A Global/Local Affinity Graph for Image Segmentation

Xiaofang Wang, Yuxing Tang, Simon Masnou, and Liming Chen *Senior IEEE Member*

Abstract

Construction of a reliable graph capturing perceptual grouping cues of an image is fundamental for graph-cut based image segmentation methods. In this paper, we propose a novel sparse global/local affinity graph over superpixels of an input image to capture both short and long range grouping cues, thereby enabling perceptual grouping laws, *e.g.*, proximity, similarity, continuity, to enter in action through a suitable graph cut algorithm. Moreover, we also evaluate three major visual features, namely color, texture and shape,for their effectiveness in perceptual segmentation and propose a simple graph fusion scheme to implement some recent findings from psychophysics which suggest combining these visual features with different emphases for perceptual grouping. Specifically, an input image is first oversegmented into superpixels at different scales. We postulate a gravitation law based on empirical observations and divide superpixels adaptively into small, medium and large sized sets. Global grouping is achieved using medium sized superpixels through a sparse representation of superpixels' features by solving a ℓ_0-minimization problem, thereby enabling continuity or propagation of local smoothness over long range connections. Small and large sized superpixels are then used to achieve local smoothness through an adjacent graph in a given feature space, thus implementing perceptual laws, *e.g.*, similarity and proximity. Finally, a bipartite graph is also introduced to enable propagation of grouping cues between superpixels of different scales. Extensive experiments are carried out on the Berkeley Segmentation Database in comparison with several state of the art graph constructions. The results show the effectiveness of the proposed approach

Xiaofang Wang, Yuxing Tang and Liming Chen are with Department of Mathematics and Computer Science, Ecole Centrale de Lyon, Ecully, 69130, France (email: {xiaofang.wang, yuxing.tang, liming.chen}@ec-lyon.fr). Simon Masnou is with Mathematics at Institut Camille Jordan, University Lyon 1, Villeurbanne, 69622, France.

This work was supported by the Chinese Scholarship Council (CSC), and by the French research agency ANR through the VideoSense project under the grant 2009 CORD 026 02 and the Visen project within the ERA-NET CHIST-ERA framework under the grant ANR-12-CHRI-0002-04.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

which outperforms state of the art graphs using 4 different objective criteria, namely PRI, VoI, GCE and BDE.

Index Terms

Image Segmentation, graph construction, sparse representation, normalized cut, superpixels.

I. Introduction

Image segmentation aims to partition an image into meaningful regions and is a fundamental step for many computer vision tasks, *e.g.*, object recognition [1], scene interpretation [2], or content-based image retrieval [3]. It proves to be extremely challenging due to the huge diversity and ambiguity of visual grouping patterns in natural scene images, in particular in presence of faint object boundaries and cluttered background (see Fig.1(a)). When no restrictive prior is imposed, segmenting an image is an inherently ill-posed task which requires incorporating prior knowledge into the algorithm and keeps attracting many researcher's attention.

A. Background

In this work, we are interested in graph-oriented methods which turn the problem of segmenting an image into a problem of partitioning a graph. They prove to be very versatile while providing the ability to encode perceptual grouping laws which play a major role in human visual perception [4], [5], [6]. However, it is also well known that the quality of the final segmentation result strongly depends on the way the initial graph is built from the input image. Building a graph requires defining its nodes and the relationships between them, *i.e.,* the edges and their weights. Numerous works have been devoted in recent years to reliable graph's building [7], [8], [9], [10], [11], [12], [13], [14], [15]. There are essentially two categories of approaches depending on whether the graph construction is unsupervised or semi-supervised/supervised. Unsupervised methods divide themselves between *static* and *adaptive* techniques. In *static graphs* the connected nodes are selected using hard decision and the pairwise similarities are computed without consideration of other data points. In *adaptive graphs*, the similarities depend on all data points, and the edges and their weights are defined simultaneously. The semi/supervised graphs either optimize different features and their combinations to measure pairwise affinities on the large image data set with manually segmented regions [7] or learn a new pairwise distance metrics using semi-supervised learning techniques [16] to obtain the relevance scores, learned from the test images as graph affinities [8] or using diffusion-based metric learning [10]. It is worth noting that, thanks to its versatility, there

are also other graph construction methods which have been applied to different tasks in computer vision. For instance, in [11], the author proposed a method to define similarity, which is context-sensitive due to that the new similarity is learned iteratively so that the neighbors of a given data influence its final similarity to other data points. Later, in [12], they proposed a framework called as co-transduction to fuse two or multiple similarities. Instead of diffusion on the original graph, in [13], [14], the authors proposed learning the similarities on tensor product graph of original graph with itself, and they also presented an efficient iterative method to approximate this diffusion process. Later, authors in [15] proposed to fuse different similarity graph using the diffusion technique on tensor product graph in [13].

Based on the previous studies on graph construction for image segmentation, one can deduce the following requirements for graph reliability [17]:

1) **High discriminating power**. Pixels from the same object are expected to be assigned large affinities or similarities, in contrast with pixels from different objects. While an object usually exhibits heterogeneous properties (color, texture, gradient, etc.), one single feature descriptor cannot comprehensively describe them all. There is a huge body of literature on the subject of features selection or their appropriate fusion. In supervised partitioning scenarios, [18] presented excellent survey. It would be pertinent to mention that in [19], multi-view learning introduced one function to model a particular view and jointly optimizes all the functions to exploit the redundant views of the same input data and improve the learning performance. However in unsupervised setting, selecting features, is a much harder problem, due to the absence of class labels. Literature on this problem is rarely studied and limited, for exceptions see e.g. [20], [21];
2) **Sparsity**. Many works on graph partitioning state that meaningful results derive from a sparse graph [5], [22] because it conveys with low memory cost yet valuable semantic information of the original high-dimensional data [17]. Furthermore, due to storage limitations and the need for efficient solving of eigenvector problems [22], it is inevitable to build a sparse graph;
3) **Adaptivity**. It happens frequently that the data (pixels/regions) is not evenly distributed. Conventional static graphs use fixed size and shape for the neighborhoods that are used for computing affinity weights, which will potentially generate erroneous associations between pixels. In contrast, it is reasonable to ask that different data points should have their corresponding adaptive neighborhood structure.

However, human vision of a scene is perceptual, making use of perceptual laws [4], *e.g.*, similarity, proximity, continuity, *etc.* Given the unsupervised context while dealing with a huge diversity of natural

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

scene images, *e.g.*, indoor and outdoor scenes, landscapes, cityscapes, plants, animals, people, objects, *etc.*, the challenge here is thus to construct a reliable graph fulfilling the aforementioned requirements while encoding some prior knowledge, *e.g.*, perceptual laws.

An attractive advantage of *static graphs* (*e.g.*, *adjacent*-graph) is that they enforce proximity, *i.e.* geometrical adjacency. However, they fail to capture long range grouping cues. In contrast, *adaptive graphs* (*e.g.*, our previously proposed ℓ_0-graph [23]) can capture long range grouping cues in a sparse way, but they tend not to emphasis sufficiently the proximity, and thereby generate isolated regions in the segmented result. However, both proximity and continuity, also known as geometrical adjacency and long range cue are critical to obtain reliable segmentation results.

B. The proposed approach and Contributions

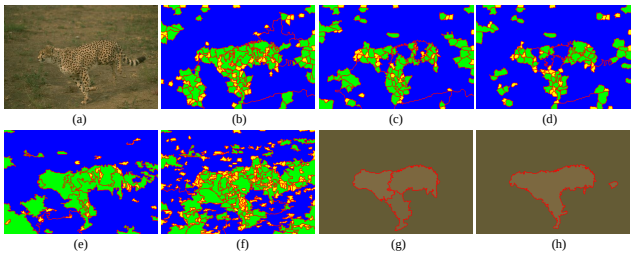


Fig. 1. Illustration of the gravitation law in perceptual grouping: (a) leopard running on the ground, (b)-(d) are superpixels of 3 different scales by Mean Shift (MS) by oversegmenting (a) using 3 parameter settings, and (e)-(f) superpixels of 2 other different scales by Felzenszwalb-Huttenlocher (FH). Superpixels are divided into small, medium and large sized sets colored in yellow, green and blue, respectively. (g) and (h) are segmented result by SAS and the proposed *GL*-graph with a number of segments $k = 4$ and $k = 2$ respectively.

In this work, we propose to construct a sparse and discriminative graph over superpixels to implement not only some obvious perceptual grouping laws, *e.g.*, proximity, similarity, but also enable some others, less straightforward, *e.g.*, continuity, to enter into action for the purpose of perceptual image segmentation. Based on empirical observations, we first postulate a gravitation law over superpixels for their perceptual grouping. Specifically, as can be seen in Fig.1 (b)-(f), in dividing broadly superpixels into small, medium and large sized sets, colored in yellow, green and blue in Fig.1, respectively, the postulated gravitation law states that:

1) Small sized superpixels are tiny regions which tend to be perceptually attracted by nearby medium or large sized superpixels while large sized superpixels are wide regions, *e.g.*, ground regions in blue in Fig.1(b), which could span as large as more than half of an image. They are structuring

visual patterns that already convey long range information and tend to strongly attract their direct medium and small sized superpixels in perceptual grouping;

2) Medium sized superpixels express long range visual grouping patterns, *e.g.*, skin spots of the leopard in green in Fig.1.(b), which need to be captured to further enable propagation of local grouping cues across long range connections;

As a result, we propose to construct an adjacent-graph over small and large sized superpixels to encode the proximity, and adopt our previously proposed sparse ℓ_0 graph over medium sized ones to capture continuity and promote sparsity. As the proposed graph can capture both local and global relationships among data points, we call it a *Global/Local Graph*, or *GL*-graph in short. Furthermore, to enable propagation of grouping cues among superpixels of different scales, we also introduce a bipartite graph which expresses relationships between pixels and superpixels.

Another important perceptual grouping law is similarity of data points within an object which can be characterized by three major perceptual visual features, namely color, texture and shape. According to a few works in psychophysics of human vision [24][25], these features jointly contribute to perceptual grouping but with different emphases. In this work, we implement this paradigm and evaluate the aforementioned three visual features in our GL graphs, through mlab and color histogram, Color LBP and SIFT-based codebooks for color, texture and shape, respectively, both individually and their weighted combinations for their effectiveness in unsupervised image segmentation.

