Integrating Local Features into Discriminative Graphlets for Scene Classification

Luming Zhang[1], Wei Bian[2], Mingli Song[1], Dacheng Tao[2], and Xiao Liu[1]

[1] Zhejiang Provincial Key Laboratory of Service Robot,
Computer Science College, Zhejiang University
{zglumg,brooksong,ender_liux}@cs.zju.edu.cn
[2] Centre for Quantum Computation and Information Systems,
University of Technology, Sydney
wei.bian@student.uts.edu.au, dacheng.tao@uts.edu.au

Abstract. Scene classification plays an important role in multimedia information retrieval. Since local features are robust to image transformation, they have been used extensively for scene classification. However, it is difficult to encode the spatial relations of local features in the classification process. To solve this problem, Geometric Local Features Integration(GLFI) is proposed. By segmenting a scene image into a set of regions, a so-called Region Adjacency Graph(RAG) is constructed to model their spatial relations. To measure the similarity of two RAGs, we select a few discriminative templates and then use them to extract the corresponding discriminative graphlets(connected subgraphs of an RAG). These discriminative graphlets are further integrated by a boosting strategy for scene classification. Experiments on five datasets validate the effectiveness of our GLFI.

Keywords: scene classification, graphlet, local features.

1 Introduction

Scene classification is an important issue for many multimedia applications, such as image retrieval and surveillance. To deal with scene classification successfully, it is essential to have proper discriminative image features. In the evolution of image analysis, many features have been proposed and they can be categorized into two groups: global features and local features. Global features, e.g., eigenspace [1], represent an image by a single vector and are hence tractable for conventional classifiers, such as Support Vector Machine(SVM) [13]. However, global features are sensitive to occlusions and clutters, which result in poor classification accuracy. In contrast to global features, local features, e.g., Scale Invariant Feature Transform(SIFT) [12], are extracted at interest points and are robust to image deformations. Different images may produce different number of local features. In order to be tractable for conventional classifiers, these local features are often integrated into an orderless bag-of-features representation. Unfortunately, as a non-structural representation, the bag-of-features representation ignores the spatial relations of local features, which prevents it from being discriminative.

B.-L. Lu, L. Zhang, and J. Kwok (Eds.): ICONIP 2011, Part III, LNCS 7064, pp. 657–666, 2011.

To encode the spatial relations of local features for scene classification, graph based local feature integration [2–7] is proposed. In [2, 3], each image is modelled as a tree and image matching is formulated into tree matching. Unfortunately, compared to general graphs, the capability of modelling regions' spatial relations by trees is limited. Felzenszwalb et al. [4] modelled the relation of different parts of an object as a spring. However, [4] relies heavily on the optimal background subtraction. In [5], Hedau et al. defined a new measure of pairwise regions based on the overlaps between regions; but just region overlaps are too simple to capture the complicated spatial relations of regions. Keselaman et al. [6] defined a graph, called Least Common Abstraction (LCA), for an object. However, LCA cannot be output to a conventional classifier, e.g., SVM [13] directly. Walk kernel [7] captures the walk structures of regions by a finite sequence of neighboring regions. Unfortunately, as demonstrated in [8], the totter phenomenon brings noise to walk kernel [7] and thus makes it less discriminative.

