Probabilistic Graphlet Cut: Exploiting Spatial Structure Cue for Weakly Supervised Image Segmentation

Luming Zhang[†], Mingli Song[†], Zicheng Liu[‡], Xiao Liu[†], Jiajun Bu[†] and Chun Chen[†]
College of Computer Science, Zhejiang University, Hang Zhou[†]
Microsoft Research, Redmond[‡]
{zglumg, brooksong, ender_liux, bjj, chenc}@zju.edu.cn[†]
zliu@microsoft.com[‡]

Abstract

Weakly supervised image segmentation is a challenging problem in computer vision field. In this paper, we present a new weakly supervised image segmentation algorithm by learning the distribution of spatially structured superpixel sets from image-level labels. Specifically, we first extract graphlets from each image where a graphlet is a small-sized graph consisting of superpixels as its nodes and it encapsulates the spatial structure of those superpixels. Then, a manifold embedding algorithm is proposed to transform graphlets of different sizes into equal-length feature vectors. Thereafter, we use GMM to learn the distribution of the post-embedding graphlets. Finally, we propose a novel image segmentation algorithm, called graphlet cut, that leverages the learned graphlet distribution in measuring the homogeneity of a set of spatially structured superpixels. Experimental results show that the proposed approach outperforms state-of-the-art weakly supervised image segmentation methods, and its performance is comparable to those of the fully supervised segmentation models.

1. Introduction

As a preprocessing operation, image segmentation is widely used in many computer vision applications, *e.g.*, image enhancement, image understanding, *etc*. Typically, these applications assume that the images are ideally segmented, *i.e.*, each segmented region covers a semantic component. However, in order to avoid overburdening users with manual work, these applications are usually built based on unsupervised image segmentation methods, which perform unsatisfactorily due to the lack of high-level cues. For example, many segmented regions partially cover one or multiple semantic objects, causing the subsequently constructed models to significantly deviate from the theoretical requirement of these applications. In addition, there are many tiny segmented regions which are treated as noises.

Inspired by the progress in image retrieval area, image-level labels can be efficiently and accurately acquired. Thus, it is possible to leverage weak supervision, *i.e.*, image-level labels, to improve image segmentation. However, weakly-supervised image segmentation is still a challenging problem due to the following two factors:

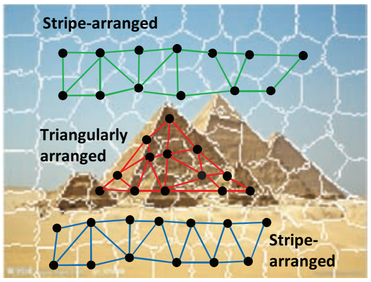


Figure 1. The superpixel mosaic of an example image

- The intrinsic ambiguity of image-level labels: compared with the pixel-level labels used in fully supervised segmentation models, image-level labels are much coarser cues which are difficult to be effectively incorporated into the segmentation model.

- The ignorance of spatial structure in measuring the homogeneity of superpixels: beyond the appearance features, the spatial structure of superpixels is also important for measuring their homogeneity. But the state-of-the-art segmentation models [1, 2] do not take it into consideration. As shown in Figure 1, the yellow pyramid and the sand have similar superpixel appearances. However, the triangularly arranged superpixels are distinctive for the pyramid, thus they should be assigned with strong homogeneity and are further encouraged to merge.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.249

To address the above two problems, we propose to learn the distribution of graphlets from image-level labels and use the learned distribution to guide the image segmentation. To capture the spatial structure of superpixels, we extract graphlets by connecting spatially neighboring superpixels, wherein the graphlets are small-sized graphs effectively capturing the neighboring structures of superpixels. Because different-sized graphlets are incomparable in Euclidean space, we project graphlets onto the Grassmann manifold and subsequently develop a manifold embedding algorithm which incorporates image-level labels into graphlets. Through the embedding, different-sized graphlets are transformed into equal length feature vectors, thus making it possible to model the distribution of graphlets. Since the learned graphlet distribution reflects the spatial structure of superpixels, we propose a new segmentation algorithm, called graphlet cut, that leverages the learned graphlet distribution.

2. Related Work

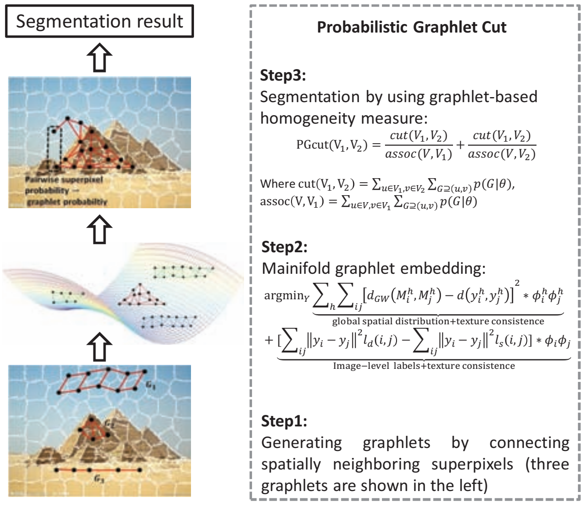
Recently, several weakly supervised image segmentation methods have been proposed, which focus on developing statistical models to transfer image-level labels into superpixels unary or pairwise potentials. Verbeek *et al.* [5] proposed an aspect model to estimate pixel-level labels for each image, which is modeled as a mixture of latent topics. Vezhnevets *et al.* [6] formulated weakly-supervised image segmentation as a multiple instances learning problem. However, both [5] and [6] fail to model the interactions between superpixels, which is important for smoothing superpixel labels. To model the relationships among superpixels, Verzhnevetz *et al.* [7] proposed a graphical model, termed multi-image model (MIM), to integrate image appearance features, image-level labels and superpixel labels into one network. To refine the MIM-based segmentation, Verzhnevetz *et al.* [8] designed an active learning scheme to select a few semantically most uncertain superpixels within an image. The selected superpixels are accurately labeled by querying an oracle database, and they guide the label inference for the remaining superpixels. Moreover, Verzhnevetz *et al.* [9] developed a parametric family of structured models, where multi-channel visual features are employed to form the pairwise potential, and the weights of each channel is computed by minimizing the discrepancy between superpixels labeled by segmentation models trained by different image sets.

