Int J Multimed Info Retr (2012) 1:239–248
DOI 10.1007/s13735-012-0013-5

REGULAR PAPER

Fast shape retrieval using a graph theoretic approach

Chunyuan Li · A. Ben Hamza

Received: 27 March 2012 / Revised: 12 April 2012 / Accepted: 1 May 2012 / Published online: 19 May 2012

Abstract A critical issue in shape retrieval systems is that when a user submits a query shape, some shapes in the database are returned relatively often, while some are returned only when submitting specific queries. Intuitively, this phenomenon yields suboptimal retrieval accuracy. In this paper, we address the shape retrieval problem by casting it into the task of identifying "authority" nodes in an inferred similarity graph and also by re-ranking the shapes. The main idea is that the average similarity between a node and its neighboring nodes takes into account the local distribution, and therefore, helps modify the neighborhood edge weight, which guides the re-ranking. The proposed approach is evaluated on both 2D and 3D shape datasets, and the experimental results show that the proposed neighborhood induced similarity measure significantly improves the shape retrieval performance. Moreover, the computational speed of the proposed method is extremely fast.

Keywords Shape retrieval · Graph theory · Similarity · Re-ranking

1 Introduction

Searching shapes more accurately and faster is one of the most important goals in computer vision. In recent years, several approaches have been proposed to optimize shape retrieval systems, from designing smart descriptors [1–4], to explore suitable similarity measure methods [5]. However, in almost all these systems, the following phenomenon exists: when a user submits a query, some shapes in the database are returned relatively often, while some are returned only given certain special queries. For example, in Fig. 1, the statistics of retrieval results using shape contexts (SC) [1] on the MPEG-7 dataset [4] are depicted. There are 70 classes with 20 shapes for each. The frequency of a given shape is displayed as the sum of appearing times in the top 40 rankings when each shape takes the role of query. In Fig. 1a, the appearing frequency is shown, where the red points display the proper rate and the blue points represent the shapes that are returned too often or too rare. Obviously, there is a considerable number of shapes appearing with abnormal times. In particular, it can be observed that the 'device' shape shown in Fig. 1b is returned 156 times, but the 'glass' shape shown in Fig. 1c is returned only four times. This observation strongly suggests that shapes appearing with too many or very few times will damage the overall retrieval result.

To tackle this problem, we propose the Neighborhood Induced Similarity (NIS), which updates the original similarity based on the neighborhood of a shape before the final ranking. Traditional shape retrieval systems compute the pair-wise similarity among shapes, from which a global ordering can be derived. By separating ranking from similarity measurement, one can leverage ranking algorithms to generate a global ordering. Just like existing re-ranking algorithm for web page [6], image [7] and video [8] retrieval, our proposed method also takes the similarity/dissimilarity/distance matrix as the input, and outputs an optimized similarity matrix for the final ranking.

However, the difference in our setting is that we aim at eliminating the undesired phenomenon stated above. We present an approach from the perspective of a graph representation, where shapes are represented as nodes, and edges encode the similarity between shapes. In this way, searching shapes is formulated as label propagation on the graph. Therefore, the edge value is very critical, since it guides the

C. Li · A. Ben Hamza (✉)
Concordia Institute for Information Systems Engineering, Concordia University, Montréal, QC, Canada
e-mail: hamza@ciise.concordia.ca

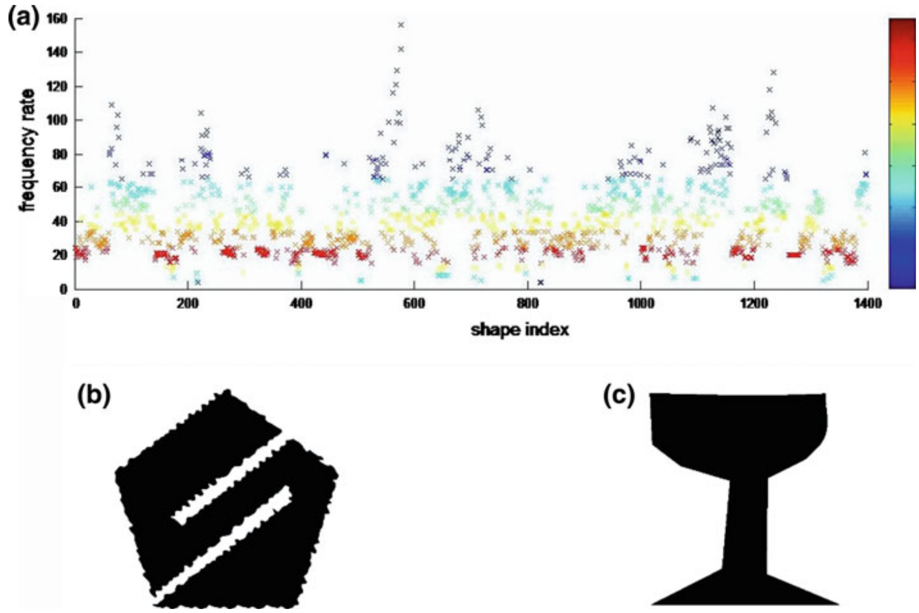

Fig. 1 Statistical results on the MPEG-7 dataset. **a** Appearing frequency, **b** 'device' shape (highest frequency), **c** 'glass' shape (lowest frequency)

propagation behavior. In our method, we consider all the nodes on the graph as a whole, and update the edge value by the average similarity of nodes; further improving the retrieval accuracy.

