Multi-level Graph Label Propagation for Image Segmentation

Ivar Vargas Belizario and Joao Batista Neto
Institute of Mathematics and Computer Science (ICMC)
University of Sao Paulo (USP), São Carlos, SP, Brazil
Email: ivar@usp.br and jbatista@icmc.usp.br

***Abstract*—This article introduces a multi-level automatic image segmentation method based on graphs and Label Propagation (LP), originally proposed for the detection of communities in complex networks, namely MGLP. To reduce the number of graph nodes, a super-pixel strategy is employed, followed by the computation of color descriptors. Segmentation is achieved by a deterministic propagation of vertex labels at each level. Several experiments with real color images of the BSDS500 dataset were performed to evaluate the method. Our method outperforms related strategies in terms of segmentation quality and processing time. Considering the Covering metric for image segmentation quality, for example, MGLP outperforms LPCI-SP, its most similar counterpart, in 38.99%. In term of processing times, MGLP is 1.07 faster than LPCI-SP.**

I. Introduction

Automatic image segmentation, still an open problem in image processing, plays an important role in tasks such as classification, object detection and tracking.

Several approaches are described in the literature for the automatic segmentation of images, raging from simple threshold, region growing or edge detection strategies to the more elaborate ones which include graph clustering, machine learning, and other hybrid approaches [1]–[3].

For graph-based methods, it is crucial to create an adequate representation of the image, that is, one in which the graph model conveys all the necessary information to generate an accurate segmentation. To this end, vertices are usually represented by pixels and edge weights are obtained by computing the similarity between pairs of vertices, according to a certain neighborhood criterion. The resulting segmentation is achieved by grouping similar vertices with graph clustering algorithms.

In a classical representation, an image with n pixels is represented by a graph where the number of vertices is equal to the number of pixels and the maximum number of edges can be of order $n(n-1)/2$. The computational cost of graph clustering methods is associated with the cardinality of the graph (number of vertices and edges). Hence, finding an optimal graph partition that determines a good segmentation is an NP-hard problem [4].

Therefore, it is crucial to have an image representation with the fewest possible number of vertices and edges, which also provides an accurate segmentation. To reduce the computational cost, the pre-segmentation of images in small homogeneous regions has been proposed. One such strategy is known as super-pixels [5]–[8].

As for the graph clustering procedure, a large number of methods has been proposed. The best approaches are: Normalized Cut [9], Fast Greedy [10], Label Propagation [11], Louvain [12] and Efficient Graph-Based Image Segmentation [4]. Most of these methods were originally formulated in the general context of clustering for data types other than images. And if at all applied to the aforementioned domain, important image information was simply ignored.

The Label Propagation method, for example, was originally proposed to detect communities in complex networks with more than 1 million vertices [13]. However, when used for image segmentation, with no knowledge of the image domain, resulted in an over-segmented image. Moreover, because of its random propagation strategy, results are not deterministic and, hence, reproducibility does not hold. This is not desirable in image segmentation.

However, Label Propagation is of $\mathcal{O}(m)$ complexity - with m being the number of edges - making it attractive to formulate faster methods that can handle the clustering of large graphs.

On the other hand, the inherent multi-level nature of both Fast Greedy and Louvain methods allows the clustering of graphs up to 100 million vertices [12]. In graph-based multi-level methods, sub-sets of vertices are clustered together to form a new vertex at the next level. This guarantees a "compression" of the graph at higher level and, hence, a gain in processing times.

We hypothesize that by combining the linear complexity of the label propagation, a multi-level and super-pixel strategy to deal with the clustering of large graphs and, above all, the inclusion of contextual information in images, a deterministic automatic and fast image segmentation approach can be attained.

In this work we propose an automatic multi-level image segmentation method based on graphs and label propagation. The pipeline of our proposal is illustrated in Fig. 1. Given a natural scene image, super-pixels are extracted, followed by the computation of color descriptors. Then, at each level, a graph is then created, where each vertex is a super-pixel and edge weights represent the similarity between adjacent super-pixels. Each vertex is assigned a distinct label. Next, the label propagation is performed. During this process, some vertices will swap labels and, eventually, neighboring vertices will have identical labels. The resulting segmentation consists of regions

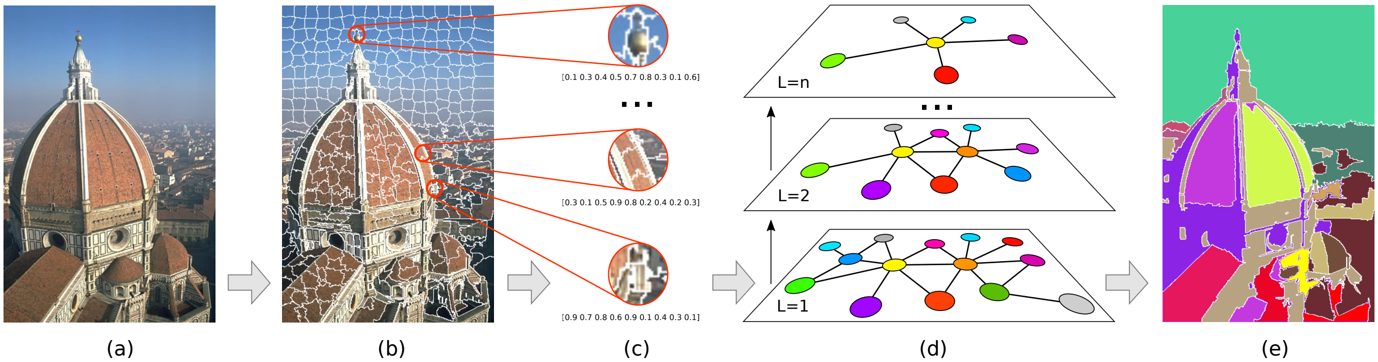


Fig. 1. The proposed method: (a) Input image; (b) super-pixel pre-segmentation; (c) feature extraction; (d) multi-level propagation where new graphs are created by merging similar super-pixels from the previous level; (e) final segmentation at the last level.

