Graph Commute Times for Image Representation

Régis Behmo[1,2] Nikos Paragios[1] Véronique Prinet[2]

[1]MAS, Ecole Centrale Paris,
Grande Voie des Vignes,
F-92295 Châtenay-Malabry Cedex, France
http://www.mas.ecp.fr/

[2]NLPR/LIAMA, Institute of Automation,
Chinese Academy of Sciences,
P.O Box 2728, Beijing 100190, China
http://kepler.ia.ac.cn/

Abstract

We introduce a new image representation that encompasses both the general layout of groups of quantized local invariant descriptors as well as their relative frequency. A graph of interest points clusters is constructed and we use the matrix of commute times between the different nodes of the graph to obtain a description of their relative arrangement that is robust to large intra class variation.

The obtained high dimensional representation is then embedded in a space of lower dimension by exploiting the spectral properties of the graph made of the different images. Classification tasks can be performed in this embedding space. We expose classification and labelling results obtained on three different datasets, including the challenging PASCAL VOC2007 dataset. The performances of our approach compare favorably with the standard bag of features, which is a particular case of our representation.

1. Introduction

The progress that has been made in the field of content-based image retrieval and object recognition during the last decade is significant. However, a widely usable solution to these problems is far from being established. Actual state-of-the-art approaches do not scale well to large numbers (tens of thousands) of object classes. One can think of a number of professional end users, such as press agencies, marketing companies or spatial data analysts, for whom both the objects contained in the image, their layout as well as their posture are of interest. Addressing all these challenges together is a difficult task.

Graph representations of visual data were quite popular at the early stages of statistical pattern recognition and computer vision. Despite their ability to describe relatively complex interactions between groups of data, represented by nodes, with a variable degree of precision the problems they raise worked against their favour and they were soon largely abandoned. The pioneer work of [20] represents shape parts of object models as graph nodes and their arrangement as edge attributes, and notes the difficulty of matching "networks" to one another. This is one of the issues that spontaneously arise when we want to compare graphs. In [10] node-to-node graph matching is modelled as an energy minimisation discrete problem and is solved by continuation. However, the $O(N^4)$ complexity of the algorithm, when N is the number of nodes, makes it intractable for real-life problems where N is of the order of a few thousands. In fact, the practical issue of computational intractability is recurrent when graphs are involved and a number of methods [14],[12] based on properties of the transition matrix, of dimension $N(N+1)/2$, however successful on synthetic data, simply cannot be efficiently implemented on standard actual computers.

From the point of view of the possible applications to computer vision, some of the most promising results in graph theory concern the spectral properties of graphs [4],[5]. In [19], graph nodes are clustered by the Laplacian cut and measures on these clusters are used to classify satellite images according to the degree of urbanization. The relationship between the graph Laplacian and the commute times between nodes [3] also opens interesting perspectives, and applications to image segmentation, video tracking [18] and dimensionality reduction [5],[16] have been demonstrated.

The state of the art of image representation is dominated by the bag of words paradigm [6],[8]: local features of interest, also called keypoints, are extracted from the image and quantized to form an histogram of codebook entries. The resulting representation discards the information relative to the spatial organisation of the keypoints and is therefore very robust to intra-class variability. The discriminative power of the interest points and the quality of the codebook ensure the ability to differentiate image classes [17],[21].

In [11], the spatial organization of the keypoints is taken into account by constructing a multi-scale bag of keypoints representation. The image representation provided by [1] contains properties of local parts as well as the spatial relations between parts. In these two examples results show that incorporating a certain level of information concerning the image layout in the representation can increase image labelling performance.

We propose in this paper a new image representation which contains information concerning both the appearance and layout of the image content. This representation is based on statistical properties of graphs of interest points. It intrinsically encodes in a loose manner the spatial arrangement of the interest points as well as their frequency relatively to a descriptor vocabulary codebook. In this respect our approach encompasses the bag of features representation.

The remainder of this paper is organized as follows: we present graph commute times and the associated results for dimensionality reduction in section 2. We then use these results in two different contexts: first, we introduce a novel high dimensional image descriptor based on commute times between groups of keypoints in the image (Section 3). Second, we classify images according to their new representation by embedding a graph of images in a new space, thereby assigning a descriptor of low dimension to each image (Section 4). Finally, we present labelling results and compare them to the state of the art in section 5, while discussion is the last section of our paper.