The contributions of the proposed approach are threefold:

1) A sparse global/local graph over superpixels of different scales is proposed to capture both short and long range grouping cues of an image, thereby enabling perceptual grouping laws, *e.g.*, proximity, similarity, continuity, to enter into action through a suitable graph cut algorithm. This is achieved in over-segmenting the input image into superpixels at different scales, postulating and implementing a gravitation law which makes use of small and large sized superpixels to encode local smoothness, *e.g.*, proximity, while medium sized superpixels to capture sparse long range grouping cues, *e.g.*, continuity, through ℓ_0 sparsity. A bipartite graph is also introduced to further enable propagation of grouping cues across superpixels of different scales.
2) Using *GL*-graph, we also evaluate three major visual grouping features, namely color, texture and shape, for their discriminating power in perceptual image segmentation, as well as simple weighted fusion schemes which implement findings from psychophysics which suggest combining color, texture and shape cues with different emphases for perceptual grouping. These evaluations are not

only conducted on the proposed *GL*-graph but also on a number of state of the art graph construction methods to shed light on how constructing discriminative graphs with suitable features and their combinations.

3) Extensive experiments are carried out on the Berkeley Segmentation Database (BSD) using 4 different criteria, namely PRI, VoI, GCE and BDE. The experimental results show the effectiveness of the proposed approach, which generate perceptually meaningful partitions and display very competitive objective results in comparison with a number of state of the art algorithms.

Paper Organization. The rest of this paper is organized as follows: in Section II we discuss the basic principles underlying standard graph construction methods in the literature. Section III presents the proposed *GL*-graph in detail and introduces the graph cut method for general image segmentation tasks. In Section IV we carry out extensive experiments on different graphs with different features, and compare the proposed graph with existing graphs as well as other state-of-the-art segmentation methods both visually and quantitatively. Finally, the conclusion is drawn in Section V.

II. Relative works

Most graph-cut approaches for image segmentation build a static graph which only models the local neighborhood relationships between data points [26]. Classical methods for selecting connected vertices are:

1) The ε-neighborhood graph (ε-graph), which connects all points whose pairwise distances are smaller than ε. The pairwise similarity is chosen almost constant, making the constructed graph unweighted. However, selecting a single ε for all nodes in the graph might not properly capture the neighborhood structure of the data points;
2) The K-nearest neighbor graph (*KNN*-graph) connects every point to all points that are among its K-nearest neighbors, and the similarity is computed using pairwise distances. The fact that the *KNN*-graph's neighborhood size is fixed may lead to include noisy edges in the neighborhood of a data point as pointed out in [27];
3) The fully connected graph connects all points with positive similarity with each other. This construction is only useful if the similarity function itself models local neighborhood relationship. In most cases, Gaussian similarity function $w(x_i,x_j) = e^{(-(x_i-x_j)^2)/(2\sigma^2))}$ is chosen to compute the similarity, where the parameter σ controls the width of the neighborhood. Obviously, a "good" σ would help in pulling intra-class objects together and in pushing interclass objects far away

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

from each other. Therefore the parameter σ is critical in generating a reliable affinity matrix by controlling the neighborhood size and scaling pairwise similarities.

The conventional graphs are insufficient to produce a desirable segmentation for they only consider local relationships between data points [8], [9], [22]. Therefore, a recent significant move consists of propagating local grouping cues across long-range spatial connections in order to improve the segmentation result. For example, the multiscale method combines fine- and coarse-level details [22], while other methods pre-segment the image into small regions to replace the pixels as graph nodes [9], [28].

Although these methods have improved the segmentation results, they remain insufficient to encode valuable long-range information into the graph. Alternatively, sparse representation theory has gained a great deal of interest in various research communities, *e.g.* face recognition [17], subspace clustering [29], [30]. However, little literature exists on exploiting its nice property, *e.g.* sparsity and long-range grouping cues, to graph construction for the task of image segmentation, with exception of work [31], [23], [32]. In context of our image segmentation problem, the basic principle is to approximate every data point, *i.e.*, superpixel in a given feature space, as a linear combination of other superpixels of the same image, which are considered as neighbors, and their pairwise similarities or affinities are computed from the corresponding representation error. Formally, such an approximation can be written as:

$$y_i = \mathbf{Y}c_i,\ c_{ii} = 0 \tag{1}$$

where $c_i \in \mathbb{R}^n$ is the sparse representation of the data point $y_i \in \mathbb{R}^m$ over the dictionary $\mathbf{Y}$ which is a matrix representation of data points. The constraint $c_{ii} = 0$ prevents the self-representation of y_i.

There is another way to derive a linear representation of a given data point by solving the following low-rank-minimization problem [30], denoted as *LRR*-graph:

$$\min\ \text{rank}(\mathbf{C})\quad s.t.\quad \mathbf{Y} = \mathbf{Y}C, \tag{2}$$

where $\mathbf{C} \in \mathbb{R}^{n\times n}$ is the coefficient matrix of a representation of $\mathbf{Y} \in \mathbb{R}^{m\times n}$ over itself. This optimization problem is not convex and difficult to solve. Fortunately, the following convex problem provides a good surrogate:

$$\min\ ||\mathbf{C}||_*\quad s.t.\quad \mathbf{Y} = \mathbf{Y}C \tag{3}$$

where $||C||_* = \text{trace}(\sqrt{C*C})$ is the nuclear norm of matrix. In real applications, for observations are noisy, a more reasonable formulation is used:

$$\min\ ||\mathbf{C}||_* + \lambda||E||_{2,1}\quad s.t.\quad \mathbf{Y} = \mathbf{Y}C + E \tag{4}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

where the $\ell_{2,1}$-norm is defined as $||E||_{2,1} = \sum_{j=1}^{n} \sqrt{\sum_{i=1}^{n} ([E]_{ij})^2}$, and the parameter $\lambda > 0$ tunes the relative influence of both terms. It can be set according to the properties of both norms, or empirically tuned.

III. Proposed Global Local affinity graph based on Superpixel and Sparse Representation

The flowchart of the proposed graph-cut approach for image segmentation is shown in Fig.2. We start by over-segmenting at different scales the input image and refer to the resultant segments as "superpixels". Various visual features, *e.g.*, color, texture and shape, are then extracted from the superpixels. A GL-graph is then constructed to capture both short and long range grouping cues through visual features of superpixels of different scales.

Finally, unlike usual unsupervised approaches like normalized cut (Ncut) [5], the image segmentation problem is solved by computing the partition of a bipartite graph obtained with the previous GL-graph while encoding the associations between pixels and superpixels.

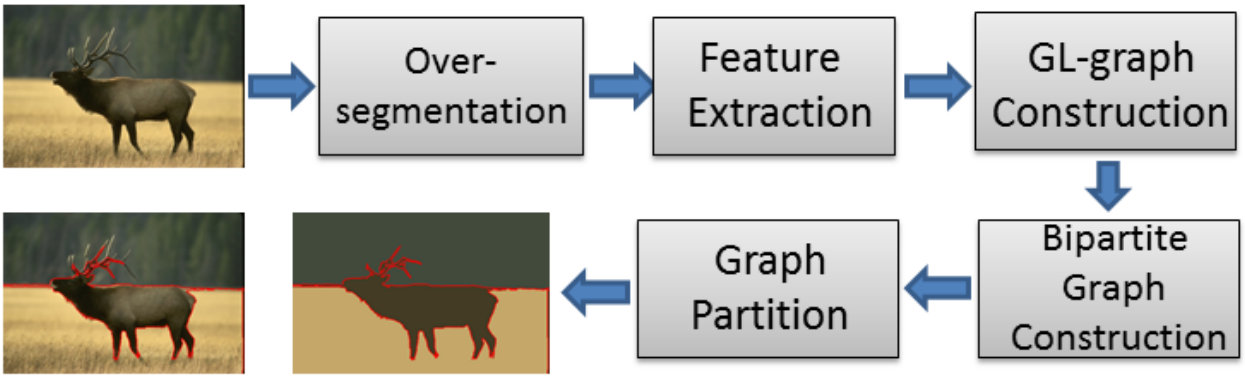


Fig. 2. The framework of our proposed graph-cut approach for image segmentation

A. Multi-scale Superpixels Generation and Representation

As pointed in [9], superpixels generated by different methods with varying parameters can capture various and multiscale visual patterns of a natural scene image. By superpixel, we mean here a connected maximal region in a segmented image. As shown in Fig. 3, an input image is oversegmented into superpixels of different scales, *e.g.*, 5 scales in the figure, using one or several state of the art segmentation methods, *e.g.* the Mean Shift algorithm (MS) [33] and the Felzenszwalb-Huttenlocher (FH) graph-based method [6] in this work. Fig. 3 shows 5 oversegmentations at 5 different scales using the same parameters as the method referred to as Segmentation by Aggregating Superpixels (SAS) [9] in the sequel. Then, to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

obtain a discriminative affinity graph, we compute for each superpixel various visual features. While any kind of region-based feature could be used, we evaluate the discriminating power of three perceptual visual features, namely color, texture and shape, which play a major role in human vision-based segmentation [24][25]. Specifically, in this work, color feature is characterized using mean value in the L*a*b space (*mLab*) and Color Histogram (*CH*) in the RGB space, texture through Local Binary Pattern (*LBP*) while shape cues using SIFT based bag-of-visual-words (*BoW*) [34][35] as shown in Fig. 3. Unlike RGB, Lab color space is designed to approximate human vision and its L component closely matches human perception of lightness. Local Binary Patterns (*LBP*) are reputed to encode micro-texture and robust to monotone light changes.

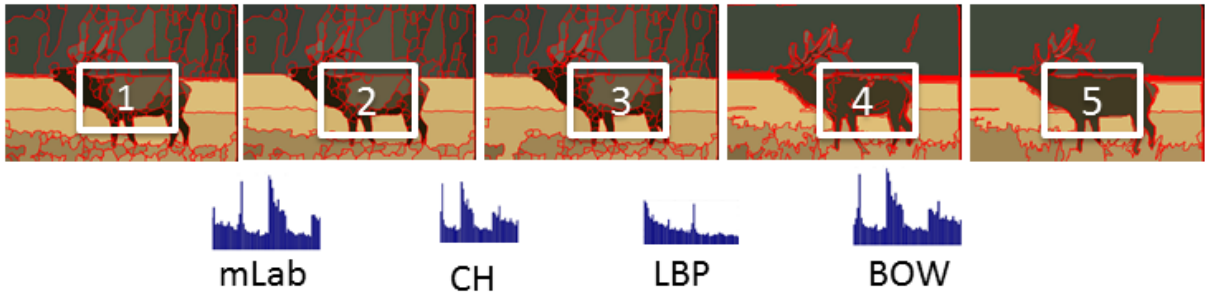


Fig. 3. Multi-scale superpixels generation and representation with three perceptual visual features: each superpixel is described by its color feature (mean value in L*a*b, *mlab*, color histogram in RGB (*CH*), texture feature (Local Binary Pattern, LBP), and shape appearance cue (Bag-of-Words) with SIFT.