Extensive experiments on two standard 2D shape datasets, one trademark image dataset and one 3D shape dataset show that the returned frequency of shapes using our method is more sound and logical than that of existing re-ranking algorithms. Furthermore, our method provides an improved performance accuracy and it is extremely fast. For example, after the adoption of NIS with SC on the MPEG-7 dataset, the returned frequency of the 'device' shape drops from 156 to 60, and the 'glass' shape rises from 4 to 44, leading to improved retrieval rates from 86.79 to 90.49 %. The main advantages of our proposed method may be summarized as follows:

- It eliminates the abnormal frequency phenomenon for shape retrieval with a graph representation.
- It provides an improved accuracy and offers a very competitive time complexity.
- It is universal to arbitrary shapes, both 2D and 3D shapes.

2 Previous work

Shape retrieval techniques may be broadly classified into two main categories: traditional matching/retrieval methods and similarity learning methods [5]. Belongie et al. [1] introduced shape contest, which is a 2D histogram representation of a shape. Ling and Jacobs [2] proposed the inner distance which modifies shape contexts by considering the geodesic distance between contour points instead of the Euclidean distance, and thus significantly improves the retrieval and classification of articulated shapes. Wei et al. [11] extracted Zernike features to describe trademark shapes. Trademark images are very complex 2D shapes, and obtaining a high performance of trademark retrieval is of paramount importance to the industry. On the other hand, the Light Field Descriptor (LFD) [12] has been reported in the literature as one of the most efficient techniques for the retrieval of 3D rigid models [13]. Our method is, however, general and not limited to any particular similarity measure or representation.

Our work is partly motivated by Bai et al.'s work [5], which adopts a graph-based transductive learning algorithm to improve the shape retrieval results. The key idea of this distance learning algorithm is to replace the original shape distance with a distance induced by geodesic paths in the manifold of known shapes. In other words, each shape is considered in the context of other shapes in its class, and the class need not be known. In this paper, we propose NIS instead of Graph Transduction in a sense of improving the shape retrieval results.

There has been a significant body of work on similarity measures-based methods. Cheng et al. [14] proposed the sparsity induced similarity measure to improve the label propagation performance [15]. Jegou et al. [16] proposed the Contextual Dissimilarity Measure (CDM) to improve the image search accuracy through improving the symmetry of the k-neighborhood relationship. Our work reassigns edge weight in a fully connected graph by the average neighborhood weight, which is in a similar spirit as CDM. Other related works on similarity measure-based methods can be found in [17]. More recently, Bronstein et al. [9] proposed a non-rigid shape retrieval approach using bags of features based on the heat kernel signature (HKS) [10], which is defined as the diagonal of the heat kernel of the Laplace–Beltrami operator on a manifold. HKS is a local shape descriptor that enjoys nice properties, including robustness to small perturbations of the shape, efficiency, and invariance to isometric transformations. However, HKS depends on the time parameter, which needs to be set a priori. In addition, the discrete heat kernel requires eigendecomposition of a typically large Laplace–Beltrami matrix. Thus, finding the eigenvalues and eigenfunctions of such a large matrix is often computationally expensive. In addition, the choice of the vocabulary size, the time parameter, and the number of eigenvalues/eigenfunctions can have an impact on the performance of the HKS-based retrieval algorithm.

3 Neighborhood induced similarity measure

In traditional shape retrieval systems, a user usually employs some distance function to compute the pair-wise similarity between two shape features, and assumes that the more similar two shapes are, the smaller their difference is. For a given query, these systems rank the shapes in the dataset as a list according to the pair-wise similarity, and present to the user several top rankings in the returned list.

3.1 Problem formulation

To measure the number of returned times of a shape in a given dataset, we first introduce the concept of *appearing frequency* of a shape. Suppose there are N shapes in a dataset, and let us consider the top t ranking shapes $R_t(n)$ in the returned list $L(n)$ of a query shape Q_n, $1 \leq n \leq N$. Obviously, the cardinality $|R_t(n)| = t$ of the set $R_t(n)$ is constant within the t-highest ranking framework. The appearing frequency of Q_n is then defined as follows:

$$f(n) = \sum_{i=1}^{N} \sum_{j \in R_t(n)} \delta_{i,j} \tag{1}$$

where

$$\delta_{i,j} = \begin{cases} 1, & \text{if } Q_n \text{ is the } j\text{-th returned shape} \\ & \quad \text{of the query } Q_i \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

We can observe that some shapes have high frequency rate, while others are returned only when submitting specific queries. These shapes are referred to as *over-returned shapes* and *never-returned shapes*, respectively, which are defined for a given neighborhood size. Both of them are considered abnormal shapes or 'bad shapes' in a shape retrieval system. Our goal is to decrease the number of these abnormal shapes. In other words, we hope that the frequency rates of each shape in the dataset would be the same constant which is relative to $|R_t(n)|$.

The main idea of our proposed method is that we would like the k-neighborhoods to have a similar distance in order to approach the same frequency rate.

3.2 Proposed neighborhood induced similarity algorithm

Let $D = (d_{ij})$ be a distance matrix computed by some shape function. We formulate the shape retrieval problem as a form of propagation on a graph, where a node's label propagates to the neighboring nodes according to their proximity. In this process, we fix the label on the query shape. Thus, the query shape acts like a source that pushes out the label to other shapes. Intuitively, we want shapes that are similar to have the same label.