One weakness of these weakly supervised segmentation methods is the low descriptive unary/pairwise potentials, resulting in many ambiguous segment boundaries. To alleviate this problem, high-order potentials among superpixels are exploited to refine image segmentation. Kohli *et al.* [2] proposed a high-order conditional random field for image segmentation, where the high-order potentials are defined over pixel sets. In [10], Rital *et al.* generalized the conventional normalized cut into hypergraph cut, where each hyperedge connects multiple spatially neighboring superpixels. However, hypergraph cut has two limitations: 1) supervision incorporation is difficult, and 2) label inference is computationally less efficient. To overcome these limitations, Kim *et al.* [1] developed a supervised high-order correlation clustering technique for image segmentation. Based on the structured support vector machine and the linear programming relaxation, both the parameter learning and segmentation process are carried out efficiently. Notably, these approaches are either unsupervised or fully-supervised, and it is difficult to transform them into a weakly supervised version. Moreover, the spatial structure of superpixels, an essential cue for measuring their homogeneity, is neglected.

3. The proposed approach

3.1. An overview

As shown in Figure 2, the proposed approach learns the distribution of graphlets [23] and then facilitates image segmentation based on the learned graphlet distribution. We first extract graphlets from each image, which capture the spatial structure of the superpixels. Then, the extracted graphlets are projected onto the Grassmann manifold and a manifold embedding algorithm is proposed to integrate image-level labels, global spatial layout, and rough geometric context, into graphlets. After embedding, graphlets are transformed into equal length feature vectors, and we use GMM to learn their distribution. The learned distribution is used to measure the homogeneity of superpixels. Finally, we propose a graphlet cut algorithm based on the homogeneity measure for image segmentation.



Probabilistic Graphlet Cut

Step3:
Segmentation by using graphlet-based homogeneity measure:

$$\mathrm{PGcut}(\mathrm{V}_1,\mathrm{V}_2) = \frac{cut(V_1,V_2)}{assoc(V,V_1)} + \frac{cut(V_1,V_2)}{assoc(V,V_2)}$$

Where $\mathrm{cut}(\mathrm{V}_1,\mathrm{V}_2) = \sum_{u\in V_1, v\in V_2} \sum_{G\supseteq(u,v)} p(G|\theta)$,
$\mathrm{assoc}(\mathrm{V},\mathrm{V}_1) = \sum_{u\in V, v\in V_1} \sum_{G\supseteq(u,v)} p(G|\theta)$

Step2:
Mainifold graphlet embedding:

$$\mathrm{argmin}_Y \underbrace{\sum_h \sum_{ij} \left[d_{GW}(M_i^h, M_j^h) - d(y_i^h, y_j^h)\right]^2 * \phi_i^h \phi_j^h}_{\text{global spatial distribution+texture consistence}} + \underbrace{\left[\sum_{ij} \|y_i - y_j\|^2 l_d(i,j) - \sum_{ij} \|y_i - y_j\|^2 l_s(i,j)\right] * \phi_i \phi_j}_{\text{Image−level labels+texture consistence}}$$

Step1:
Generating graphlets by connecting spatially neighboring superpixels (three graphlets are shown in the left)

Figure 2. The pipeline of the proposed approach

3.2. Graphlet extraction and representation

An image usually contains millions of pixels. To avoid the high computational cost caused by such gigantic amount of pixels in image segmentation, these pixels are clustered into superpixels and further associated with their spatial structure. And since graph is a natural and powerful tool to describe the relationships between objects, usually the region adjacency graph (RAG) is adopted to model the superpixels and their spatial structures, *i.e.*,

$$\mathcal{G} = (V, E) \tag{1}$$

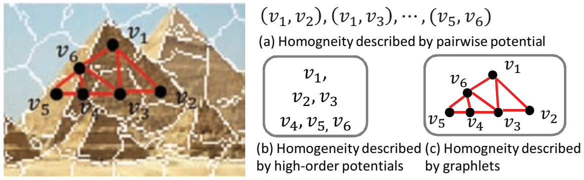
where V is a set of vertices, each representing a superpixel; E is a set of edges, each connecting pairwise spatially adjacent superpixels.