B. Global/local affinity graph construction

As explained in subsection I-B, we postulate the gravitation law from empirical observations on superpixels and broadly divide them into *small*, *medium* and *large* sized sets for their perceptual grouping. *Adjacency*-graph is used for both small and large sized superpixels with respect to their spatial neighbors to capture local smoothness while ℓ_0-graph is applied to medium sized pixels. The final result is a sparse Global/Local graph, namely *GL*-graph, as illustrated in Fig.(5), which implement proximity, long-range continuity and similarity in the same framework.

Specifically, given an input image I, and a collection of superpixels $S_l = \{s_1, s_2, ..., s_N\}$ at a given scale l, a GL-graph is built in a given feature space, *e.g.*, mLab, using the superpixels as graph nodes. Superpixels are divided adaptively into three disjoints sets: *small*, *medium* and *large* sized ones. The *small* sized superpixels can be directly defined using the minimum area parameter involved in the oversegmentation algorithms used for the computation of the superpixels. To decide the *large* sized superpixels, we first

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

sort all the superpixels areas in an ascending order, then we compute the cumulative sum $\mathscr{C}(s_l)$ of the reordered areas. Fig.4 illustrates the graph of this cumulative sum for superpixels of 5 different scales. Calculating the second derivative of each curve, we identify its maximal value, and the corresponding area is chosen as threshold value (see the corresponding blue mark on the cumulative graphs in Fig.4). This simple procedure seems rather robust in our experiments. Indeed, they depict almost the same performance when the threshold for deciding the large sized superpixels varies in a range close to the inflection identified by the aforementioned procedure through second derivative.

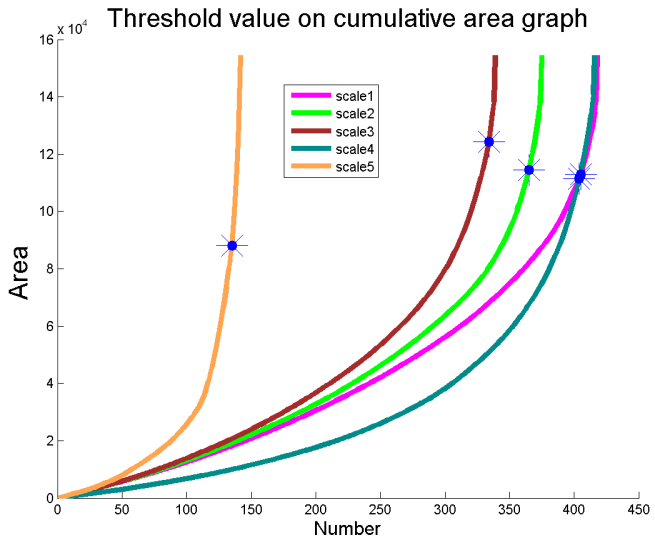


Fig. 4. Illustration of the adaptive threshold selection of large regions.

Building a ℓ_0-graph for medium-sized superpixels. Our previously proposed ℓ_0-graph in [23] is applied to *medium*-sized superpixels in order to capture long range grouping cues, using Eq.(1) to approximate every *medium*-sized superpixel from other *medium*-sized ones in a given feature space, *e.g.*, *mLab*.

However, Eq.(1) is generally underdetermined and can have an infinite number of solutions whereas we seek to build a sparse image graph in line with the requirements expressed in subsection I-A. It turns out that the sparsest solution of Eq.(1) measured in the sense of ℓ_0-norm is unique and conveys the most meaningful information of a signal [36].

Formally, this sparsest solution can be written as the following minimization problem:

$$\min \ ||c_i||_0 \quad s.t. \quad y_i = \mathbf{Y}c_i, \ c_{ii} = 0 \tag{5}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

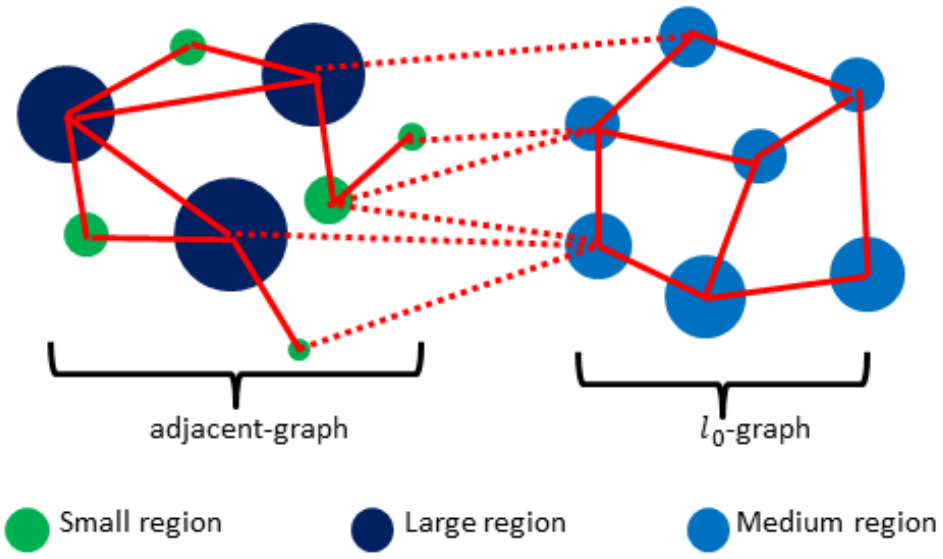


Fig. 5. Illustration of the *GL*-graph's structure: for each over-segmentations, all the superpixels are divided into three sets: *small* (the green dots), *medium* (the blue dots) and *large* (the ink blue dots) according to their area. Over *small* and *large* sets, all data points will connect to their adjacent neighbors, while over *medium* set, each data point will search its neighbors all over the set. Note that bold red lines represent undirected edges connecting data points within sets, while the dashed red lines describe the edges connecting data points between two different sets.

where $\|\cdot\|_0$ denotes the ℓ_0 norm, which counts the number of nonzero values in a vector.

However, the problem of finding the sparsest solution of linear equations is NP-hard. Nevertheless, there are many sparse approximation methods, the most two common ones being the ℓ_1-norm approximation and the orthogonal matching pursuit (OMP).

The ℓ_1-norm can be used to approximate the ℓ_0-norm:

$$\min \ ||c_i||_1 \quad s.t. \quad y_i = \mathbf{Y}c_i, \ c_{ii} = 0 \tag{6}$$

under the condition if the solution sought is sparse enough [17], [37], [38]. However, within our context of image segmentation using superpixels, such a condition is not necessarily satisfied, given the fact that the number of superpixels given by an oversegmentation is quite limited, *e.g.*, a few hundreds, and even less for *medium* sized superpixels. Furthermore, because of the huge diversity of natural scene images, the dictionary,*i.e.*,the data point representation matrix $\mathbf{Y} = [y_i, ..., y_N] \in \mathbb{R}^{d \times N}$ in Eq.(1), could be very unbalanced, for instance with far much more sky superpixels than others, thus missing to be overcomplete for some visual patterns. As a result, we keep to solve Eq.(5) using the ℓ_0-norm but make

use of orthogonal matching pursuit (OMP) to seek an approximation of the sparsest solution. Experimental results discussed later on in section IV are in line with our analysis and provide further support in favor of our choice of the ℓ_0-sparsity.

OMP is a simple and fast greedy method for approximately solving the ℓ_0-norm sparse formulation through the following optimization problem:

$$\tilde{c}_i = \text{argmin}_{c_i} \left\{ \|y_i - \mathbf{Y}c_i\|_2^2,\ \|c_i\|_0 \leq L,\ c_{ii} = 0 \right\} \tag{7}$$

where the parameter L controls the sparsity of the representation. The OMP takes linear time $O(NL)$ with the N representing total number of entries in the dictionary $\mathbf{Y}$, and the L be the maximal number of coefficients for each input data atom y_i.

Once achieved a sparse representation for each data point whose nonzero elements are expected to indicate superpixels from a same object, these superpixels will be considered as graph neighbors of the given data point. The next step of the algorithm is to define the similarity matrix W using the sparse reconstruction error:

$$r_{ij} = \|y_i - c_{ij}y_j\|_2^2. \tag{8}$$

The similarity coefficient w_{ij} between superpixels s_i, s_j is defined as

$$w_{ij} = \begin{cases} 1 & \text{if } i = j \\ 1 - (r_{ij} + r_{ji})/2 & \text{if } i \neq j. \end{cases} \tag{9}$$

Building an adjacency-graph for small and large sized superpixels with respect to their neighbors. As for the superpixels in the *small*- and *large*-sized sets, every superpixel is connected to all its adjacent superpixels, denoted as *adjacency*-graph. Traditionally, the pairwise similarities are computed with the Gaussian kernel function which is influenced greatly by the choice of the standard deviation σ [9], [5]. In our combining scheme, it is hard to decide adaptively the value of σ in order to maintain the same order of magnitude with ℓ_0-graph. Therefore, we adopt the same principle as for ℓ_0-graph to compute the similarities: given a superpixel s_i associated with its corresponding feature vector x_i and the matrix-representation $\mathscr{D}$ of all its adjacent neighbors, we try to represent x_i as a linear combination of elements in $\mathscr{D}$. In practice, we solve the following optimization problem:

$$\tilde{c}_i = \text{argmin}_{c_i} ||x_i - \mathbf{D}c_i||_2 \tag{10}$$

Once a minimizer $\tilde{c}_i$ has been computed, the similarities between a superpixel and its graph neighbors are computed as in (8) and (9).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

C. Fusing GL-graphs of different visual features and different scales

In summary, for each scale of oversegmented superpixels $S_l = \{s_1, ..., s_N\}$, and its associated feature matrix $[x_1, ..., x_N]$, we construct a *GL*-graph $\mathscr{G}_l$. In this work, as explained in subsection III-A, we aim to evaluate the effectiveness of three major perceptual visual features, namely color, texture and shape, for their discriminating power, and therefore generate for each of them f_k a similarity matrix W^{f_k}. Furthermore, following [24], [25] which suggest combining color, texture and shape cues with different emphases, we implement a simple weighted sum as in Eq.(11) to fuse these similarities into a single affinity matrix.

$$w_{ij} = \sum_{k=1}^{m} (\beta^{f_k} w_{ij}^{f_k}) \tag{11}$$

where β^{f_k} is a weight assigned to feature f_k, which controls this feature's importance, and $w_{ij}^{f_k}$ denotes the similarity of superpixels s_i and s_j with feature f_k. For comparison, a baseline fusion scheme as defined in Eq.(12) is also used.

$$w_{ij} = \sqrt{\sum_{k=1}^{m} (w_{ij}^{f_k})^2} \tag{12}$$

To fuse all scales of superpixels, we plug each scale affinity matrix W_l corresponding to its *GL*-graph $\mathscr{G}_l$ into a block diagonal multiscale affinity matrix W_{ss} like [22] as follows:

$$W_{ss} = \begin{pmatrix} W_1 & & 0 \\ & \ddots & \\ 0 & & W_l \end{pmatrix} \tag{13}$$

Note that this multiscale affinity matrix of superpixels gathers all the informative intra-scale similarities for grouping. Furthermore, in packing them diagonally, we are ready also to enable propagation of long-range grouping cues across scales, which is achieved by constructing and diagonalizing a pixel-superpixel graph, or bipartite graph, as introduced in the next subsection.