2. Commute times in a graph

Graph-based representations have been considered in the context of computer vision: a practical way to describe the structure of a graph is to use the matrix of distances between graph nodes. Distances between nodes are frequently defined as the length of the shortest path that separates them. However, in problems where the presence or the accuracy of graph nodes is uncertain, as it will be the case here, the shortest path distance lacks robustness and does not provide any statistical information about the structure of the graph. In this respect the notion of *commute times* between graph nodes is preferable.

Let us denote $\Gamma = (V, E, \Omega)$ a weighted graph where V is the set of vertices, or nodes, indexed by $i \in \{1 \dots N\}$, $E \subset V \times V$ is the set of weighted edges and Ω is the $N \times N$ weighted symmetric adjacency matrix:

$$\Omega(i,j) = \begin{cases} w(i,j) & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $w(i,j) = w(j,i)$ is the weight of edge (i,j). Given $1 \leq i_0 \leq N$, we define the random walk $(Y_n)_{0 \leq n}$ started at i_0 as follows:

$$Y_0 = i_0, \quad (2)$$

$$P\left[Y_{n+1} = j \,|\, Y_n = i\right] = \begin{cases} \frac{w_{ij}}{d_i} & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $d_i = \sum_{k=1}^{N} w_{ik}$ is the degree of node i. The *hitting time* $HT(i,j)$ is defined as the average number of steps of the random walk (Y_n) started at node i required to reach node j for the first time, and the *commute time* is the "symmetrized" hitting time:

$$HT(i,j) = E\left[min\left\{n : Y_n = j\right\} \,|\, Y_0 = i\right] \quad (4)$$

$$CT(i,j) = HT(i,j) + HT(j,i) \quad (5)$$

Note that the commute time is a metric and that it can take infinite values if the graph is not connected. It has been shown ([3], see [18] for a summary) that the commute time matrix CT can be expressed as a function of the eigenvectors and eigenvalues of the normalised Laplacian of the graph which is defined as the $N \times N$ matrix $\mathcal{L}$:

$$\forall i,j \in [1,N],\ \mathcal{L}(i,j) = \begin{cases} 1 - \frac{w_{ii}}{d_i} & \text{if } i = j \\ \frac{-w_{ij}}{\sqrt{d_i d_j}} & \text{if } i \neq j \end{cases} \quad (6)$$

We denote $(\phi_1, \dots \phi_N)$ the eigenvectors of $\mathcal{L}$ associated to the eigenvalues $(\lambda_1, \dots \lambda_N)$. We can demonstrate (see appendix A) that the eigenvalues of $\mathcal{L}$ are non-negative. We consider the case when the graph is connected, i.e: for every pair of nodes there exists a path that links them. In this case, it has been proven (see [4]) that there is only one zero eigenvalue. We can thus write $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_N$. It has been proved [3],[18] that the elements of the commute time matrix can be expressed as follows:

$$\forall i,j,\ CT(i,j) = vol \sum_{k=2}^{N} \frac{1}{\lambda_k} \left(\frac{\phi_k(i)}{\sqrt{d_i}} - \frac{\phi_k(j)}{\sqrt{d_j}} \right)^2 \quad (7)$$

$$\text{with } vol = \sum_{k=1}^{N} d_k \quad (8)$$

where $\phi_k(i)$ denotes the i^{th} coordinate of eigenvector k.

Thus, the only operation required to compute the commute time matrix is the extraction of the eigenvectors and eigenvalues of $\mathcal{L}$. We will use these results in section 3 to obtain a representation of the spatial layout of the interest points in the image, and thus a representation of the image structure.

As emphasized by [16], it is possible to view the eigenvectors (ϕ_k) of $\mathcal{L}$ as functions on the vertices of the graph. In this light, equation 7 can thus be considered

as an L^2 distance function between vectors of coordinates $\sqrt{\frac{vol}{d_i}}\left(\frac{\phi_2(i)}{\sqrt{\lambda_2}}\cdots\frac{\phi_N(i)}{\sqrt{\lambda_N}}\right)$. In equation 7 we can neglect the terms corresponding to high eigenvalues (low values of $\frac{1}{\lambda_k}$) and obtain an embedding of the graph nodes in a space of arbitrary dimension inferior to N. The sharper the increase of the sequence $(\lambda_k)_{1<k\leq N}$ the better the approximation.

In section 4 we will apply this method to the dimensionality reduction of the image representation.