We create a graph $G = (\mathcal{V}, \mathcal{E})$ where the node set $\mathcal{V}$ represent all the shapes in the dataset, both query and the others. The element of the edge set $\mathcal{E}$ represent the similarity between nodes. We propagate the labels through the edges. Larger edge weights allow labels to travel through more easily. The propagation process stops when a user-specified number of nodes are labeled. Likewise, the frequency of a node $\boldsymbol{v}_i \in \mathcal{V}$ is defined as the sum of the labeled times after each node in the graph propagates its label to its neighborhood. Interestingly, we find that the frequency of a node is equal to the degree of the node if we cut off the edges that no label travels. A subgraph of the newly obtained graph G_0 includes some dense graph G_{dense} and sparse graph G_{sparse}. Obviously, G_{dense} consists of nodes with high frequency (or degree) V_{high}, and G_{sparse} consists of nodes with low frequency V_{low}. Viewed in this fashion, our target is to obtain a well-distributed graph.

Now assume that the graph is fully connected with the following weights computed by a Gaussian kernel:

$$w(\boldsymbol{v}_i, \boldsymbol{v}_j) = \exp\left(-\frac{d_{ij}}{\alpha^2}\right), \tag{3}$$

where α is a bandwidth hyper-parameter and it is determined empirically. In the sequel, we set α to 100.

The k-nearest neighbors of a given node $\boldsymbol{v}_i$ are the nodes $\text{NN}_k(i)$, in which the nodes $\boldsymbol{v}$ and $\boldsymbol{v}_i$ are connected by an e.g., if the edge weight between $\boldsymbol{v}$ and $\boldsymbol{v}_i$ is among the k-th largest from $\boldsymbol{v}_i$ to other nodes, i.e.

$$\text{NN}_k(i) = \{\boldsymbol{v} : \max_k w(\boldsymbol{v}_i, \boldsymbol{v})\} \tag{4}$$

The above-mentioned problem of frequency rate suggests a solution which reassigns weight. Intuitively, we would like the k-neighborhoods to have similar weights in order to eliminate 'bad shapes'.

Let us consider the neighborhood of a given node defined by its $|\text{NN}_k(i)|$ nearest neighbors. The value k is a compromise between computation cost and quality of retrieval result. The larger the value of k, the more expensive the computation. In general, k needs to be greater than 20 to prevent the system from being over constrained due to possible noise in the original measure.

We define the neighborhood weight or similarity $s(i)$ as the mean weight of a given node $\boldsymbol{v}_i$ to the nodes of its neighborhood:

$$s(i) = \frac{1}{k} \sum_{\boldsymbol{x} \in \mathrm{NN}_k(i)} w(\boldsymbol{v}_i, \boldsymbol{x}) \tag{5}$$

and it is computed for each node. Subsequently, we define a new weight between two nodes as follows:

$$w^{\star}(\boldsymbol{v}_i, \boldsymbol{u}_k) = w(\boldsymbol{v}_i, \boldsymbol{u}_k) \frac{\bar{s}}{(s(i)s(j))^{1/2}}, \tag{6}$$

where $\bar{s}$ is the geometric mean neighborhood similarity obtained by

$$\bar{s} = \prod_i s(i)^{1/n}. \tag{7}$$

Thus, we reassign the graph weight and propagate the query label according to the new weight. Note that the terms $\bar{s}$ and $s(i)$ do not impact the nearest neighbors of a given node.

3.3 Relation to contextual dissimilarity measure

Given the visual word vector $\boldsymbol{\xi}_j$ of a query, similar images in the database are represented by vector(s) $\boldsymbol{\xi}_j$ minimizing $d(\boldsymbol{\xi}_i, \boldsymbol{\xi}_j)$, where the relation $d(\cdot, \cdot)$ is a distance on the visual word vector space. Note that the weighting scheme previously described can be seen as part of the distance definition. The contextual dissimilarity measure (CDM) updates the given distance $d(\cdot, \cdot)$ (e.g., Manhattan distance) by applying two weighting factors η_i and η_j that depend on the vectors and between which the distance is computed:

$$\mathrm{CDM}(\boldsymbol{\xi}_i, \boldsymbol{\xi}_j) = d(\boldsymbol{\xi}_i, \boldsymbol{\xi}_j)\eta_i \eta_j. \tag{8}$$

Our proposed NIS approach is a graph representation. We encode the edge weights as similarity between shapes, and the retrieval is a one-step label propagation from the query to other shapes. Our goal is to update the edge weight instead of distance. It is worth pointing out that CDM is iterative, whereas the proposed method is non-iterative. Moreover, we aim at re-ranking the shape retrieval result, though the edge weighting factor is learned in a similar fashion as CDM.

4 Applying neighborhood-induced similarity

The goal of shape retrieval is to recall shapes that are relevant to the query. In this section, we explain the intuition behind the use of NIS to improve the relevancy of shape ranking results.

4.1 Distance measure

A reliable measurement of shape similarity is critical to the performance of NIS since it determines the underlying graph structure. Measuring pair-wise similarity can be categorized into two main classes: (1) compute direct difference in features extracted from shapes, which are invariant to rotation and robust to certain degree of deformation, such as skeletons, moments, and Fourier descriptors; (2) perform matching to find the detailed point-wise correspondences to compute these differences [1,2]. Our algorithm is, however, general and not limited to any particular similarity measure or representation.

4.2 Nodes with high or low frequency

In the weighted similarity graph, NIS works on nodes with high or low frequency to improve the shape re-ranking results. We illustrate the process in Fig. 2. In a dense graph, for V_{high}, there are relatively more query nodes that can propagate their label to it, which indicates that edge weights between V_{high} and other nodes are larger. However, we could obtain several reduced weights by NIS since the overall average weight is smaller. Thus, a more concise and relevant set of query candidate shapes is searched. In a sparse graph, for V_{low}, in the opposite way, we increase the candidate set by promoting the edge weight.

