High-Dimensional Spectral Feature Selection for 3D Object Recognition Based on Reeb Graphs

Boyan Bonev[1], Francisco Escolano[1], Daniela Giorgi[2], and Silvia Biasotti[2]

[1] University of Alicante, Spain
{boyan,sco}@dccia.ua.es
[2] IMATI CNR, Genova, Italy
{daniela,silvia}@ge.imati.cnr.it

Abstract. In this work we evaluate purely structural graph measures for 3D object classification. We extract spectral features from different Reeb graph representations and successfully deal with a multi-class problem. We use an information-theoretic filter for feature selection. We show experimentally that a small change in the order of selection has a significant impact on the classification performance and we study the impact of the precision of the selection criterion. A detailed analysis of the feature participation during the selection process helps us to draw conclusions about which spectral features are most important for the classification problem.

1 Introduction

Although feature selection (FS) plays a fundamental role in pattern classification [1], there are few studies about this topic in structured patterns, mainly when graphs are not attributed (pure structure). One exception is the work of Luo et al. [2] where different spectral features are investigated, but for embedding purposes. Regarding application areas, graph-based descriptors have been used for 3D object retrieval and classification. In this paper we study Reeb graphs [3] obtained from different functions. What is the role of each function? What is the role of each spectral feature, beyond the ones studied so far? Answering these questions, through an information-theoretic [4] method, is the main contribution of this paper. Not less important is the successful multi-class classification of unattributed graphs, using only structural information.

2 Reeb Graphs

Given a surface $\mathcal{S}$ and a real function $f : \mathcal{S} \rightarrow \mathbb{R}$, the *Reeb graph* (RG) [5] represents the topology of $\mathcal{S}$ through a graph structure whose nodes correspond to the critical points of f. When f is differentiable, the critical points are located in correspondence of topological changes of S, such as birth, join, split and death of connected components of the surface. Hence, RGs describe the *global* topological structure of $\mathcal{S}$, while also coding *local* features identified by f. RGs are becoming popular in several application domains including shape comparison, segmentation and visualisation. A detailed overview of mathematical properties, computational techniques and applications

E.R. Hancock et al. (Eds.): SSPR & SPR 2010, LNCS 6218, pp. 119–128, 2010.

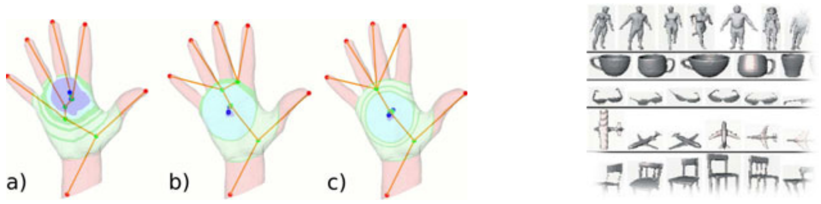

Fig. 1. Left: Extended Reeb graphs. Right: some samples of the 3D shapes database [9].

of Reeb graphs is presented in [6]. The graph representation we adopt in this paper is the *Extended Reeb Graph* (ERG) proposed in [7,3] for triangle meshes representing closed surfaces embedded in $\mathbb{R}^3$. The salient feature of ERG is the approximation of the RG by using a fixed number of level sets (63 in this paper) that divide the surface into a set of regions; critical regions, rather than critical points, are identified according to the behaviour of f along level sets; ERG nodes correspond to critical regions, while the arcs are detected by tracking the evolution of level sets.

The most interesting aspect of RGs is their parametric nature. By changing f, we have different descriptions of the same surface S that highlight different shape properties. Here we choose three alternative scalar functions f, namely the integral geodesic distance defined in [8] and the two distance functions $f(\mathbf{p}) = ||\mathbf{p} - \mathbf{b}||_2$, with $\mathbf{b}$ the center of mass and the center of the sphere circumscribing the triangle mesh respectively. Fig. 1 exemplifies our three ERG representations on a hand model, namely a) using geodesic distance [8], b) the distance from the mass center, and c) from the center of the circumscribing sphere (Fig. 1-left).

