Improving Affinity Matrices by Modified Mutual kNN-Graphs

Peter Kontschieder, Michael Donoser and Horst Bischof

Institute for Computer Graphics and Vision
Graz University of Technology, Austria
{*kontschieder,donoser,bischof*}*@icg.tugraz.at*

Abstract
The recent progress in describing affinities between images or objects by means of shape, appearance or texture allows the exploitation of inherently emerging redundancies for improvement of retrieval tasks. We propose a two-way normalization and analysis scheme which aims on (a) modeling object interdependence by neighborhood incorporation and (b) retrieval improvement by subsequent analysis from a modified mutual k nearest neighbor graph. We provide a general and flexible approach which may be either applied for improving retrieval quality or as base for semi-supervised classification, clustering or dimensionality reduction methods. The presented experiments demonstrate that our approach yields to significant improvements on a broad variety of data sets, including the highest ever reported bullseye score of 93.40% on the MPEG-7 database.

1 Introduction

Information retrieval and query-by-example methods have attracted a lot of researchers in the last decade. It is the task of returning the most similar exemplars in huge databases given a single query. In this paper we focus on content based image retrieval, where we analyze the similarities of given input images. Nevertheless the introduced methods and discussions can be related to any other query-by-example applications as e. g. retrieving documents, music or multimedia data.

In general image retrieval methods can be split in two different steps. First, features have to be extracted from the images analyzing color, texture, shape or context. Second a similarity measure analyzing differences in images features has to be defined. Then given a query image, all other images of the database are sorted based on their similarity score to the query image. Mostly, the returned images are subjectively analyzed if they share the same content to measure the quality of the retrieval method. If ground truth is provided for the data set, i. e. each image can be assigned to one of a pre-defined set of labels, quantitative measures like the bullseye-score (ratio between true positives to false positives within the first N returned images) can be calculated.

There is a vast amount of methods for finding features in images and for defining the similarity scores. Recently some effort was also put on post-processing the obtained similarity scores by analyzing the estimated similarities between all given images to increase the discriminability between different image groups as e. g. in [8, 12]. This post-processing step is the main focus of this paper which is based on the main idea to capture the underlying structure of our high-dimensional data for improving retrieval scores. In general such methods can be also interpreted as updating the $N \times N$ affinity matrix W consisting of the pre-calculated similarities W_{ij} between all given input data points (images).

Strategies for improving these affinity matrices mainly consist of two different steps: normalizing the affinity matrix W and capturing the underlying structure of the data points e. g. by building nearest neighbor graphs and analyzing shortest paths. Surprisingly, many papers ignore the important point of how to normalize the given affinity matrix. Distances between images are not necessarily metric and need not to be scaled in a range from 0 to 1. Different authors like [10] already outlined the importance of the normalization which can have severe effects on retrieval results. This is also demonstrated in the experimental section.

After normalization a bottom-up strategy tries to capture the underlying structure of the data by defining a neighborhood for each data-point in terms of a neighborhood graph. Conventional nearest neighbor graphs are supposed to model the local relation between each data point (or vertex) i and its k nearest neighbors (kNN) or all points within distances $w(i,j) < \varepsilon,\ j = 1 \dots N$ (εNN), respectively. Depending on k or ε, the resulting graph $G_{NN} = (V, E)$ is typically very sparse, compared to a fully connected graph constructed from an affinity matrix. Both approaches do not guarantee a connected neighborhood graph which is for example required for a subsequent dimensionality reduction step. A solution to this problem is to construct k-edge minimal spanning trees, which ensures that the graph is entirely connected. Unfortunately building the MST-graph is an NP hard problem. Since we focus on retrieval application in this paper we allow unconnected graphs because the partition mainly highlights non-related categories in our task.

This paper discusses different possibilities for normalizing the affinity matrix and demonstrates their influence on a subsequent retrieval step. Furthermore, we propose to use a modified version of a mutual k-nearest neighbor graph to capture the underlying structure which yields improved performance on all analyzed data-sets. Since normalization and subsequent adaption of the affinity matrix constitutes the basis for many other applications, the proposed method could also be useful for semi-supervised classification tasks [11], non-linear dimensionality reduction methods like ISOMAP [7] or LLE [5] and for clustering methods [9, 2].

The outline of this paper is as follows. Section 2.1 shortly subsumes already existing normalization strategies and introduces a novel method for improving given distance matrices. Section 2.2 describes the usage of a modified mutual k-nearest neighbor graph as a more robust strategy for analyzing the underlying data structure. Finally Section 3 shows an exhaustive experimental evaluation concerning different normalization and analysis strategies applied to three different data sets. The improved performance is e. g. shown by achieving a retrieval score of 93.4% on the MPEG-7 database, which is by far the best ever reported score outperforming recent state-of-the-art [8].