5 Experimental results

In this section, we conduct extensive experiments to validate the performance of the proposed method. We first investigate the impact of the neighborhood size and then show the improved shape retrieval results by handling the frequency problem on 2D shapes. Then, we conduct more experiments on complex 2D shapes from a trademark image dataset, and also on a 3D shape dataset. In our experiments, we used SC, IDSC for 2D shapes, Zernike features for the trademark images, and LFD for 3D models to generate the distance measure and construct the similarity graph. Finally, we show that our method is computationally fast by deriving the overall computational cost.

5.1 Results on 2D shapes

Datasets: The experiments are performed on two shape data sets, the MPEG-7 dataset and Tarri dataset. The former consists of 1,400 silhouette images grouped into 70 categories. Each category contains 20 different shapes. The latter is comprised of 50 object categories, 20 shapes per category; so there are 1,000 images in total.

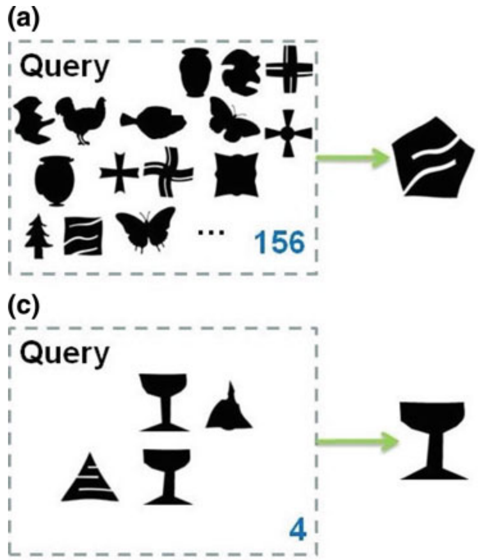

Fig. 2 Two nodes in the similarity graph, e.g., 'device' and 'glass' shapes. By updating the edge weight, NIS identifies a highly relevant set of candidate query shapes. **a** 156 query shapes for 'device', 15 shapes are shown, **b** 60 query shapes for 'device' with NIS, 8 shapes are shown; **c** 4 query shapes for 'glass', **d** 44 query shapes for 'glass' with NIS, 8 shapes are shown

Evaluation criterion: The performance is measured using the so-called bull's eye score to evaluate the accuracy of the proposed method. Every shape in the dataset is compared to all other shapes, and the number of shapes from the same class among the 40 most similar shapes is reported. The bull's eye retrieval rate is the ratio of the total number of shapes from the same class to the highest possible number (which is 20 × 1,400 on MPEG-7). Thus, the best possible rate is 100 %.

5.1.1 Neighborhood size of NIS

One important parameter of the proposed method is the neighborhood size, k. We evaluate the accuracy of the method for different values of k. It results in disparate accuracy when the size is relatively small. Figure 3 shows the percentage of correct results on MPEG-7 dataset. We observe that the accuracy surges significantly when the size of neighborhood is small, namely 7 or 8 in this case, then slightly decreases as the neighborhood enlarges. The peak accuracy is obtained when the neighborhood size is equal to 30. Considering that a small neighborhood contributes to lower computational cost, while a larger one provides a better accuracy in a limited range. We set the size k to 30 in the experiments as a trade-off value.

5.1.2 Improving 2D shape retrieval

We evaluate the NIS performance for shape retrieval on MPEG-7 dataset and Tarri dataset. The proposed method is able to improve the shape contexts retrieval rates from

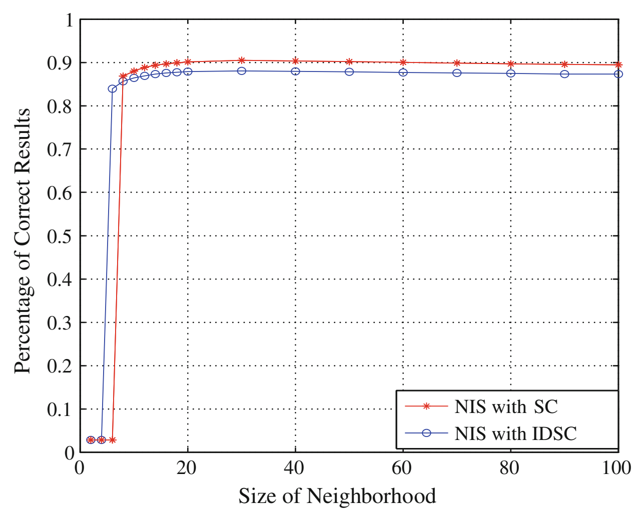

Fig. 3 Impact of neighborhood size for the MPEG-7 dataset

86.79 to 90.49 %, and the IDSC from 84.68 to 88.05 % on the MPEG-7 dataset. It also improves shape contexts retrieval rates from 94.17 to 97 % on Tarri dataset.

In order to visualize the gain in retrieval rates by our method, we conduct a series of experiments on the MPEG-7 dataset. Here we vary the neighborhood size k from 1 to 40 instead of setting it to 40 as in the bull's eye evaluation. Figure 4 shows that our similarity measure outperforms the Shape Contexts and IDSC on both MPEG-7 and Tarri datasets. Note that each class has 20 shapes, so the curve increases for $k > 20$.

