AN APPLICATION OF GRAPH COMMUTE TIMES TO IMAGE INDEXING

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

[1]MAS, Ecole Centrale Paris,
Grande Voie des Vignes, F-92295 Châtenay-Malabry Cedex, France,
[2]NLPR/LIAMA, Institute of Automation, Chinese Academy of Sciences,
P.O Box 2728, Beijing 100190, China

ABSTRACT

In this paper we provide an overview of an image representation approach based on the description of layout and appearance properties of groups of features. In each image a graph of quantized features of interest is constructed. The features that are assigned to the same codebook bin are then grouped to produce a collapsed graph; the image contentn is represented by the matrix of commute times of this collapsed graph.

This novel image descriptor can be used to label satellite image databases; we demonstrate the relevance and the efficiency of our approach by addressing classification problems on a dataset of 0.6m resolution Quickbird images.

Index Terms— Image mining, classification, spectral graph theory

1. INTRODUCTION

In the field of satellite imaging interpretation, the means for a human agent to access and to process the available acquired data are not able to cope with the breadth and the quality of the data itself. This situation is paradoxical because it means we, as a scientific community, actually receive too much information to value it according to its true worth. The bottleneck that we face is the representation of the visual content of the satellite images. An automatic method for reliably describing the content of image subregions would allow us to index the image databases and to perform content-based queries on them. This, in turn, would open the door to precise automatic statistical measures and would therefore expand our large-scale analysis capability.

We apply here a resolutely novel image representation introduced in [1] that takes into account both the local appearance of regions as well as their relative layout. It is based on the measure of spectral properties of a graph built on a sparse set of interest points sampled in the image. These properties represent the distances between groups of interest points, where distance is computed in terms of similarity and spatial proximity. The relative importance of the appearance and the layout in the representation can be defined by two parameters; we observe that the bag-of-visterms [2], which dismisses all spatial information from the image representation, is a particular case of our approach. The idea of using attributed graph to represent image content has been introduced before [3]. However the approach of [3] is based on pixels groups and as such cannot really be applied to high resolution data.

Our representation was designed with the specific goal of content-based image retrieval in mind: regions that display similar but not exactly identical features and layouts should nonetheless have close representations. On the other hand, the information contained in the representation should be sufficiently rich to be able to discriminate between a large variety of visual classes. As a matter of fact, our approach is able to address both problems of intra and inter class variability.

2. METHODOLOGY

The construction of our image representation proceeds in several steps, described in full details in [1]. **First**, we sample interest points from the image. The choice of the detector/descriptor pair is arbitrary and should be made in accordance with the application and the type of visual data considered. Points can be extracted in a dense or sparse fashion, can be described by a wide array of possible descriptors and may be subject to certain invariances such as rotation, scale and/or affine transformations [4].

Second, we build the *feature graph* of the image: it is an unoriented weighted graph in which each interest point is a node and the nodes that are likely to belong to the same visual part are all the more strongly connected. We consider that interest points that belong to the same visual parts have close spatial positions and similar descriptors. Therefore we decide to connect each node i to its M nearest neighbours according to the distance:

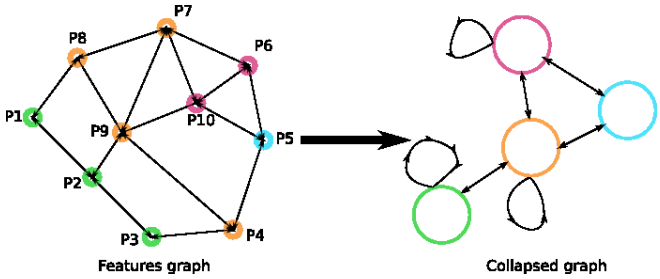
$$\Delta(i,j) = \Delta_{desc}(i,j)^{\alpha}\Delta_{geo}(i,j)^{1-\alpha} \quad (1)$$

Δ_{desc} is the distance function defined to measure the similarity of the feature descriptors. Δ_{geo} is the "spatial" or "geographical" distance between interest points coordinates as

image pixels (x, y), possibly normalised by the feature scale σ: $\Delta_{geo}(X_i, X_j) = \sqrt{\frac{(x_i-x_j)^2+(y_i-y_j)^2}{\sigma_i \sigma_j}}$.

The relative contributions of the appearance and the spatial proximity is weighted by $\alpha \in [0, 1]$. Changing the value of M determines the connectivity of the feature graph and the typical scale of the object subparts that the graph structure will capture. α and M are the only parameters of our approach that need to be defined experimentally.

Third, the nodes of the feature graph that are assigned to the same codebook entries are grouped together to produce a *collapsed graph*. The quantisation is made according to a codebook of fixed size K that was built offline. Each node of this graph represents a codebook entry and the weight of the edge $w_{kk'}$ between two nodes k, k' is equal to the sum of the weights w_{ij} of the edges $i \rightarrow j$ that join nodes that were assigned to codebook entries k and k' in the feature graph. The collapse can be simply illustrated by figure 2 and the following equation, in which different colours represent different codebook bins: $w'_{\blacksquare\blacksquare} = \sum w_{\blacksquare\blacksquare}$.