2 Affinity Matrix Manipulation and Analysis

In this section we discuss different approaches how to improve given similarity matrices for the task of information retrieval by manipulating the corresponding affinity matrices. After a short introduction of currently applied normalization methods in Section 2.1, we describe our normalization strategy which tries to overcome the problem of proper affinity matrix generation by incorporation of neighborhood information. Unlike previous approaches, our method is based on the interdependence of each data point with its respective neighborhood and not only its single (geodesic) neighbors as proposed in [8, 2].
In Section 2.2 we focus on the analysis of the previously normalized affinity matrix. We propose a method based on a modified mutual k nearest neighbor graph which exploits the preceding normalization scheme.

2.1 Affinity Matrix Normalization

Given an $N \times N$ distance matrix A, the corresponding affinity matrix W may be derived by element-wise normalization according to

$$W_{i,j} = \exp\left(-\frac{A_{i,j}^2}{\sigma_{i,j}^2}\right) . \tag{1}$$

Many methods were introduced to determine the normalization parameter σ. We briefly subsume and enumerate some selected methods (with roman numbers) which we compare to our approach in the experimental section.

(I)
The two most common approaches make use of the matrices' variance $\sigma^2 = var(A)$ for equal normalization of all elements or determine
(II)

$$\sigma^2 = \sqrt{RC} , \tag{2}$$

where R is the sum-vector of A's rows and C is the sum-vector of A's columns.

(III)
Zelnik-Manor and Perona present another simple and yet powerful method in [10] for the selection of $\sigma_{i,j}$ by choosing

$$\sigma_{i,j} = \sigma_i \sigma_j \tag{3}$$

with

$$\sigma_i = A(i, i_K) , \tag{4}$$

where i_K is the K'th neighbor of object i. They suggested to choose $k = 7$ in their experimental part.

(IV)
In [1], normalization using the symmetric Graph Laplacian is described with

$$W = D^{-1/2} A D^{-1/2} , \tag{5}$$

where D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{N} A_{ij}$.

(V)
Another approach which may also be found in [1] employs a Random Walk Graph Laplacian

$$W = D^{-1} A \tag{6}$$

that again makes use of the previously described diagonal matrix D.

The normalization scheme used by Yang et al. in [8] employs an adaptive kernel size, based on the mean distance to each objects' k-neighborhood. They derive

$$\sigma_{i,j} = r \cdot \text{mean}(\{knn\_dist(x_i), knn\_dist(x_j)\}) \tag{7}$$

where mean($\{knn\_dist(i), knn\_dist(j)\}$) denotes the mean distance of all k-nearest neighbor distances of samples i and j. Thus, the mean value is calculated from the set consisting of $2 \cdot k$ nearest neighbor distances of i and j. The determination of r and k is done empirically. Please note that this normalization scheme is only listed for the sake of completeness since during experimental evaluation in Section 3 it was only used in combination with its proposed analysis method in [8].

(VI)
In [2], Fischer et al. introduce a matrix preprocessing step where the affinity matrix distances for each object pair $A_{i,j}$ are replaced with their *total dissimilarities* $W_{i,j}$, which are defined as follows: Let $G = (V, E, d')$ be a graph established on a symmetrized distance matrix with $|V|$ vertices and nonnegative edge weights $d'_{i,j}$ on edge $e(i,j) \in E$. Given the set $\mathcal{P}_{i,j}$ which contains all paths from i to j, the effective dissimilarity $d'^{p}_{i,j}$ is defined for each path $p \in \mathcal{P}_{i,j}$ by the maximum distance on its path. The total dissimilarity for (i,j) is then defined as the minimum of all effective dissimilarities

$$W_{i,j} := \min_{p \in \mathcal{P}_{i,j}} \Big\{ \max_{1 \le h \le |p|-1} d'_{p[h],p[h+1]} \Big\} \,. \tag{8}$$