with the same label. Before moving into the next level, the number of super-pixels is naturally reduced by grouping all super-pixels with the same label of the current level into a new super-pixel. The descriptors are also updated. Notice that this process does not dictated a fixed number of levels. It iteratively builds a new level and only stops when the number of super-pixels between contiguous levels no longer changes.

The main contributions of this work are the following:

- A multi-level strategy to create graphs that represent images in order to simplify the propagation of labels;
- Introduction of two deterministic traversal strategies to propagate labels suitable for the image context;
- A fast iterative multi-level method for color images segmentation based on label propagation.

The remainder of the paper is structured as follows. Section II brings the theoretical background. Section III presents related work on graph-based image segmentation. Section IV details the proposed method named Multi-level Graph Label Propagation (MGLP). Materials and methods are given in Section V. Experimental results and comparisons are given in Section VI. Conclusions are finally drawn in Section VII.

II. Theoretical Background

This section presents the theoretical background that underpins the proposed segmentation method: super-pixels and graph clustering methods.

A. *Super-pixels*

Super-pixel methods are a relatively recent proposal [14]–[16] whose aim is to group pixels according to a certain similarity criterion. The quality of super-pixels can be measured by their ability to adjust precisely to the image boundaries and their main importance is associated not only with the reduction of computational costs of the underlying image analysis tasks but also to provide a better perceptual meaning of the image, as they carry more information (such as texture and shape) than a single pixel [15].

Among the various existing super-pixel methods, two deserve special attention due to their linear computational cost and high quality of the results. Simple Linear Iterative Clustering (SLIC) [15] is a method based on the k-means clustering algorithm, where the pixels are interactively clustered in order to minimize the distance between the pixels and the super-pixel centroid. The Simple Not Iterative Clustering (SNIC) [16] is presented as an improvement of the SLIC method [15], both in quality and in computational cost. SNIC uses a queue that prioritizes the distance between the pixels and the centroids of the super-pixels. In comparison to SLIC, SNIC does not require iterations, which makes it even faster.

The main steps of the SNIC algorithm are as follows: (i) given an image with N pixel, and a side of initial length S, $k = N/S^2$ regular super-pixels of dimensions $S \times S$ are created; (ii) initialize the k centroids with color and spatial information of the pixels located in the center of each regular super-pixel. (iii) insert the centroids in queue Q, whose priority is the distance between the pixels and the centroids of super-pixels; (iv) while Q is not empty, remove an element e from Q associated with the nearest centroid C_k, add the pixel e in the super-pixel and update the centroid C_k. Compute priorities based on distances between the g pixels neighbors of e and the centroid C_k. Then, insert the g pixels with their respective priorities in Q.

B. *Graph Clustering Methods*

Fast Greedy (FG) [10] is a multi-level greedy agglomerative algorithm based on the Modularity measure, which assesses the quality of dividing a graph into communities. Ideally, a good division is one in which a high number of edges between vertices in the same community and a low density of edges between members of different communities occur. The FG method applies a greedy strategy to approximate an ideal division of the graph by maximizing Modularity. One of the main advantages of FG is the low computational cost. For sparse graphs, the complexity is $\sim \mathcal{O}(n\ log^2\ n)$. Another important feature is that FG automatically determines the number of clusters in the graph. The main disadvantage in some applied domains is its non-deterministic nature.

The Lovain (LV) [12] method is a greedy multi-level algorithm - also based on Modularity - for the detection of communities in large graphs. The LV method also seeks to maximize Modularity and consists of two iterative phases: (i) Optimization of Modularity: initially, each vertex is labeled as a community. The vertices can swap communities when a gain in Modularity occurs. (ii) Aggregation: a new graph is created, observing the new arrangement of vertices in the communities created in the first phase. Both (i) and (ii) are processed until no further gain in Modularity occurs. The complexity of the LV method is linear $\mathcal{O}(n)$ for graphs with $n \approx m$, where n is the number of vertices and m the number of edges. For dense graphs, it is approximately $\mathcal{O}(n \, log \, n)$. The major drawback of the method, when applied in the image segmentation domain, is over-segmentation.

The Label Propagation (LP) [11] method was also originally proposed to detect communities in complex networks. The LP algorithm has a simple formulation: (i) Initialize vertices with different labels. (ii) Iteratively traverse the vertices randomly and propagate the labels according to the highest frequency of identical labels in the neighborhood of each vertex. Stop the algorithm when most neighbors have the same label. The complexity of LP method is $\mathcal{O}(m)$, where m is the number of edges. This property makes it suitable for clustering graphs with up to 1 million vertices.