3. Image representation

Our image representation consists in computing properties of a graph built on interest points of the image; interest points are first collected in the image to constitute the set of nodes of what we call a "feature graph". Then the corresponding "collapsed graph" is built by associating each node of the feature graph to an entry of a descriptor codebook. Finally the symmetric matrix of commute times between the nodes of this collapsed graph is computed to obtain the final image representation which encodes both the relative frequency of the codebook entries to which the features are associated, as in the bag of features representation, as well as their spatial proximity, in a sense that will be defined.

3.1. Keypoints detection and description

Here, the selection of the keypoint detection and description strategy is directly linked to the nature of the investigated data, and not to the algorithm itself. All possible combinations of feature detectors and descriptors can be used in the context of our approach, as long as it is possible to compute a distance between descriptors and to create a descriptor codebook i.e: a quantization of the descriptor space. In particular, the selection of the invariances (rotation, scale, affine, illumination, etc.) to which the detectors and descriptors are subject should be investigated in detail. To this end we forward the reader to work dedicated to comparing the performances of various feature detectors and descriptors [13],[15].

In each image a variable number N of features $(X_i)_{1\leq i\leq N}$ is collected. We define these features as: $\forall i, X_i = (x_i, y_i, \sigma_i, D_i)$, where (x_i, y_i) is the feature position in the image, σ_i its scale and D_i its descriptor vector. Moreover, we denote by k_i the index of the codebook entry associated to D_i: $k_i = argmin_{1\leq k\leq K}(||Q_k - D_i||)$, where $(Q_k)_{1\leq k\leq K}$ is our codebook.

3.2. Feature graph

We want to arrange the set of interest points extracted from the image in a graph that will preserve their general layout, the "feature graph". More precisely, we want to strongly connect nodes that are likely to belong to the same object. To this end, we consider that features belonging to the same object have close spatial positions as well as descriptor vectors. We will therefore connect graph nodes for which a certain distance Δ function of the spatial and content proximity will be small. We chose to define this distance Δ between features X_i and X_j as the weighted product of their normalised spatial distance and their descriptor distance:

$$\Delta(X_i, X_j) = \Delta_{desc}(X_i, X_j)^{\alpha}\Delta_{geo}(X_i, X_j)^{1-\alpha} \quad (9)$$

$$\Delta_{desc}(X_i, X_j) = ||D_i - D_j|| \quad (10)$$

$$\Delta_{geo}(X_i, X_j) = \sqrt{\frac{(x_i-x_j)^2+(y_i-y_j)^2}{\sigma_i\sigma_j}} \quad (11)$$

Parameter α can be adjusted to construct feature graph that depend more or less on the spatial layout and the descriptors similarity. Its optimal values will depend on the image classes (see section 5). Naturally the definition of Δ_{desc} depends on the type of features and could be chosen to be a sum of squared differences or a χ^2 distance for instance; see [21] for a performance review of the different possible distances. Moreover, we should note that the definition of Δ proposed in this paper can be amended to encode other types of distances between features as well.

The Δ distance[1] will be used to determine the presence of edges between graph nodes as well as their weight: we connect each node to its M closest neighbours and each edge weight is defined as $w(i,j) = e^{\frac{-\Delta(X_i,X_j)}{\sigma}}$, where σ is a normalisation factor chosen appropriately (in practice σ depends only on the descriptor distance Δ_{desc}). It should be noted that each node is connected to *at least* M other nodes: see figure 2 for an illustration.

A feature graph is represented in figure 1: the graph nodes have been embedded in a space of dimension 3 according to section 2. The difference in colour of the nodes belonging to the same object (e.g: water) can be explained by the fact that $\alpha \neq 0$.

The feature graph in itself can hardly be used to describe the image for various reasons. In particular, the feature graph representation is not unique because the order of the interest points is arbitrary, so any representation based on the transition matrix or the commute time matrix can be ruled out. Also, if we decided to rely on the graph representation to compare images we would have to use graph matching techniques, such as [12], which quickly become intractable when it comes to graphs containing thousands of nodes, as it is the case here. Still, the information contained in the commute times matrix is a powerful description of the structure of the feature graph, thus of the image itself. Consequently, we will base our representation on the matrix of commute times of a normalised graph, in which each node represents a cluster of similar features.

[1] Δ does not satisfy the conditions to be a metric, but for its conveniency we shall nonetheless use the term "distance".