D. Bipartite Graph Construction and Partition

To map the relationships between pixels and superpixels and enable propagation of grouping cues across superpixels of different scales, we build a bipartite graph which consists of two parts describing the pixel-superpixel and superpixel-superpixel relationships, respectively. Fig. 6 illustrates the structure of such a bipartite graph which encodes the information between pixels and superpixels in blue lines, and the information between superpixels in yellow ones. In particular, taking into account the demand of sparsity for a good-quality graph, pixels are only connected to the superpixels to which they belong.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

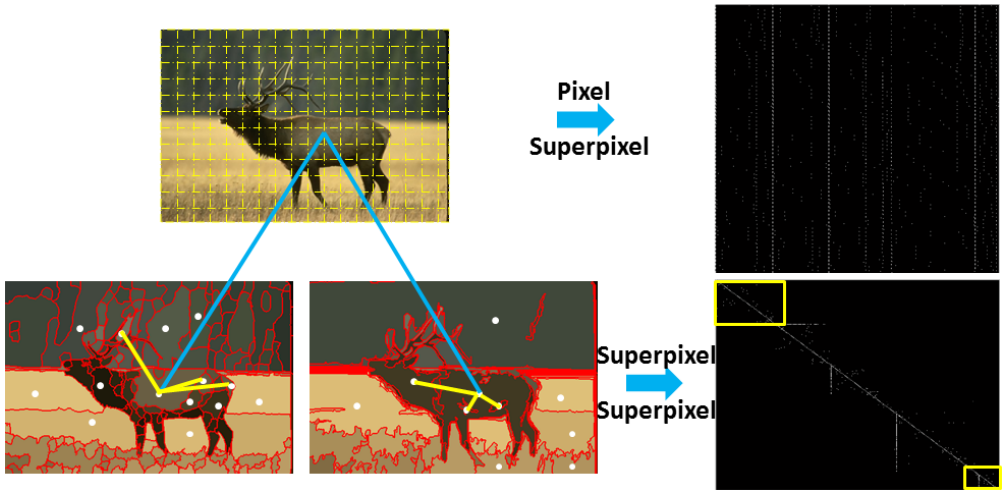


Fig. 6. Illustration of the construction of an unbalanced bipartite graph over multi-scale over-segmentations: a yellow dot denotes a pixel, and a white dot denotes a superpixel. The blue lines show that each pixel is only connected to its corresponding superpixel in each scale of over-segmentations which is represented as a pixel-superpixel affinity matrix (upper block matrix), while the yellow lines show undirected edges representing the relationships between two superpixels, represented by a superpixel-superpixel affinity matrix (lower block matrix).

More precisely, let $\mathscr{G}_{\mathscr{B}} = \{\mathscr{U}, \mathscr{V}, B\}$ denote the bipartite graph, where $\mathscr{U} = I \cup S$, $\mathscr{V} = S$, I is the set of pixels and S the set of superpixels. $B = \begin{bmatrix} W_{IS} \\ W_{SS} \end{bmatrix}$, with $W_{IS} = (b_{ij})_{|I| \times |V|}$, and $b_{ij} = \gamma$, if pixel i belongs to superpixel j (in our experiments, we set $\gamma = 10^{-3}$), $b_{ij} = 0$ otherwise. W_{SS} is the affinity graph between superpixels computed in section 3.2. Note that the resultant bipartite graph is highly sparse[1] because of its unbalanced nature. Furthermore, superpixels sharing a large number of pixels are likely to be grouped together, thanks to connections between pixels and their superpixels containing them, thus enabling propagation of grouping cues across scales.

Given an undirected graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ with a similarity matrix W and a number of segments k, various techniques can be applied to group the data points into k different clusters, as cuts [5], maximum-flow techniques [39] and spectral clustering algorithms [26], [40]. Among these methods, spectral clustering

[1]In the bipartite graph, a pixel is connected to only l superpixels for l over-segmentations of an image, we used $l = 5$ *or* 6 for our experiments . For instance, an image named 2092, it is oversegmented into 5 scales with 123, 121, 105, 209 and 53 superpixels, respectively, thus resulting in 611 superpixels in total. The pixel-superpixel graph's size is 154401×611. The total number of nonzero elements in this graph is 154401×5. The percentage of nonzero elements can be viewed as a measurement of sparsity, *i.e.*, $\frac{5}{611} = 0.0081$ Additionally, the unbalanced structure of the constructed bipartite graph $\mathscr{G}_{\mathscr{B}} = \{\mathscr{U}, \mathscr{V}, B\}$ makes the graph further sparser. The final bipartite graph has the size $|\mathscr{U}| = |\mathscr{V}| + |I| = (154401 + 611) > |I| = 481 \times 321 = 154401$.

algorithms have been proven successful in many applications, and in particular image segmentation [5]. They have been in recent years a major trend to achieve clusters from a sparse graph, mainly using representations as linear combinations of eigenvectors of the Laplacian matrix. Basically, spectral clustering consists of partitioning the graph using eigenspaces associated with the following generalized eigen problem [5]:

$$L\mathbf{f} = \lambda D\mathbf{f}, \tag{14}$$

where $L = D - W$ denotes the graph Laplacian, and $D = diag(W\mathbf{1})$ with $\mathbf{1}$ a vector with all components equal to 1. The Lanczos method [41] and the partial SVD [42] can be applied to solve the above eigen problem.

However finding eigenvalues of large matrices is in general computationally demanding, for example, given the bipartite graph $\mathscr{G}_B$, it takes $O(k(|\mathscr{U}|+|\mathscr{V}|)^{3/2})$ [9] for the Lanczos method and the partial SVD. Note that the bipartite graph in our case is unbalanced, i.e. $|\mathscr{U}| = |\mathscr{V}| + |I|$, and $|I| >> |\mathscr{V}|$ in general, which gives $|\mathscr{U}| >> |\mathscr{V}|$. We use the Transfer Cuts method [9] which has been proposed to solve efficiently the unbalanced bipartite graph partitioning problem. Interestingly, Transfer Cuts solve a problem which has similar form as (14), but holds on a much smaller graph over superpixels only

$$L_{\mathscr{V}}\mathbf{f} = \lambda D_V\mathbf{f}, \tag{15}$$

where $L_{\mathscr{V}} = D_{\mathscr{V}} - W_{\mathscr{V}}$, $D_{\mathscr{V}} = diag(B^\top\mathbf{1})$, and $W_{\mathscr{V}} = B^\top D_{\mathscr{U}}^{-1}B$, $D_{\mathscr{U}} = diag(B\mathbf{1})$. Note that solving (14) takes linear time $O(k|\mathscr{V}|^{2/3})$ with a small constant.

IV. Experiments and Analysis

The proposed *GL*-graph is extensively evaluated on the Berkeley Segmentation Database in comparison with the state of the art.

A. Image Database and Evaluation Metrics

All experiments are carried out on the Berkeley Segmentation Database (BSD) [43], which includes 300 images and the corresponding ground truth data (each image has at least 4 human annotations). It is divided into a training set which contains 200 images and a test set including 100 images. Each image's size is 481×321. Four standard measurements are used for quantitative evaluation: the Probabilistic Rand Index (PRI) [44], the Variation of Information (VoI) [45], the Global Consistency Error (GCE) [46], and the Boundary Displacement Error (BDE) [47].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

The Probabilistic Rand Index (PRI) measures the fraction of pixel pairs whose labels are consistent between the segmentation result and the ground truth. In practice, PRI can be computed in a simple form. Let S_{ground} and S_{test} be two clusterings of the same image with different number of clusters, and let n_{ij} be the number of points in the ith cluster of S_{ground} and the jth cluster of S_{test}. N is the total number of pixels of the image. The similarity between the two clusterings is:

$$PRI(S_{ground}, S_{test}) = \left\{ \binom{N}{2} - 1/2\{\sum_i (\sum_j n_{ij})^2 + \sum_j (\sum_i n_{ij})^2 - \sum\sum n_{ij}^2\} \right\} \Big/ \binom{N}{2} \tag{16}$$

The value of PRI ranges from 0 (when there is no intersection at all between S_{ground} and S_{test}) to 1 when the two clusterings are actually the same.

The Volume of Information (VoI) computes the amount of information loss/gain between the compared images, and can therefore measure the extent to which one image can explain the other, with lower values representing greater similarity. Formally, it is defined as:

$$VoI(S_{ground}, S_{test}) = H(S_{ground}) + H(S_{test}) - 2I(S_{ground}, S_{test}) \tag{17}$$

where H and I represent respectively the entropies of and the mutual information between the two clusterings, see [45] for more details.

The Global Consistency Error (GCE) computes the degree to which two segmentations are mutually consistent. Let $R(S_{ground}, p_i) \Delta R(S_{test}, p_i)$ denote the symmetric difference between $R(S_{ground}, p_i)$ (the subregion of S_{ground} containing the pixel p_i) and $R(S_{test}, p_i)$ (the subregion of S_{test} containing the pixel p_i). Let $|\cdot|$ denote set cardinality. The non symmetric local consistency error is defined as:

$$E(S_{ground}, S_{test}, p_i) = \frac{|R(S_{ground}, p_i) \Delta R(S_{test}, p_i)|}{|R(S_{ground}, p_i)|} \tag{18}$$

and the global consistency error is obtained by symmetrization and averaging:

$$GCE(S_{ground}, S_{test}) = \frac{1}{N} \min\{\sum_i E(S_{ground}, S_{test}, p_i), \sum_i E(S_{test}, S_{ground}, p_i)\} \tag{19}$$

GCE is valued in $[0, 1]$, where the null value indicates of course that both segmentations are equivalent.