III. Related Work

Efficient Graph-Based Image Segmentation (EGBIS) [4] was introduced for automatic image segmentation. It creates regions, consisted of vertices which represent location and color information, by means of merging operations. Edge weights are computed by measuring the distance among vertices. Initially, each pixel is a component. Then, those linked by edges with the lowest weights are gradually merged. This confers EGBIS a multi-level behavior. The complexity is approximately linear $\sim O(m)$, where m is the number of edges. A variation of the EGBIS method [17] represents vertices as super-pixels. The authors highlight three reasons for using super-pixels instead of single pixels, as they: i) convey information such as shape and texture; ii) drastically reduces the number of vertices and, therefore, drops the computational cost; and iii) add more coherence and robustness to the result.

Automatic image segmentation with FG clustering and super-pixel [14] has already been conducted [5], [6]. The authors used super-pixels to convey color and texture information and concluded that colors descriptors were more effective in segmenting natural scene images.

The LV method has also been used for automatic segmentation [18]. This is a pixel-based strategy which uses the pixel intensity only. The work was further extended to include super-pixels [8]. The authors proposed a strategy to solve the over-segmentation problem by combining pairs of similar regions.

LP has also been employed to automatic segmentation [5]. According to the authors, the results were not as accurate as that obtained with the FG algorithm. The main reason being the over-segmentation. Like Fast Greedy, the LP approach, as implemented, was non-deterministic. This was due to the random strategy adopted for the propagation of labels.

IV. The proposed Method - MGLP

The major clustering methods FG, LV and LP show pros and cons when applied as image segmentation solutions. They have linear complexity, for example. When combined with super-pixel, more information is conveyed and the quality of segmentation improves. The major drawbacks are over-segmentation and, especially, the non-deterministic nature, which is not desirable in image segmentation. Particularly, no multi-level implementation for LP method is available, unlike FG, LV and EGBIS.

We introduce a method based on the LP clustering, that is both multi-level and deterministic, while preserving the low computational cost. In this section we detail the main steps of the proposed Multi-level Graph Label Propagation (MGLP) method as depicted in Fig. 1. The steps described in sections IV-A and IV-B are performed at the first level only, whereas the others are executed at all levels.

A. Super-pixels Pre-segmentation

Let N and S be, respectively, the number of pixel of an image and the side's length of a squared super-pixel. Hence, $R = N/S^2$ super-pixel can be extracted from the image, each representing a vertex of the graph. As the value of S changes, so does the number of vertices in the graph, which impacts the segmentation result. To establish the most appropriate value of S is an issue discussed in our experiments.

In this work, we opted for the SNIC super-pixel method [16] due to its precise fit along image boundaries and low processing time, as described in sub-section II-A.

B. Feature Extraction

We devised a color descriptor named CM9 that operates over each super-pixels. It is so called because it computes the first three statistical moments (mean, variance and skewness) for each of the three color channels in the CIELAB color space, yielding a 9-dimensional features vector. All CM9 descriptors are then normalized to zero mean and unit variance. Each super-pixel and its associate feature vector are then assigned a distinct label.

C. Graph Building

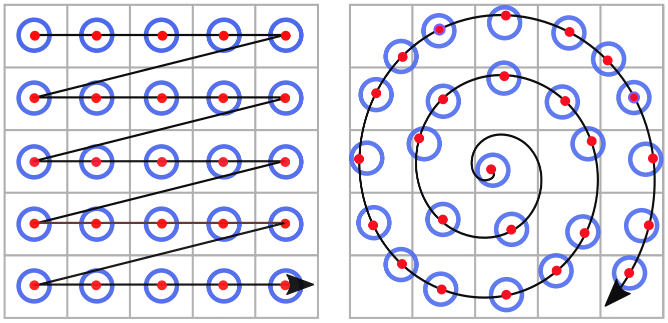
At each level, an indirect weighted graph is created. The vertices are represented by a CM9 descriptor and edges are determined according to a similarity function $F(.)$ among all j neighbors of vertex i. The edge weights are computed by

$$W_{ij} = \begin{cases} F(i,j), & \text{If} \;\; j \in N(i) \;\; \text{and} \;\; F(i,j) \geq T \\ 0, & \text{else} \end{cases} \tag{1}$$

where $N(i)$ represents all neighbors of vertex i, T is the threshold that delimits the weight of the edges and the similarity function $F(.)$ is the Gaussian function, given by

$$GAU(i,j) = e^{\frac{-d(i,j)^2}{2\sigma^2}} \tag{2}$$

where σ is set to 0.5 for all experiments and $d(i,j)$ is the Euclidean distance.



(a) (b)

Fig. 2. Traversal strategies for super-pixel label propagation. a) CON, b) SPI

D. Super-pixels Traversal

To ensure optimal propagation of the labels, we have defined two deterministic strategies for traversing the super-pixels, as illustrated in Fig. 2: (i) CON, traversal in convolution form, that starts at the top left-hand super-pixel of the image and continues from left to right, top to bottom (Fig. 2a); (ii) SPI, traversal in a spiral shape-like manner, starting at the central super-pixel, until all super-pixels are visited (Fig. 2b).

E. Label Propagation

We redefined the random propagation method LP introduced by [5], [11]. Our propagation proposal, as explained in section II-B, is deterministic and performed just once, in a single pass. In the original LP method this is an iterative procedure. During traversal, the label of each vertex is updated according with the following equation:

$$f(i) = \underset{j \in N(i)}{\operatorname{argmax}} \left(\sum L_j(\delta(i,j)) \right), \tag{3}$$

where i is the vertex whose label has to be updated, j represents the neighboring vertices of i, L_j represents the label of j, $f(i)$ returns a new label with the highest frequency among the labels of the j neighbors of vertex i and $\delta(i,j)$ counts the frequency of label L_j, given by

$$\delta(i,j) = \begin{cases} 1, & \text{If } W_{ij} \geq T, \\ 0, & \text{else} \end{cases} \tag{4}$$

where T is the threshold parameter introduced in IV-C.

