Pattern Recognition 83 (2018) 353–364

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Improving context-sensitive similarity via smooth neighborhood for object retrieval

Song Bai[a], Shaoyan Sun[b], Xiang Bai[a,*], Zhaoxiang Zhang[c], Qi Tian[d]

[a] Huazhong University of Science and Technology, China
[b] University of Science and Technology of China, China
[c] CAS Center for Excellence in Brain Science and Intelligence Technology, CASIA, China
[d] University of Texas at San Antonio, United States

ARTICLE INFO

Article history:
Received 30 October 2017
Revised 15 March 2018
Accepted 1 June 2018

Keywords:
Object retrieval
Context-sensitive similarity
3D shape
Re-ranking
Rank aggregation

ABSTRACT

Due to the ability of capturing the geometry structure of data manifold, context-sensitive similarity has demonstrated impressive performances in the retrieval task. The key idea of context-sensitive similarity is that the similarity between two data points can be more reliably estimated with the local context of other points in the affinity graph. Therefore, neighborhood selection is a crucial factor for those algorithms, which affects the performance dramatically. In this paper, we propose a new algorithm called Smooth Neighborhood (SN) that mines the neighborhood structure to satisfy the manifold assumption. By doing so, nearby points on the underlying manifold are guaranteed to yield similar neighbors as much as possible. Moreover, SN is adjusted to tackle multiple affinity graphs by imposing a weight learning paradigm, and this is the primary difference compared with related works which are only applicable with one affinity graph. Finally, we integrate SN with Sparse Contextual Activation (SCA), a representative context-sensitive similarity proposed recently. Extensive experimental results and comparisons manifest that with the neighborhood structure generated by SN, the proposed framework can yield state-of-the-art performances on shape retrieval, image retrieval and 3D model retrieval.

1. Introduction

Object retrieval [1] is an important topic in pattern recognition [2], computer vision [3], multimedia computing [4,5] and machine learning, which has been investigated for decades. A typical retrieval system receives a query data as its input, and outputs the searching results which are expected to be visually similar to the given query. Therefore, the crucial issue in object retrieval is to define a reliable similarity between the query object and the database elements. In most cases, visual descriptors that are robust to common deformations (e.g., rotation, occlusion, illumination) are designed to assign each object a vectorial representation. Then, the pairwise matching between objects can be done in the Euclidean distance.

In recent years, context-sensitive similarity [6,7] has attracted much attention due to its superior performances in object retrieval, which does not solely rely on the pairwise matching. These approaches have a very diverse nomenclature, such as contextual dissimilarity measure [8], graph transduction [9–11], affinity learning [12,13], ranking list comparison [2,14,15], re-ranking [16–18]. However, the inherent principles of most those algorithms are almost the same, that is, the similarity between two data points can be more accurately measured by taking the underlying manifold structure into account. In order to specify the differences among them systematically, Donoser and Bischof, [19] provide a generic framework called Diffusion Process and a thorough comparison of most aforementioned algorithms experimentally. Diffusion process is usually operated on an affinity graph, with the nodes representing data points and the edge weights denoting the pairwise similarities between two adjacent nodes. In fact, the affinity graph defines a data manifold implicitly, then the similarities are diffused along the geodesic path of the manifold.

As one of the most important conclusions quoted from Donoser and Bischof [19], it is crucial to constrain the diffusion process locally, since diffusion process is susceptible to noise edges in the affinity graph. The experimental observation supports the "locality" assumption in manifold learning [20], that each data point and its neighbors lie on a linear patch of the manifold. It means that only quite short distances are reliable since they tend to associate

* Corresponding author.
E-mail addresses: songbai.site@gmail.com (S. Bai), sunshy@mail.ustc.edu.cn (S. Sun), xbai@hust.edu.cn (X. Bai), zhaoxiang.zhang@ia.ac.cn (Z. Zhang), qi.tian@utsa.edu (Q. Tian).

https://doi.org/10.1016/j.patcog.2018.06.001
0031-3203/

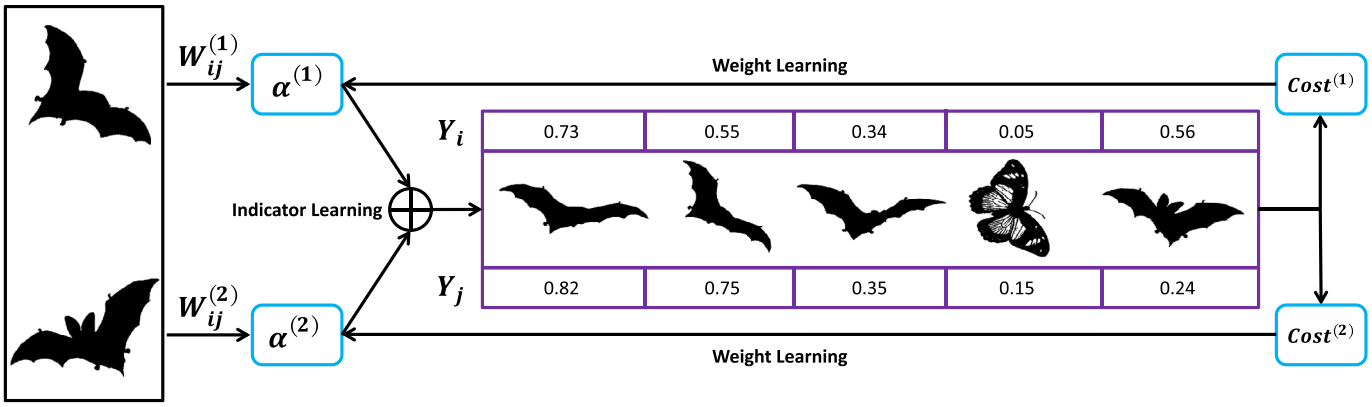

Fig. 1. The illustration of the proposed method with two similarity measures. For each pair of objects (x_i, x_j) having similarity $W_{ij}^{(v)}$ $(v = 1, 2)$, we learn their indicator functions on neighbor selections, which is constrained by the similarities. Then, the resulted costs can be used, in turn, to learn the weights of those similarities.