The collapsed graph is a structure that can be used to compare different images, contrary to the feature graph. The matrix of distances between graph nodes is an appropriate choice to represent the structure of the collapsed graph, but it requires a definition of this distance, just like different possible definitions of a metric exist in a euclidean space. We could simply use the transition matrix of the graph or the matrix of shortest paths between graph nodes. However, in problems where the presence or the accuracy of graph nodes is uncertain, as it is 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.

2.1. Graph commute times

Considering a random walk on the nodes of the collapsed graph started at node k with a transition probability proportional to the edge weights, the commute time $CT(k, k')$ between graph nodes k, k' is defined as the average number of steps required to reach k' for the first time and then to come back to k (see [5], [6] for details). Note that commute times can take infinite values when the graph is not connected. It has been shown ([6], [7] for a summary) that the matrix of commute times CT can be computed as a function of the eigenvectors $(\phi_k)_{1 \leq k \leq K}$ and eigenvalues $(\lambda_k)_{1 \leq k \leq K}$ of the Laplacian $\mathcal{L}$ of the graph:

$$\forall k, k' \in [1, K]\,, \tag{2}$$

$$\mathcal{L}(k, k') = \begin{cases} 1 - \frac{w'_{kk'}}{d_k} \text{if } k = k' \\ \frac{-w'_{kk'}}{\sqrt{d_k d'_k}} \text{if } k \neq k' \end{cases} \tag{3}$$

$$CT(k, k') = vol \sum_{i=2}^{N} \frac{1}{\lambda_i} \left(\frac{\phi_i(k)}{\sqrt{d_k}} - \frac{\phi_i(k')}{\sqrt{d_{k'}}} \right)^2 \tag{4}$$

$$\tag{5}$$

with: $d_k = \sum_i w'_{ki}$ and $vol = \sum_{i=1}^{K} d_i$. Our image representation χ is a normalisation of the $K \times K$ commute time matrix: $\chi(k, k') = exp\left(\frac{-CT(k,k')}{K}\right)$. For $M = 0$ the only non-zero terms are the diagonal elements that correspond to quantised features located in the image and χ is equal to the binary bag-of-visterms.

The obtained representation is of dimension $K(K+1)/2$ with K of the order of a few hundreds to a few thousands. It is thus time to note that equation 4 can also be viewed as an embedding of the nodes of the graph in a space in which coordinate $i-1$ of node k is equal to $\sqrt{\frac{vol}{\lambda_i d_k}} \phi_i(k)$. We sort the eigenvalues of $\mathcal{L}$ by increasing order: $0 = \lambda_1 < \lambda_2 \leq \ldots \lambda_K$. The dimensionality of the embedding space can thus be arbitrarily reduced by considering only the first D (with $D < K$) eigenvalues. We can therefore considerably reduce the dimensionality of χ by considering each image as a node in a graph and by embedding the nodes of the graph in a space of low dimension.

3. RESULTS

3.1. Dataset and Parameters

We tested our approach on two datasets composed of high resolution (0.6m) optical panchromatic Quickbird images realised in the area of Beijing (China). For each dataset and each category, half of the images will be used to train our classifier and the other half will constitute our testing dataset.

1. Our first dataset is composed of 251 images of size 512×512 containing either portions of road or vegetation areas. Certain images were arbitrarily assigned to one of the two classes despite the fact that they contained instances both of vegeation and roads.

2. The second dataset, illustrated in figure 1, is composed of 878 images of size 200×200 coming from seven classes: (1) big buildings, (2) golf fields, (3) greenhouses, (4) small industry, (5) fields, (6) dense urban, (7) residential area.

We employed a rotation- and scale-invariant Speeded Up Robust Features (SURF) detector as well as the associated descriptor [8] to build our feature graphs. The features extracted from our training images were clustered by k-means relatively to a codebook of size $K = 500$ previous to the execution of the algorithm. The image representations are embedded in a space of dimension $D = 20$. The image classification step is realized by 1 VS 1 AdaBoost. We set the values of $M = 2$ and $\alpha = 0.5$ so as to obtain optimal performances.

3.2. Performances

Experiments on the first dataset, which is relatively simple, are meant to demonstrate the validity of our approach. Figure 2 represents the embedding of the image representations in a space of dimension 2.

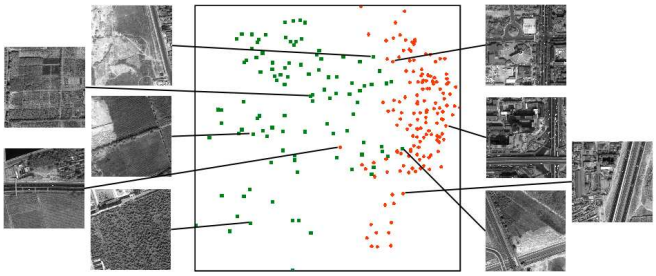


Fig. 2: Dataset 1: "Vegetation" versus "Road" image classification. ($\alpha = 0.5, M = 2$)

We can see that a linear SVM classifier in that space can successfully classify most images. Moreover, images that are close to the separator contain either roads crossing vegetation areas, or large vegetation areas bordered by roads. As a matter of fact, binary classification in the embedding space of dimension 20 results in a 96% good classification rate. These results are in fact closer to total recall if we take into account the fact that certain images are wrongly labelled. Generally speaking, the quality of the results demonstrate the validity and the relevance of our approach.