F. Super-pixel and Descriptor Update

After the label propagation step, super-pixels with the same labels are merged and the CM9 descriptors are updated as follows:

$$V_i = \left[\frac{1}{|e|} \sum_{j \in e} (v_{j1}), ..., \frac{1}{|e|} \sum_{j \in e} (v_{jm}) \right] \tag{5}$$

where V_i is the new descriptor, e represents the set of descriptors belonging to super-pixels with the same label, $|e|$ is the number of super-pixels with the same label and m is descriptor dimension.

The iterative multi-level approach works as follows: if the number of super-pixels in contiguous levels is the same, stop the algorithm; otherwise, create a new level and run the algorithm from the Graph Building step, as described in section IV-C.

The pseudo-code for our iterative multi-level approach is given in Algorithm 1.

Algorithm 1: Multi-level Graph Label Propagation - MGLP

Input : I: image; T: threshold; S: super-pixel side length.
Output: image segmentated.

```
/* (A) Super-pixels Pre-segmentation     */
R ← Superpixels(I, S) ;                  // SNIC
/* (B) Feature Extraction                */
V ← FeatureExtraction(R)
L_i ← i, i ∈ V;                          // different labels
nextlevel ← True
while nextlevel do
    /* (C) Graph Building                */
    {G = (V, E, W)} ← GraphBuilding(R, T)
    /* (D) Super-pixel Traversal         */
    X ← SuperpixelsTraversal(R)
    /* (E) Label Propagation             */
    foreach i ∈ X do
        L_i = f(i) ;                     // Eq(3)
    /* (F) Super-pixel/Descriptor Update */
    {R, V} ← Update(R, V, L)
    /* Stop Evaluation                   */
    nextlevel ← StopEvaluation()
return R
```

V. Materials and Methods

This sections describes the evaluation metrics and the dataset used in this work. We also describe how to select a reference image from the dataset to perform quantitative analysis.

A. Evaluation Metrics

Three metrics commonly used to quantitatively assess segmentation have been considered: the first is based on the coverage of regions; the second, on distances among the object contours and the third, a measure based on probabilities.

*1) **Covering**:* let S_T be the segmentation result and S_G a reference segmentation (ground truth). The coverage of regions (Covering) [19], [20] quantifies the coverage of the regions of S_T in relation to the regions of S_G. It is defined by:

$$Covering(S_T \rightarrow S_G) = \frac{1}{A} \sum_{R \in S_G} |R| \max_{R' \in S_T} \{\mathcal{O}(R, R')\}, \tag{6}$$

$$\mathcal{O}(R, R') = \frac{|R \cap R'|}{|R \cup R'|} = \frac{|R \cap R'|}{|R| + |R'| - |R \cap R'|}, \tag{7}$$

where A is the number of pixels in the image; R and R' are regions of S_G and S_T, respectively; $\mathcal{O}(R, R')$ represents the overlap between the R and R' regions and $|R|$ and $|R'|$ the number of pixels in the R and R' regions, respectively. Eq. 6 returns values in the range $(0, 1)$. The higher the value, the higher the similarity among the regions of S_T and S_G.

*2) **Boundary-Based Measures**:* the quantitative assessment for boundary-based segmentation [21]–[24] is estimated by calculating the minimum distances between the pairs of points of 2 sets of boundaries: (i) BT, the boundaries of some segmentation method S_T and (ii) BG, the boundaries of a ground-truth reference segmentation S_G. Similarity is attained by computing the Precision (P), Recall (R) and BF1-Score, given by:

$$P = \frac{1}{|BT|} \sum_{p \in BT} [Matched(p, BG) \leq \theta] \tag{8}$$

$$R = \frac{1}{|BG|} \sum_{p \in BG} [Matched(p, BT) \leq \theta] \tag{9}$$

$$BF1 - Score = 2 \times \frac{P \times R}{R + P}, \tag{10}$$

The function $Matched(.)$, in P, traverses the p points of BT in search of points near to the BG boundaries, according to a maximum distance θ. If $[.]$ is true, it returns 1, otherwise 0. The function $Matched(.)$, in R traverses the p points of BG searching for points near the BT boundaries. $|BT|$ and $|BG|$ are the number of points in boundaries BT and BG, respectively.

*3) **Probabilistic Rand Index (PRI)**:* the PRI metric [25], [26] computes the probability of a pair of pixels (i, j), belonging to a segmentation from method S_T, having consistent labels in the set of k reference S_{G_k} segmentation (ground-truth). The PRI metric is defined as:

$$PRI(S_T, \{S_{G_k}\}) = \frac{1}{A} \sum_{i<j} [c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})] \tag{11}$$

where c_{ij} is the probability of pixels i and j having the same label in segmentation S_T and p_{ij} corresponds to the probability of pixels i and j sharing the same label in the set of reference segments S_{G_k} and A is the total number of pixel pairs. Function PRI returns values in the range of $(0, 1)$. The higher the values, the higher the similarity between S_T and S_{G_k}.

B. Dataset

The dataset BSDS500 [26] has been employed to evaluate the proposed segmentation MGLP method. It contains 500 images of natural scenes, each containing 5-10 man-made segmented samples. The dataset consists of 200 images for training, 200 images for testing and 100 images for validation. Images are of dimension 481x321, yielding graphs with a maximum $154,401$ vertices. The manual segmentation of most BSDS500 samples vary widely, as they are subject to the perception of each human.