Figure 1. The nodes of the feature graph are embedded in $\mathbb{R}^3$ following the approach described in section 2, with $\alpha = 0.5$ and $M = 10$. They are represented here as (R,G,B) values. Parts of the graph between which commute times are high have very different colours. Graph edges are not shown for the sake of readability. (best viewed in colours) [9]

3.3. Collapsed graph

The matrix of commute times of the feature graph cannot be used as an image descriptor, but it is possible to compute instead the matrix of commute times between *groups* of interest points, where each group corresponds to a codebook entry. In this perspective, we would compute the commute times between the distributions of nodes θ_k, $k \in [1, K]$, with:

$$\forall i \in [1, N], \theta_k(i) = \begin{cases} \frac{d_i}{vol_k} & \text{if } k_i = k \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

$$vol_k = \sum_{\substack{i=1 \\ k_i=k}}^{N} d_i \quad (13)$$

Given a random walk (Y_n) started at a random node Y_0 following distribution θ_k, we want to determine the average number of steps required to reach a point of $\theta_{k'}$ and come back to θ_k for the first time. This comes down to computing the hitting time:

$$HT(\theta_k, \theta_{k'}) = E\left[\min\left\{n : \theta_{k'}(Y_n) \neq 0\right\} \mid Y_0 \sim \theta_k\right] \quad (14)$$

The resolution of this problem with the Laplacian of the graph is a difficult problem and, to our knowledge, there exists no closed form solution that can be implemented in a computationally feasible way. However, we can approximate the commute times between distribution of nodes by computing the commute times between the nodes of the collapsed graph Γ_c. Γ_c is the graph that is obtained by grouping ("collapsing") the nodes of Γ associated to the same codebook entry into a single node (see figure 2). The collapsed graph contains thus K nodes (where K is the size of the codebook) and we define the weight $\omega_{kk'}$ of the edge between nodes k, k' as:

$$\omega_{kk'} = \sum_{\substack{i=1 \\ k_i=k}}^{N} \sum_{\substack{j=1 \\ k_j=k'}}^{N} w_{ij} \quad (15)$$

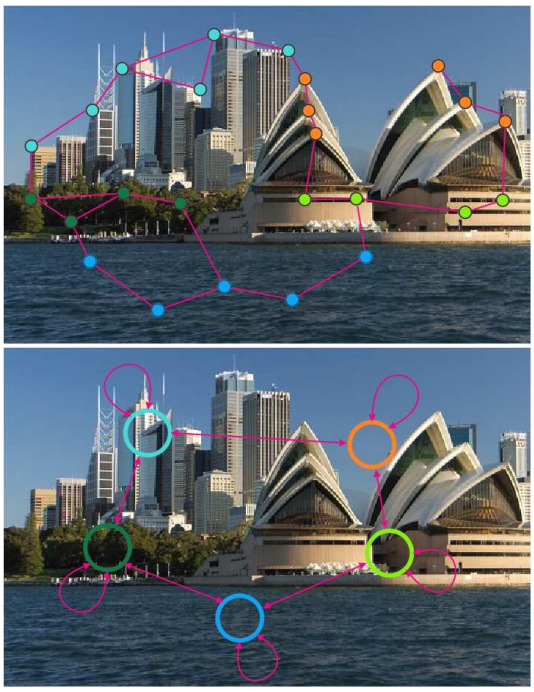


Figure 2. Toy example of collapsed graph, with parameters $N = 25$, $M = 2$, $K = 5$. (best viewed in colours)

The idea underlying the collapsed graph is to describe the proximity of image regions: how can we represent the fact that roads frequently stretch through urban areas in satellite images, or that water features often lie between sand and sky features in pictures of coastal scenes? We measure this notion of proximity by computing the commute times between different groups of features, each group containing features that were assigned to the same codebook entry of the features vocabulary.

Once the collapsed graph has been built we compute the commute times between each pair of nodes as described in section 2. The hypothesis required to obtain result illustrated by equation 7 is that the graph (in our case the collapsed graph) should be connected. If this is not the case, we first compute the commute times between nodes belonging to common connected components of the collapsed graph. Then the commute times between nodes belonging to different connected components is set to infinity. Moreover, the commute time from one node to itself is set to 0 if it is present in the collapsed graph, and infinity otherwise.