(VII)
We propose a method encoding the neighborhood interdependence in $\sigma_{i,j}$ which allows to pull intra-class objects together ($\sigma_{i,j} \gg A_{i,j}$) and push extra-class objects aside ($\sigma_{i,j} \ll A_{i,j}$). $\sigma_{i,j}$ is self-assembling and takes the current distance $\frac{A_{i,j}}{2}$ as base term which is extended with a linear combination derived from the cross neighborhood of i. The cross neighborhood is defined by the relation of i to all other objects j and vice-versa, which delivers the coefficients for the aforementioned linear combination. With higher mutual interdependence between i and j, the base term will be substantially increased. The coefficients are derived from a voting system which requires the neighborhood ranking of A. This ranking is established by simply sorting all rows of A with non-decreasing order, which delivers the sorted distances and their respective indices $K(i,j)$. The normalization parameter is now calculated as

$$\sigma_{(i,K(i,j))} = \underbrace{\frac{A_{(i,K(i,j))}}{2}}_{\text{base term}} + \underbrace{\sum_{l=j}^{N} \overbrace{\frac{1}{l-j+1} \cdot \frac{2}{K(K(i,l),i)}}^{\text{voting term}} A_{(K(i,l),i)}}_{\text{cross neighborhood term}} \quad \forall i = 1 \dots N, j = 2 \dots N \tag{9}$$

and has the following properties:

- The neighborhood-influence to object i is determined from both, ranking and distance to other objects
- Depending on the mutual closeness of all considered pairs (i,j) the respective voting term is either rewarding or penalizing the current constellation
- $\sigma_{i,j}$ typically grows faster when objects i and j belong to the same neighborhood/cluster

As illustrated in Figure 1, the voting term finds its peak for directly neighboring points. In contrast, all points that are far away from the currently considered sample in terms of its k-neighborhood are substantially damped and thus negligible.

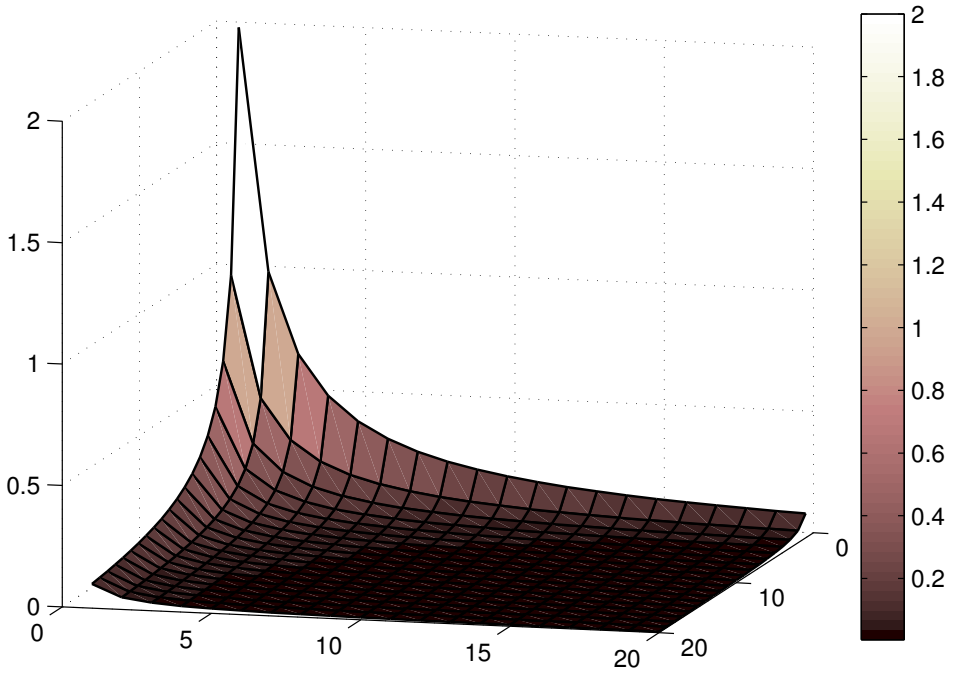

Figure 1: Illustration of proposed voting term.

2.2 Affinity Matrix Analysis

Normalization should ideally provide well-defined and embedded sub-manifolds representing separable clusters of matching objects. Unfortunately, there are no ideal normalization methods which is why we need to perform an analysis-step after normalization, to optimize object retrieval rates or clustering results.

Such analysis is typically performed using graph based methods which aim on capturing the locally underlying structures. Connected regions in such graphs are then exploited to reflect the interdependence between objects, i. e. by finding shorter paths between matching objects than between non-matching objects. In [8], Yang et al. try to group intra-class objects through graph transduction by propagating the normalized models through existing objects and state the analogy to the computation of geodesics in the underlying manifold.