Due to this high variability, we have developed a strategy to select a single reference manual segmentation, in order to carry out a quantitative assessment of the proposed method. We add all contours of all manual segmented samples of a particular image to produce a combined contour image. We then apply Eq. 10 to select the manual reference segmentation with the highest BF1-Score value. In this scenario, the manual segmentation correspond to the contours of BT and the image of the combined contour, correspond to the term BG, as explained in the sub-section V-A2. This process was carried out for all 200 test images and all 100 validation images of the dataset. The former was necessary for the parameter setup. The latter, for the evaluation of the proposed method.

The source code was written in C/C++[1]. We have performed our experiments on a Linux workstation (Intel Core i7-4810MQ CPU 2.80GHz x 4 with 16GB memory).

VI. Experimental Results

In this section we present quantitative and qualitative results. We also describe the scheme adopted for defining the most appropriate parameter values as part of the MGLP method, namely: the squared super-pixel side of length S and the threshold T.

A. Assessment of Parameters

The performance of the MGLP method is influenced by the 2 aforementioned parameters. The following experiment aims to show this behavior as we change the values of S and T and, hence, come up with the most appropriate value for both parameters.

The experiment setup is as follows: parameter S is evaluated in the range 2–50, 1 increment, yielding 49 distinct values. Similarly, parameter T is evaluated in the value range 0.0–0.95, with 0.05 increment, totaling 20 values. Combining both set of values and considering the 200 test samples present in the BSDS500 dataset, our method was executed $392,000$ times. MGLP was executed for both traversal strategies, CON and SPI.

Quantitative results are shown in Fig. 3. The plots reveal that the best results were those in which S and T fluctuate in the range 15–30 and 0.20–0.30, respectively. As for the running time, we notice a considerable drop for values of S greater than or equal to 18. The larger the values of S, the fewer the number of vertices in the graph. The choice of the label propagation traversal strategy (either CON or SPI) did not play a significant impact in terms of quality and processing times. They were both very similar. A qualitative outcome for the proposed method is illustrated in Fig. 4. It shows the progress of the segmentation at each level of propagation.

B. Comparison with Related Methods

We compared our MGLP method with other similar segmentation methods based on graphs and complex networks. The following methods were chosen: EGBIS [4], SUTP-FG [5], [6], LV [18] e LPCI [5].