Figure 3 shows the relationship between the commute times computed in the collapsed graph and the experimental commute times between the distribution of nodes associated to the different codebook entries in the feature graph. The correlation is not linear but suffices to justify our approximation

The $K \times K$ symmetric matrix CT_c of commute times of the collapsed graph is a representation of the image that we

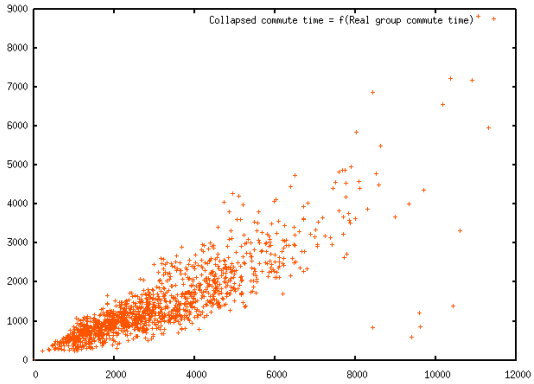


Figure 3. Commute times in the collapsed graph as a function of the empirically measured commute times between node groups in the feature graph. Commute times measures can be acquired by randomly walking on the graph.

normalise to obtain the final representation χ:

$$\forall k, k' \in [1, K]\,,\ \chi(k, k') = exp\left(\frac{-CT(k, k')}{K}\right) \quad (16)$$

This normalisation is done in order to obtain a consistent representation for which: $\chi(k, k') = 0$ if no feature is associated to the codebook entries k or k'; $\chi(k, k) = 1$ if codebook entry k exists in the collapsed graph.

Therefore, if the feature graph is entirely disconnected, the matrix χ contains only zero values except on its diagonal, where $\chi(k, k) = 1$ if at least one feature is assigned to codebook entry k. In this particular case our image representation is equal to a binary bag of features. This will allow us to compare our image representation to the bag of features representation simply by changing the value of M (Section 5).

It should be noted that despite its high dimensionality, χ can be made memory-efficient by taking into account its sparsity since in practice it often contains less than 10% of non-zero values. We will nonetheless reduce the dimensionality of χ in order to proceed to the classification step.

4. Dimensionality reduction and image classification

We apply the dimensionality reduction method described in section 2 to a fully connected graph in which the nodes are the image descriptors (χ_i) and the weight μ_{ij} between two nodes (i, j) is a function of the proximity of the image descriptors:

$$\forall i, j, \mu_{ij} = exp\left(\frac{-1}{\tau}\sum_{p,q=1}^{K}\frac{|\chi_i(p, q) - \chi_j(p, q)|}{\chi_i(p, q) + \chi_j(p, q)}\right) \quad (17)$$

where τ is a normalisation factor appropriately selected (for instance the mean proximity between image descriptors). The output of this step is a set of points in a low dimensional space, each one of them corresponding to an image. Of course, this dimensionality reduction method can be applied to any kind of high dimensional image representation or data description.

5. Image classification and labelling results

We evaluate the quality of our image representation by trying to complete classification and labelling tasks on three different datasets. Each dataset is equally divided in a training set and a test set and performance is reported on the test set only. The datasets are detailed below.

The **high resolution satellite image dataset** is composed of 128 images containing roads and 103 images of vegetation. The 60 cm panchromatic images were acquired by satellite Quickbird in the area of Beijing, China.

The **indoor scene dataset** is a subset of the Fei-Fei & Perona [8] dataset. It is composed of 930 images belonging to one of four classes: bedroom, kitchen, living room and office. The training and testing sets contain respectively 464 and 466 images. Each test image has to be classified in one of the four classes.

The **vehicle dataset** is a subset of the PASCAL VOC2007 classification challenge dataset [7]. It contains 1331 challenging images coming from the aeroplane, boat, bicycle, bus, motorbike, and train classes: these images display a high intra class variability and heavy background clutter. Object instance(s) can come from one or more classes in each image.

In our experiments we use scale and rotation-invariant SURF (Speeded Up Robust Features, [2]) of dimension 64. The number of points extracted with this detector is typically of the order of $1000 - 2000$. The features quantization can be done in various ways [17] but we chose to use simple k-means on the set of features collected in 10% of the training images. Bag of features approaches have been shown to be more efficient with codebook sizes of the order of a thousand [17]; however, we have seen that if K is the codebook size the dimension of the image descriptor will be $K(K + 1)/2$. Thus a value of $K = 500$ seems to be a good compromise between dimensionality and computational tractability.

The dimensionality of the representations is reduced to 20. Images are then assigned real valued predictions for each class by summing one versus one linear SVM contributions. In the classification task each image is assigned to the class for which the prediction is highest. In the labelling task each image has a real valued prediction for each class and receiver operator characteristic (ROC) curves can be drawn by changing the threshold on the prediction values.