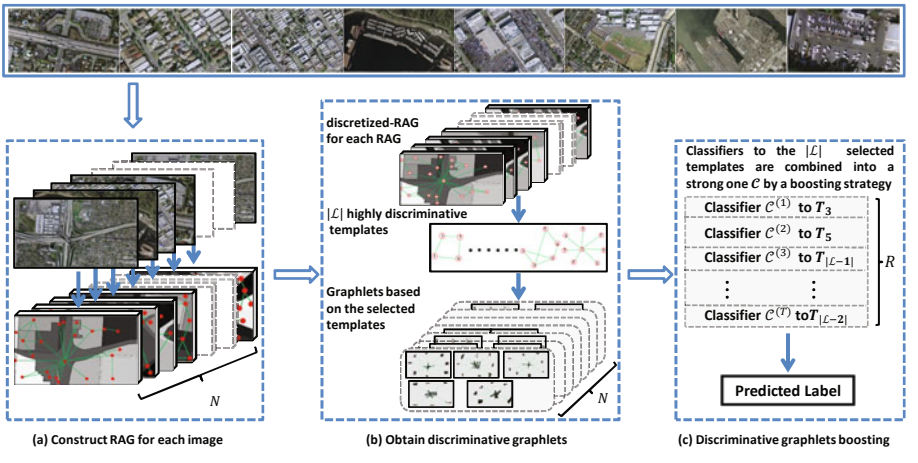

Fig. 1. The flowchart of our GLFI

To solve or at least reduce the aforementioned problems, a new local feature integration method GLFI is proposed for scene classification. As shown in Fig. 1, first of all, each scene image is segmented into a set of regions. To model the spatial relations of these regions, a graph called RAG is constructed subsequently (Fig.1(a)). Then, to measure a pair of RAGs, it is straightforward to compare all their pairwise graphlets. Unfortunately, based on graph theory, the number of graphlets of an RAG is huge, making the graphlet enumeration computational intractable. Towards an efficient measure, it is necessary to select a few discriminative graphlets for comparison. As the number of candidate graphlet for selection is huge, aiming at fewer candidates, we obtain templates by discretizing the continuous labels of graphlets into discretized ones, then only highly discriminative templates are selected and further used to extract the corresponding discriminative graphlets (Fig.1(b)). Finally, these discriminative graphlets are integrated by a boosting strategy for scene classification (Fig.1(c)).

2 Region Adjacency Graph(RAG)

A scene image usually contains millions of pixels. If we treat each pixel as a local feature, high computational cost will make scene classification computational intractable. Fortunately, a scene image can be represented by a set of clusters because pixels are usually highly correlated with their spatial neighbors, wherein each cluster consists of neighboring pixels with consistent color intensities. Thus, we propose RAG to represent a scene image by a set of regions and encode their spatial relations in a labelled graph.

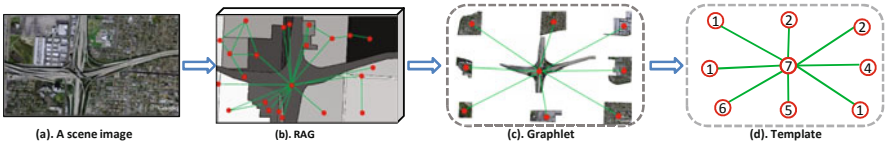

Fig. 2. The flowchart from a scene image(a) to its RAG(b) and further to its graphlet(c) and its template(d)

As shown in Fig.2(a, b), given a scene image I, we segment it into a set of regions $\{r_1, r_2, \cdots, r_M\}$ (Unsupervised Fuzzy Clustering(UFC) [17] based segmentation is applied because of its stability), and an RAG G is constructed to model a scene image I, i.e.,

$$G = (V, E, H, L, h, l) \tag{1}$$

where $V = \{v_1, v_2, \cdots, v_M\}$ is a finite set of vertices, v_i represents region r_i; $h : V \to H$ is a function assigning a label to each $v \in V$, i.e., $h(v)$ is a row vector representing the RGB histogram of the region corresponding to v; $l : V \to L$ is a function assigning an index to each vertex $v \in V$, i.e., $l(v)$ means the region corresponding to v is obtained from the $l(v)$-th segmentation(multiple segmentations are applied); $E = \{(v_i, v_j) | v_i, v_j \in V \wedge l(v_i) = l(v_j) \wedge v_i \sim v_j\}$ is a set of edges, $v_i \sim v_j$ means two regions corresponding to v_i and v_j are spatial adjacent.

As shown in Fig.2(c), given an RAG G, we call S a graphlet of G if S is a connected subgraph of G. For two graphlets S and $S^{'}$, they are isomorphic [8], denoted by $S \cong S^{'}$, if there exists a bijection $\varphi : V \to V^{'}$ such that for each $u, v \in V$,$(u, v) \in E$ iff $(\varphi(u), \varphi(v)) \in E^{'}$ and $h(u) = h(\varphi(u^{'}))$. If $S \cong S^{'}$ and $S^{'} \subseteq G^{'}$, we call S subgraph isomorphic to $G^{'}$ or, $G^{'}$ supergraph isomorphic to S, denoted by $S \sqsubseteq G^{'}$.