For the proposed analysis step we first describe three different neighborhood graphs which belong to the group of geometric random graphs. For the considered graphs $G(V, E)$, let $|V| = N$ be the number of vertices $v \in V$, corresponding to the number of objects of the affinity matrix. The respective edge weights E for vertices (v_i, v_j) are determined by the normalized affinity matrix distances $w_{i,j}$, where $w_{i,j} = 0$ means that there is no connection between vertices (v_i, v_j). The amount of established connections between the vertices may now be regularized with respect to the chosen type of neighborhood graph:

- The ε-neighborhood graph $G_\varepsilon(N, \varepsilon)$ connects all vertices (v_i, v_j) if $w_{i,j} \leq \varepsilon$
- The (symmetric) k nearest neighbor graph $G_{k\text{NN}}(N, k)$ connects all vertices (v_i, v_j) if $v_i \in k\text{NN}(v_j)$ or $v_j \in k\text{NN}(v_i)$

- The mutual k nearest neighbor graph $G_{\mathrm{M}k\mathrm{NN}}(N,k)$ connects all vertices (v_i, v_j) if $v_i \in k\mathrm{NN}(v_j)$ and $v_j \in k\mathrm{NN}(v_i)$

Here $k\mathrm{NN}(v_j)$ denotes the set of k nearest neighbors of vertice v_j. The use of ε-neighborhood graphs requires data distributions on same scales which means that the object points in the underlying regions have to form tight clusters. k nearest neighbor graphs are able to connect regions with different scales, since only the absolute neighborhood ranking is of interest for the connection of two points. k nearest neighbor and ε-neighborhood graphs are often used in analyses methods while mutual k nearest neighbor graphs are widely ignored despite their interesting properties.

The mutual k nearest neighbor graph does not allow to mix differently scaled regions and may thus be considered as hybrid form of ε-neighborhood graphs and k nearest neighbor graphs since it allows clustering of regions with both, high or low density.

The major limitation of the mutual k nearest neighbor graph lies in its inherent inability to generate single clusters from mixed density regions, especially if there are isolated regions or single isolated objects within a cluster. This is why we introduce a *non-symmetric mutual neighborhood criterion* which allows edge construction in a modified mutual k nearest neighbor graph $G_{\mathrm{mM}k\mathrm{NN}}(N,k)$ between v_i and v_j only if

$$v_j \in k\mathrm{NN}(v_i) \quad \text{and} \quad v_i \in ck\mathrm{NN}(v_j). \tag{10}$$

The introduction of an asymmetry coefficient c allows to control the range of neighborhood incorporation during the mutual k nearest neighbor graph construction. This relaxation is also beneficial when the clusters have a high local density (which our proposed normalization scheme in Section 2.1 aims on) and $k \ll m$ with m being the number of cluster members.

In addition to this modification in the graph construction, we exploit possible asymmetries of the normalized input matrix W and exchange all edge weights in $G_{\mathrm{mM}k\mathrm{NN}}(i,j)$ with the smallest known distances from $G_{\mathrm{mM}k\mathrm{NN}}(i,j), W(i,j)$ or $W(j,i)$.

3 Experiments

In this section we show that the proposed normalization strategies and the subsequent affinity matrix analyses give remarkable improvements on a variety of different affinity matrices. After a short description of how each matrix is constructed we list its retrieval scores for the presented normalization strategies (roman numbers) in combination with the respective analysis method. Each table lists the retrieval scores in percent and provides the experimentally determined additional parameters which delivered the highest scores. The parameterization of the algorithm proposed by Yang et al. in [8] is taken as described in their paper with a window size of 300, $r = 0.27$, $k = 10$ and a propagation iteration number of 5000.

3.1 Olivetti face database

Each of the first 100 images of the Olivetti face database with size 64×64 was smoothed with a Gaussian kernel ($\sigma = 0.5$), and then reproduced 9 times as rotated ($-10°$, $0°$ and $+10°$) and scaled version (0.9, 1.0 and 1.1) to generate a total number of 900 images. To remove background clutter, a central window with size 50×50 was extracted from each image and then normalized to have zero mean and variance of 0.1.

For construction of the 900×900 distance matrix, the sum of squared pixel differences was calculated for each image pair. Thus, the resulting matrix consists of 10 classes with 90 objects per class. For the Olivetti face database we list the retrieval rates which are obtained within the true number of possible matches. According to this evaluation the constructed input distance matrix has a retrieval score of 39.78%, which means that there is a mean recognition of approximately 36 out of 90 faces for each query face.