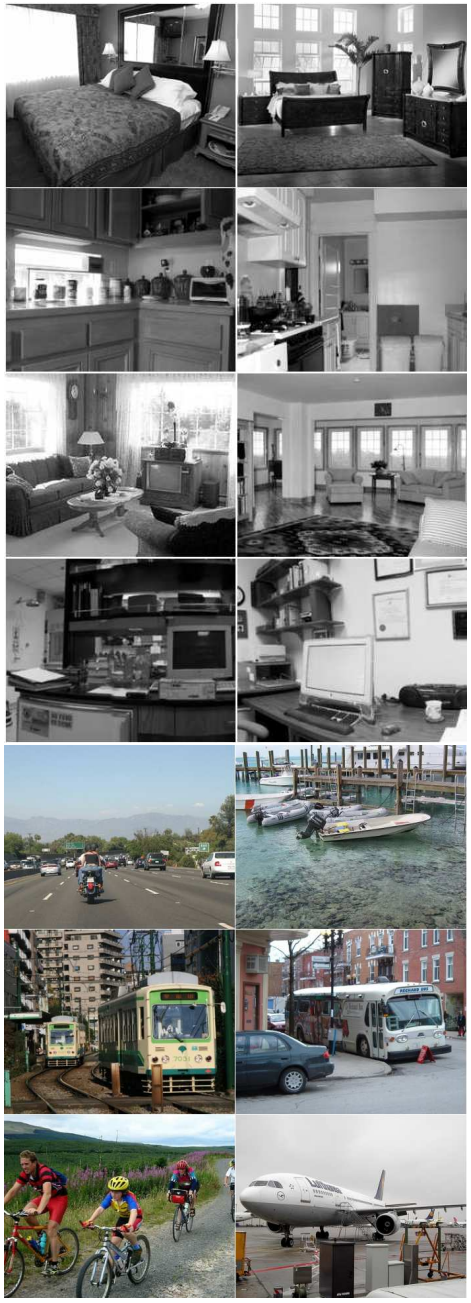


Figure 4. Sample images from the Fei-Fei & Perona indoor scene classes dataset [8] and the vehicles subset of the PASCAL VOC2007 challenge [7]

5.1. Parametric evaluation and comparison with the bag of features

We first validate our approach by computing our image representations on the **high resolution satellite image dataset**. The parameters are assigned default values: $\alpha = 0.5$, $M = 4$. After the dimensionality reduction embedding we can plot the first two coordinates of the representation. As we can see (figure 5) the separation between road images and vegetation images is sharp. In fact, the images that lie at the border of the separation are ambiguously or incorrectly annotated: they are either road images containing a lot of vegetation or vegetation images containing relatively small roads or straight paths. Correct classification rates are nonetheless very good: 93.20% and 97.30% for the vegetation and road images respectively.

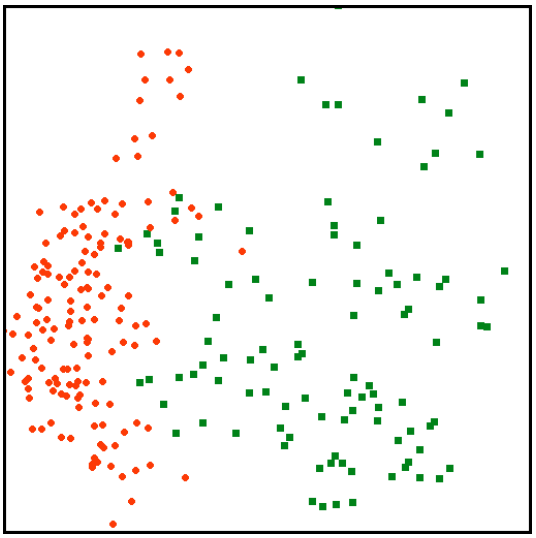


Figure 5. High resolution satellite image database: vegetation images (green squares) and road images (orange circles) after embedding of the image representations in two dimensions.

The quantitative contribution of our approach can be observed in the classification results of the **indoor scene dataset** as a function of M, the minimum number of connections per node in the feature graph (see figure 6). The value $M = 0$ corresponds to a binary histogram of quantized local descriptors, aka: the bag of features representation. As M is increased the feature graph becomes more connected and the information due to the layout of the different groups of nodes gains greater importance in the image representation (outside the diagonal of χ) relatively to the histogram of quantized features (diagonal of χ). What we observe is that an increase of M causes variations in the classification performances. These variations can be positive or negative, depending on the classes and the value of M. This reveals two phenomena: first, it shows that taking into account the image layout can raise the ambiguity between image classes that have similar bag of features representations (see bedroom and office classes). For the two others (kitchen and living room) adding spatial information only increases confusion: the content of these images is spatially too chaotic and our image representation is an overkill compared to the simple bag of features. Second, the extent