3 Discriminative Graphlets Selection

Based on the definition of RAG, the similarity of a pair of scene images I and $I^{'}$ depends on their corresponding RAGs G and $G^{'}$. To measure the similarity between G and $G^{'}$, it is straightforward to compare all their pairwise graphlets.

However, 1). based on graph theory, the number of graphlets of an RAG is $\mathcal{O}(M^M)$(usually $M > 50$); 2). non-discriminative graphlets make no contribution to scene classification. Therefore, it is necessary to select a few discriminative graphlets for scene classification.

Towards an efficient selection of discriminative grpahlets, a three-step method is developed: firstly, we obtain a small set of templates from the training RAGs and accordingly derive the class label of template. Then, a few discriminative templates are selected. Finally, these discriminative templates are used to extract the corresponding discriminative graphlets.

3.1 Template and Its Class Label

As shown in Fig.2(d), to obtain a template, a codebook $H^D = [h_1^D, h_2^D, \cdots, h_P^D]$ is generated by k-means [13] on all the training vertex labels firstly, then the continuous label $h(v)$ of vertex v is discretized into $h^D(v)$ by:

$$h^D(v) = \arg\min_{h \in H^D} ||h(v) - h|| \tag{2}$$

where $||\cdot||$ is the Euclidean norm. Based on H^D and h^D, given an graphlet S, we define its corresponding template T is obtained by mapping $g : S \to T$, where

$$T = \{V, E, H^D, L, h^D, l\} \tag{3}$$

Since template is a label-discretized graphlet, the number of candidate templates for selection is much smaller than that of graphlets, thus it is feasible to select a few discriminative ones for scene classification. Before selecting discriminative templates, we need to measure template's discrimination, i.e., how accuracy of a template predicting the class labels of scene images. As a label-discretized graphlet, template describes the spatial relations of local features in an approximate manner, to accurately predict the class label of template T, given an RAG G , it is necessary to find graphlets in G corresponding to T. Formally, we call graphlets S satisfying T, if $g(S) = T$, and graphlets of G satisfying T are collected into $G(T)$, i.e.,

$$G(T) = \{S | S \subseteq G \wedge g(S) = T\} \tag{4}$$

If $G(T) \neq \emptyset$, each graphlet $S \in G(T)$ can be represented as a vector $h(S)$, i.e.,

$$h(S) = \cup_{v \in S}[h(v)] \tag{5}$$

where $\cup[\cdot]$ is a row-wise vector concatenation operator.

Based on (5), given a set of training RAGs $\mathcal{G} = \{G_1, G_2, \cdots, G_N\}$ and a template T, we obtain a set of feature vectors $\mathcal{H} = \{h(S) | S \in G(T) \wedge G \in \mathcal{G}\}$, and further train a SVM classifier [13] $\mathcal{C}$ based on $\{\mathcal{H}, \mathcal{K}\}$, where $\mathcal{K}$ is the set of class labels corresponding to RAGs in $\mathcal{G}$. Based on trained SVM classifier $\mathcal{C}$, given an RAG G and a template T, the class label $k \in \{1, 2, \cdots, K\}$ of graphlet $S \in G(T)$ is obtained based on the posterior probability $P(G \to k|S)$ output from $\mathcal{C}$, i.e.,

$$S \to \arg\max_k P(G \to k|S) \tag{6}$$

Since there may be more than one graphlets in G satisfying template T, i.e., $|G(T)| \geq 1$, the label of $G(T)$ is derived from a multiple classifiers combining strategy [19] (under the sum rule), i.e., the posterior probability for $G(T)$ belonging to class $k \in \{1, 2, \cdots, K\}$ is:

$$P(G \to k|G(T)) = (1 - Z)P(G \to k) + \sum_{i=1}^{Z} P(G \to k|S_i) \tag{7}$$

where $Z = |G(T)|$; $P(G \to k)$ is the probability of RAG G belonging to class k (computed from the training RAGs), i.e.,

$$P(G \to k) = \frac{|G \to k \wedge G \in \mathcal{G}|}{N} \tag{8}$$

Based on (7), the class label of $G(T)$ is obtained by:

$$G(T) \to \begin{cases} \arg\max_k P(G \to k|G(T)) & \text{if } G(T) \neq \emptyset \\ 0 & \text{if } G(T) = \emptyset \end{cases} \tag{9}$$

where $G(T) \to 0$ means decision cannot be made on $G(T)$.