The Boundary Displacement Error (BDE) measures the average displacement error of boundary pixels between two segmentation results. More precisely, it defines the error of one boundary pixel as the

distance between the pixel and its closest boundary pixel in the other image. Denoting

$$d(p_i, B_2) = \min_{p \in B_2} \|p_i - p\| \tag{20}$$

the distance of a boundary point $p_i \in B_1$ to the boundary set B_2, and N_1, N_2 the total number of points in the boundary sets B_1 and B_2, BDE is defined as:

$$BDE(B_1, B_2) = \frac{\sum_i^{N_1} d(p_i, B_2)/N_1 + \sum_i^{N_2} d(p_i, B_1)/N_2}{2} \tag{21}$$

A value of BDE close to zero is a good indication that both segmentations are similar.

B. Experimental Setup

Using the framework depicted in Fig.2, the proposed GL-graph is first evaluated through a single visual feature in comparison with several state of the art graphs. It is then further evaluated when fusing visual features as proposed by psychophysicists and several global graphs to capture different grouping cues. This means that only the GL-graph construction is evaluated and compared while keeping the same all the other steps, *e.g.*, over-sgementation, feature extraction, bipartite graph construction and graph partition using spectral clustering. Please refer to section III for further details.

The state of the art graphs studied are the *adjacent*-graph as in SAS [2][9] and four popular global graphs, namely *KNN*-graph [3] [48], ℓ_1-graph[4] [29], *LRR*-graph (Low Rank Representation)[5] [30], and ℓ_0-graph [23]. For each method, the parameters are tuned to achieve the best performance:

1) *adjacent*-graph: the standard deviation of the Gaussian kernel function is defined as $\sigma = 20$;
2) *KNN*-graph: we adopt Euclidean distance as the similarity metric, and use Gaussian kernel function to compute the weights of edges, with $\sigma = 20$ as [9]. Various numbers of neighbors are tested;
3) ℓ_1-graph: we construct the graph following the method in [29]. Since the affinity matrix is asymmetric, we replace it with $\tilde{W} = (|W| + |W|^T)/2$;
4) ℓ_0-graph: we derive the graph and symmetrize it following [23]. Parameters are chosen as in [23];
5) *LRR*-graph: we construct the *LRR*-graph and symmetrize it as for the ℓ_1-graph following [30]. We set the balance parameter $\lambda = 0.18$;

[2]http://www.ee.columbia.edu/ln/dvmm/SuperPixelSeg/

[3]http://cns.bu.edu/ lgrady/software.html

[4]http://www.cis.jhu.edu/ ehsan/

[5]https://sites.google.com/site/guangcanliu/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

6) *GL*-graph: the threshold value for defining small regions is empirically set to 300 pixels and the threshold for large regions is decided adaptively as explained in subsection III-B. Performances with various L are presented.

As explained in subsection III-A, following the findings of psychophysicists, we evaluate three major perceptual visual features, namely color using the mean value of a superpixel in L*a*b denoted as $mLab \in \mathbb{R}^3$ or RGB color histogram denoted as $CH \in \mathbb{R}^{256}$, texture through Uniform Color Local Binary Pattern[6] [49] denoted as $CLBP^{u2} \in \mathbb{R}^{177}$, and shape cues using the Bag-of-Visual-Words (BoW). In the experiments, we compute the scale invariant feature transform (SIFT)[7][34] at each pixel and then perform the vector quantization by fast K-means[8] to construct the visual vocabulary. The number of clustering centers is 100, 150, 200 and 300, denoted as $BoW100$, $BoW150$, $BoW200$, $BoW300$, respectively.

C. Experimental results using single visual feature

This experiment aims to compare the quality of the proposed *GL*-graph with 5 other state of art graph constructions and highlights the discriminating power of each visual feature. Table I tabulates the performance of the 6 tested graph construction methods over each visual feature. One can observe from Table I that:

1) Regardless of visual features, the performances of global graphs are essentially stable when the number of clusters k increases. This property is mainly due to the fact that global graphs choose each node's neighbors by searching globally, which enables the constructed graph to capture long-range grouping cue. This is in contrast with the local *ad jacent* graph which is more sensitive than global graphs to the number of segments k and to the used visual feature. It is worth mentioning that such property of global graph makes it promising for practical applications in object recognition, image annotation, *etc.*
2) The family of sparsity-based graphs (*e.g.*, ℓ_0-graph) has better performance than rank minimization graph (*LRR*-graph) or ℓ_1-graph. The reason is that in a ℓ_0-graph, each node has very few neighbors, which makes the graph much sparser compared with ℓ_1-graph and *LRR*-graph, see Table II where scores with various values of the parameter L are reported;

[6]http://www.cse.oulu.fi/CMV/Downloads/LBPMatlab

[7]http://www.vlfeat.org/

[8]https://gforge.inria.fr/frs/?group_id=2151

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

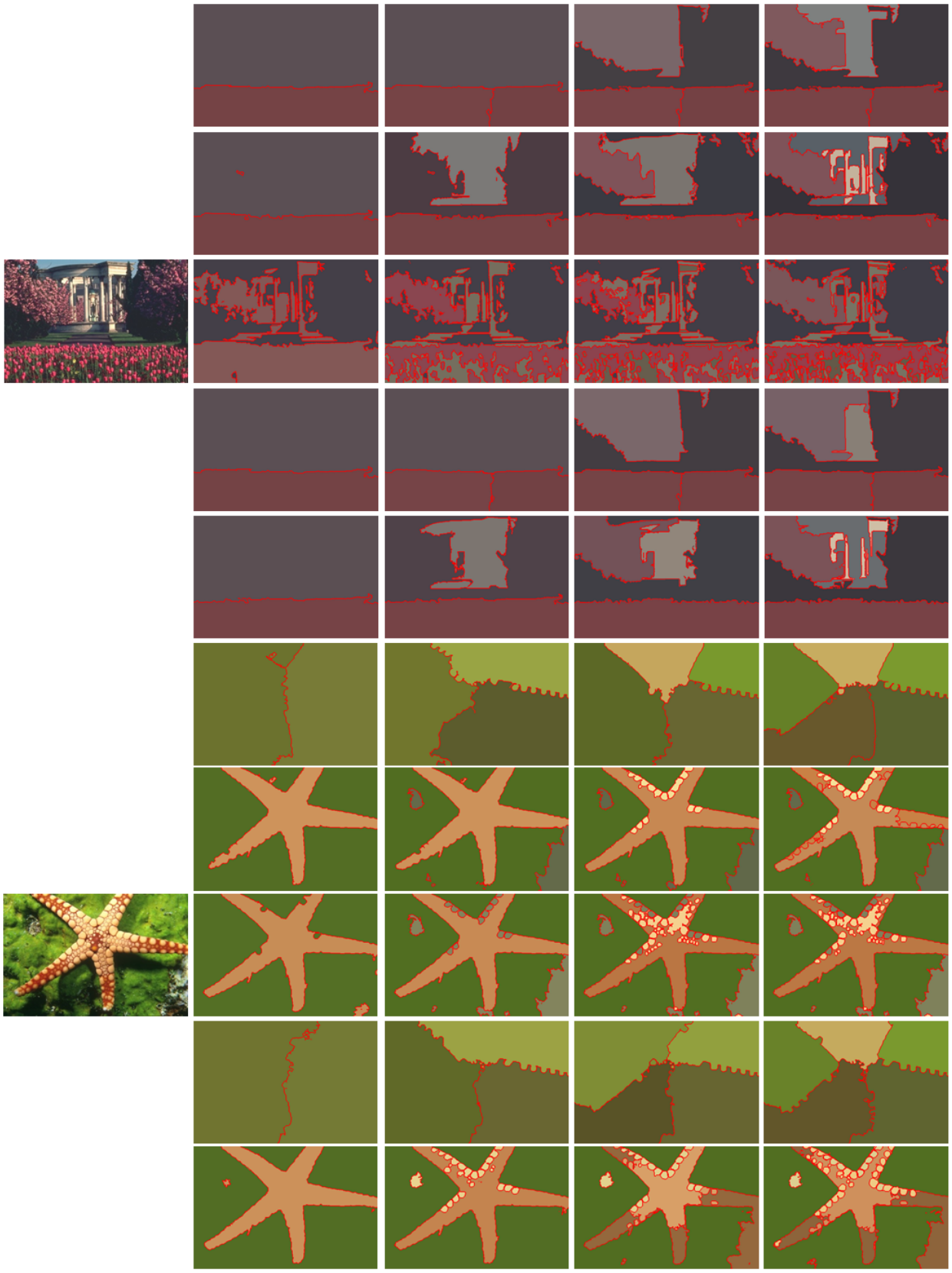


Fig. 7. Visual comparison of the results obtained with the *GL*-graph and with other graphs. Each line from top to bottom corresponds to the segmentation result obtained with the following graphs: *adjacency*, *kNN*, ℓ_1, *LRR*, and the proposed *GL*-graph. From left to right, the results for various choices of $k = 2, 3, 4, 5$. Note that the result is segmented using the most appropriate feature for each kind of graph according to Table I, e.g., we use the feature *mixing* for the *adjacency*-graph and the *KNN*-graph.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

TABLE I

PERFORMANCE VALIDATION FOR THE PROPOSED *GL*-GRAPH AND OTHER TYPES OF GRAPHS, USING VARIOUS FEATURES, ON THE BERKELEY SEGMENTATION DATABASE TEST SET. FOUR METRICS ARE USED: PRI, VOI, GCE AND BDE. FOR EACH GRAPH, THE BEST PERFORMANCE OVER FEATURES IS HIGHLIGHTED. NOTE THAT THE BEST PERFORMANCE RESULT IS COMPUTED BY MAXIMIZING PRI OF EACH IMAGE OVER ALL ITS EVALUATION RESULTS RANGING FROM 2 TO 40.