Dataset 2 is more representative of a true use case as it contains image instances coming from a greater number of classes. We visualize in tables 1, 2 and figure 3.2 the influence of parameters α and M on the classification performances for each class.

We observe that modifying the structure of the feature graph has an influence on the classification performance. Moreover, different classes have different optimal parameters values. As a rule of thumb, we can say that high values of α (i.e: higher influence of the layout information on the image representation) produce better performances. It is interesting to observe the performance evolution as a function of parameter M, as the case $M = 0$ is equivalent to the bag-of-visterms representation. We notice that adding some graph structure can greatly boost the good classification rate for classes such as *fields* ($+38.36\%$) or *greenhouses* ($+10.9\%$); more generally speaking, for each class the classification performances can be improved by setting a particular value of M.

4. CONCLUSIONS

The approach detailed in this paper aims at describing an image representation that encompasses both the content appearance and the general layout of the content. The representation is realised in a sufficiently loose way to cope with large intra-class variation but on the other hand is more precise than the orderless bag of features, resulting thus in an increase in performance for classification tasks.

Even if there is not one single set of parameters for which our approach improves over the bag-of-features for all classes, we have shown that incorporating information about the layout of regions of interest in the image representation can be a major improvement for certain classes.

5. REFERENCES

[1] R. Behmo, N. Paragios, and V. Prinet, "Graph commute times for image representation," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[2] P. Quelhas, F. Monay, J-M Odobez, D. Gatica-Perez, and T. Tuytelaars, "A thousand words in a scene," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, no. 9, pp. 1575–1589, 2007.

[3] S. Aksoy, "Modeling of remote sensing image content using attributed relational graphs," in *SSPR/SPR*, 2006, pp. 475–483.

[4] 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.

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

[6] F. Chung and S. T. Yau, "Discrete green's functions," *Journal of Combinatorial Theory Series A*, vol. 91, no. 1-2, pp. 191–214, 2000.

[7] H. Qiu and E. R. Hancock, "Clustering and embedding using commute times," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, no. 11, pp. 1873–1890, 2007.

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

(a) Big buildings (68) (b) Golf field (104) (c) Greenhouses (111) (d) Small industry (122) (e) Fields (52) (f) Dense urban (238) (g) Residential area (183)

Fig. 1: Dataset 2. The number of images of each class is indicated in parentheses.

	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Big buildings	91.18	85.29	91.18	91.18	88.24	88.24	85.29	91.18	91.18	88.24	88.24
Golf field	88.46	94.23	88.46	84.62	92.31	90.38	92.31	84.62	90.38	84.62	90.38
Greenhouses	32.73	32.73	38.18	50.91	38.18	49.09	50.91	74.55	67.37	74.55	58.18
Small industry	83.61	81.97	78.69	81.97	80.33	80.33	83.61	83.61	81.97	80.33	83.61
Fields	73.08	80.77	76.92	65.38	61.54	57.69	57.69	65.38	69.23	73.08	76.92
Dense urban	94.12	96.64	97.48	95.8	96.64	95.8	94.96	94.12	94.96	96.64	96.64
Residential area	89.01	91.21	91.21	90.11	90.11	89.01	94.51	91.21	86.81	91.21	91.21
Average	81.69	83.29	83.29	83.53	82.38	82.85	84.45	86.52	85.61	86.98	86.28

Table 1: Dataset2: performance evaluation as a function of parameter α ($M = 2$)

	0	1	2	3	4	5	6	7	8	9	1
Big buildings	88.24	76.47	88.24	82.35	91.18	91.18	88.24	91.18	88.24	91.18	91.18
Golf field	92.31	94.23	90.38	90.38	94.23	96.15	96.15	90.38	94.23	88.46	92.31
Greenhouses	74.55	85.45	49.09	78.18	65.45	54.55	34.55	58.18	58.18	45.45	61.82
Small industry	75.41	83.61	80.33	80.33	81.97	83.61	78.69	83.61	85.25	85.25	83.61
Fields	42.41	50.00	57.69	73.08	73.08	76.92	73.08	73.08	69.23	73.08	80.77
Dense urban	97.48	93.28	95.8	94.96	95.8	94.96	94.96	95.8	95.8	94.96	95.8
Residential area	91.21	85.71	89.01	87.91	95.6	93.41	92.31	92.31	95.6	92.31	93.41
Average	85.62	85.62	82.85	86.52	88.11	86.73	82.83	86.28	87.19	84.44	87.65

Table 2: Dataset2: performance evaluation as a function of parameter M ($\alpha = 0.5$)