3 Features from Graph Spectra

The design of the feature extraction process is the most important part in a subsequent classification task. Concerning the characterization of a graph $G = (V, E)$, the degree distribution is a major source of statistical information. For instance, testing whether a graph is scale-free or not is posed in terms of checking whether its degree distribution follows the power law [10]. A more elaborate feature is the *subgraph node centrality* [11], which quantifies the degree of participation of a node i in structural subgraphs. It is defined in terms of the spectrum of the adjacency matrix $\mathbf{A}$, i.e. $C_S(i) = \sum_{k=1}^{n} \phi_k(i)^2 e^{\lambda_k}$, where $n = |V|$, λ_k the k-th eigenvalue of $\mathbf{A}$ and ϕ_k its corresponding eigenvector. In this regard, ϕ_n (the eigenvector corresponding to the largest eigenvalue) is the so called *Perron-Frobenius eigenvector*. The components of the latter vector denote the degree of importance of each node in a connected component and they are closely related to subgraph centrality [11]. Furthermore, the magnitudes $|\phi_k|$ of the (leading) eigenvalues of $\mathbf{A}$ have been been experimentally validated for graph embedding [2]. Besides the study of the adjacency matrix, it is also interesting to exploit the *spectrum of the Laplacian* $\mathbf{L} = \mathbf{D} - \mathbf{A}$ or the *spectrum of the normalized Laplacian* $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$, where $\mathbf{D}$ is the diagonal degree matrix. These spectra encode significant structural information. For instance, $\lambda_2 \leq n/(n-1)\ , n \geq 2$;

in addition, the multiplicity of the trivial eigenvalue yields the number of connected components. in the case of $\mathcal{L}$ we have $\lambda_k \leq 2,\ 2 \leq k \leq n$, and the Laplacian spectrum plays a fundamental role in the design of regularization graph kernels. Such kernels encode a family of dissimilarity measures between the nodes of the graph. Regarding the eigenvectors of the Laplacian, the *Friedler vector*, that is, the eigenvector corresponding to the first non-trivial eigenvalue, ϕ_2 in connected graphs, encodes the connectivity structure of the graph (actually its analysis is the core of graph-cuts methods) and it is related to the Cheeger constant. In addition, both the eigenvectors and eigenvalues of the Laplacian are key to defining a metric between the nodes of the graph, namely the *commute time*, $CT(i,j)$. It is the average time taken by a random walk starting at i to reach j and then returning. If we use the un-normalized Laplacian, we have that $CT(i,j) = vol \sum_{k=2}^{n}(1/\lambda_k)(\phi_k(i) - \phi_k(j))^2$, where vol is the volume of the graph, that is, the trace of $\mathbf{D}$. In the normalized case $CT(i,j) = vol \sum_{k=2}^{n}(1/\lambda_k)(\phi_k(i)/\sqrt{d_i} - \phi_k(j)/\sqrt{d_j})^2$, where d_i and d_j are the degrees of i and j respectively. Since the commute time is a metric, and because of its utility for graph embedding [12], the path-length structure of the graph is partially encoded. Finally, considering diffusion kernels on graphs, which belong to the family of regularization kernels, the analysis of the diffusion process itself yields a valuable source of information concerning the structure of the graph. A recent characterization of the diffusion process is the *the flow complexity trace* [13], a fast version of polytopal complexity [14]. The complexity trace encodes the amount of heat flowing through the edges of G for a set of inverse temperatures β: from $\beta = 0$ (no flow) to $\beta \rightarrow \infty$ (flow equal to $2|E|$) there is a phase-transition point. More precisely, the instantaneous flow for a given β is $F(G;\beta) = \sum_{i=1}^{n}\sum_{j \neq i}^{n} A_{ij}(\sum_{k=1}^{n}\phi_k(i)\phi_k(j)e^{-\beta\lambda_k})$ and the trace element for this inverse temperature is the instantaneous complexity $C(G;\beta) = \log_2(1 + F(G;\beta)) - \log_2(n)$ where the final term is for the purpose of size normalization.