An image usually contains multiple semantic components, each spanning several superpixels. Given a superpixel set, two observations can be made. First, the appearance and spatial structure of the superpixels collaboratively contribute to their homogeneity. Second, the more their appearance and spatial structure correlate with a particular semantic object, the stronger their homogenneity. As shown in Figure 1, the superpixel set in the sky region and the superpixel set in the sand region have similar spatial structure but different superpixel appearance, thus they should be assigned with different homogeneities. Compared with the stripe-distributed yellow superpixels, the strip distributed blue superpixels appear more common in semantic objects, such as lake and river, which indicates they are low correlated with any particular semantic object, thus should be assigned with a weaker homogeneity. On the other hand, the pyramid-covered and the sand-covered superpixel sets have similar superpixel appearance but different spatial structure, thus they should also be assigned with different homogeneities. Compared with the stripe distributed yellow superpixels, the triangularly distributed yellow superpixels are unique for the Egyptian pyramid, thus they should be assigned with a stronger homogeneity.

We propose to use graphlets to capture the appearance and spatial structure of superpixels. The graphlets are obtained by extracting connected subgraphs from an RAG. The size of a graphlet is defined as the number of its constituent superpixels. In this work, only small-sized graphlets are adopted because: 1) the number of all the possible graphlets is exponentially increasing with its size; 2) the graphlet embedding implicitly extends the homogeneity beyond single small-sized graphlets. (as shown in Sect. 3.3); 3) empirical results show that the segmentation accuracy stops increasing when the graphlet size increases from 5 to 10, thus small-sized graphlets are descriptive enough. Let T denote the maximum graphlet size, we extract graphlets of all sizes ranging from 2 to T. The graphlet extraction is based on depth-first search, which is computationally efficient. Besides, our approach is also storage efficient. Given 50 superpixels in an image, and assuming the average superpixel degree is 5 and the maximum graphlet size is also 5, there are $50 * 5^5/5! + \cdots + 50 * 5^2/2! \approx 4300$ graphlets, which, after embedding, are transformed into 4300 low-dimensional feature vectors. Thus the required storage space is very small.

Note that graphlets extend the non-structural homogeneity of superpixels [1, 2]. As shown in Figure 3, both the pairwise and high-order potentials represent the homogeneity of orderless superpixels, whereas the graphlet represents the homogeneity of spatially structured superpixels. If we ignore graphlet topology, the proposed graphlet-based homogeneity reduces to the high-order potential homogeneity.

A quantitative description of graphlets is necessary for a computational segmentation model. Given a t-sized graphlet, we characterize the appearance of its superpixels as a matrix M_r. Each row of M_r is a 137-dimensional feature vector extracted from a superpixel, *i.e.*, a 128-dimensional histogram of gradient (HOG) [11] combined with a 9-dimenional color moment [12]. And, for the spatial structure of superpixels, within a t-sized graphlet, we use a $t \times t$-sized matrix to represent it as:



(a) Homogneity described by pairwise potential

(b) Homogeneity described by high-order potentials

(c) Homogneity described by graphlets

Figure 3. Different types of superpixel homogeneity

$$M_s(i, j) = \begin{cases} \theta(R_i, R_j) & \text{if } R_i \text{ and } R_j \text{ are spatially adjacent} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\theta(R_i, R_j)$ is the angle between the positive horizontal direction and the vector from the center of superpixel R_i to the center of superpixel R_j. Based on M_r and M_s, a t-sized graphlet can be represented by a $t \times (137 + t)$ matrix, *i.e.*,

$$M = [M_r, M_s] \tag{3}$$

According to the differential geometry theory [13], each matrix can be regarded as a point on the Grassmann manifold. To measure the distance between graphlets on the Grassmann manifold, their Golub-Werman distance is defined as:

$$d_{GW}(M, M') = ||M_o - M_o^{'}||_2 \tag{4}$$

where M_o and $M_o^{'}$ denote the orthonormal basis of M and M' respectively.

3.3. Manifold graphlet embedding

As mentioned in Sect. 3.2, the appearance and spatial structures of semantically-consistent superpixels reflect strong homogeneity. Thus, it is necessary to integrate category information into graphlets in measuring the homogeneity of superpixels. To this end, a manifold embedding algorithm is proposed to encode image-level labels into graphlets. Besides image-level labels, two supplementary cues are also incorporated into the embedding. The first is the global spatial layout, and the second is the geometric context.

To incorporate the global spatial layout information, we would like our embedding scheme to maximally preserve the relative distances between the graphlets. This is helpful to expand the homogeneity of superpixels across individual graphlets. As shown in the left of Figure 4, preserving the relative distances in the embedding process encodes global spatial layout into graphlets, which implicitly extends the homogeneity beyond the individual small-sized graphlets.

As demonstrated by Vezhnevets *et al.* [6], rough geometric context [14] effectively complements image-level labels for image segmentation. Rough geometric context means categorizing each pixel in an image into ground, different oriented vertical regions, non-planar solid, or porous. This motivates us to integrate geometric context information into the embedding process. Intuitively, a graphlet with consistent geometric context should reflect stronger homogeneity. As shown in the right of Figure 4, graphlet G_1 has more consistent geometric context than graphlet G_2, thus superpixels within G_1 should be assigned with stronger homogeneity than those within G_2.