3.2 Selecting Discriminative Templates

In the extreme case, a template T is optimal if $\exists k \in \{1, 2, \cdots, K\}$, the following two conditions are satisfied:

C1: $P(G(T) \to k|G \to k) = 1$
C2: $P(G \to k|G(T) \to k) = 1$

where C1 maximize the descriptive ability of template T, and C2 maximize the discriminative ability of template T. However, as proved in [13], in the case of noisy training data, such optimal template may not always exist. Therefore, it is necessary to search for a set of sub-optimal templates, i.e., $\exists k \in \{1, 2, \cdots K\}$, such that:

C3: $P(G(T) \to k) \geq \min(P(G \to k))$
C4: $P(G \to k|G(T) \to k) \geq \alpha * P(G \to k)$

To satisfy C3, we obtain a set of discretized RAGs $\mathcal{G}^D = \{G^D|G^D = g(G) \wedge G \in \mathcal{G}\}$ based on (2), then the frequency of template T is computed by counting how many $G^D \in \mathcal{G}^D$ are supergraph isomorphic to T, i.e.,

$$P(G(T)) = \frac{|T \sqsubseteq G^D \wedge G^D \in \mathcal{G}^D|}{N} \tag{10}$$

Based on (10), the frequency of a template belonging to class $k \in \{1, 2, \cdots, K\}$ is computed by:

$$P(G(T) \to k) = \frac{|T \sqsubseteq G^D \wedge G^D \in \mathcal{G}^D \wedge G^D = g(G) \wedge G \to k|}{N} \tag{11}$$

For a template T, a larger $P(G(T) \rightarrow k)$ means T has a higher generalization ability towards class k. In our approach, an efficient frequent subgraph mining algorithm, FSG [18], is employed to output templates whose $P(G(T) \rightarrow k) \geq \min(P(G \rightarrow k))$

To satisfy C4, given a template T, its measure of discrimination is defined as largest discrimination towards class $k \in \{1, 2, \cdots, K\}$, i.e.,

$$disc(T) = \max_k \left[\frac{P(G \rightarrow k|G(T))}{P(G \rightarrow k)} \right] \tag{12}$$

where denominator is computed based on (8); the numerator is computed based on (7). Template whose $disc(T) < \alpha$ is regarded as a less discriminative one. Based on C3+C4, we present the algorithm of discriminative template selection in Table 1.

Table 1. Discriminative Template Selection(Algorithm 1)

input: A set of training data $\mathcal{D} = \{G_i, k_i\}_{i=1}^N$; Threshold α;
output: A set of discriminative template $\mathcal{L}$;

begin:
1. For each RAG G_i in $\mathcal{D}$, obtain the corresponding discretized-RAGs G_i^D and save them into $\mathcal{G}^D$;
2. Conduct FSG on $\mathcal{G}^D$ to output templates T whose $P(G(T) \rightarrow k) \geq \min(P(G \rightarrow k))$ into $\mathcal{L}$;
3. **for** each template $T \in \mathcal{L}$
   if $disc(T) < \alpha$, then $\mathcal{L} \leftarrow \mathcal{L} \setminus T$;
   end for;
Return $\mathcal{L}$;
end

3.3 Extracting Discriminative Graphlet

Each template $T \in \mathcal{L}$ (output from Algorithm 1) is discriminative. Thus given an input RAG G, we conduct depth-first-search on G, and graphlets of G satisfying T are extracted for scene classification. It is noticeable that, vertices in RAG are of low degree, i.e., less than 5 on average, so its computational is approximately linear increasing with the number of vertices in RAG G.

4 Discriminative Graphlets Boosting

To integrate the extracted discriminative graphlets for scene classification, a boosting strategy is developed. In detail, for each template $T \in \mathcal{L}$, a SVM classifier $\mathcal{C}$ is trained as described in Section 3.1. Based on $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{L}|}$, we develop a multi-class boosting algorithm to integrate the $|\mathcal{L}|$ weak classifiers $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{L}|}$ into a strong one $\boldsymbol{\mathcal{C}}$. We present the algorithm of discriminative graphlets boosting in Table 2.