Class	$M=0$	$M=4$
Aeroplane (126)	0.855	0.855
Bicycle (127)	0.722	0.743
Boat (100)	0.752	0.762
Bus (89)	0.722	0.726
Motorbike (125)	0.813	0.842
Train (134)	0.749	0.786

Table 1. Area under curve (AUC) scores of the vehicle classes from the PASCAL VOC2007 challenge for $M=0$ (binary bag of features) and $M=4$.

to which the proximity between image regions should be taken into account varies between classes: a low value of M means that only the interactions between regions that are both spatially close and very similar will be integrated into the image representation.

The influence of parameter α in the construction of the feature graph can be seen on figure 6. A value of $\alpha=1$ means that connection between interest points will depend only on the similarity of their descriptors: this leads to feature graphs containing several disconnected subgraphs in which the most similar interest points tend to be grouped. On the contrary, a value of $\alpha=0$ means that only the spatial organisation of the interest points will decide on the connections of the feature graph. Again, this quantitative comparison shows that capturing the information of the layout of the interest points is not evenly important for all image classes. Adjusting the α parameter can lead to substantial performance gain but is not critical. For the kitchen class, these measures confirm the previous observation that adding information about the spatial organisation in the image representation is superfluous.

Finally we tested our approach on the challenging **vehicle classes** of the PASCAL VOC2007 challenge. The addressed task is that of labelling, since objects coming from different classes can be contained in each image. Our approach with parameters $\alpha=0.5$ and $M=4$ compares well with the binary bag of features: the area under curve (AUC) of the ROC is increased by up to 3.7 points in the case of the train class. Admittedly, this increase is not major: this is due to the fact that our representation is better suited for image classification than for labelling, as it cannot cope with the localisation of objects in images.

6. Conclusions

In this paper we have presented a new compact, self-contained representation of image content. It is based on the commute times between groups of descriptors and encompasses both geometry and content. Such representation adapts naturally to the observed image content and can be used for indexing, retrieval or recognition. It can also be adapted to specific tasks by properly choosing the interest point detectors and descriptors, using a customised definition of the weights and the connections of the feature graph, and setting appropriate values for the parameters α and M.

Understanding the relationship in a formal way between the full and the collapsed graph in terms of commute times is an ongoing research effort towards the solidification of this new image representation. Such an action will lead toward the representation of parts of images in the purpose of completing object localisation tasks.

A. Spectrum of the normalised Laplacian matrix

The normalised Laplacian $\mathcal{L}$ can be written as a function of the un-normalised Laplacian L:

$$\forall i,j,\ L(i,j) = \begin{cases} d_i - w_{ii} \text{ if } i=j \\ -w_{ij} \text{ otherwise} \end{cases} \quad (18)$$

$$\mathcal{L} = T^{-1/2} L T^{-1/2} \quad (19)$$

Since L is symmetric its eigenvalues are real. Moreover, L is (weakly) diagonally dominant: $\forall i, |L(i,i)| = \sum_{j\neq i} |L(i,j)|$. Gershgorin's circle theorem states that all eigenvalues of L are non-negative and therefore L is positive semi-definite. Thus $\mathcal{L}$ is positive semi-definite as well because the d_i are non-zero in a connected graph. We also know that the eigenvalues of $\mathcal{L}$ are real because $\mathcal{L}$ is symmetric, so they are non-negative.

References

[1] J. Amores, N. Sebe, and P. Radeva. Context-based object-class recognition and retrieval by generalized correlograms. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 2007. 2

[2] B. Bay, T. Tuytelaars, and L. J. Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, 2006. 5

[3] F. Chung and S. T. Yau. Discrete green's functions. *J. Comb. Theory Ser. A*, 2000. 1, 2

[4] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997. 1, 2

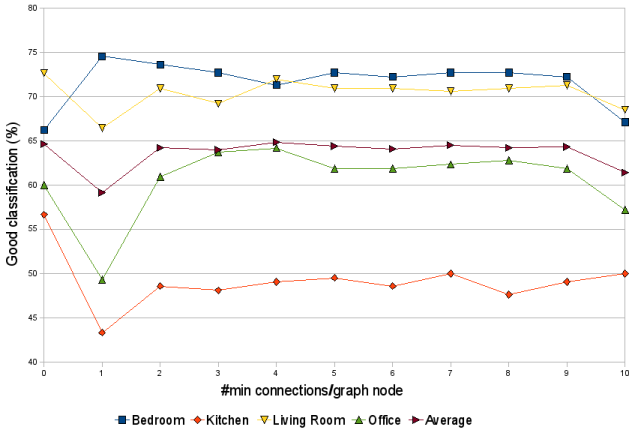
[5] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis: Special issue on Diffusion Maps and Wavelets*, 2006. 1