To capture the above three cues, namely, image-level labels, global spatial layout, and geometric context, we propose a manifold embedding algorithm with the objective function defined as:

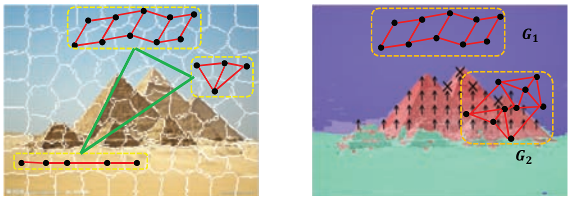


Figure 4. Left: example of preserving global spatial layout; Right: adding rough geometric context into graphlets, ground(green), sky(blue), different oriented vertical regions(red), non-planar solid('x')

$$\arg\min_Y \underbrace{\sum_h \sum_{ij} [d_{GW}(M_i^h, M_j^h) - d_E(y_i^h, y_j^h)]^2 * \phi_i^h \phi_j^h}_{global\ spatial\ layout+geometric\ context}$$

$$+ \underbrace{[\sum_{ij} ||y_i - y_j||^2 l_s(i,j) - \sum_{ij} ||y_i - y_j||^2 l_d(i,j)] * \phi_i \phi_j}_{Image-level\ labels+geometric\ context} \tag{5}$$

The first term $\sum_{ij}[d_{GW}(M_i^h, M_j^h) - d_E(y_i^h, y_j^h)]^2$ describes the discrepancy between pairwise graphlet distances on the Grassmann manifold [21] and those in the Euclidean space. The minimization of this term will maximally preserve the global spatial arrangement of the graphlets. The second term $\phi_i^h \phi_j^h$ enforces the geometric context constraint on graphlets. That is, graphlets with more consistent geometric context are assigned with smaller weights. The third term $\sum_{ij} ||y_i - y_j||^2 l_s(i,j) - \sum_{ij} ||y_i - y_j||^2 l_d(i,j)$ encodes image-level labels into pairwise graphlets. That is, the proximity of two graphlets in feature space should be consistent with their image-level labels.

The variables in (5) are defined as follows. M_i^h and M_j^h denote two identical-sized graphlets; y_i^h and y_j^h are their low-dimensional representations; l_s is a function that measures the similarity of two graphlets. l_d is a function that measures the difference between two graphlets. Let $b(G)$ denote the C-dimensional row vector containing the class label of the image corresponding to graphlets G. Denote $\vec{N} = [N^1, N^2, \cdots, N^C]^T$ where N^c is the number of images for category c, then $l_s(i,j) = \frac{[b(G_i) \cap b(G_j)]\vec{N}}{\sum_c N^c}$ and $l_d(i,j) = \frac{[b(G_i) \oplus b(G_j)]\vec{N}}{\sum_c N^c}$. ϕ_i reflects the geometric context consistency of the i-th graphlet, which is implemented as the i-th graphlet entropy, *i.e.*, $\phi_i = -\sum_j g_i(j) \log_2 g_i(j)$, where $g_i(j)$ is percentage of the j-th geometric context corresponding to graphlet G_i. ϕ_i^h is the geometric context obtained from the i-th graphlet in the h-th training image.

Denoting $D_{GW}^h = [d_{GW}(M_i^h, M_j^h)]$ as the matrix whose entry $d_{GW}(M_i^h, M_j^h)$ is the Golub-Werman distance between the i-th and j-th identical-sized graphlets extracted from the h-th image. Its inner product matrix is obtained by:

$$\tau(D_{GW}^h) = -R_{N_h} S_{GW}^h R_{N_h}/2 \tag{6}$$

where $(S_{GW}^h)_{ij} = (D_{GW}^h)_{ij}^2$, and $R_{N_h} = \mathbf{I}_{N_h} - \vec{e}_{N_h} \vec{e}_{N_h}^T / N$ which is the centralization matrix. $\mathbf{I}_{N_h}$ is an $N_h \times N_h$ identity matrix, $\vec{e}_{N_h} = [1, 1, \cdots, 1]^T \in \mathbb{R}^{N_h}$, N is the number of all training graphlets, and N_h the number of graphelts from the h-th training image.

Thus the first part of (5) can be rewritten as:

$$\begin{aligned} &\arg\min_Y \sum_h \sum_{ij} [d_GW(M_i^h, M_j^h) - d_E(y_i^h, y_j^h)]^2 * \phi_i^h \phi_j^h \\ &= \arg\min_Y \sum_h ||\tau(D_{GW}^h) - \tau(D_Y^h)||^2 * \phi_i^h \phi_j^h \\ &= \arg\max_Y \sum_h tr(Y^h \tau(D_{GW}^h \Phi^h)(Y^h)^T) \\ &= \arg\max_Y tr(Y \tau(D_{GW}) \Phi) Y^T \end{aligned} \tag{7}$$

where $\Phi = [\phi_i, \phi_j]$ is an $N \times N$ matrix; and D_{GW} is a block diagonal matrix, the h-th diagonal block is D_{GW}^h.

The second part in (5) can be reorganized into:

$$\begin{aligned} &\arg\max_Y [\sum_{ij} ||y_i - y_j||^2 l_d(i,j) - \sum_{ij} ||y_i - y_j||^2 \\ &l_s(i,j)] * \phi_i \phi_j = \arg\max_Y tr(YRY^T) \end{aligned} \tag{8}$$

where $R = [-\vec{e}_{N-1}^T, \mathbf{I}_{N-1}]^T W_1 [-\vec{e}_{N-1}^T, \mathbf{I}_{N-1}] + \cdots + [\mathbf{I}_{N-1}, -\vec{e}_{N-1}^T]^T W_N [\mathbf{I}_{N-1}, -\vec{e}_{N-1}^T]$, and W_i is an $N \times N$ diagonal matrix whose h-th diagonal element is $[l_s(h,i) - l_d(h,i)] * \phi_h \phi_i$.