Normalization method		No analysis	Yang et al. [8]	εNN-graph	kNN-Graph	Ours
	without normalization	39.78	54.67	49.36 ($\varepsilon = 16$)	50.98 ($k = 6$)	62.88 ($k = 6, c = 1$)
I	Variance of data	39.78	38.46	50.18 ($\varepsilon = 0.91$)	50.19 ($k = 6$)	62.33 ($k = 6, c = 1$)
II	$\sqrt{RC}$	40.65	54.67	51.48 ($\varepsilon = 0.007$)	49.11 ($k = 6$)	61.09 ($k = 6, c = 1$)
III	kth NN [10] ($k = 8$)	41.11	45.86	61.34 ($\varepsilon = 0.58$)	61.23 ($k = 6$)	62.02 ($k = 6, c = 1$)
IV	Symmetric Graph Laplacian [1]	41.27	56.75	53.02 ($\varepsilon = 0.48$)	52.30 ($k = 5$)	**63.01** ($k = 6, c = 1$)
V	Random Walk Graph Laplacian [1]	43.70	56.77	53.00 ($\varepsilon = 0.48$)	51.48 ($k = 6$)	62.77 ($k = 6, c = 1$)
VI	Total dissimilarities [2]	49.16	45.86	49.16 ($\varepsilon = 19$)	49.16 ($k = 90$)	50.88 ($k = 60, c = 10$)
VII	Ours	41.13	54.09	56.07 ($\varepsilon = 0.58$)	57.98 ($k = 5$)	61.64 ($k = 6, c = 1$)

Table 1: Retrieval scores on Olivetti face database

The results in Table 1 demonstrate that analysis by the symmetricmutual k nearest neighbor graph clearly outperforms conventional k nearest neighbor or ε neighborhood graphs as well as the graph transduction method proposed by Yang et al. in terms of retrieval scores. In combination with normalization by the symmetric graph Laplacian we obtain a boost of 23.23% in retrieval score which yields to a mean recognition of approximately 57 out of 90 faces per query face. We also want to mention that different normalization strategies yield to different results with the observation that our proposed analysis method always delivers the highest retrieval scores, independent of the applied normalization scheme.

3.2 Ukbench database

The similarity matrix for the UKbench database [4] is the result of a standard vocabulary tree scoring function. SIFT keys are extracted in each input image and a pre-trained vocabulary tree is employed to quantize the features into a set of visual words. We utilize the standard *tf-idf* (term frequency-inverse document frequency) weighting as proposed in [6] to compute the similarity score.

We take a subset with size 1400×1400 of this database (which corresponds to 350 classes with 4 objects per class) since the evaluation time for the approach of Yang et al. in [8] takes several hours evaluation cycle for each database, although the window size is set to 300. The used matrix has an initial retrieval score of 81.55% within the first 4 (true) matches.

Normalization method		No analysis	Yang et al. [8]	εNN-graph	kNN-Graph	Ours
	without normalization	81.55	87.36	81.46 ($\varepsilon = 0.19$)	81.46 ($k = 3$)	88.39 ($k = 2, c = 4$)
I	Variance of data	25.25	25.21	25.25 ($\varepsilon = 1.0$)	25.25 ($k = 3$)	25.25 ($k = 3, c = 1$)
II	$\sqrt{RC}$	82.32	90.55	82.21 ($\varepsilon = 1.413 \cdot 10^{-4}$)	82.21 ($k = 3$)	89.34 ($k = 2, c = 4$)
III	kth NN [10] ($k = 3$)	87.98	88.59	89.46 ($\varepsilon = 0.64$)	87.59 ($k = 3$)	91.39 ($k = 2, c = 3$)
IV	Symmetric Graph Laplacian [1]	82.73	87.71	82.64 ($\varepsilon = 0.6$)	82.64 ($k = 3$)	89.38 ($k = 2, c = 3$)
V	Random Walk Graph Laplacian [1]	84.27	87.73	81.88 ($\varepsilon = 0.67$)	81.46 ($k = 3$)	88.54 ($k = 2, c = 4$)
VI	Total dissimilarities [2]	80.25	80.82	80.25 ($\varepsilon = 0.21$)	80.25 ($k = 3$)	80.25 ($k = 3, c = 2$)
VII	Ours	86.45	90.71	90.09 ($\varepsilon = 0.95$)	90.07 ($k = 3$)	**91.48** ($k = 3, c = 1$)

Table 2: Retrieval scores on UKbench database

The proposed normalization and analysis methods outperform all other combinations in terms of retrieval scores. We obtain an increase in the retrieval score within the true matches from 9.93%.