adjacent-graph	PRI↑	VoI↓	GCE↓	BDE↓	*KNN*-graph	PRI↑	VoI↓	GCE↓	BDE↓
mlab	**0.8264**	**1.7537**	**0.1935**	**12.7985**	**mlab**	**0.8290**	**2.0732**	**0.2316**	**12.1872**
CH	0.8133	1.9811	0.2204	13.9598	CH	0.8016	2.7882	0.3229	14.4206
$CLBP^{u2}$	0.8133	1.9811	0.2204	13.9598	$CLBP^{u2}$	0.8016	2.7882	0.3229	14.4206
BoW100	0.8106	1.9983	0.2301	14.7859	BoW100	0.7862	3.2387	0.3440	16.1826
BoW150	0.8112	2.0210	0.2302	14.9699	BoW150	0.7891	3.1858	0.3385	16.2013
BoW200	0.8104	2.0179	0.2286	14.7858	BoW200	0.7899	3.2239	0.3402	15.3621
BoW300	0.8113	1.9954	0.2285	14.9503	BoW300	0.7871	3.2063	0.3383	16.1223
ℓ_1-graph	PRI↑	VoI↓	GCE↓	BDE↓	*LRR*-graph	PRI↑	VoI↓	GCE↓	BDE↓
mlab	**0.8036**	**2.9053**	**0.3079**	**12.7745**	mlab	0.8155	1.8788	0.2071	13.7015
CH	0.7710	2.8919	0.3012	13.5910	**CH**	**0.8153**	**1.8794**	**0.2068**	**13.6949**
$CLBP^{u2}$	0.7710	2.8919	0.3012	13.5910	$CLBP^{u2}$	**0.8153**	**1.8794**	**0.2068**	**13.6949**
BoW100	0.6963	2.9473	0.3691	19.1577	BoW100	0.8148	1.8809	0.2072	13.6680
BoW150	0.7009	2.9678	0.3695	19.6824	BoW150	0.8146	1.8864	0.2084	13.7504
BoW200	0.7046	2.9428	0.3702	24.5510	BoW200	0.8140	1.8838	0.2083	13.7732
BoW300	0.7096	2.8871	0.3495	23.2067	BoW300	0.8147	1.8901	0.2078	13.6894
ℓ_0-graph	PRI↑	VoI↓	GCE↓	BDE↓	*GL*-graph	PRI↑	VoI↓	GCE↓	BDE↓
mlab	0.8141	2.2969	0.2470	12.2632	mlab	0.8230	2.0848	0.2260	11.7124
CH	**0.8185**	**2.2426**	**0.2622**	**12.8445**	**CH**	**0.8266**	**1.9585**	**0.2204**	**12.0042**
$CLBP^{u2}$	0.8152	1.8793	0.2068	13.6948	**$CLBP^{u2}$**	**0.8266**	**1.9584**	**0.2204**	**12.0043**
BoW100	0.7896	2.7465	0.3057	15.7107	BoW100	0.7970	2.4072	0.2545	15.2672
BoW150	0.7878	2.7624	0.2994	15.5692	BoW150	0.7959	2.4067	0.2542	14.7353
BoW200	0.7859	2.7847	0.3050	15.2595	BoW200	0.7991	2.3744	0.2521	15.1711
BoW300	0.7872	2.7346	0.2968	15.1443	BoW300	0.7997	2.3612	0.2502	15.4163

3) The proposed *GL*-graph combines local graph and ℓ_0-graph's nice properties. It achieves the best performances in Table I in comparison with the *adjacency*-graph and the ℓ_0-graph. As shown in Table II, it is however somewhat sensitive to the parameter L.

Regarding the discriminating power of visual features, it can be seen from Table I that:

1) both color and texture cues, *i.e.*, *mLab*, *CH*, *LBP*, show their better discriminating power over all the graphs in comparison with shape cues, *i.e.*, BoW features;
2) Color through *mlab* or *CH* outperforms texture, *i.e.*,*LBP*, for all kinds of graph except the *LRR*-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

graph and *GL*-graph on which *LBP* has equivalent performance with color. It proves thereby to be a faithful grouping cue;

Remark that these findings are in perfect accordance with those of psychophysicists on human vision-based segmentation which suggest that appearance grouping cues, *i.e.*, color and texture, outweight shape-based ones [25] while human vision makes joint use of color and texture for image segmentation but with asymmetric role in favor of color [24]. These findings will be fully explored in the fusion scheme as explained in subsections IV-D.

TABLE II

QUANTITATIVE SCORES FOR DIFFERENT VALUES OF THE PARAMETER *L* FOR THE *GL*-GRAPH OVER THE BERKELEY SEGMENTATION DATABASE TEST SET.

Sparsity (CH)	PRI↑	VoI↓	GCE↓	BDE↓
L=2	0.8213	2.1111	0.2453	13.2554
L=3	0.8185	2.2426	0.2622	12.8445
L=4	0.8195	2.2958	0.2645	12.4510
L=5	0.8177	2.3079	0.2622	12.2648
L=6	0.8185	2.3086	0.2631	12.8950
L=7	0.8190	2.2913	0.2624	12.8970
L=8	0.8185	2.3253	0.2667	12.1963

Obtaining visually meaningful results requires inevitably the careful tuning of the number of segments k. We show in Fig. 7 the different performances of the graphs for various values of k and the following observations can be made: 1) the *adjacent*-graph considers only the local structure of image, which leads to wrong segmentations (see the results segmented in first row for each image) when the objects cover a large part of the image. 2) the ℓ_1-graph tends to oversegment the image (see third rows for every example in Fig. 7), due to its high sensitivity to noise and outliers, which is a convenient skill for face recognition [37], but not for image segmentation; 3) unlike the graph based on sparse minimization, which finds the sparse representation of every point, the *LRR*-graph finds a global lowest rank representation, therefore further enforces the global structure over the data points. However, as pointed in [50], *LRR*-graph often produces a dense graph which fails to meet the demand of sparsity for a desirable graph.

D. Results on fusing different graphs and visual features

The experimental results shown in subsection IV-C in perfect accordance with the findings of psychophysicists on human vision-based segmentation [25] strengthen the simple weighted sum fusion scheme as defined in Eq.(11) in subsection III-C [24] which enables combining color, texture and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

shape cues with different emphases. Specifically, following both the findings of psychophysics and the experimental results shown in subsection IV-C, we empirically implement several fusion schemes, namely fusion schemes combining color and texture features as well as those combining color, texture and shape at the same time. When color and texture cues are jointly used, more weight is given to color-based affinities than those of texture-based one; When all the three visual features are used at the same time, shape receives less weight in comparison with color and texture. As a baseline, we also implement the baseline fusion scheme as defined in Eq.(12) which gives an equal weight to each kind of visual grouping cues. As can be seen from the Table III, very competitive results are achieved by the proposed *GL*-graph when fusing color, texture and shape with different emphases.

TABLE III

QUANTITATIVE PERFORMANCE OF THE PROPOSED METHOD (*GL*-GRAPH) WITH SIMPLE WEIGHTED SUM FUSION SCHEME.

Methods	PRI↑	VoI↓	GCE↓	BDE↓
$\sqrt{(LBP^2+mlab^2+SIFT^2)}$	0.8332	1.8890	0.1998	10.7904
$\sqrt{(LBP^2+CH^2+SIFT^2)}$	0.8355	1.8716	0.2048	10.9985
(0.4LBP+0.6mlab)	0.8355	1.8965	0.1765	10.9157
(0.4LBP+0.6CH)	0.8363	**1.6776**	**0.1727**	11.0456
(0.4LBP+0.4mlab+0.2SIFT)	0.8368	1.8347	**0.1706**	10.8552
(0.4LBP+0.4CH+0.2SIFT)	0.8381	1.8753	0.1741	**10.6787**
(0.3LBP+0.5mlab+0.2SIFT)	**0.8384**	1.8012	0.1934	**10.6633**
(0.3LBP+0.5CH+0.2SIFT)	**0.8383**	**1.7927**	0.1958	11.4088

We showed in the previous section that that different graphs capture different affinities between superpixels. Given a visual feature, *e.g.*, color *mLab*, fusion can also be carried out at the graph level in combining the proposed *GL*-graph with other ones, *e.g.*, ℓ_1, *KNN*, *LRR*, and in averaging their affinities. Specifically, the segmentation framework as defined in section III is kept the same, the fusion only takes place at the graph level of medium sized superpixels. Table IV reports the experimental results of such graph level fusion schemes. As can be seen from Table IV, when the input image is simply segmented into two clusters ($k = 2$), the baseline, *i.e.*, the proposed *GL*-graph, outperforms all three combinations. However, when the number of clusters is increased, *i.e.*, $k = 10, 30, 40$, all three combinations outperform the baseline *GL*-graph. These results suggest that, when the number of clusters is increased, new connections are brought in by other global graphs, *i.e.*, ℓ_1, *KNN*, *LRR*, definitively contribute to improve the segmentation result. Furthermore, both *KNN* and *LRR* graphs prove to bring more complementary information with respect to the *GL*-graph than the ℓ_1 graph.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

TABLE IV

QUANTITATIVE COMPARISON OF DIFFERENT COMBINATIONS OF TWO GLOBAL GRAPHS, ASSOCIATING WITH *ad jacent*-GRAPH OVER THE BERKELEY SEGMENTATION DATABASE.