Based on above formulation, we can reorganize the objective function as:

$$\begin{aligned} & \arg\max_Y Y(\tau(D_{GW})\Phi + R)Y^T \\ & = \arg\max_Y YZY^T \ s.t. \ YY^T = \mathbf{I}_N \end{aligned} \qquad (9)$$

where $Z = \tau(D_{GW})\Phi + R$ is an $N \times N$ matrix, and $YY^T = \mathbf{I}_N$ is a term to uniquely determine Y. Note that the embedding in (9) can only handle identical-sized graphlets. Assuming the maximum graphlet size is T, the embedding is repeated T times.

We note that (9) is a quadratic programming with quadratic constrains that can be solved through eigenvalue decomposition, which has a time complexity of $\mathcal{O}(N^3)$. However, Z is a large-sized matrix because usually $N > 50,000$. Therefore it is computational intractable to solve (9) using a global once-for-all eigenvalue decomposition on matrix Z. Instead, we decompose the eigenvalue decomposition into a set of sub-problems. Particularly, we first solve an initial embedding $Y^{(0)}$ using (9) under $N^{(0)}$ training graphlets, where $N^{(0)} << N$. Then we use coordinate propagation [15] to transmit the embedded coordinates to the new graphlets. The coordinate propagation is carried out fast based on the iterative algorithm proposed by Xiang *et al.* [15].

3.4. Probabilistic graphlet cut

After the embedding process, graphlets of different sizes are transformed into d-dimensional feature vectors. To employ these post-embedding graphlets for image segmentation, we train a standard GMM to model their distribution. Given an post-embedding graphlet $f(G_{test})$ from the test image, the homogeneity of its superpixels is computed by:

$$p(f(G_{test})|\theta) = \sum\nolimits_{k=1}^{K} w_k \mathcal{N}(f(G_{test})|\mu_k, \Sigma_k) \qquad (10)$$

where $\theta = \{w_k, \mu_k, \Sigma_k\}$ are the GMM parameters learned by using expectation maximization from training data, and the Gaussian component number K is set to 5.

Next, we apply the graphlet-based homogeneity measure for image segmentation in the normalized cut framework. The proposed approach improves the conventional normalized cut in two aspects. First, the conventional normalized cut measures the similarity between superpixels using the distance between their appearance feature vectors, whereas our approach measures their similarity by taking into consideration their spatial structures. Second, conventional normalized cut fails to incorporate supervision, while our approach integrates image-level labels, global spatial layout, and rough geometric context to refine the segmentation process. The objective function of our graphlet-guided normalized cut is given below:

$$PGcut(V_1, V_2) = \frac{cut(V_1, V_2)}{assoc(V, V_1)} + \frac{cut(V_1, V_2)}{assoc(V, V_2)} \qquad (11)$$

where V_1 and V_2 are two disjoint sets of superpixels. The three terms $cut(V_1, V_2)$, $assoc(V, V_1)$ and $assoc(V, V_1)$ are defined in the next paragraph. (11) can be solved in the same way as the conventional normalized cut. Note that 2-way cut is presented in (11) and the multi-way variant can be derived straightforwardly by following [16].

The numerator in (11) measures the cost of removing all edges spanning superpixel sets V_1 and V_2, *i.e.*,

$$\begin{aligned} cut(V_1, V_2) &= \sum\nolimits_{u \in V_1, v \in V_2} w(u, v) \\ &= \frac{1}{|G|} \sum\nolimits_{u \in V_1, v \in V_2} \sum\nolimits_{G \supseteq (u,v)} p(G|\theta) \end{aligned} \qquad (12)$$

where $w(u, v)$ is the relationship between superpixel u and v. The term $G \supseteq (u, v)$ contains all the parent graphlets of superpixel pair (u, v), and $1/|G|$ functions as a normalization factor.

The two denominators in (11) respectively accumulate connections from superpixels in set V_1 and V_2 to the entire superpixels, *i.e.*,

$$\begin{aligned} assoc(V, V_1) &= \sum_{u \in V, v \in V_1} w(u, v) \qquad (13) \\ &= \frac{1}{|G|} \sum_{u \in V, v \in V_1} \sum_{G \supseteq (u,v)} p(G|\theta) \end{aligned}$$

$$\begin{aligned} assoc(V, V_2) &= \sum_{u \in V, v \in V_2} w(u, v) \qquad (14) \\ &= \frac{1}{|G|} \sum_{u \in V, v \in V_2} \sum_{G \supseteq (u,v)} p(G|\theta) \end{aligned}$$

Similar to many segmentation methods such as [6], which assign semantics to segmented regions, our approach can also label semantics for each superpixel. Particularly, we first learn a multi-label SVM based on the d-dimensional post-embedding graphlets and the category labels of the images from which the graphlets are extracted. Given a test graphlet G_{test}, based on the probabilistic output of SVM [17], we obtain its probability of belonging to semantic class $c : p(G_{test} \to c)$, and the semantic label of segmented region R is computed by maximum majority voting of all its spatially overlapping graphlets:

$$\arg\max_c \sum\nolimits_{G_{test} \cap R \neq \emptyset} p(G_{test} \to c) \qquad (15)$$

The detailed implementation procedure of the proposed probabilistic graphlet cut is given in Table 1.