4 Feature Selection

4.1 Mutual Information Criterion

In *filter feature selection* methods, the criterion for selecting or discarding features does not depend on any classifier. We estimate the mutual information (MI) between the features set and the class label, provided that we tackle a supervised classification problem: $I(\boldsymbol{S};\boldsymbol{C}) = H(\boldsymbol{S}) - H(\boldsymbol{S}|\boldsymbol{C})$. Here $\boldsymbol{S}$ is a matrix of size $m \times n$ and $\boldsymbol{C}$ of size $m \times 1$ where m is the number of samples and n the number of features of the feature subset. Traditionally the MI has been evaluated between a single feature and the class label. Here we calculate the MI using the entire set of features to select. This is an important advantage in FS, as the interactions between features are also taken into account [1]. The entropies $H(\cdot)$ of a set with a large n number of features can be efficiently estimated using the k-NN-based method developed by Leonenko [15]. Thus, we take the data set with all its features and determine which feature to discard in order to produce the smallest decrease of $I(\boldsymbol{S_{n-1}};\boldsymbol{C})$. We then repeat the process for the features of the remaining feature set, until only one feature is left. A similar information-theoretic selection approach is described in detail in [16]. They use minimal spanning trees for entropy estimation, while in this work we use the method of Leonenko which is simpler and allows us to vary the precision of the estimation by using the Approximate Nearest Neighbours algorithm [17].

4.2 Entropy Estimation

A simple way to understand the k-NN entropy estimation proposed by Leonenko [15] is to look at the Shannon entropy formula $H(X) = -\int f(x)\log f(x)dx$, as an average of $\log f(x)$, being $f(x)$ an existing pdf. The estimation of $\widehat{\log f(x)}$ would allow the estimation of $\hat{H}(X) = -N^{-1}\sum_{i=1}^{N}\widehat{\log f(x)}$. For this purpose the probability distribution $P_k(\epsilon)$ of the distance between a sample x_i and its k-NN is considered. If a ball of diameter ϵ is centered at x_i and there is a point within distance $\epsilon/2$, then there are $k-1$ other points closer to x_i and $N-k-1$ points farther from it. The probability of this to happen is $P_k(\epsilon)d\epsilon = k\binom{N-1}{k}\frac{dp_i(\epsilon)}{d\epsilon}p_i^{k-1}(1-p_i)^{N-k-1}$ being p_i the mass of the ϵ-ball and $p_i(\epsilon) = \int_{||\xi-x_i||<\epsilon/2} f(\xi)d\xi$.

The expectation of of $\log p_i(\epsilon)$ is $E(\log p_i) = \int_0^\infty P_k(\epsilon)\log p_i(\epsilon)d\epsilon$ that is $=$ $k\binom{N-1}{k}\int_0^1 p^{k-1}(1-p)^{N-k-1}\log p \cdot dp = \psi(k) - \psi(N)$, where $\psi(\cdot)$ is the well-known digamma function. If assumed that $f(x)$ is constant in the entire ϵ-ball, then the approximation $p_i(\epsilon) \approx \frac{V_d}{2^d}\epsilon^d\mu(x_i)$ can be formulated. Here d is the dimension and V_d is the volume of the unit ball $\mathcal{B}(0,1)$, defined as $V_d = \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2}+1)}$. From the previous approximation and using the expectation of $\log p_i(\epsilon)$, we have the approximation $\log f(\epsilon) \approx \psi(k) - \psi(N) - dE(\log\epsilon) - \log\frac{V_d}{2^d}$, and finally,

$$\hat{H}(X) = -\psi(k) + \psi(N) + \log\frac{V_d}{2^d} + \frac{d}{N}\sum_{i=1}^{N}\log\epsilon_i \tag{1}$$

is the estimation of $H(X)$, where $\epsilon_i = 2||x_i - x_j||$ is twice the distance between the sample x_i and its k-NN x_j. It is suggested that the error for Gaussian and uniform distributions is $\sim k/N$ or $\sim k/N\log(N/k)$.

4.3 Experimental Setup

In this work each sample is originally a 3D object represented by a triangle mesh. From each 3D object, three types of graphs (Sec. 2) are extracted (labeled in the figures as a) *Sphere*, b) *Baricenter* and c) *Geodesic*). Only the structural graph information is used for classification. For each graph, 9 different measures (listed in the area plots in Fig. 4) are calculated, as described in Sec. 3. They are transformed into histograms after normalizing them by the volume of the graph. Commute time is normalized twice, 1) linearly and 2) quadratically. Histograms are used in order to characterize the graph without dependence on the number and order of nodes. Only the complexity flow curve is not histogrammed, for the sake of order preservation. Since there is no optimal way to select the number of bins, we perform several different binnings on each measure (2, 4, 6 and 8 bins). All histograms form a bag of features, of size $9 \cdot 3 \cdot 20 = 540$ features (see Fig. 2). We let the FS process decide which binning from which measure and from which graph to discard.