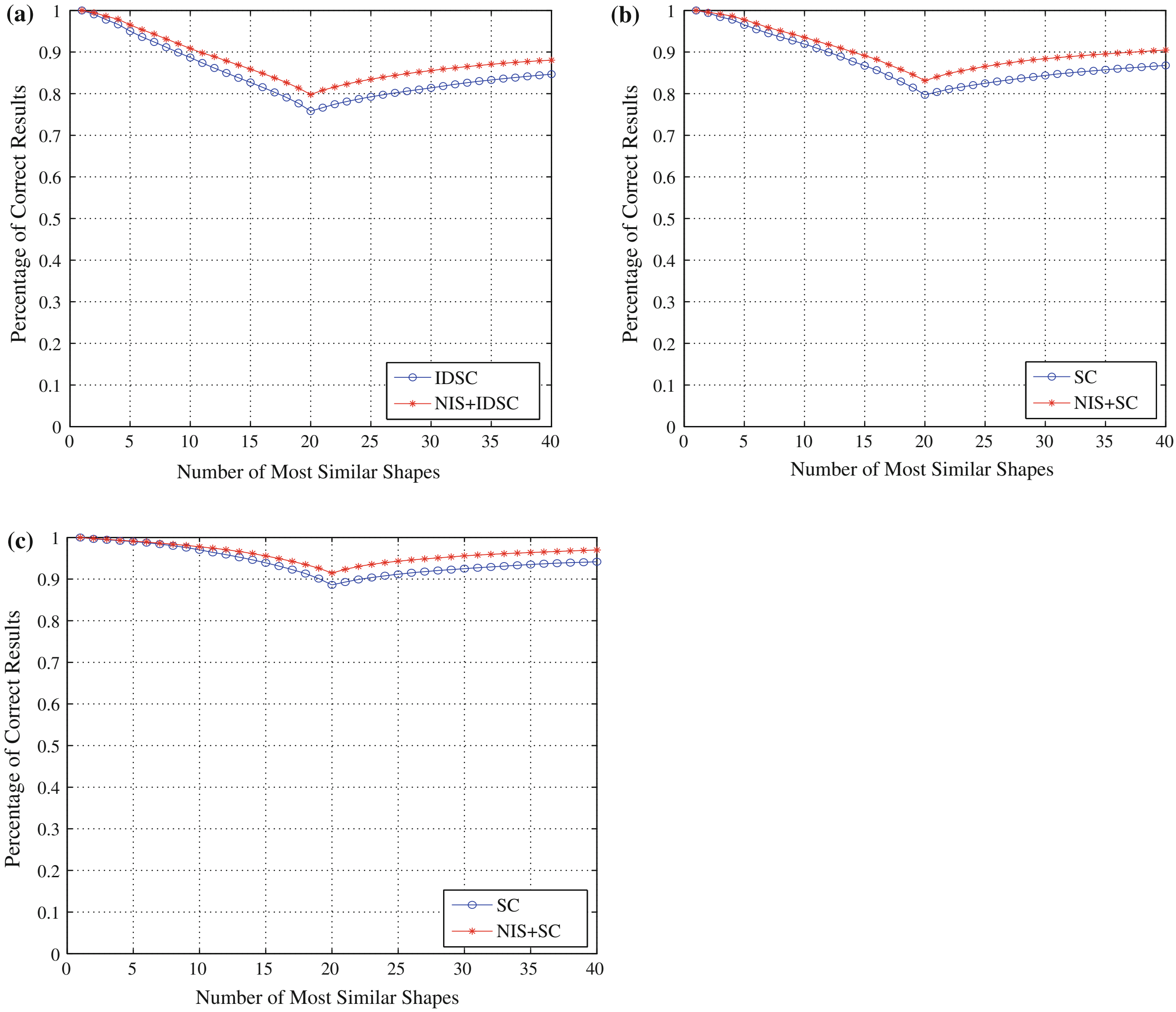

Fig. 4 **a** A comparison of retrieval rates between IDSC (*blue circle*) and by the proposed method (*red asterisk*) for MPEG-7 dataset. **b** A comparison of retrieval rates between SC (*blue circle*) and by the proposed method (*red asterisk*) for MPEG-7 dataset. **c** A comparison of retrieval rates between SC (*blue circle*) and by the proposed method (*red asterisk*) for Tarri dataset (color figure online)

Figure 5 illustrates some typical queries for which NIS significantly improves the results on MPEG-7 dataset. We place the query in the first column, and the most relevant returns in a row beside it excluding the query itself. For each group of comparison, the query with no NIS (upper row) is often irrelevantly retrieved, whereas the query with NIS (second row) returns the most relevant shapes. It shows that only one result is correct for the query 'glass'. It instead retrieves eight elephants as the most similar shapes among the top ten, since it confuses the glass bottom with the elephant crus and trunk. However, the proposed method deliberately increases the distance from a global perspective. It can accurately distinguish the difference with only one mistake.

The results of the query 'lizard' are equally convincing. No correct shape is returned besides the query itself without NIS, and our method correctly retrieves seven among top ten. For the results of the query 'tree', it mistakenly retrieves camel for 'tree', since the camel's hump is more similar to the serrated shape of the pine tree. Nevertheless, all retrieval results are correct using the new distance learned via our method.