Table 1. Probabilistic Graphlet Cut

```
//training stage:
input: a set of training images $\{I_1, I_2, \cdots, I_H\}$ associated
with image-level labels $\{c_1, c_2, \cdots, c_H\}$;
output: trained embedding model and a multi-class SVM;
1. Construct RAG for each image and extracted graphelts
from these RAGs; then use manifold graphelt embedding to
transformeach graphlet into $d$-dimensional feature vector
according to (5);
2. Using GMM to model the statistics of these training
post-embedding graphlets;
3. Learn a multi-label SVM based on the post-embedding
graphlets and the image-level labels;
//test stage:
input: a test image $I_{test}$ and its image-level label $c_{test}$; the
number of segmented regions $L$;
output: a segmentation mask of $I_{test}$;
1. Construct RAG for $I_{test}$ and extracted its graphelts; Using
the trained manifold embedding model to represent each
graphelt by a $d$-dimensional feature vector;
2. Using $(11)$ to partition $I_{test}$ into $L$ disjoint sets; infer the
semantics of each segmented region in $I_{test}$ accordingly
based on $(15)$;
```

4. Experimental results and analysis

In this section, we validate the effectiveness of the proposed approach for weakly-supervised segmentation based on four sets of experiments. The first set of experiments compares our approach with representative segmentation algorithms. The second set of experiments evaluates the individual components of our approach. Discussion of parameter setting is given in the third part. The last part analyzes the segmentation results of our approach on SIFT-flow [3].

4.1. Comparison with the state of the art

In this experiment, we compare our approach with four segmentation methods including two weakly-supervised segmentation methods: multi-image model (MIM) [7] and its variant(GMIM) [9], as well as two fully-supervised segmentation algorithms: TextonBoost (TB) [18] and hierarchical conditional random field (HCRF) [19].

To compare our approach with the existing weakly supervised segmentation methods, we carry out experiments on SIFT-flow, because the objects are of diversified structures and the image-level labels are off-the-shelf. Additionally, it is important to compare our approach with fully supervised segmentation methods, because the comparative results show how effectively the image-level labels facilitate image segmentation. To this end, we also experiment on PASCAL VOC 2008 [4]. Note that, only foreground objects are labeled in the VOC 2008. To obtain background labels, for each image we manually assign one of sky, road, indoor as its background label. We further combine the foreground label and the background label as the image-level label. On both data sets, we use standard training and test splits.

Table 2. Average per-class measure from the five compared methods on SIFT-flow and PASCAL VOC 2008

	MIM	GMIM	TB	HCRF	Our
SIFT-flow	14%	21%	24%	31.22%	27.73%
VOC 2008	8.11%	9.24%	13.2%	20.1 %	14.87%

The segmentation performance is evaluated by average-per-class measure, which averages the correctly classified pixels per-class over all classes. In Table 2, we report the segmentation performance of the five compared methods. There are two observations. First, on both data sets, our approach significantly outperforms the other two weakly-supervised segmentation methods: MIM and G-MIM, demonstrating that image-level labels are more effectively encoded by our model. Second, our approach outperforms TextonBoost on both datasets, and performs comparably to HCRF on VOC 2008. This demonstrates that, even though image-level labels are much coarser cues compared with pixel-level labels, if exploited effectively they can boost segmentation performance to the same extent as pixel-level labels.

4.2. Step-by-step model justification

This experiment justifies the effectiveness of the three main components in the proposed approach, *i.e.*, graphlet extraction, manifold graphlet embedding, and the probabilistic segmentation model.

To justify the effectiveness of graphlets for weakly-supervised segmentation, two experimental settings are adopted to weaken the description power of the graphlets. First, we reduce graphlets to superpixels, that is, 1-sized graphlet which captures no spatial structure of superpixels.Second, we remove the structure term M_s from (3). In Figure 5, we present the segmentation results under the two experimental settings. We can see that segmentation using superpixels or non-structural graphlets results in many ambiguous segment boundaries.

To justify the effectiveness of manifold graphlet embedding, three experimental settings are used. In the first setting, we remove the geometric context term $\phi_i^h \phi_j^h$ and $\phi_i \phi_j$ from the objective function (5). In the second setting, we transform our approach into an unsupervised version by abandoning the image-level label encoding term from (5). In the third setting we transform our approach into an unsupervised version by replacing the manifold embedding with kernel PCA, where the kernel is defined as $k(M, M') = ||M^T M'||_F^2$. We present segmentation results under the three experimental settings in Figure 5. By comparing with the ground truth, we can see that segmentation by removing the geometric context term results in large incorrectly labeled regions. This demonstrates the impor-

Table 3. Average per-class accuracy on PASCAL VOC 2008

aeroplane	bicycle	bird	boat	bottle	bus	car	chair	cow	diningt
21.11%	15.23%	12.34%	23.12%	16.13%	18.24%	13.45%	16.22%	12.98%	11.18%
diningt.	dog	horse	motorbike	person	pottedp.	sheep	sofa	train	tv
14.66%	17.21%	10.01%	15.37%	9.98%	16.12%	17.35%	17.10%	8.14%	10.15%

