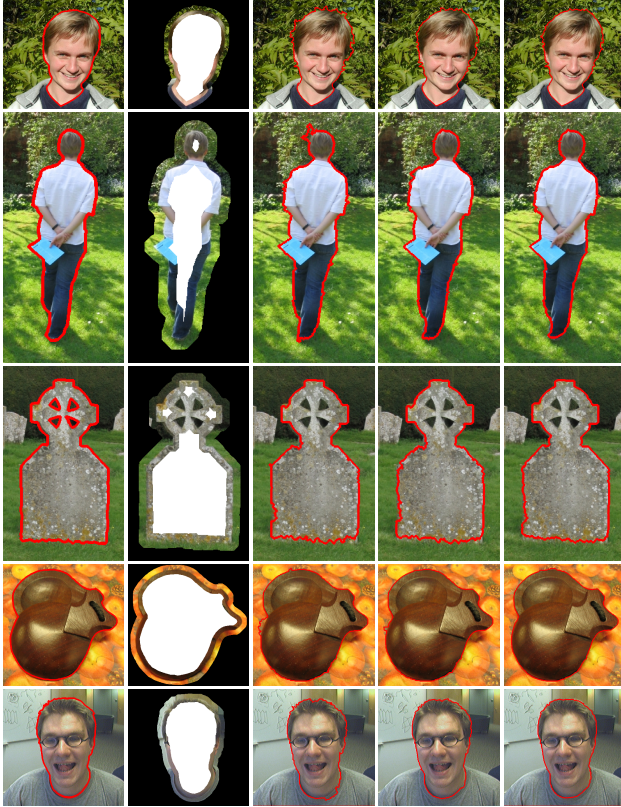
Figure 7. From left to right: Ground-truth, the tri-map images (seeds and the unknown region) provided by the Grabcut dataset, and segmentations resulting from PWS, RW and our approach.
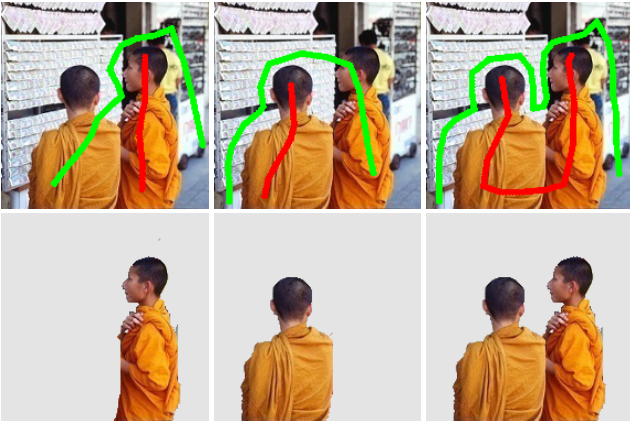

Figure 8. Selecting different objects from the image by exploiting the seed sensitivity of the Laplacian Coordinates. First row: multiple selections are given as input to the method. Bottom row: the corresponding segmentations.

$K_j \in K = \{K_1, K_2, ..., K_N\}$, instead. Assuming that C is a positive constant, we set $b_i^{(j)} = C$, $i \in K_j$, $b_i^{(j)} = -C$, $i \in (K \backslash K_j)$, zero, otherwise. Finally the segmentation $y^{(j)}$

(a binary image) is performed by

$$y^{(j)} = \bigcap_{\substack{p=1,...,N \\ p \neq j}} (x^{(j)} > x^{(p)}),$$

where $>$ is computed for all pixels of the image.

Figure 9 depicts the result of applying Laplacian Coordinates to segment multiple regions. Color strokes mark the objects (strokes with the same color correspond to the same region), from which Laplacian Coordinates generates the segmentation in multiple regions.
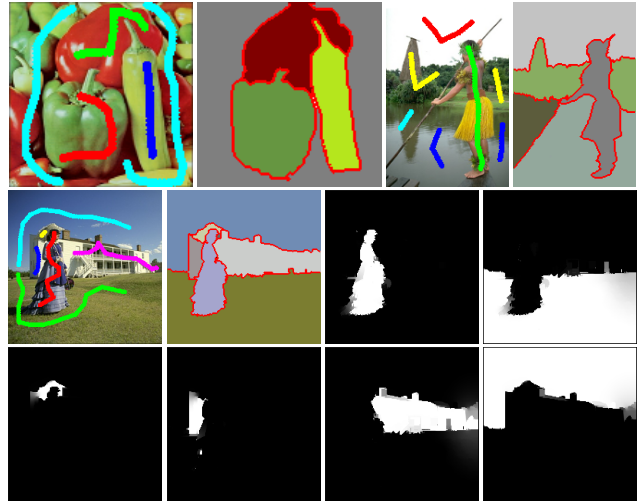

Figure 9. Extension of the Laplacian Coordinates (2) for multiple segmentation. First row: multiple seeds are sketched as colored strokes, from which Laplacian Coordinates produced the multiple segmented regions. Middle and bottom row: sketched seeds, the final segmentation and the six solution vectors $\mathbf{x}^{(j)}$ that give rise to the multiple segmentation.