[1]The MGLP source code is available at https://ivarvb.github.io/mglp/

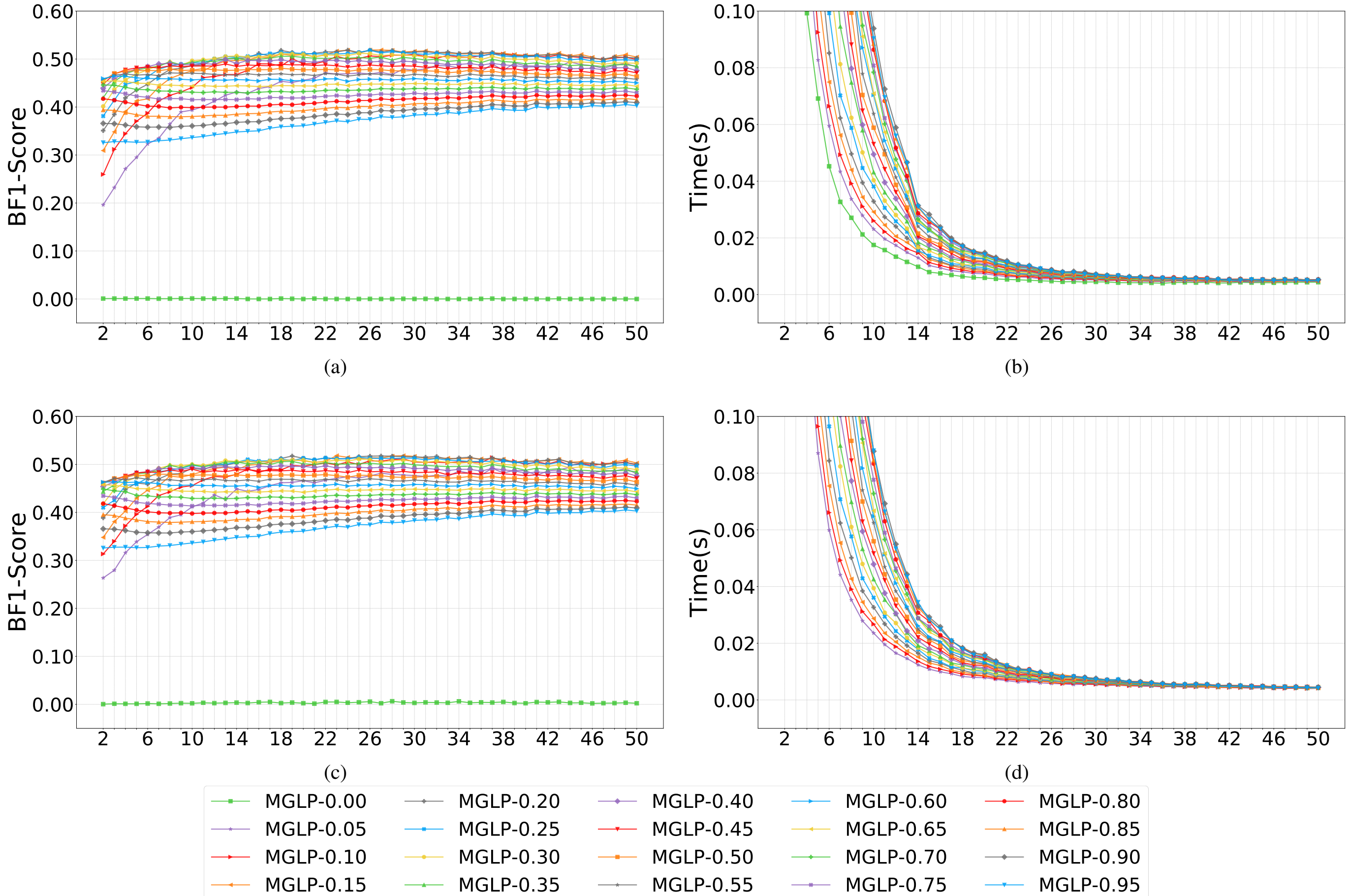


Fig. 3. Evaluation of parameters S and T. (a) BF1-Score for CON, (b) Time for CON, (c) BF1-Score for SPI, (d) Time for SPI. X axis represents S. Accuracy and processing time are shown in the Y axis.

Based on the outcome of the previous experiment, we adopted the mean value for both S and T MGLP parameters. Hence, we set $S = 18$ and $T = 0.25$. As for the other methods, we adopted the parameter setup suggested by the authors. To avoid bias, comparison was conducted with the validation subset (100 samples), whereas the parameter tuning was performed with the 200 test samples.

As the original EGBIS-P method [4] is a pixel-based segmentation method, we implemented a super-pixel version called EGBIS-SP. This contributes for a fairer comparison. The following configuration was adopted: SNIC super-pixel method, with initial side of lengths $S = 18$; use of the proposed CM9 descriptor. The edges were created with the first neighbors of each vertex and Euclidean distance as the similarity function.

As proposed by the authors, the SUTP-FG [5], [6] method was performed with the following configuration: SUTP [14] as the super-pixel method, with an initial side equal to $S = 10$ and a neighborhood radius of 5. Super-pixels were described by a three-dimensional color descriptor based on the CIELAB average value of all pixels belonging to a super-pixel. The weights of the edges were created with Euclidean distances less than or equal 6.

The LV method [18], also pixel-based, was also adapted for fairness. We implemented LV-SP, a super-pixel-based version, configured as follows: SNIC super-pixel method, with initial side of length $S = 18$; CM9 descriptor; neighborhood radius 4, similarity function GAU and threshold $T = 0.25$.

The LPCI-SP, an adaptation of the SUTP-FG method, was run with the following configuration: SNIC super-pixel method with initial side length $S = 18$, descriptor CM9, neighborhood radius 3, similarity function GAU and threshold $T = 0.25$.

The results are presented in Table I. To distinguish between the SPI and CON traversal used, the methods are referred as MGLP-SPI and MGLP-CON, respectively. The proposed method yields better results for all metrics, except for PRI which shows EGBIS-P as the best option. As in the previous experiment, both MGLP-SPI and MGLP-CON show similar results, with a slight advantage for the former.

Among all the methods, LPCI-SP is the one that best resemble our proposal. However, we outperform LPCI-SP both is quality and processing time. As for the processing time, the reasons are two-fold: (i) the MGLP requires fewer edges are they are created with the first neighbors of each vertex only; (ii) the multi-level strategy accelerates the propagation of labels. Qualitative results are presented in Fig. 5.