Table 4. Performance decrease of component replacement

Component replacement	SIFT-flow	VOC 2008
Superpixel as graphlet	4.21%	3.43%
Non-structral graphlet	3.36%	2.77%
Remove geometric context term	6.54%	5.43%
Remove image-level label term	3.31%	2.58%
Replace graph. embed. with kernel PCA	5.12%	4.67%
Normalized cut with 2-sized graphelts	4.49%	4.54%

Table 5. Segmentation accuracy and time consumption per image under different maximum graphlet size T

T	Seg. Acc	Seg. Time	T	Seg. Acc	Seg. Time
1	7.76%	3.23s	6	14.65%	124.56s
2	8.56%	6.78s	7	14.76%	278.56s
3	9.87%	15.67s	8	14.87%	675.89s
4	10.23%	32.45s	9	14.87%	1345.77s
5	13.45%	66.65s	10	14.87%	2688.73s

tance of incorporating geometric context into the segmentation process. Besides, segmentation without image-level label supervision performs less satisfactorily. This shows that image-level labels contribute positively to image segmentation. Furthermore, very poor segmentation results are observed when kernel PCA is adopted because both geometric context and image-level label supervision are neglected.

To justify the effectiveness of the probabilistic segmentation model, we restrict the graphlet size to two and thus only binary relationships of superpixels are exploited in the normalized cut based segmentation. As shown in Figure 5, segmentation with 2-sized graphlets results in numerous over-segmented patches, because of the limited superpixel label smoothing capability of 2-sized graphlets. Beyond the analysis of the sample segmentation results, the statistics in Table 4 shows the performance degradation caused by the above component replacements, which clearly demonstrates the indispensability and inseparability of components in our approach.

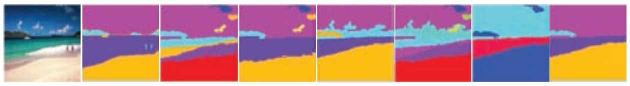


Figure 5. Example of segmentation results under functionally reduced component(First column: original photo. Second column: ground truth. Third column: superpixel→graphlet. Fourth column: non-structural graphlets. Fifth column: remove image-level labels in the embeddding. Sixth column: graphlet embedding→ kernel PCA. Seventh columun: graphlet cut→2-sized graphlets. Last column: proposed method)

4.3. Effects of maximum graphlet size

The maximum graphlet size T influences significantly the segmentation results. In Table 1, we present segmentation accuracy, time consumption corresponding to T ranging from 1 to 10, on VOC 2008 [4]. We do not experiment with T larger than 10 because the segmentation takes too long, for example, longer than one hour to segment an 1024×768 image. From Table 5, we have two observations. First, segmentation accuracy increases moderately as T goes up from 1 to 6, and remains table as T goes up further from 7 to 10. This implies that 6-sized graphlet is adequately descriptive to capture the homogeneity of superpixels. Second, segmentation time increases exponentially as the graphlet size goes up. Therefore it is better to keep T small. In our experiments, we typically set T to 6.

4.4. Segmentation results analysis

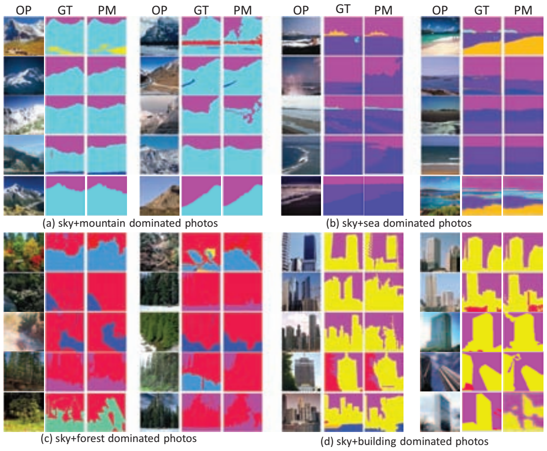
We first categorize the photos into six groups according to their semantics, and then present segmentation results of each group. As shown in Figure 6, we make the follow observations. The first observation is that, the proposed approach performs satisfactorily on sky+mountain, sky+sea, sky+forest, and sky+building dominated photos. This is because: 1) the objects are quite well semantically distributed. For example, the sun region is embedded in the sky region, the car region is contained in the road region, *etc*. Such semantic relationships are well captured by the proposed graphlets; 2) The objects in these groups are sufficiently large relative to the superpixel size, thus not many superpixels span multiple objects; 3) photos in SIFT-flow are accurately assigned with multiple image-level labels. The proposed graphlet embedding method effectively leverages the image-level labels to refine the segmentation process.

The second observation is that for the architecture+windows dominated photos, the proposed approach accurately labels the architecture area and the sky area, but fails to localize small-sized components, such as windows, cars, and pedestrians. This is because a moderate sized superpixel may contain multiple small objects. To maintain the geometric consistency, its semantic label is forced to be the surrounding one.

The third observation is that for the road+car dominated photos, the proposed approach performs acceptably. The main challenge is that the cars are too small in size, and our method only identifies a few of them.

5. Conclusions and future work

In this paper, we have presentd a weakly supervised image segmentation method by learning the distribution of s-



(a) sky+mountain dominated photos (b) sky+sea dominated photos (c) sky+forest dominated photos (d) sky+building dominated photos

Figure 6. Example segmentation results on SIFT-flow (OP: original photo, GT: ground truth, PM: proposed method)

patially structured superpixel sets. We introduced the notion of graphlet that captures the spatial structures of superpixels. To integrate image-level labels, a manifold embedding technique is proposed to transform different-sized graphlets into equal length feature vectors. The embedding allows us to use GMM to learn the distribution of the embedded graphlets, which is used to measure the homogeneity of superpixels for image segmentation. An important property of such homogeneity measure is that it takes the spatial structure of the superpixels into consideration..