Again, the modified mutual k nearest neighbor graph is favored over all other matrix analysis methods. The suboptimal scores for normalization method (I) are due to the low variance ($\sigma^2 = 7.16 \cdot 10^{-5}$) of the data which causes a normalization of all data points to the *unit* value.

3.3 MPEG-7 CE-Shape-1 database

The MPEG-7 CE-Shape-1 database is widely used for testing the performance of shape descriptors [3]. It consists of 1400 silhouette images which are grouped into 70 classes with 20 objects per class. In Figure 2, we show one sample object of each class.
The retrieval rate is measured by the so-called bullseye score which counts all matching objects within the 40 most similar rated candidates. Since each class consists of 20 objects, the retrieved score is normalized with the highest possible number of hits (20×1400).

The shape descriptor IDSC+DP (inner distance shape context with dynamic programming matching) proposed by Ling et al. in [3] obtains a bullseye score of 85.40%. In [8], Yang et al. presented an improvement based on IDSC+DP and boosted the retrieval score to 91.00% by learning graph transduction. All presented normalization and analysis evaluations are based on the distance matrix generated from IDSC+DP.

Normalization method		No analysis	Yang et al. [8]	εNN-graph	kNN-Graph	Ours
	without normalization	85.40	91.00	84.82 ($\varepsilon = 70$)	89.55 ($k = 7$)	90.60 ($k = 7, c = 3$)
I	Variance of data	72.42	24.67	71.53 ($\varepsilon = 1.0$)	82.35 ($k = 7$)	83.57 ($k = 7, c = 3$)
II	$\sqrt{RC}$	85.82	91.42	85.45 ($\varepsilon = 0.05$)	89.08 ($k = 7$)	90.34 ($k = 7, c = 3$)
III	kth NN [10] ($k = 8$)	88.44	91.18	91.98 ($\varepsilon = 0.67$)	92.81 ($k = 6$)	**93.40** ($k = 6, c = 2$)
IV	Symmetric Graph Laplacian [1]	85.95	90.96	85.51 ($\varepsilon = 0.57$)	89.85 ($k = 7$)	90.68 ($k = 7, c = 3$)
V	Random Walk Graph Laplacian [1]	86.18	90.96	85.12 ($\varepsilon = 0.57$)	89.62 ($k = 7$)	90.53 ($k = 7, c = 3$)
VI	Total dissimilarities [2]	77.81	80.09	77.81 ($\varepsilon = 70$)	78.41 ($k = 25$)	78.41 ($k = 26, c = 5$)
VII	Ours	86.39	88.97	90.86 ($\varepsilon = 0.81$)	91.56 ($k = 6$)	92.11 ($k = 6, c = 2$)

Table 3: Retrieval scores on MPEG-7 database

Table 3 again approves the applicability of our proposed normalization and analysis scheme by delivering the highest ever reported bullseye scores of 93.4% (III) and 92.11% (VII). The attained improvements range up to 7.98% compared to the retrieval score of the input matrix. The results in this table also clearly demonstrate that using the modified mutual k nearest neighbor graph for the analysis step can be favored over the other methods for most normalization schemes. In Figure 3 we show some examples which illustrate the impact of the introduced affinity matrix post-processing approach. We only show the ranking of the first 20 objects - note that the bullseye score allows matching objects to be found within the first 40 objects.

3.3.1 Behavior with parameter variation in modified mutual k nearest neighbor graph

Manifold structure analysis using the modified mutual k nearest neighbor graph requires the parameterization of two parameters. Once a reasonable neighborhood is covered with the combination of c (neighborhood incorporation factor) and k (neighborhood size), the retrieval scores are quite stable.

Figure 2: Silhouette shapes from the MPEG7 database. Each item represents one of the 70 classes.