5 Experimental Results and Analysis

To demonstrate the advantage of our GLFI, we experiment on five datasets: Scene15 [9], Scene67 [20], Caltech256 [14], PASCAL VOC 2009 [15] and LHI [16]. Details of the five datasets are presented in Table 3.

Table 2. Discriminative Graphlets Boosting(Algorithm 2)

input: A set of training RAGs and their corresponding labels: $\{G_j, k_j\}_{j=1}^{N}$;
A set of weak classifiers $\{\mathcal{C}_i\}_{i=1}^{|\mathcal{C}|}$; Iteration number of boosting R;
output: A strong classifier: $\boldsymbol{\mathcal{C}}(G)$;

begin:
1. Set the training RAG weights $w_j = \frac{1}{N}$, $j = 1, 2, \cdots, N$;
2. **for** $t = 1, 2, \cdots, R$
(a).Select a weak classifier $\mathcal{C}^{(t)}$ from $\{\mathcal{C}_i\}$: $\arg\min_{\mathcal{C}^{(t)} \in \{\mathcal{C}_i\}} \sum_{j=1}^{N} w_j \cdot \prod(G_j(T_i) \nrightarrow k)$;
(b).Compute weighted training error: $err^t = \frac{\sum_{j=1}^{N} w_j \cdot \prod(G_j(T_i) \nrightarrow k)}{\sum_{j=1}^{N} w_j}$;
(c). $a^t \leftarrow \log \frac{(1-err^t)}{err^t} + \log(K-1)$;
(d). Update the training RAG weight: $w_j \leftarrow w_j \cdot \exp[a^t \cdot \prod(G_j(T) \nrightarrow k)]$;
(e). Re-normalize w_j;
end for;
Return $\boldsymbol{\mathcal{C}}(G) = \arg\max_k \sum_{t=1}^{T} a^t \cdot \prod(G(T_i) \rightarrow k)$;
end

Table 3. Details of the five datasets

Dataset	# of categories.	# of images.	# of training images	# of test images
Scene15	15	4485	100 per category	rest per category
Indoor67	67	15620	80 per category	20 per category
Caltech256	256	30,607	50 per category	rest per category
VOC2009	20	14,743	7,054	7,689
LHI	5	20	N/A	N/A

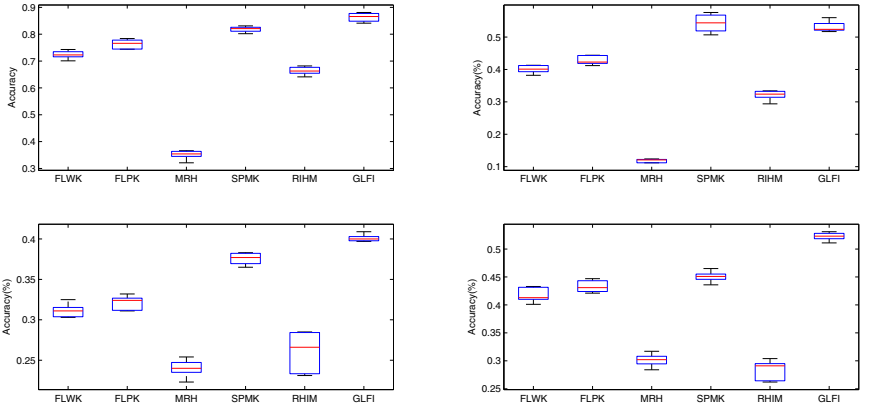

Fig. 3. Classification accuracy of the compared methods on Scene15(top left), Scene67(top right), Caltech256(bottom left) and PASCAL VOC 2009(bottom right)

5.1 GLFI versus Representative Local Features Integration Methods

In Fig. 3, we compare our GLFI with five representative local feature integration methods, i.e., fixed length walk kernel(FLWK) [7], fixed length path kernel

(FLPK) [8], multiresolution histogram(MRH) [10], spatial pyramid matching kernel(SPMK) [9] and region-based hierarchical image matching(RHIM) [2]. The experimental settings are as follows: the lengths of FLWK [7] and FLPK [8] are tuned from 2 to 10; for MRH [10], we smooth images with RBF kernels of 15 gray levels; for SPMK [9], each image is decomposed into over 1 million SIFT [12] features of 16×16 pixel patches computed over a grid with spacing of 8 pixels, then a codebook of size 400 is generated by k-means [13]; for our GLFI, the times of multiple segmentations, $\max(L)$, is tuned from 2 to 7, and the iteration number of boosting, R, is set to 200.