5. Conclusion

In this paper we introduce Laplacian Coordinates, a novel seed-based image segmentation technique which has several advantages when compared with existing methods. Besides its simple mathematical formulation, Laplacian Coordinates is easy to implement, guarantees a unique solution, and outperforms existing methods with respect to well established quantitative measures popularly used in the context of image segmentation. Laplacian coordinates also holds high accuracy in terms of image boundary fitting capability for the object segmentation task. All those properties render Laplacian Coordinates an interesting and compelling seed-based image segmentation technique.

Acknowledgment

References

[1] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *IEEE Conf. on Comp. Vision and Pattern Recog. (CVPR)*, pages 811–818, 2013. 1

[2] N. T. N. Anh, J. Cai, J. Zhang, and J. Zheng. Robust interactive image segmentation using convex active contours. *IEEE Transactions on Image Processing*, 21(8):3734–3743, 2012. 2

[3] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011. 3, 5, 6

[4] X. Bai and G. Sapiro. Geodesic matting: A framework for fast interactive image and video segmentation and matting. *Int. J. Comput. Vision*, 82(2):113–132, 2009. 2

[5] Y. Boykov and G. Funka-Lea. Graph cuts and efficient n-d image segmentation. *International Journal of Computer Vision*, 70(7):109–131, 2006. 1, 2

[6] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112, 2001. 2

[7] W. Casaca, A. Paiva, E. Gomez-Nieto, P. Joia, and L. G. Nonato. Spectral image segmentation using image decomposition and inner product-based metric. *Journal of Mathematical Imaging and Vision*, 45(3):227–238, 2013. 2, 3

[8] C. Couprie, L. Grady, L. Najman, and H. Talbot. Power watershed: A unifying graph-based optimization framework. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(7):1384–1399, 2011. 1, 2, 5

[9] T. Cour, F. Bénézit, and J. Shi. Spectral segmentation with multiscale graph decomposition. In *Proc. of the IEEE CVPR*, pages 1124–1131, 2005. 3

[10] J. Cousty, G. Bertrand, L. Najman, and M. Couprie. Watershed cuts: Minimum spanning forests and the drop of water principle. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(8):1362–1374, 2009. 1, 2, 5

[11] T. A. Davis and W. W. Hager. Dynamic supernodes in sparse cholesky update/downdate and triangular solves. *ACM Trans. Math. Softw.*, 35(4), 2009. 4

[12] F. J. Estrada and A. D. Jepson. Benchmarking image segmentation algorithms. *International Journal of Comput. Vision*, 85(2):167–181, 2009. 6

[13] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006. 1, 2, 5

[14] L. Grady. Targeted image segmentation using graph methods. In O. Lézoray and L. Grady, editors, *Image Processing and Analysis with Graphs*, pages 111–135. CRC Press, 2012. 1, 3

[15] V. Jankovic. Quadratic functions in several variables. *The Teaching of Mathematics*, VIII:53–60, 2005. 4

[16] T. H. Kim, K. M. Lee, and S. U. Lee. Learning full pairwise affinities for spectral segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(7):1690–1703, 2013. 3

[17] V. S. Lempitsky, P. Kohli, C. Rother, and T. Sharp. Image segmentation with a bounding box prior. In *ICCV*, pages 277–284. IEEE, 2009. 2

[18] S. Maji, N. K. Vishnoi, and J. Malik. Biased normalized cuts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2057–2064. IEEE, 2011. 2

[19] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int. Conf. Computer Vision (ICCV)*, volume 2, pages 416–423, July 2001. 5

[20] M. Meilă. Comparing clusterings: an axiomatic view. In *Proceedings of the 22nd international conference on Machine learning*, ICML '05, pages 577–584, New York, NY, USA, 2005. ACM. 5

[21] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. (SIGGRAPH 04)*, 23(3):309–314, 2004. 1, 2, 5

[22] O. Sorkine. Differential representations for mesh processing (the state of the art report). *Computer Graphics Forum (Eurographics)*, 25(4):789–807, 2006. 1

[23] R. Unnikrishnan, C. Pantofaru, and M. Hebert. Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):929–944, 2007. 5

[24] S. Vicente, V. Kolmogorov, and C. Rother. Graph cut based image segmentation with connectivity priors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 0, pages 1–8. IEEE Computer Society, 2008. 1

[25] B. Wang and Z. Tu. Affinity learning via self-diffusion for image segmentation and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2312–2319, 2012. 1

[26] K. Xu, H. Zhang, D. Cohen-Or, and Y. Xiong. Dynamic harmonic fields for surface processing. *Computer and Graphics*, 33(3):391–398, 2009. 1

[27] J. Zhang, J. Zheng, and J. Cai. A diffusion approach to seeded image segmentation. In *Proceedings of the IEEE CVRP*, pages 2125–2132. IEEE, 2010. 2