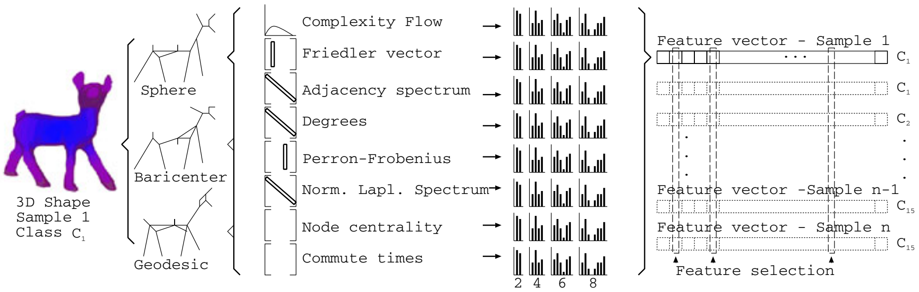

Fig. 2. The process of extracting from the 3D object the three graph representations, unattributed graph features, histogram bins, and finally selecting the features

5 Results and Discussion

The experiments are performed on the pre-classified 3D shapes database [9]. It consists of 15 classes × 20 objects. Each one of the 300 samples is characterized by 540 features, and has a class label $l \in \{$*human, cup, glasses, airplane, chair, octopus, table, hand, fish, bird, spring, armadillo, buste, mechanic, four-leg*$\}$; see Fig. 1-right.

5.1 Classification Error

The errors are measured by 10-fold cross validation (10-fold CV). In Fig. 3 we show how MI is maximized as the number of selected features grows, and its relation to the decrease in error. The figure shows how a high number of features degrades the classification performance. For the 15-class problem, the optimal error $(23, 3)$ is achieved with a set of 222 features. This error is lower for 10 classes $(15, 5\%)$, 5 classes (6%) and 2 classes problems (0%). These results depend on the classifier used for measuring the error. However the MI curve, as well as the selected features, do not depend on the classifier, as it is a purely information-theoretic measure.

5.2 Features Analysis

Several different unattributed graph measures are used in this work. We aim to determine which measures are most important and in which combinations. In Fig. 4-left we show the evolution of the proportion of selected features. The coloured areas in the plot represent how much a feature is used with respect to the remaining ones (the height on the Y axis is arbitrary). For the 15-class experiment, in the feature sets smaller than 100 features, the most important is the Friedler vector, *in combination* with the remaining features. Commute time is also an important feature. Some features that are not relevant are the node centrality and the complexity flow. Turning our attention to the graphs type, all three appear relevant. In Fig. 4-right we show the proportion of features selected for the 222-feature set, which yielded the lowest error in our experiments. (The dashed vertical line in Fig. 4-left also shows the 222-feature set) In the plot representing the selected binnings we can see that the four different binnings of the features do have importance for graph characterization.

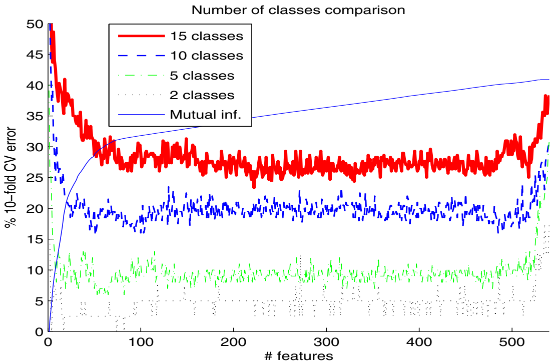

Fig. 3. Classification errors

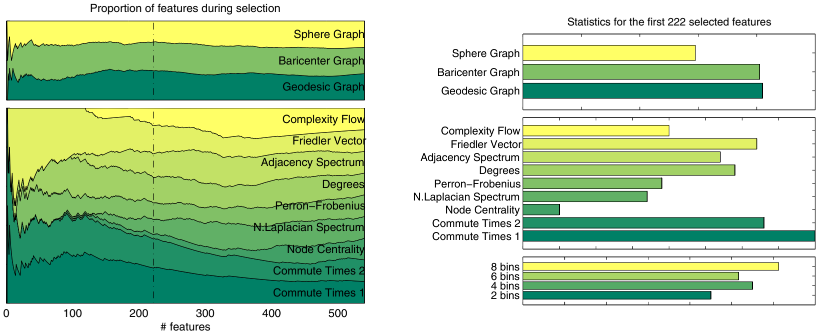

Fig. 4. Feature selection on the 15-class experiment (left) and the feature statistics for the best-error feature set (right)