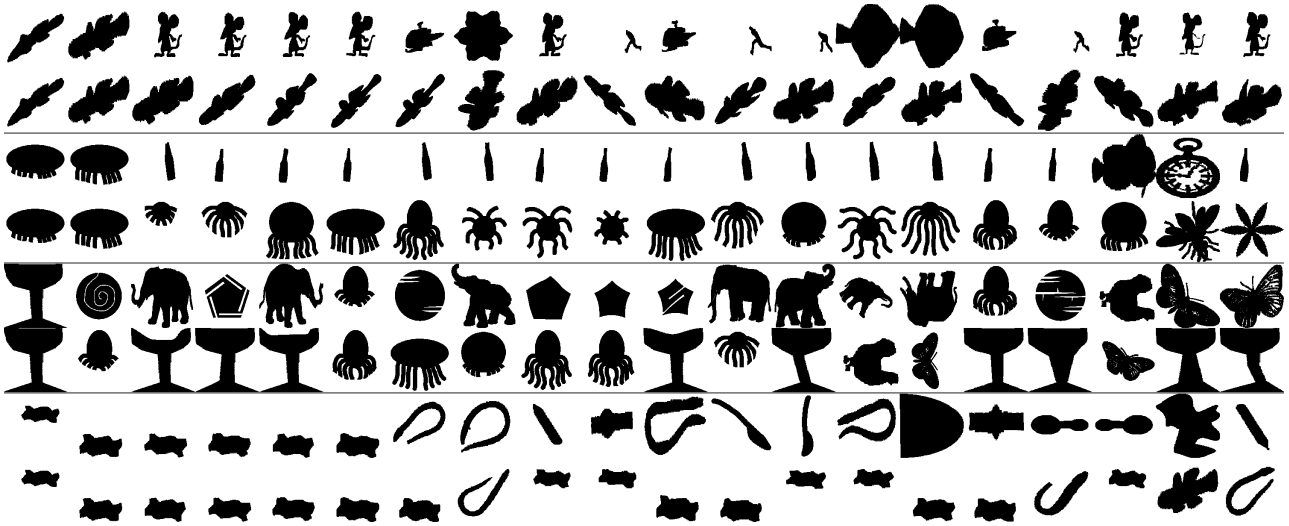

Figure 3: Comparison of retrieval results before (odd rows) and after improving affinity matrix (even rows) on samples from MPEG-7 database. The first column shows the query shape followed by its best 19 matching objects.

The obtained scores for different values of k and c after normalization with the $k = 8$th nearest neighbor (III) are shown in Table 4.

c vs. k	4	5	6	7	8
1	65.61	80.49	88.88	90.28	92.03
2	89.64	92.66	**93.40**	93.36	92.99
3	90.99	93.19	93.01	92.95	92.73
4	91.40	93.02	92.95	92.75	92.52

Table 4: Variations of parameters c and k at modified mutual kNN graph construction on MPEG7-CE database after normalization with Zelnik-Manor and Perona (III).

3.3.2 Retrieval rates performance

Although the proposed normalization method gives a slightly minor retrieval score on the MPEG-7 database compared to the normalization of Zelnik-Manor and Perona **(III)**, the ranking within the true number of matches (first 19 neighbors to each query object) is absolutely competitive. Figure 4 gives an overview on the recognition within the first 40 matches and clearly approves the performance of our normalization strategy within the first 20 true cluster members.

3.4 Synthetic halfmoon dataset

In addition to the previously shown results on real datasets we want to give an impression on the neighborhood incorporation using our normalization scheme in corporation with the subsequent matrix analysis on a synthetic half-moon data set.
Figure 5 shows the retrieval for the bold query object when either Euclidean distances or our normalization scheme is applied. In addition to getting 100% correct retrieval rate, we can also distinguish between intra class objects (denoted by crosses) and extra class objects, which cannot be accessed by the modified mutual k nearest neighbor graph (denoted by circles). Therefore we conclude that our method may also be used for preprocessing i. e. in clustering or dimensionality reduction.

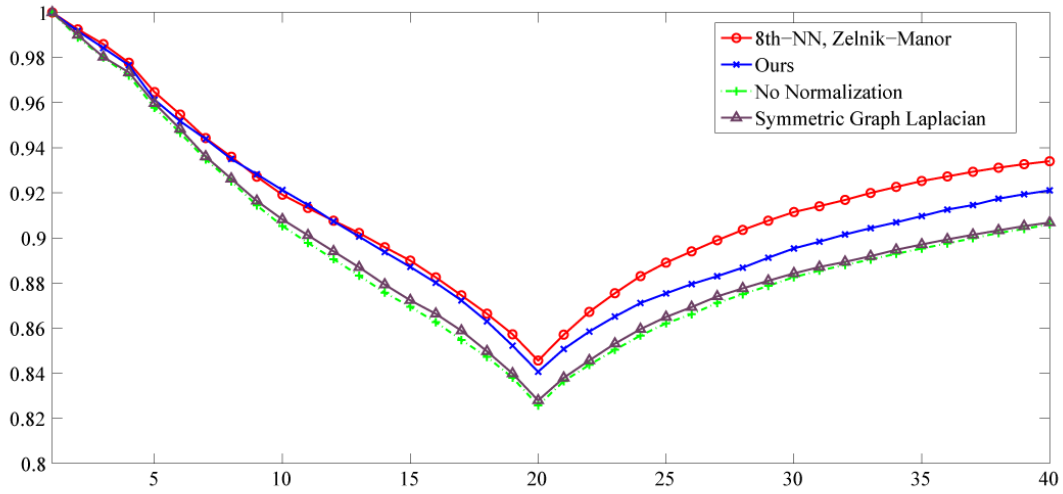

Figure 4: Mean retrieval rates vs. number of most similar ranked shapes from MPEG-7 data set.