In Table 4, we present the classification accuracy of each category on PASCAL VOC 2009 . As seen, our GLFI outperforms the three compared graph based local feature integration methods significantly on most categories, which is consistent with our theoretical analysis in Section 1.

Table 4. Averaged classification accuracy of 20 categories on PASCAL VOC 2009(%)

	aero	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
FLWK	72.2	40.6	41.2	42.1	23.9	56.6	39.8	44.3	47.2	20.2
FLPK	73.1	42.8	44.3	50.4	22.7	57.1	41.2	43.9	43.5	22.3
RHIM	60.8	22.1	25.3	33.2	11.3	34.6	30.1	26.3	30.2	13.2
GLFI	**75.6**	**54.1**	**60.6**	**58.2**	**33.4**	**65.2**	**56.5**	**56.4**	**48.5**	**37.7**
	dining	dog	horse	motor	person	potted	sheep	sofa	train	tv
FLWK	32.3	33.1	42.2	44.3	**76.6**	27.3	30.9	26.7	63.8	44.4
FLPK	33.7	34.4	44.5	44.5	73.2	**29.6**	**32.1**	28.4	65.3	46.7
RHIM	13.4	22.1	26.4	25.4	56.8	9.6	17.6	10.2	44.8	30.1
GLFI	**47.7**	**43.2**	**60.2**	**63.2**	74.6	29.4	31.3	**40.2**	**77.3**	**51.1**

5.2 Influence of Different Segmentation Settings

In retrospect to the proposed GLFI, we notice that the influence of segmentation operation in the construction of RAG is nonnegligible. To evaluate scene classification under different segmentation settings, based on (12), we report the frequent template's(output from Step2 of Algorithm 1) meausre of discrimination under benchmark-segmentation, deficient-segmentation, and over-segmentation. We experiment on PASCAL VOC 2009 [15] beacause its segmentation benchmark is helpful to make a precise comparison.

As shown in Fig. 4, templates from benchmark-segmentation achieves the highest discrimination, with the highest *disc* value of 35.4, followed by the over-segmentation 33.7 and deficient-segmentation 31.2. The explanations are as follows: 1).the benchmark segmentation is obtained by manually annotation, which encodes the high-level semantic understanding, thus it is unavoidable that UFC [17] may be less accurate than the benchmark segmentation; 2).in contrast with deficient-segmentation, more regions are obtained in over-segmentation setting, so it is rarer for one region spans several components, fewer discriminative components are neglected.

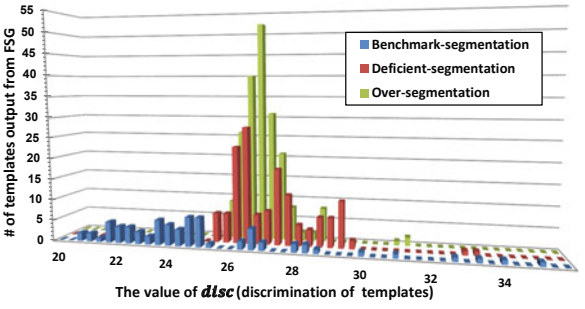

Fig. 4. disc value of templates under 3 different segmentation settings

5.3 Visualization of the Discriminative Graphlets

A unique property of our GLFI is the "transparency" of the scene classification model. As shown in Fig. 5, we visualize the the most discriminative graphlets of aerial images in LHI [16]. As seen, discriminative graphlets from different categories have different structure pattern, which further validates the intuition of our GLFI.