Combinations (mlab)	PRI↑	VoI↓	GCE↓	BDE↓
$k = 2$				
baseline: *GL*-graph	0.6205	2.0445	0.1240	25.0000
$adjacency + \ell_0 + KNN$	0.5646	2.0936	0.0960	43.7168
$adjacency + \ell_0 + \ell_1$	0.5276	2.1655	0.1001	47.4737
$adjacency + \ell_0 + LRR$	0.5732	2.1191	0.1138	43.4317
$k = 10$				
baseline: *GL*-graph	0.7456	2.1730	0.2381	15.0301
$adjacency + \ell_0 + KNN$	0.7851	1.9744	0.2290	14.5649
$adjacency + \ell_0 + \ell_1$	0.7518	2.0892	0.2404	16.8827
$adjacency + \ell_0 + LRR$	0.7892	1.9773	0.2306	14.7932
$k = 30$				
baseline: *GL*-graph	0.7703	2.3802	0.2350	13.5401
$adjacency + \ell_0 + KNN$	0.7968	2.3235	0.1988	12.8590
$adjacency + \ell_0 + \ell_1$	0.7900	2.3705	0.2166	13.3426
$adjacency + \ell_0 + LRR$	0.7964	2.3166	0.1904	12.8149
$k = 40$				
baseline: *GL*-graph	0.7752	2.5688	0.2301	13.5003
$adjacency + \ell_0 + KNN$	0.7957	2.4623	0.1845	12.8569
$adjacency + \ell_0 + \ell_1$	0.7911	2.4922	0.2005	13.2135
$adjacency + \ell_0 + LRR$	0.7951	2.4603	0.1743	12.8511

E. Comparison with state-of-the-art algorithms

Our work follows a similar, yet not identical, strategy as the SAS algorithm [9], i.e., building a bipartite graph over multiple superpixels and pixels, then using Tcuts for image segmentation. The main difference between both methods is the affinity graph construction. In SAS, adjacent neighborhoods of superpixels are used, and the pairwise superpixel similarity is computed by the Gaussian weighted Euclidean distance in the color feature space. In our method, we build a *GL*-graph combining classical spatial homogeneity of objects and long range clustering based on sparse representation over multiple types of features and multi-scale superpixels, making the constructed graph having the characteristics of a long range neighborhood topology, yet with sparsity and high discriminative power. Fig.8 shows various segmentation results obtained with either the SAS method (second image of each experiment), or with our algoritm (third image). Notice that the results of SAS are the best results reported by the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

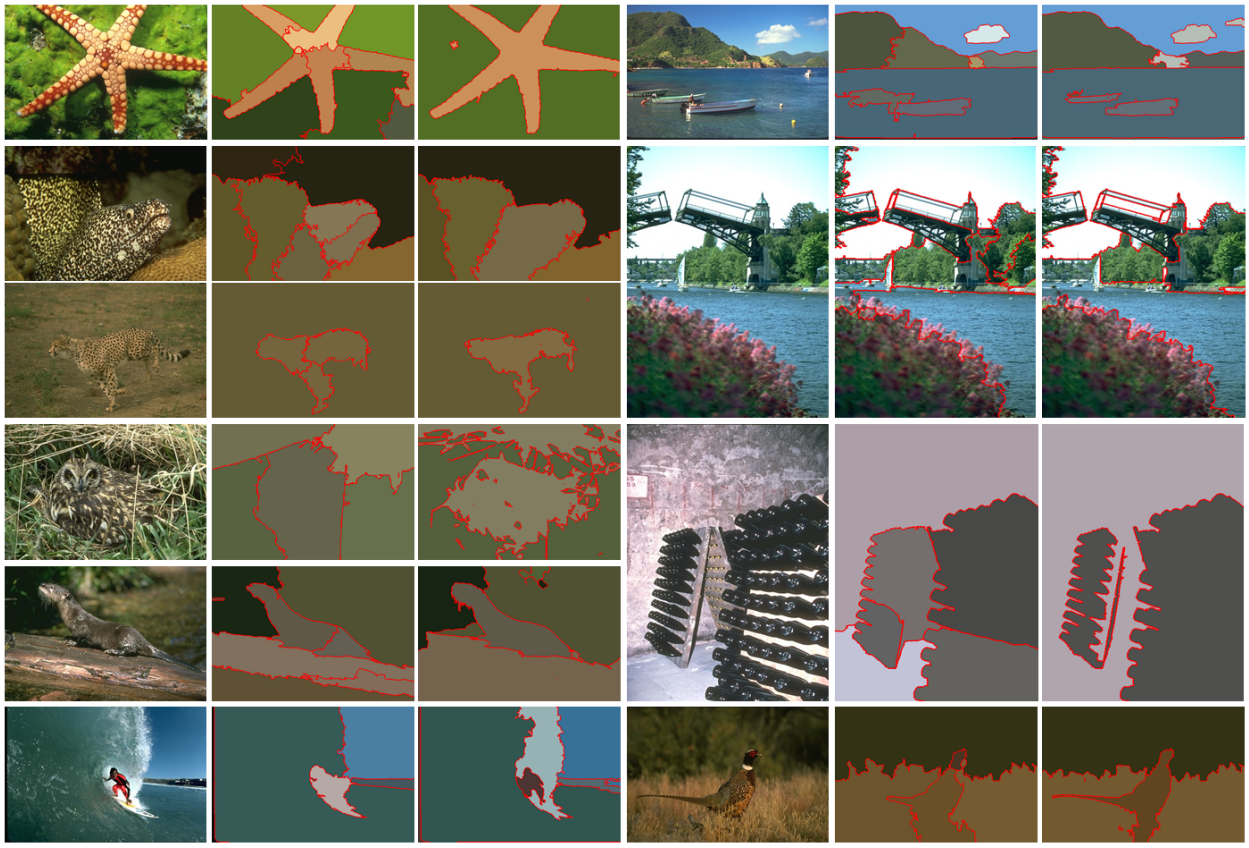


Fig. 8. Visual comparison with SAS. For each experiment, the second image shows the results of SAS, and the third image is obtained with our method. Our results require significantly less tuning for k and are visually better in general, in particular often more accurate.

authors, and require a careful tuning of the number of segments k (e.g. for *starfish*, *owl* and *leopard*, $k = 11, 4, 5$ respectively). For our method that takes into account the global information, a desirable result can be usually achieved with either k=2, 3, or 4 (e.g. for *starfish*, *owl* and *leopard*, $k = 2$). Especially, compared with SAS, our method achieves a correct segmentation even in the difficult cases where: 1) The detected object is highly textured (this is for instance the case of *starfish*, *moray eel*, *leopard*, and *owl*), and the background may be highly unstructured. In the particularly difficult case of the *owl* image, our method segments it correctly while the segmentation provided by SAS is not meaningful; 2) The object and its surrounding are quite similar in color or texture (*river otter*, *leopard* and *bird*). For example, the SAS algorithm oversegments the river otter and the leopard into several parts, while our method yields a correct segmentation. 3) Objects of the same type appear in a large, possibly disconnected, region of the image, as for instance the bottles or the mountain. SAS is not competitive with our method for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

long-range grouping, hence it tends to split the object into different parts (e.g. the bottles into 4 parts and the mountain into 4 parts). On the contrary, our proposed method can derive the right partition.

TABLE V

QUANTITATIVE COMPARISON OF THE PROPOSED METHOD (*GL*-GRAPH) WITH STATE-OF-THE-ART METHODS OVER THE BERKELEY SEGMENTATION DATABASE.

Methods	PRI↑	VoI↓	GCE↓	BDE↓
NCut [5]	0.7242	2.9061	0.2232	17.15
JSEG [51]	0.7756	2.3217	0.1989	14.40
MNCut [22]	0.7559	2.4701	0.1925	15.10
NTP [52]	0.7974	2.113	0.2171	13.58
TBES [53]	0.8000	1.7600	N/A	N/A
UCM [43]	0.8100	**1.6800**	N/A	N/A
SDTV [54]	0.7758	1.8165	0.1768	16.24
LFPA [8]	0.8146	1.8545	0.1809	12.21
Context-sensitive (mlab) [11]	0.7937	3.9174	0.4165	**9.9046**
$Cotransduction_{(mlab+LBP)}$ [12]	0.8083	2.3644	0.2681	14.1972
TPG [14]	0.8227	1.7696	N/A	N/A
FusionTP [15]	0.7771	3.3089	0.3654	13.2428
SAS [9]	0.8319	1.6849	0.1779	11.29
ℓ_0-graph [23]	0.8355	1.9935	0.2297	11.1955
GL-graph	**0.8384**	1.8012	0.1934	**10.6633**

We also report quantitative comparison with SAS and other standard benchmarks: Ncut [5], JSEG [51], Multi-scale Ncut (MNcut) [22], Normalized Tree Partitioning (NTP) [55], Saliency Driven Total Variation (SDTV) [54], Texture and Boundary Encoding-based Segmentation (TBES) [53], Ultrasound Contour Map (UCM) [43], Learning Full Pairwise Affinity (LFPA) [8], SAS [9], Context-sensitive [11], Co-transduction [12], Tensor Product Graph (TPG) [14], and Fusion with TPG [15]. The results are shown in Table V, where we highlight in bold the best two results for each qualitative criterion.

Most of the average scores of the benchmark methods are collected from [9], [8] and [14], with exception of [11], [12], [15], the graphs proposed in which are for task of shape retrieval and visual tracking. Nevertheless, we compare with their graph construction methods, by only replacing the *GL*-graph in our segmentation framework, while keeping other settings such as multi-scale superpixels and bipartite graph structure the same, for the sake of fairness. From Table V, we can observe that only with one feature mlab, the context-sensitive graph [11] has very promising performance. It is worth to mention that the graph construction techniques proposed in [11], [12], [15], [14], are with very high computational

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

cost and even hardly acceptable for the bottom-up segmentation, which is usually pre-process for high-level compute vision task, e.g. object recognition and detection. We can see that our method ranks first for PRI, VoI, GCE, and BDE, in particular the gain is significant for PRI and BDE.

Fig. 9. Visual segmentation examples by the proposed method: all images are segmented into 2 regions (k=2). Note that the salient objects or parts can be segmented accurately, such as the plane, boat, flower with insects, elephants, hill. Even multiple objects with large inner color variation can be segmented correctly, as the cactus flowers or the men in water.

Additionally, to demonstrate the advantage of our algorithm in practical applications, we present visual segmentation results of our method with k =2. As can be seen in Fig. 9, our method tends to first segment the most salient objects in the image even in the following cases where: 1) the detected object is tiny (see the *aeroplane*, the *boat*); 2) multiple objects are needed to segment in the same image (as in both middle rows); 3) the color of background and object are quite similar (see the last row).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2015.2397313, IEEE Transactions on Image Processing

F. Algorithm time complexity

The framework of the proposed algorithm as depicted in Fig.2 includes steps of oversegmentation, feature extraction, GL-graph and bipartite graph construction and graph partitioning. They are all coded as Matlab routines. The time complexity of OMP for GL-graph construction is analyzed in section III-B. The time complexity of graph partitioning using Transfer cut is analyzed in section III-D. Using a standard computer (Intel Core (TM) 2.3GHz CPU with 16G memory) to segment an image from BSD, *e.g.*, *"2092.jpg"*, generating multi-scale superpixels with MS and FH takes 4.68 seconds, extracting all the visual features listed in Section III-A takes 1872.23 seconds, building the bipartite graph with a single feature requires 2.12 seconds, of which the superpixel graph constructed by the proposed GL-graph only lasts 0.12 seconds for a graph with size 123×123, 0.11 seconds for 121×121, 0.13 seconds for 105×105, 0.39 seconds for 209×209, and 0.03 seconds for 53×53; cutting the bipartite graph into 11 clusters, the computational time is 0.87 seconds.