The comparison of frequency rate is illustrated in Fig. 6, where the red squares denote the proposed method and the blue crosses otherwise. Since each class has 20 shapes and we consider the top 40 for each query, the optimal frequency rate ranges from about 20–40 and the ideal dots form a band-like

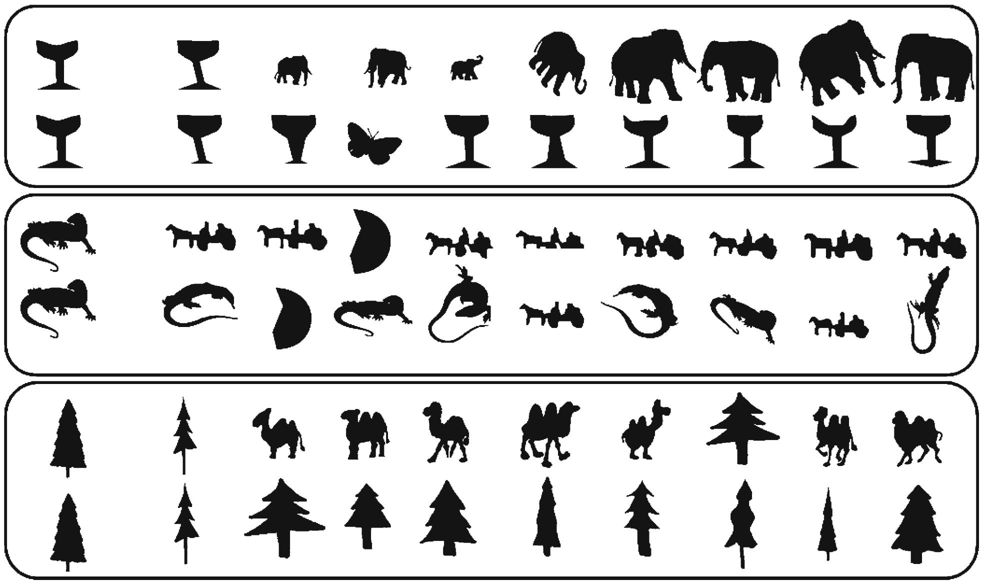

Fig. 5 First column shows query shapes. Remaining columns show the most similar shapes retrieved by SC (*odd row numbers*) and by our method (*even row numbers*)

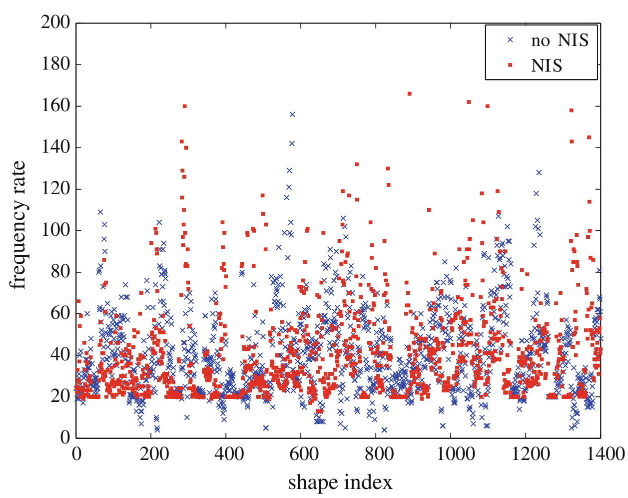

Fig. 6 Frequency rate without NIS (*blue points*) and with NIS (*red points*) (color figure online)

region which is parallel to the x-axis. The performance is considered unsatisfactory provided that the deviation of frequency rate from the region is large. After all, a large deviation implies over-returned and never-returned. We observe that the rate of query shapes gather towards the ideal region in a considerable number. The most notable improvement of the proposed method is that the frequency rates seldom fall below the boundary of 20, unlike the frequency rate without NIS. This clearly demonstrates that NIS can effectively regulate the overall similarity of each shape and reduce the disturbance of 'bad shapes'.

5.2 Results on more complex 2D shapes

In this experiment, we investigate the NIS performance for improving the retrieval results on Wei's Trademark Image dataset with 1,003 images in 14 classes [11]. These include Apple, Fan, Abstract Circle1, Abstract Circle2, Encircled Cross, Abstract Saddle, Abstract Saddle, Abstract Sign, Triangle, Bat, Peacock, Abstract Flowers, Rabbit, Snow Flake, and Miscellaneous. Figure 7 displays sample images from the Trademark Image dataset. Traditional descriptor-based methods applied on these complex shapes still maintain a modest performance. However, the industry is in urgent need for a satisfactory retrieval performance because it will save human consumption of comparing one by one to avoid reduplication. In this setting, the neighborhood size is set to 85.

In our comparative analysis, we used the Precision/Recall curve to measure the retrieval performance. Ideally, this curve should be a horizontal line at unit precision. For each query image, we use the first 108 return trademark images with descending similarity rankings (i.e., ascending Euclidean distance ranking), dividing them into 9 groups accordingly. However, in order to obtain a more objective picture of the performance, we plot the average performance of 20 query images of the same class. We show the result in Fig. 8, where Zernike features [3] are first extracted to calculate the original distance matrix, and the proposed algorithm is used to obtain the accuracy-improved matrix. Again, the overall retrieval performance is improved by NIS.