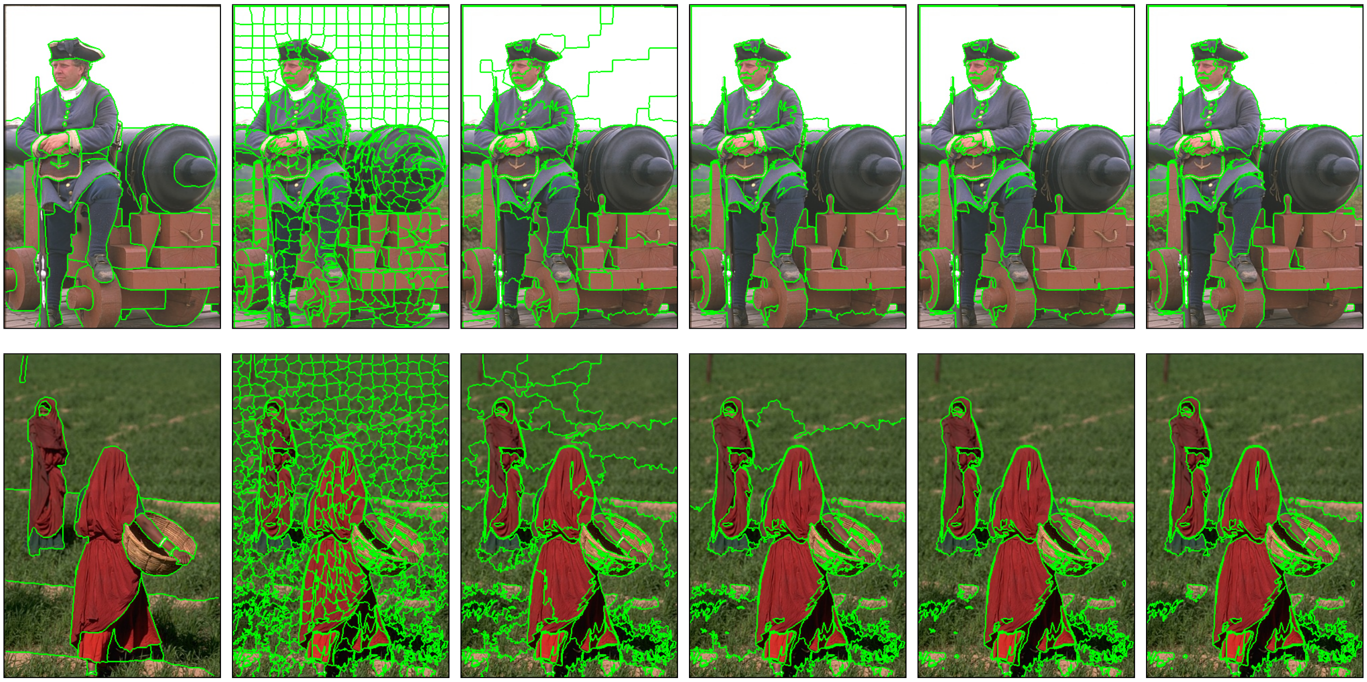


Fig. 4. Qualitative results for SPI traversal strategy with $S = 18$ and $T = 0.25$. First column: ground truth; 2nd-6th columns: resulting segmentation for levels 1-5, respectively.

TABLE I
COMPARISON WITH OTHER METHODS. THE TWO BEST RESULTS ARE HIGHLIGHTED FOR EACH METRIC.

Method	Covering Mean (std)	BF1-Score Mean (std)	PRI Mean (std)	Time Mean (s)
EGBIS-P	0.4844 (± 0.1598)	0.4717 (± 0.1428)	**0.8143** (± 0.1181)	0.2354
EGBIS-SP	0.4746 (± 0.1073)	0.4735 (± 0.1150)	0.7982 (± 0.1372)	**0.0004**
LPCI-SP	0.3754 (± 0.1057)	0.4665 (± 0.1288)	0.7858 (± 0.1532)	0.0103
LV-SP	0.4625 (± 0.1004)	0.4993 (± 0.1333)	0.7956 (± 0.1386)	0.0646
SUTP-FG	0.4973 (± 0.1770)	0.4868 (± 0.1642)	0.7547 (± 0.1316)	16.0452
MGLP-CON (**our**)	**0.5173** (± 0.1831)	**0.4965** (± 0.1435)	0.7993 (± 0.1346)	**0.0096**
MGLP-SPI (**our**)	**0.5263** (± 0.1777)	**0.5023** (± 0.1406)	**0.8063** (± 0.1320)	**0.0096**

VII. CONCLUSION

In this article, we presented MGLP, a fast and accurate multi-level method for automatic image segmentation, based on a re-formulation of the label propagation method originally used to identify communities in complex networks. Our method incorporates 2 deterministic propagation strategies that take into account information from the image domain. Several experiments carried out with the BSDS500 dataset have shown that the proposed method is faster and more accurate when compared with similar graph-based methods. Due to its linear complexity, we plan to extend its application to 3D data, by swapping super-pixels to super-voxels.

ACKNOWLEDGMENT

This research was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

REFERENCES

[1] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*. Cengage Learning, 2014.

[2] F. Garcia-Lamont, J. Cervantes, A. López, and L. Rodriguez, "Segmentation of images by color features: A survey," *Neurocomputing*, vol. 292, pp. 1 – 27, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218302364

[3] N. M. Zaitoun and M. J. Aqel, "Survey on image segmentation techniques," *Procedia Computer Science*, vol. 65, pp. 797 – 806, 2015, international Conference on Communications, management, and Information technology (ICCMIT'2015). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050915028574

[4] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, Sep 2004.

[5] O. Cuadros, G. Botelho, F. Rodrigues, and J. B. Neto, "Segmentation of large images with complex networks," in *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, Aug 2012, pp. 24–31.

[6] O. A. C. Linares, G. M. Botelho, F. A. Rodrigues, and J. B. Neto, "Segmentation of large images based on super-pixels and community detection in graphs," *IET Image Processing*, vol. 11, no. 12, pp. 1219–1228, 2017.