V. Conclusion

In this paper, we introduced a sparse global/local graph which encodes in a sparse way the perceptual grouping laws, e.g., proximity, similarity, and continuity. Unlike classical methods, our *GL*-graph is able to encode adaptively both local and global homogeneity of an object via fusing two types of graphs: the *adjacent*-graph and the sparse ℓ_0-minimization based graph built separately on three different classes of superpixels, i.e. enforcing proximity and similarity over small and large sized superpixels and encoding long range similarity on medium sized ones. Moreover, the discriminative power of the *GL*-graph is further enhanced by fusing several different features over multi-scale superpixels. The derived *GL*-graph is plugged into an efficient graph-cut method for unsupervised image segmentation. Extensive validations on the BSD data set show that our method yields very competitive qualitative and quantitative segmentation results compared to state-of-the-art methods.

As future extension of our work, it could be interesting to be able to learn an optimal fusion scheme that combines color, texture and shape cues using training data. That would be an interesting step toward semi-supervised image segmentation.

References

[1] Y. J. Lee and K. Grauman, "Object-graphs for context-aware category discovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1–8.

[2] M. P. Kumar and D. Koller, "Efficiently selecting regions for scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3217–3224.

[3] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and texture-based image segmentation using em and its application to content-based image retrieval," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1998, pp. 675–682.

[4] M. Wertheimer, "Laws of organization in perceptual forms," in *A sourcebook of Gestalt Psychology*, 1938, pp. 71–88.

[5] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[6] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[7] J. M. C. Fowlkes, D. R. Martin, "Learning affinity functions for image segmentation: Combining patch-based and gradient-based approaches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2003, pp. 54–64.

[8] T. H. Kim, K. M. Lee, and S. U. Lee, "Learning full pairwise affinities for spectral segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2101–2108.

[9] Z. Li, X. Wu, and S. Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 789–796.

[10] B. Wang and Z. Tu, "Affinity learning via self-diffusion for image segmentation and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2312–2319.

[11] X. Bai, X. Yang, L. J. Latecki, W. Liu, and Z. Tu, "Learning context-sensitive shape similarity by graph transduction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 5, pp. 861–874, 2010.

[12] X. Bai, B. Wang, C. Yao, W. Liu, and Z. Tu, "Co-transduction for shape retrieval," *Image Processing, IEEE Transactions on*, vol. 21, no. 5, pp. 2747–2757, 2012.

[13] X. Yang and L. J. Latecki, "Affinity learning on a tensor product graph with applications to shape and image retrieval," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2369–2376.

[14] X. Yang, L. Prasad, and L. J. Latecki, "Affinity learning with diffusion on tensor product graph," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 28–38, 2013.

[15] Y. Zhou, X. Bai, W. Liu, and L. J. Latecki, "Fusion with diffusion for robust visual tracking," in *Advances in Neural Information Processing Systems*, 2012, pp. 2978–2986.

[16] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Inf. Process. Syst. (NIPS) 16*, 2004, pp. 321–328.

[17] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[18] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv preprint arXiv:1304.5634*, 2013.

[19] N. Chen, J. Zhu, F. Sun, and E. P. Xing, "Large-margin predictive latent subspace learning for multiview data analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 12, pp. 2365–2378, 2012.

[20] V. Roth and T. Lange, "Adaptive feature selection in image segmentation," in *Pattern Recognition*. Springer, 2004, pp. 9–17.

[21] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2439–2446.

[22] T. Cour, F. Bénézit, and J. Shi, "Spectral segmentation with multiscale graph decomposition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 1124–1131.

[23] X. Wang, H. Li, S. Masnou, and L. Chen, "A graph-cut approach to image segmentation using an affinity graph based on ℓ_0- sparse representation of features," in *Proc. IEEE Int. Conf. Image Process.*, 2013, pp. 4019–4023.

[24] M. S. Toni P.Saarela, "Combination of texture and color cues in visual segmentation," *Vision Research*, pp. 58:59–67, 2012.

[25] M.Peterson and B.Gibson., "Shape recognition contributions to figure-ground organization in three dimensional displays," *Cognitive Psychology*, pp. 25 :383–429, 1993.

[26] V. Ulrike, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[27] X. Yang and L. J. Latecki, "Affinity learning on a tensor product graph with applications to shape and image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 2369–2376.

[28] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2439–2446.

[29] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[30] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[31] X. Zhang, Z.L.Wei, J. Feng, and L. Jiao, "Sparse representation-based spectral clustering for sar image segmentation," *SPIE*, pp. 08–06, 2011.

[32] X. Wang, H. Li, Simon, and L. Chen, "Sparse coding and mid-level superpixel-feature for l0-graph based unsupervised image segmentation," in *International Conference on Computer Analysis of Images Patterns (CAIP)*, 2013.

[33] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[34] D. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, 1999, pp. 1150–1157 vol.2.

[35] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2439–2446.

[36] M. Elad, *Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing*. Springer, 2010. [Online]. Available: http://dx.doi.org/10.1007/978-1-4419-7011-4

[37] J. Wright, A. Yang, A. Ganesh, S.S.Sastry, and M.Yi, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210 –227, Feb. 2009.

[38] P. Breen, "Algorithms for sparse approximation," *Project, School of Mathematics University of Edinburgh*, 2009.

[39] A. V. Goldberg and R. E. Tarjan, "A new approach to the maximum flow problem," *J. ACM*, vol. 35, pp. 921–940, Oct. 1988.

[40] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[41] G. H. Golub and C. F. Van Loan, *Matrix computations (3rd ed.)*. Baltimore, MD, USA: Johns Hopkins University Press, 1996.

[42] H. Z. Xiaofeng, X. He, C. Ding, H. Simon, and M. Gu, "Bipartite graph partitioning and data clustering," in *Proc. the tenth Int. Conf. on Inf. and Knowl. Manag. (CIKM)*, 2001, pp. 25–32.

[43] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[44] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 929–944, Jun. 2007.

[45] M. Meila, "Comparing clusterings: an axiomatic view," in *Proc. Int. Conf. Machine Learning*, 2005, pp. 577–584.

[46] D. R. 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. IEEE Int. Conf. Comput. Vis.*, 2001, pp. 416–425.

[47] J. Freixenet, X. Muñoz, D. Raba, J. Martí, and X. Cufí, "Yet another survey on image segmentation: Region and boundary information integration," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 408–422.

[48] L. J. Grady, "Space-variant computer vision: a graph-theoretic approach," Ph.D. dissertation, Boston, MA, USA, 2004.

[49] C. Zhu, C. Bichot, and L. Chen, "Multi-scale color local binary patterns for visual object classes recognition," in *Int. Conf. on Pattern Recognit.*, 2010, pp. 3065–3068.

[50] L. Zhuang, H. Gao, Z. Lin, Y. Ma, X. Zhang, and N. Yu, "Non-negative low rank and sparse graph for semi-supervised learning," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2012, pp. 2328–2335.

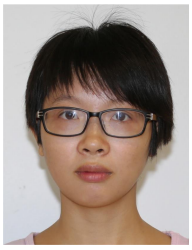
[51] D. Yining and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 8, pp. 800–810, Aug. 2001.

[52] J. Wang, H. Jiang, Y. Jia, X.-S. Hua, C. Zhang, and L. Quan, "Regularized tree partitioning and its application to unsupervised image segmentation," *TIP*, vol. 23, no. 4, pp. 1909–1922, 2014.

[53] S. Rao, H. Mobahi, A. Y. Yang, S. Sastry, and Y. Ma, "Natural image segmentation with adaptive texture and boundary encoding," in *Asian Conf. on Comput. Vis.*, 2009, pp. 135–146.

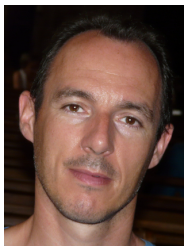
[54] M. Donoser, M. Urschler, M. H., and H. Bischof, "Saliency driven total variation segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 817–824.

[55] J. Wang, Y. Jia, X. Hua, C. Zhang, and L. Quan, "Normalized tree partitioning for image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.



Xiaofang Wang received the B.S. and M.S. degrees in biomedical engineering from Central South University, Changsha, Hunan, China. She is currently working toward the PhD degree in computer science from Ecole Centrale de Lyon, Ecully, France. Her current research interests include image/video processing , medical image segmentation and analysis, multiple object tracking, and object detection and recognition.

Yuxing Tang received the B.S. and M.S. degrees from the Department of Information and Telecommunication Engineering, Beijing Jiaotong University, Beijing, China. He is currently a PhD candidate in the Department of Mathematics and Computer Science, Ecole Centrale de Lyon, Ecully, France. His research interests are computer vision and machine learning; in particular models for visual category recognition and detection.



Simon Masnou received the B.S., M.Eng degree from Telecom Paris, France in 1992. He received the M.S. degree in Applied Mathematics from University Paris 9, France in 1993. From 1993 to 1995 he held a teaching position in Gabon. In 1998, he obtained a Ph.D. degree in Mathematics from University Paris 9, France. In 1999, he was a Post-Doctoral Fellow at the Scuola Normale Superiore di Pisa, Italy. From 1999 to 2009 he was an Assistant Professor at University Paris 6, France. Since September 2009, he is a Professor of Mathematics at Institut Camille Jordan, University Lyon 1, France. Since 2013, he is Head of the ”Mathematical Modeling and Scientific Computing” group at the Institut Camille Jordan. His research interests include applications of calculus of variations and geometric measure theory to image and video processing.

Liming Chen was awarded a joint B.Sc. degree in mathematics and computer science from the University of Nantes in 1984. He obtained a M.Sc. degree in 1986 and a Ph.D. degree in computer science from the University of Paris 6 in 1989. He first served as associate professor at the Université de Technologie de Compiègne, then joined Ecole Centrale de Lyon as Professor in 1998, where he leads an advanced research team on multimedia computing and pattern recognition. From 2001 to 2003, he also served as Chief Scientific Officer in a Paris-based company, Avivias, specialized in media asset management. In 2005, he served as scientific multimedia expert in France Telecom R&D China. He has been head of the department of mathematics and computer science from 2007.

Prof. Liming Chen has taken out three patents, authored more than 100 publications and acted as chairman, PC member and reviewer in a number of high profile journal and conferences since 1995. He has been a (co)-principal investigator on a number of research grants from EU FP program, French research funding bodies and local government departments. He has directed more than 15 Ph.D. theses. His current research spans from 2D/3D face analysis and recognition, image and video analysis and categorization, to affect analysis both in image, audio and video.