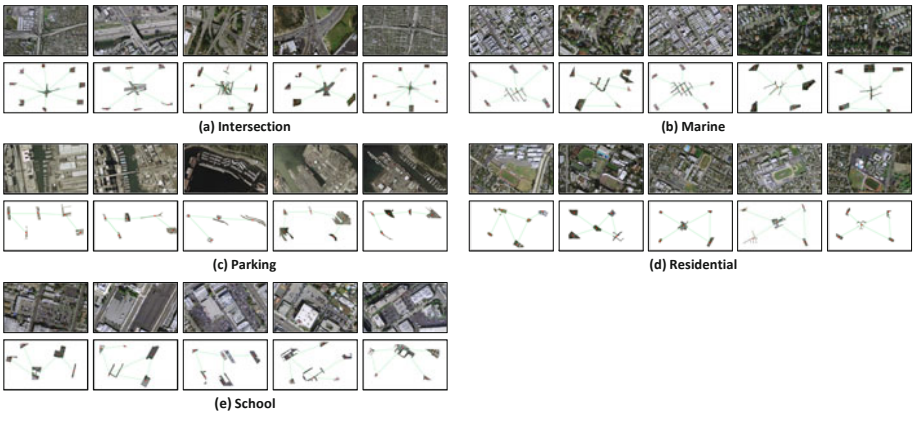

Fig. 5. Visualized discriminative graphlets

6 Conclusions

In this paper, a new local feature integration method GLFI is proposed for scene classification. First, an RAG is constructed to encode the geometric property and color intensity distribution of scene image. Then, the discriminative graphlets are selected from the RAGs. Finally, these discriminative graphlets are integrated by a boosting strategy for scene classification. Extensive experiments on five datasets validate the effectiveness of our GLFI.

Acknowledgments. This work is supported by National Natural Science Foundation of China (60873124), Program for New Century Excellent Talents in University (NCET-09-0685), and the Natural Science Foundation of Zhejiang Province (Y1090516).

References

1. Yuan, X., Zhu, H., Yang, S.: IEEE Workshop on Applications of Computer Vision and the IEEE Workshop on Motion and Video Computing, pp. 54–59 (2005)
2. Todorovic, S., Ahuja, N.: Region-based hierarchical image matching. IJCV (2007)
3. Demirci, et al.: Object recognition as many-to-many feature matching. IJCV 69(2) (2006)
4. Felzenszwalb, et al.: Pictorial structure for object recognition. IJCV 61(1) (2005)
5. Hedau, V., et al.: Matching images under unstable segmentations. In: CVPR (2008)
6. Keselman, Y., et al.: Generic Model Abstraction from Examples: TPAMI, 1141–1156 (2005)
7. Harchaoui, Z., Bach, F.: Image Classification with Segmentation Graph Kernels. In: CVPR, pp. 1–8 (2007)
8. Sherashidze, N., et al.: Efficient Graphlet Kernels for Large Graph Comparison. In: International Conference on Artificial Intelligence and Statistics, pp. 488–495 (2009)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: ICCV, pp. 2169–2178 (2006)
10. Hadjidemetriou, E., et al.: Multiresolution Histograms and Their Use for Recognition. TPAMI, 831–847 (2004)
11. Cao, L., Fei-fei, L.: Spatially Coherent Latent Topic Model for Concurrent Segmentation and Classification of Objects and Scenes. In: ICCV, pp. 1–8 (2007)
12. Porway, J., Wang, K., Yao, B., Zhu, S.C.: Scale-invariant shape features for recognition of object categories, 90–96 (2004)
13. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley Interscience (2000)
14. Griffin, G., Holub, A., Perona, P.: (2007), `http://authors.library.caltech.edu/769`
15. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: (2009), `http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html`
16. Yao, B., Yang, X., Zhu, S.-C.: Introduction to a Large Scale General Purpose Ground Truth Dataset: Methodology, Annotation Tool, and Benchmarks. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) EMMCVPR 2007. LNCS, vol. 4679, pp. 169–183. Springer, Heidelberg (2007)
17. Xiong, X., Chan, K.L.: Towards An Unsupervised Optimal Fuzzy Clustering Algorithm for Image Database Organization. In: ICPR (2000)
18. Kuramochi, M., Karypis, G.: An Efficient Algorithm for Discovering Frequent Subgraphs. TKDE, 1038–1051 (2004)
19. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On Combining Classifier. TPAMI, 226–239 (1998)
20. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR (2009)