Based on the above experimental results, our algorithm is validated to improve shape retrieval in 2D, both on standard

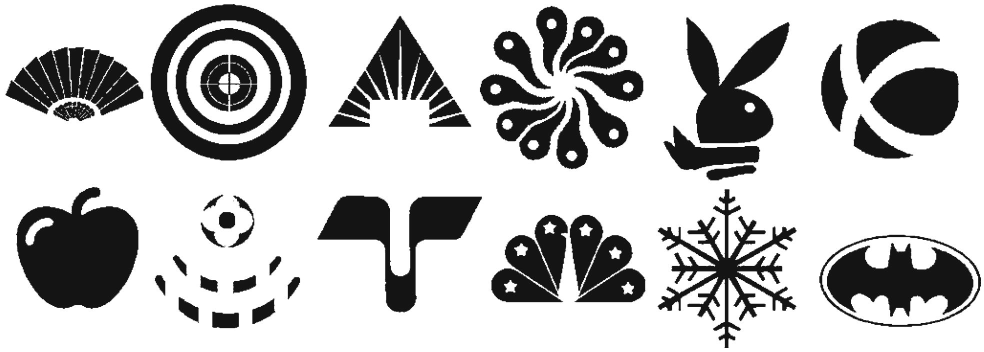

Fig. 7 Sample images from the trademark shape dataset

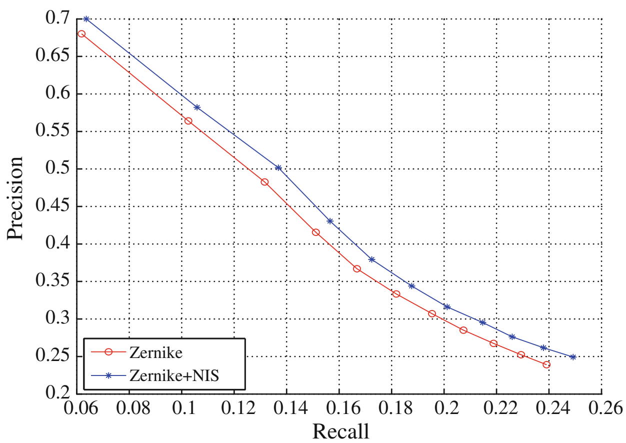

Fig. 8 NIS improves the overall retrieval performance for the trademark image dataset. *Blue line* is with NIS and *red line* is without NIS (color figure online)

datasets and on trademark images with large variation among individuals.

5.3 Results on 3D shapes

We tested the performance of the proposed matching algorithm using the McGill Shape Benchmark (http://www.cim.McGill.ca/shape/benchmark). This publicly available benchmark database provides a 3D shape repository, which contains 255 objects that are divided into ten categories, namely, 'Ants', 'Crabs', 'Spectacles', 'Hands', 'Humans', 'Octopuses', 'Pliers', 'Snakes', 'Spiders', and 'Teddy Bears'. Sample models from this database are shown in Fig. 9.

The evaluation of the retrieval results is based on the following quantification measures. These measures range from 0 to 100 %, and higher values indicate better performance.

- *Nearest neighbor (NN):* The percentage of queries where the closest match belongs to the query's class.
- *First tier (FT):* The recall for the $(\kappa - 1)$ closest matches, where κ is the cardinality of the query's class.
- *Second tier (ST):* The recall for the $2(\kappa - 1)$ closest matches, where κ is the cardinality of the query's class.
- *Discounted cumulative gain:* A statistic that correct results near the front of the retrieval list are weighted more heavily than correct results near the end under the assumption that a user is most interested in the first results.

We compare our method with the 3D Light Field Distribution (LFD) method. As in the previous experiments, the distance matrix calculated via LFD is chosen as the input of NIS, and the neighborhood size is set to 48. The corresponding scores of each method for each class of the database as well as the overall scores for the complete database are shown in Table 1. We render numbers in bold if our method is superior or equivalent to LFD. Obviously, in most cases, NIS has a positive effect on 3D shapes re-ranking. Though not all the entries with NIS outperform LFD in the comparative results, it is understandable that NIS works in a given statistical range. For example, 'Hands', the worst class according to the result, will still lead to effective shape re-ranking based upon the First Tier measure.

5.4 Computational cost

We compared the computation time of our method with the graph transduction approach [5]. Both algorithms are implemented in MATLAB running on a Pentium Dual Core 1.73 GHz PC. The MATLAB implementation of the graph transduction is publicly available in (http://happyyxw.googlepages.com/democodeeccv). The entire retrieval process of the proposed algorithm takes about 5.05 s, which is extremely faster than the graph transduction's 2.18 h.

We also analyzed the computational complexity of the proposed similarity measure approach. The time to compute the neighborhood distance is $\mathcal{O}(kN)$, to compute geometric

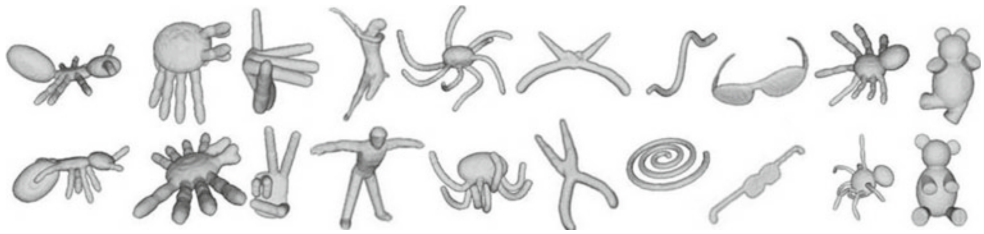

Fig. 9 Sample shapes from McGill Articulated Shape Database. Only two shapes for each of the 10 classes are shown