These conclusions concerning the relevance of each feature cannot be drawn without performing some additional experiments with different groups of graph classes. For this purpose in Fig. 5 we present four different 3-class experiments. The classes share some structural similarities, for example the 3 classes of the first experiment have a head and limbs. Although in each experiment the minimum error is achieved with very different numbers of features, the participation of each feature is highly consistent with the 15-class experiment. The Friedler vector is always the most important for feature sets smaller than 100. On the other hand, the commute time measure is not important for feature sets smaller than 20, but then it becomes as important as the Friedler vector. The main difference among experiments (Fig. 5) is that node centrality seems to be more important for discerning among elongated sharp objects. Although all three graph types are relevant, the *sphere graph* performs best for blob-shaped objects.

5.3 The Impact of Feature Selection

In Fig. 3 it is obvious that, if the number of features used for classification is too low, the error is higher, and on the other hand if the number of features is too high, the

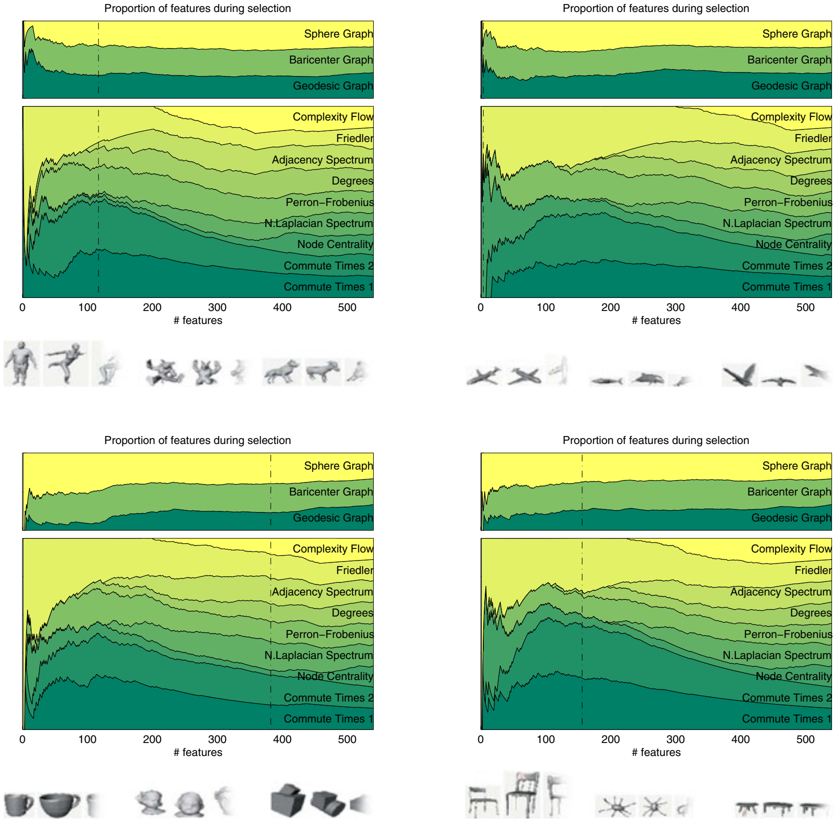

Fig. 5. Feature Selection on 3-class experiments: Human/Armadillo/Four-legged, Aircraft/Fish/Bird, Cup/Bust/Mechanic, Chair/Octopus/Table

error could also rise. However this depends on the order of features are added. In this work we use mutual information as the evaluation criterion because it is related to the minimization of the Bayesian error. What would happen if a worse criterion is used? To what extent the precision of the mutual information estimation is important? What is its impact on the final classification error?

Following we present some experiments which answer these questions. All of them refer to the 15-classes experiment. In order to vary the precision of the mutual information criterion we change the error bound ϵ of the ANN algorithm which is used for entropy estimation. ANN builds a kd-tree structure, whose cells are visited in increasing order of distance from the query point. A stop condition of the search algorithm occurs when the distance is closer than an error bound ϵ. This premature stop can save computational time, as shown in Fig. 6-right. It also causes a decrease in the precision of the k-NN computation. Thus, the entropy estimation, and so, the mutual information estimations, are degraded. To what extent? This is shown in Fig. 6-left. It is interesting to see that the error bound $\epsilon = 0$ yields significantly better feature selection results, in terms of 10-fold Cross Validation error. Also, the increment of the error bound is not linear with respect to the increment of the 10-fold CV error.