[6] G. Csurka, C. Dance, J. Willamowski, L. Fan, and C. Bray. Visual categorization with bags of keypoints. In *European Conference on Computer Vision (ECCV) International Workshop on Statistical Learning in Computer Vision*, 2004. 1

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 5, 6

M	Bedroom (108)	Kitchen (105)	Liv.Ro. (145)	Office (108)	Average (466)
0	66.2	56.67	72.66	60.00	64.63
1	74.54	43.33	66.44	49.30	59.14
2	73.61	48.57	70.93	60.93	64.20
3	72.69	48.10	69.20	63.72	63.98
4	71.30	49.05	71.97	64.19	64.85
5	72.69	49.52	70.93	61.86	64.41
6	72.22	48.57	70.93	61.86	64.09
7	72.69	50.00	70.59	62.33	64.52
8	72.69	47.62	70.93	62.79	64.20
9	72.22	49.05	71.28	61.86	64.31
10	67.13	50.00	68.51	57.21	61.40

α	Bedroom (108)	Kitchen (105)	Liv.Ro. (145)	Office (108)	Average (466)
0	72.22	46.19	68.17	62.79	62.91
0.1	71.76	48.57	69.55	62.79	63.77
0.2	72.22	48.57	69.90	61.86	63.77
0.3	70.37	48.10	69.90	61.86	63.23
0.4	71.30	50.00	69.90	61.86	63.88
0.5	70.37	50.48	70.59	60.47	63.66
0.6	71.76	47.62	70.59	62.79	63.88
0.7	72.69	48.57	71.63	63.26	64.74
0.8	71.76	50.00	69.90	60.93	63.77
0.9	70.37	51.90	69.90	63.26	64.41
1.0	69.44	52.86	69.44	60.47	62.69



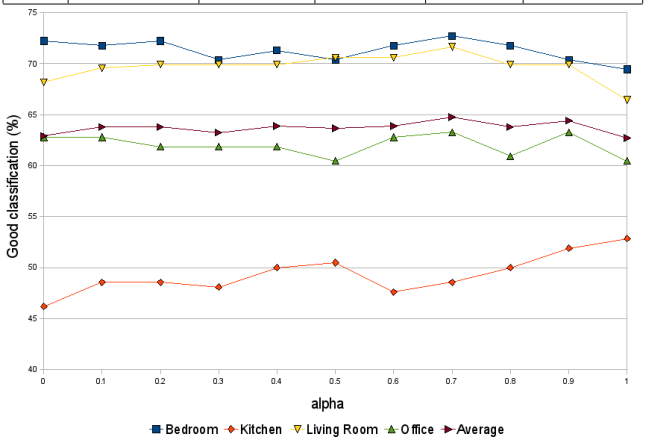


Figure 6. Good classification (in %) as a function of the minimum number of edges per node in the feature graph M and of parameter α. The number of test images per class is indicated in brackets.

[8] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2005. 1, 5, 6

[9] M. Field. http://www.mattfield.com. 4

[10] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Pattern Analysis and Machine Intelligence (PAMI)*, 1996. 1

[11] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2006. 2

[12] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *International Conference of Computer Vision (ICCV)*, 2005. 1, 3

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision (IJCV)*, 2003. 3

[14] B. Luo and H. E. R. Wilson, R. C. A spectral approach to learning structural variations in graphs. *Pattern Recogn.*, 39(6):1188–1198, 2006. 1

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *Proc. of British Machine Vision Conference (BMVC)*, 2002. 3

[16] F. Meyer. Learning and predicting brain dynamics from fmri: a spectral approach. SPIE, 2007. 1, 2

[17] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *European Conference on Computer Vision (ECCV)*, 2006. 1, 5

[18] H. Qiu and E. R. Hancock. Clustering and embedding using commute times. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 2007. 1, 2

[19] C. Unsalan and K. L. Boyer. A theoretical and experimental investigation of graph theoretical measures for land development in satellite imagery. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 2005. 1

[20] P. H. Winston. Learning structural descriptions from examples. Technical report, 1970. 1

[21] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. In *International Journal of Computer Vision (IJCV)*, 2007. 1, 3