with short geodesic distances along the data manifold. As a consequence, the nodes that diffusion process selects to spread the similarities on the affinity graph are usually the neighbors of the query which have large similarities (or small dissimilarities) with it. Hence, it is of great importance to construct robust neighborhood structures so that diffusion process can be performed in a proper way.

The simplest way to establish the neighborhood structure is k-Nearest Neighbor (kNN) rule. Given a certain query, kNN rule selects K nodes with the largest edge weights to the query as its neighborhood. Some variants of kNN are also proposed, such as ϵ-neighbors, symmetric kNN, Mutual kNN [21] (also named as reciprocal kNN in [22,23]). As extensively proven in [21], kNN is prone to including noise edges and nodes, thus leading to unsatisfactory retrieval performances. To overcome its defect, Dominant Neighborhood (DN) is proposed in [12] based on the analysis of dominant sets, and Consensus of kNN (CN) is proposed in [24] by exploiting the consensus information of kNN.

However, although these neighborhood analysis algorithms are embedded into some variants of diffusion process, they themselves do not capture the geometry of the data manifold. That is to say, they cannot preserve the property of *local consistency* that nearby points on the manifold are guaranteed to yield the same neighbors. For example, it usually occurs that two points belong to the same dense cluster, while they have no common neighbors if kNN rule or DN is used. In context-based retrieval, this problem is first proposed in [25], and later emphasized again in [19]. Nevertheless, they only alleviate the problem to a certain extent by localizing the diffusion process using kNN on both sides (query side and database side), and do not intend to tackle the problem seriously.

Moreover, previous works are only applicable with one affinity graph. When multiple affinity graphs are given, it becomes more challenging to accurately construct the neighborhood structures. Therein, the difficulties lie in two aspects. First, it is problematic to determine the weights of affinity graphs, which can distinguish the discriminative capacity of different similarity measures. Second, it is hard to aggregate the neighborhood structures generated with different affinity graphs, especially considering that retrieval is usually defined as an unsupervised task without prior knowledge. Of course, one can simply use a linear combination of multiple affinity graphs with equal weights. However, as demonstrated in our experiments, it is a suboptimal solution since the complementary nature among multiple similarities is neglected.

In this paper along with its earlier conference version [26], we propose an algorithm called Smooth Neighborhood (SN) specifically for neighborhood structure mining. As illustrated in Fig. 1, the motivation of SN is that the indicator functions can be defined to reveal the behavior of neighbor selection on affinity graphs thus yielding a selecting cost for each graph, then the resulted costs can be used, in turn, to learn the weights of those affinity graphs in an unsupervised manner. Apart from related works, our primary contributions can be divided into three parts:

1. SN enables the neighbor selection to vary smoothly with respect to the local geometry of the data manifold, thus the input similarity can be sufficiently reflected in the behavior of neighbor selection.
2. SN is suitable to deal with more than one affinity graph. It learns a shared neighborhood structure and the importance of multiple affinity graphs in a unified framework. Therefore, the weight learning procedure and the neighborhood aggregation procedure can be done simultaneously.
3. Instead of using some heuristic rules that stem from empirical observations (e.g., Mutual kNN), we give a formal formulation to SN and derive an iterative solution to the optimization problem with proven convergence.

Compared with the conference version, the work (1) gives a deeper analysis on the motivation and the difference from relevant works; (2) supplements the properties of SN, such as the proof of convergence and convexity; (3) provides more thorough experimental evaluations with different types of data pattern, such as 3D model retrieval.

The rest of paper is organized as follows. In Section 2, we review some representative algorithms which have a close relationship with SN. The formulation and optimization of SN are given in Section 3. In Section 4, the effectiveness of SN is verified with thorough experiments and comparisons on shape retrieval, image retrieval and 3D model retrieval. Conclusions are given in Section 5.

2. Related work

Tremendous developments in context-sensitive similarities advance image and shape retrieval remarkably. A family of algorithms called diffusion process is proposed in the literature, such as Graph Transduction [9], Locally Constrained Diffusion Process (LCDP) [25], Locally Constrained Mixed Diffusion (LCMD) [27], Tensor Product Graph Diffusion (TPG) [12], Shortest Path Propagation (SSP) [10], Graph-PageRank [17], etc. In the survey paper [19], most of these approaches are elegantly summarized in a unified framework.

As shown in [19], a proper selection of neighbors ensures the diffusion process to work well in real cases. However, most variants of diffusion process use k-Nearest Neighbor (kNN) rule for its simplicity. Although [19] also uses kNN rule, it points out that it is still an open issue to select a reasonable local neighborhood. Re-