The differences in the classification performance are due to small differences in the feature sets. For example, the difference among the feature sets yielded by $\epsilon = 0$ and $\epsilon = 1$ are significant (see Fig. 7,-top-left). Then, before the error bound ϵ arrives the 0.5 value, the feature sets remain very similar. Other significant changes in the feature sets are plotted in Fig. 7. Each one of the figures compares two different feature selection processes, as a consequence of different ϵ values. The first process is represented as a coloured area plot, and the second one is represented with black solid lines.

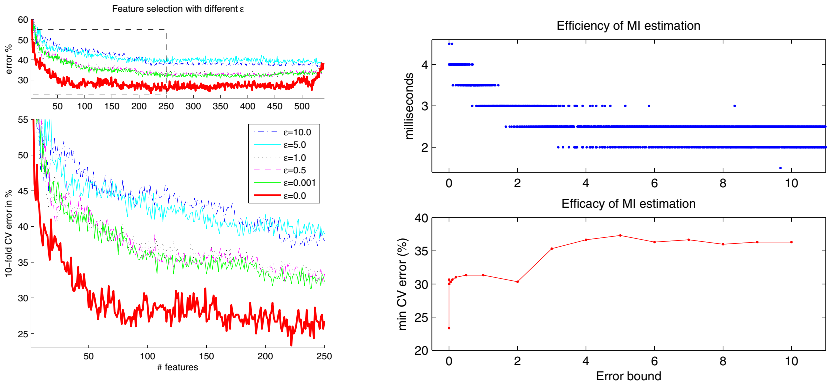

Fig. 6. Left: the 10-fold CV errors yielded by several feature selection runs, with different ANN error bound values (ϵ). Right-top: the milliseconds it takes (on a 1.6GHz Intel Centrino processor and DDR2 RAM) to evaluate the mutual information between the 300 samples with 540 features, and the 15 class labels. Right-bottom: the minimal errors achieved in the 15-class feature selection, for different Error bound (ϵ) values.

The most important differences are observed in the early stage of feature selection (before the first 200 features are selected). After that, the proportion among the different features selected converges, because there are no more features left for selecting. It is the early stage of the selection process which strongly conditions the maximum error which could be achieved, as shown in Fig. 6-left: a good run ($\epsilon = 0$) yields an error plot which decreases to $23, 3\%$, and after that increases to $37, 33\%$. A run which yields poor results is the case of $\epsilon = 0.5$, for instance. In this case the error decreases progressively until achieving $37, 33\%$, but none of the feature subsets produces a lower error.

It is also worth observing that the node centrality and the Friedler vector features are always important in the beginning of the process, disregarding the precision of the feature selection criterion. Shortly after the beginning, commute times start to play an important role. Regarding the Reeb graph types, most of the features are selected from the "sphere graph" type.