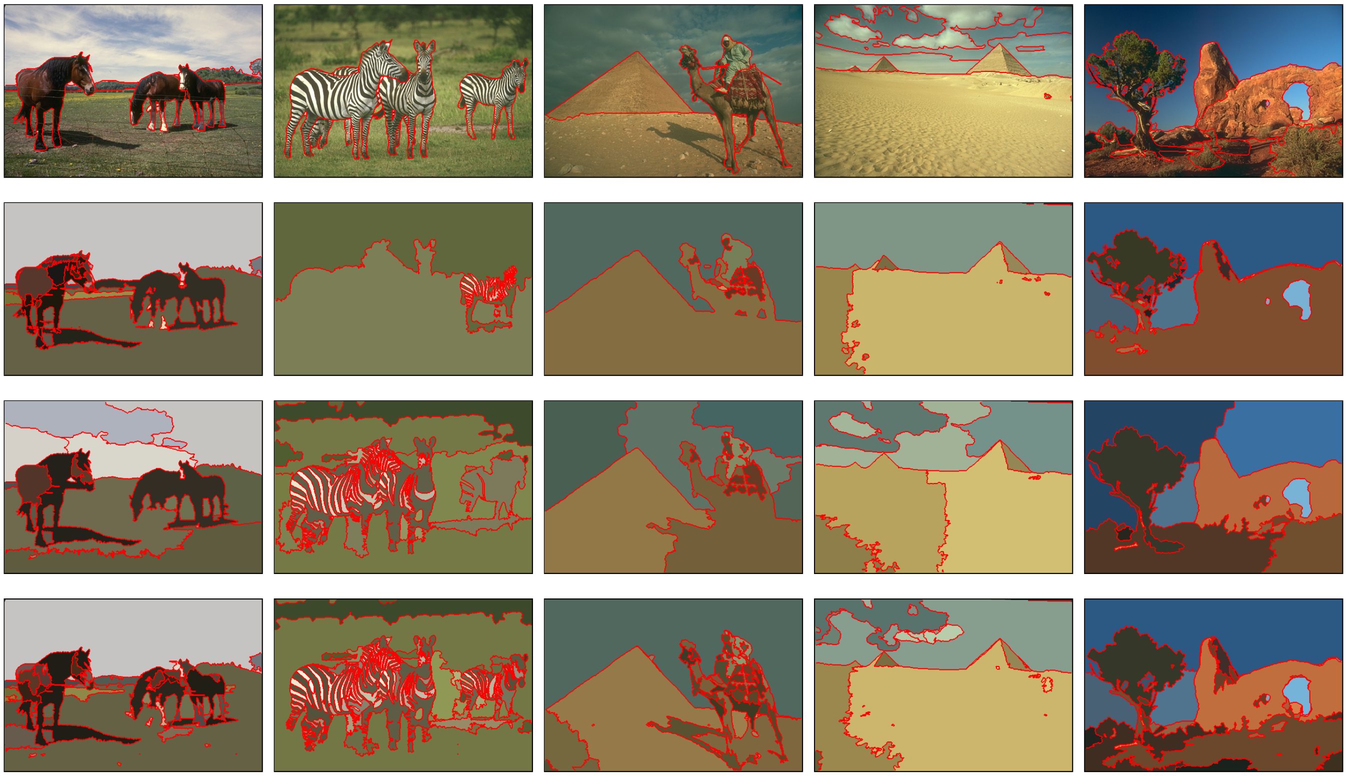


Fig. 5. Qualitative results. By row: Ground truth, SUTP-FG, LV-SP and MGLP-SPI (**our**).

[7] C. Cigla and A. A. Alatan, “Efficient graph-based image segmentation via speeded-up turbo pixels,” in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, September 2010, pp. 3013 –3016.

[8] T. Nguyen, M. Coustaty, and J. Guillaume, “A new image segmentation approach based on the louvain algorithm,” in *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*, Sep. 2018, pp. 1–6.

[9] Jianbo Shi and J. Malik, “Normalized cuts and image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, Aug 2000.

[10] A. Clauset, M. E. J. Newman, and C. Moore, “Finding community structure in very large networks,” *Phys. Rev. E*, vol. 70, p. 066111, Dec 2004.

[11] U. N. Raghavan, R. Albert, and S. Kumara, “Near linear time algorithm to detect community structures in large-scale networks,” *Phys. Rev. E*, vol. 76, p. 036106, Sep 2007.

[12] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 10, p. 10008, Oct. 2008.

[13] L. Šubelj and M. Bajec, “Unfolding communities in large complex networks: Combining defensive and offensive label propagation for core extraction,” *Phys. Rev. E*, vol. 83, p. 036103, Mar 2011. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.83.036103

[14] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, “Turbopixels: Fast superpixels using geometric flows,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, Dec 2009.

[15] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “Slic superpixels compared to state-of-the-art superpixel methods,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, Nov 2012.

[16] R. Achanta and S. Süsstrunk, “Superpixels and polygons using simple non-iterative clustering,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 4895–4904.

[17] M.-M. Cheng, Y. Liu, Q. Hou, J. Bian, P. Torr, S.-M. Hu, and Z. Tu, “Hfs: Hierarchical feature selection for efficient image segmentation,” in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 867–882.

[18] Y. Mourchid, M. E. Hassouni, and H. Cherifi, “A new image segmentation approach using community detection algorithms,” in *2015 15th International Conference on Intelligent Systems Design and Applications (ISDA)*, Dec 2015, pp. 648–653.

[19] D. Crevier, “Image segmentation algorithm development using ground truth image data sets,” *Comput. Vis. Image Underst.*, vol. 112, no. 2, pp. 143–159, Nov. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.cviu.2008.02.002

[20] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, “From contours to regions: An empirical evaluation,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2294–2301.

[21] Qian Huang and B. Dom, “Quantitative methods of evaluating image segmentation,” in *Proceedings., International Conference on Image Processing*, vol. 3, Oct 1995, pp. 53–56 vol.3.

[22] D. R. Martin, C. C. Fowlkes, and J. Malik, “Learning to detect natural image boundaries using local brightness, color, and texture cues,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, May 2004.

[23] F. J. Estrada and A. D. Jepson, “Benchmarking image segmentation algorithms,” *International Journal of Computer Vision*, vol. 85, no. 2, pp. 167–181, Nov 2009. [Online]. Available: https://doi.org/10.1007/s11263-009-0251-z

[24] D. Dogra, A. Majumdar, and S. Sural, “Evaluation of segmentation techniques using region area and boundary matching information,” *Journal of Visual Communication and Image Representation*, vol. 23, no. 1, pp. 150 – 160, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320311001210

[25] R. Unnikrishnan and M. Hebert, “Measures of similarity,” in *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION’05) - Volume 1*, vol. 1, Jan 2005, pp. 394–394.

[26] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, “Contour detection and hierarchical image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, May 2011.