In the future, we will investigate an active-learning [8]-based graphlet selection scheme to accelerate image segmentation, and a new semi-supervised [22] segmentation framework that simultaneously decomposes an image into regions and determines their semantics.

6. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (61170142), by the National Key Technology R&D Program under Grant (2011BAG05B04), by the Program of International S&T Cooperation (2013DFG12841), and by the Fundamental Research Funds for the Central Universities (2013FZA5012). M. Song is the corresponding author.

References

[1] Sungwoong Kim, Sebastian Nowozin, Pushmeet Kohli, and Chang D. Yoo, Higher-Order Correlation Clustering for Image Segmentation, *in Proc. of NIPS*, pp. 1530–1538, 2011.

[2] Pushmeet Kohli, Lubor Ladicky, Philip H. S. Torr, Robust Higher Order Potentials for Enforcing Label Consistency, *IJCV*, 82(3): 302–324 ,2009.

[3] Ce Liu, Jenny Yuen, Antonio Torralba, Nonparametric scene parsing: label transfer via dense scene alignment, *in Proc. of CVPR*, pp. 1972-1979, 2009.

[4] Everingham, M.a.V.G., L. and Williams, C. K. I. and Winn, J. and Zisserman, A., The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results., 2008.

[5] Jakob J. Verbeek, Bill Triggs, Region Classification with Markov Field Aspect Models, *in Proc of CVPR*, pp. 1–8, 2007.

[6] A. Vezhnevets and J. M. Buhmann, Towards Weakly Supervised Semantic Segmentation by Means of Multiple Instance and Multitask Learning, *in Proc of CVPR*, pp. 3249–3256, 2010.

[7] Alexander Vezhnevets, Vittorio Ferrari, Joachim M. Buhmann, Weakly Supervised Semantic Segmentation with a Multi-Image Model, *in Proc of ICCV*, pp. 643–650, 2011.

[8] Alexander Vezhnevets, Joachim M. Buhmann, Vittorio Ferrari, Active Learning for Semantic Segmentation with Expected Change, *in Proc of CVPR*, pp. 3162-3169, 2012.

[9] Alexander Vezhnevets, Vittorio Ferrari, Joachim M. Buhmann, Weakly supervised structured output learning for semantic segmentation, *in Proc of CVPR*, pp. 845–852, 2012.

[10] Rital, S., Hypergraph cuts and unsupervised representation for image segmentation, *Fundamenta Informaticae*, pp. 153–179, 2009.

[11] N. Dalal, B.Triggs, Histograms of Oriented Gradients for Human Detection, *in Proc. of CVPR*, pp. 886–893. 2005.

[12] M. Stricker, M. Orengo, Similarity of Color Images, *Storage and Retrieval of Image and Video Databases*, pp. 381–392, 1995.

[13] Xinchao Wang, Zhu Li, Dacheng Tao, Subspaces Indexing Model on Grassmann Manifold for Image Search, *IEEE T-IP*,20(9):2627–2635, 2011.

[14] Derek Hoiem Alexei A. Efros Martial Hebert, Geometric Context from a Single Image, *in Proc. of CVPR*,pp. 1–8, 2009.

[15] Shiming Xiang, Feiping Nie, Yangqiu Song, Changshui Zhang, Chunxia Zhang, Embedding new data points for manifold learning via coordinate propagation, *Knowlege and Information Systems*,19(2):159–184, 2008.

[16] Jianbo Shi and Jitendra Malik, Normalized Cuts and Image Segmentation, *IEEE T-PAMI*,22(8):888–905, 2000.

[17] G. Wahba, Computing Regularization Paths for Learning Multiple Kernels , *Advances in kernel methods, MIT Press*, pp. 69–88, 1999.

[18] J. Shotton, J. Winn, C. Rother, and A. Criminisi, TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-Class Object Recognition and Segmentation, *Proc. of ECCV*, pp. 1–15, 2006.

[19] Lubor Ladicky, C.R., Philip H.S. Torr and Pushmeet Kohli, Associative Hierarchical CRFs for Object Class Image Segmentation, *Proc. of ICCV*, pp. 739–746, 2009.

[20] Zaïd Harchaoui, Francis Bach, Image Classification with Segmentation Graph Kernels, *in Proc. of ICCV*, pages: 1–8, 2007.

[21] Yi Yang, Feiping Nie, Dong Xu, Jiebo Luo, Yueting Zhuang, Yunhe Pan, A Multimedia Retrieval Framework based on Semi-Supervised Ranking and Relevance Feedback, *IEEE T-PAMI*, 34(5), pages: 723–742, 2012

[22] Xiao Liu, Mingli Song, Dacheng Tao, Zicheng Liu, Luming Zhang, Jiajun Bu, Chun Chen, Semi-supervised Node Splitting for Random Forest Construction, *in Proc. of CVPR*, 2013.

[23] Luming Zhang, Mingli Song, Qi Zhao, Xiao Liu, Jiajun Bu, Chun Chen, Probabilistic Graphlet Transfer for Photo Cropping, *IEEE T-IP*, 21(5), pages: 2887–2897, 2013.