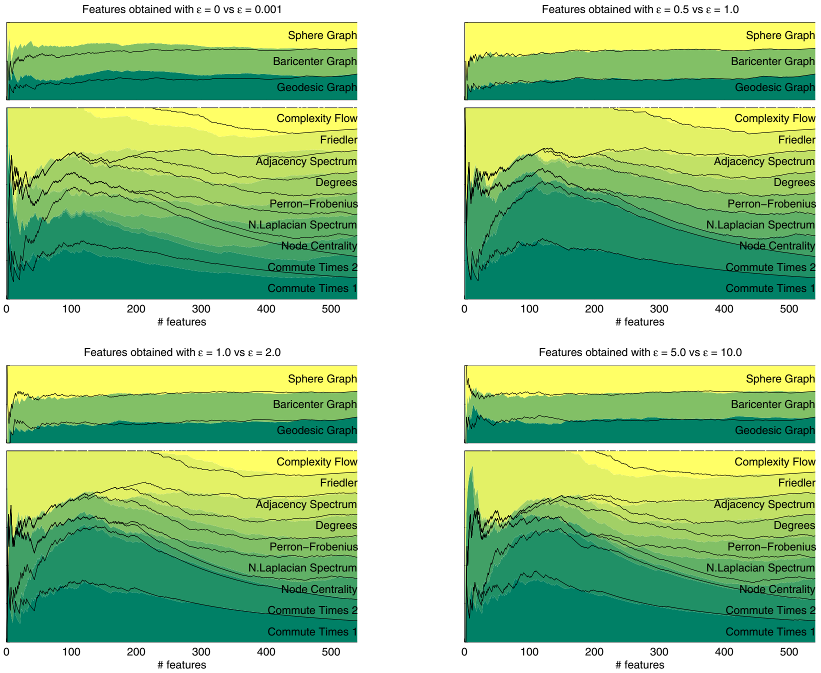

Fig. 7. Four feature selection comparisons of different pairs of ϵ values. The first feature selection process is represented as a coloured area plot, while the second one is plotted with black solid lines.

6 Conclusions

The contributions of this work to graph classification are twofold. Firstly, it demonstrates the feasibility of multi-class classification based on purely structural spectral features. Secondly, an information-theoretic feature analysis suggests that similar features are selected for very different sets of objects. Moreover, the feature selection experiments show that even if the precision of the selection criterion is degraded, the most important features are still the same.

On the other hand this paper demonstrates some important effects of feature selection. In the first place we prove that the precision of the mutual information estimation has a great impact on the final classification performance. The same experiments show how very small changes in the order of the selected features can also affect the classification result. Working with the maximum precision available is key to minimizing the classification error.

As future work we consider using attributed and directed graphs for improving classification accuracy. We also find it necessary to use a wider range of graph features, as well as other kinds of graph extraction methods.

References

1. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
2. Luo, B., Wilson, R., Hancock, E.: Spectral embedding of graphs. Pattern Recognition 36(10), 2213–2223 (2003)
3. Biasotti, S.: Topological coding of surfaces with boundary using Reeb graphs. Computer Graphics and Geometry 7(1), 31–45 (2005)
4. Escolano, F., Suau, P., Bonev, B.: Information Theory in Computer Vision and Pattern Recognition. Springer, New York (2009)
5. Reeb, G.: Sur les points singuliers d'une forme de Pfaff complètement intégrable ou d'une fonction numérique. Comptes Rendus 222, 847–849 (1946)
6. Biasotti, S., Giorgi, D., Spagnuolo, M., Falcidieno, B.: Reeb graphs for shape analysis and applications. Theoretical Computer Science 392(1–3), 5–22 (2008), doi:10.1016/j.tcs.2007.10.018.
7. Biasotti, S.: Computational Topology Methods for Shape Modelling Applications. PhD thesis, Universitá degli Studi di Genova (May 2004)
8. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.L.: Topology matching for fully automatic similarity estimation of 3D shapes. In: SIGGRAPH 2001, Los Angeles, CA, pp. 203–212 (2001)
9. Attene, M., Biasotti, S.: Shape retrieval contest 2008: Stability of watertight models. In: SMI 2008, pp. 219–220 (2008)
10. Barabási, A.L., Bonabeau, E.: Scale-free networks. Scientific American 288, 50–59 (2003)
11. Estrada, E., Rodriguez, J.A.: Subgraph centrality in complex networks. Physical Review E 71(5) (2005)
12. Qiu, H., Hancock, E.R.: Clustering and embedding using commute times. IEEE Transactions on PAMI 29(11), 1873–1890 (2007)
13. Escolano, F., Giorgi, D., Hancock, E.R., Lozano, M.A., Falcidieno, B.: Flow complexity: Fast polytopal graph complexity and 3d object clustering. In: GbRPR, pp. 253–262 (2009)
14. Escolano, F., Hancock, E.R., Lozano, M.A.: Birkhoff polytopes, heat kernels and graph complexity. In: ICPR, pp. 1–5 (2008)
15. Leonenko, N., Pronzato, L., Savani, V.: A class of rényi information estimators for multidimensional densities. The Annals of Statistics 36(5), 2153–2182 (2008)
16. Bonev, B., Escolano, F., Cazorla, M.: Feature selection, mutual information, and the classification of high-dimensional patterns. Pattern Analysis and Applications (February 2008)
17. Mount, D., Arya, S.: Ann: A library for approximate nearest neighbor searching (1997)