Table 1 Quantitative measure scores of the retrieval methods

# Queries	Method	NN (%)	FT (%)	ST (%)	DCG (%)
Overall database	Ours	84.16	46.26	62.18	83.72
	LFD	84.61	44.69	59.28	82.74
Ants	Ours	90	53.6	77.78	88.42
	LFD	93.33	54.56	76.11	89.33
Crabs	Ours	93.33	50.89	65.33	86.33
	LFD	93.33	45	60.11	84.08
Spectacles	Ours	76	51.52	66.08	88.45
	LFD	100	50.56	65.60	88.34
Hands	Ours	80	28.25	40.50	75.36
	LFD	90	28	42.25	75.44
Humans	Ours	79.31	40.79	54.94	81.26
	LFD	79.31	39.35	53.03	80.69
Octopuses	Ours	60	26.88	41.93	72.30
	LFD	48	24	35.36	68.47
Pliers	Ours	100	75.50	87.25	97.58
	LFD	100	75.25	87.25	97.47
Snakes	Ours	76	26.14	33.92	71.68
	LFD	68	20.64	25.28	66.53
Spiders	Ours	70.97	41.94	65.45	81.18
	LFD	74.12	42.77	65.35	82.35
Teddy bears	Ours	100	66.50	86.50	95.64
	LFD	100	66.75	82.50	94.59

mean is $\mathcal{O}(N)$, and to update the weight is $\mathcal{O}(N)$. There is an additional cost for ranking the scores at the end, which is $\mathcal{O}(k \log N)$. Thus, the total complexity of our method amounts to $\mathcal{O}(k \log N + (k+2)N)$. since $k \ll N$, the overall time complexity of our algorithm is bounded by $\mathcal{O}(N^2)$. It is worth pointing out that the complexity of our approach is at least one order smaller than the complexity $\mathcal{O}(TN^3)$ of the graph transduction algorithm, where T is the number of iterations.

6 Conclusions and future work

In this paper, we proposed a novel similarity measure for eliminating abnormal shapes in shape retrieval systems. The proposed NIS measure takes into account the hidden local structure by using the average neighborhood similarity in a graph representation. We tested the proposed similarity measure on the commonly used MPEG-7 and Tarri datasets, a trademark image dataset, and a 3D shape dataset. The experimental results demonstrated the efficiency of the proposed method both in 2D and 3D, even on shapes with large variations. In addition, we showed that the proposed method is computationally fast.

Future research directions include further exploration of the frequency problem on shape classification as well as clustering. We also plan to combine sparse representation with the proposed method in order to achieve much better retrieval results.

Acknowledgments The authors would like to thank the anonymous reviewers and the Section Editor for helpful and very insightful comments. This work was supported in part by an NSERC Discovery grant.

References

1. Belongie S, Malik J, Puzicha J (2002) Shape matching and object recognition using shape contexts. IEEE Trans Pattern Anal Mach Intell 24(4):509–522
2. Ling H, Jacobs D (2007) Shape classification using the inner-distance. IEEE Trans Pattern Anal Mach Intell 29(2):286–299
3. Li S, Lee M-C, Pun C-M (2009) Complex Zernike moments features for shape-based image retrieval. IEEE Trans Syst Man Cybern 39(1):227–237
4. Sebastian TB, Klein PN, Kimia BB (2004) Recognition of shapes by editing their shock graphs. IEEE Trans Pattern Anal Mach Intell 26(5):550–571
5. Bai X, Yang X, Latecki LJ, Liu W, Tu Z (2010) Learning context sensitive shape similarity by graph transduction. IEEE Trans Pattern Anal Mach Intell 32(5):861–874
6. Brin S, Page L (1998) The anatomy of a large-scale hypertextual Web search engine. Proc Int Conf World Wide Web 7 30(1–7):107–117
7. He X, Ma W-Y, Zhang H (2002) Imagerank: spectral techniques for structural analysis of image database. Proc IEEE Int Conf Multimedia Expo 1:25–28
8. Latecki L, Lakamper R, Eckhardt U (2000) Shape descriptors for non-rigid shapes with a single closed contour. Proc IEEE Conf Comput Vis Pattern Recognit 1:424–429
9. Bronstein AM, Bronstein MM, Guibas L, Ovsjanikov M (2011) Shape Google: geometric words and expressions for invariant shape retrieval. ACM Trans Graph 30(1)

10. Sun J, Ovsjanikov M, Guibas L (2009) A concise and provably informative multi-scale signature-based on heat diffusion. Comput Graph Forum 28(5):1383–1392
11. Wei C-H, Li Y, Chau W-Y, Li C-T (2009) Trademark image retrieval using synthetic features for describing global shape and interior structure. Pattern Recognit 42(3):386–394
12. Chen D-Y, Tian X-P, Shen Y-T, Ouhyoung M (2003) On visual similarity based 3D model retrieval. Comput Graph Forum 22(3):223–232
13. Siddiqi K, Zhang J, Macrini D, Shokoufandeh A, Bouix S, Dickinson S (2008) Retrieving articulated 3-D models using medial surfaces. Mach Vis Appl 19(4):261–275
14. Cheng H, Liu Z, Yang J (2009) Sparsity induced similarity measure for label propagation. Proc IEEE Int Conf Comput Vis, Kyoto, Japan, pp 317–324
15. Xiaojin Z (2005) Semi-supervised learning with Graphs, PhD thesis, CMU
16. Jegou H, Schmid C, Harzallah H, Verbeek J (2010) Accurate image search using the contextual dissimilarity measure. IEEE Trans Pattern Anal Mach Intell 32(1):2–11
17. Yu J, Amores J, Sebe N, Radeva P, Tian Q (2008) Distance learning for similarity estimation. IEEE Trans Pattern Anal Mach Intell 30(3):451–462
18. Jing Y, Baluja S (2008) VisualRank: applying PageRank to large-scale image search. IEEE Trans Pattern Anal Mach Intell 30(11):1877–1890
19. Shilane P, Min P, Kazhdan M, Funkhouser T (2004) The princeton shape benchmark. In: Proceedings of shape modeling, International, pp 167–178