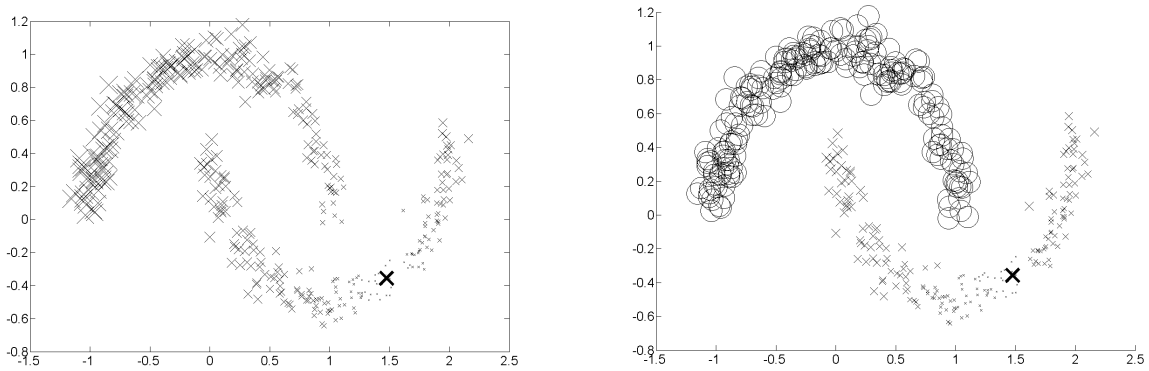

Figure 5: Retrieval rankings: Left figure shows ranking when Euclidean distances are applied. The right figure shows the retrieval after proposed normalization and analysis method. (Increasing marker size denotes increasing distances and thus decreasing ranking. Circles denote non-reachable data points.)

4 Conclusion

In this work we presented a two-way scheme for normalization and analysis of given affinity matrices to improve information retrieval. In the first step we replace the original object distances with normalized distances that are generated from neighborhood votes which incorporate the interdependence between similar objects. With this approach we naturally extend the idea of using the geodesic neighbor on the data manifold. In our analysis step we employed a modified mutual k nearest neighbor graph which allowed to capture the locally underlying object structures. Unlike other neighborhood graphs, the presented one enables modeling of regions that are composed from different scales and densities. The experimental section showed that our method significantly improves the retrieval rates on a variety of real life and synthetic data sets, i. e. achieving the highest ever reported score on the MPEG-7 database. Since normalization and analysis of the affinity matrix also constitutes the basis for applications like semi-supervised classification and dimensionality reduction, we will focus our future work on evaluating our proposed method in this area.

References

[1] Fan R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92) (Cbms Regional Conference Series in Mathematics)*. American Mathematical Society, February 1997.

[2] Bernd Fischer, Volker Roth, and Joachim M. Buhmann. Clustering with the connectivity kernel. In *Advances in Neural Information Processing Systems (NIPS)*, page 2004. MIT Press, 2004.

[3] Haibin Ling and D.W. Jacobs. Shape classification using the inner-distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(2):286–299, Feb. 2007.

[4] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2161–2168, 2006.

[5] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

[6] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. IEEE Intern. Conf. on Computer Vision (ICCV)*, pages 1470–1477, 2003.

[7] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[8] Xingwei Yang, Xiang Bai, Longin Jan Latecki, and Zhuowen Tu. Improving shape retrieval by learning graph transduction. In *Proc. European Conf. on Computer Vision (ECCV)*, pages 788–801, 2008.

[9] D. Yankov and E. Keogh. Manifold clustering of shapes. In *Proc. Intern. Conf. on Data Mining (ICDM)*, pages 1167–1171, Dec. 2006.

[10] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1601–1608. MIT Press, 2004.

[11] Xiaojin Zhu. *Semi-supervised learning with graphs*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2005.

[12] Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In Tom Fawcett, Nina Mishra, Tom Fawcett, and Nina Mishra, editors, *Proc. Intern. Conf. on Machine Learning (ICML)*, pages 912–919. AAAI Press, 2003.