

Unsupervised Manifold Learning Using Reciprocal kNN Graphs in Image Re-Ranking and Rank Aggregation Tasks

Daniel Carlos Guimarães Pedronette¹, Otávio A. B. Penatti^{2,3},
and Ricardo da S. Torres³

¹ *Dept. of Statistics, Applied Math. and Computing, State University of São Paulo (UNESP), Rio Claro, Brazil*

² *RECOD Lab - Institute of Computing (IC), University of Campinas (UNICAMP), Campinas, Brazil*

³ *SAMSUNG Research Institute, Campinas, Brazil*

Abstract

In this paper, we present an unsupervised distance learning approach for improving the effectiveness of image retrieval tasks. We propose a *Reciprocal kNN Graph* algorithm that considers the relationships among ranked lists in the context of a k -reciprocal neighborhood. The similarity is propagated among neighbors considering the geometry of the dataset manifold. The proposed method can be used both for re-ranking and rank aggregation tasks. Unlike traditional diffusion process methods, which require matrix multiplication operations, our algorithm takes only a subset of ranked lists as input, presenting linear complexity in terms of computational and storage requirements. We conducted a large evaluation protocol involving shape, color, and texture descriptors, various datasets, and comparisons with other post-processing approaches. The re-ranking and rank aggregation algorithms yield better results in terms of effectiveness performance than various state-of-the-art algorithms recently proposed in the literature, achieving bull's eye and MAP scores of 100% on the well-known MPEG-7 shape dataset.

Keywords: content-based image retrieval; re-ranking; rank aggregation

1. Introduction

The development of multimedia technologies for creating and sharing digital contents has triggered an exponential increase of image collections. Traditional search approaches based on image metadata can be unfeasible for large collections, since much human intervention is required for image annotation. Content-Based Image Retrieval (CBIR) systems has emerged as a promising alternative, aiming at retrieving the images which are the most similar to a given query.

The effectiveness of CBIR systems is very dependent on the distance measure adopted. Images are often modelled as high dimensional points in an Euclidean

space, and the distance among them is usually measured by Euclidean distances. In this scenario, CBIR systems often consider only pairwise image analysis, that is, compute similarity measures considering only pairs of images, ignoring the information encoded in the relations among several images. On the contrary, the user perception considers the query specification and responses in a given *context*. In view of that, there has been significant research [45, 44, 13, 26, 14] on improving the distance measures in CBIR systems, replacing pairwise similarities by more global affinity measures that consider the relationships among images. The overall goal of these methods is to mimic the human behavior on judging the similarity among objects by taking into account the *context* of the search process. As previously observed [43, 41], an effective distance measure should describe the relationship between the query and retrieved objects in the context of the whole collection.

Therefore, how to capture and utilize the intrinsic manifold structure of a collection becomes a central problem in the vision and learning community [14]. A common recent approach is manifold learning, mainly based on non-linear dimensionality reduction techniques. The idea is to explicitly construct a new embedding space with a corresponding metric which is more faithful to the manifold structure and hence induces a better distance/similarity measure. The manifold learning algorithms are able to learn distances between data points that correspond to geodesic distances on the data manifold [45]. In other words, the new distances are estimated considering a walk along the geometric structure of the dataset.

In this paper, we propose an unsupervised learning algorithm based on *Reciprocal kNN Graph*. The proposed algorithm improves the effectiveness of image retrieval through re-ranking and rank aggregation tasks by taking into account intrinsic geometry of the dataset manifold. The capacity of considering geometry of the dataset manifold is illustrated in Figures 1, 2, and 3. We illustrate the Two-Moon dataset, comparing the Euclidean distance with the proposed *Reciprocal kNN Graph*. One point is selected as a labeled point (marked with a triangle) in each moon. In the following, all other data points are assigned to the closest labeled point, determining their color. Figure 1 illustrates the classification computed by the Euclidean distance. Figure 2 illustrates the ideal classification (with points in red and blue) considering the dataset manifold. The Euclidean distance does not consider the geometry structure of the dataset. As it can be observed, the extremities of the moons are misclassified. Figure 3 illustrates the distances learned by the *Reciprocal kNN Graph*, after only one iteration. We can observe that several points were corrected compared with the Euclidean distance. The arrows in Figure 3 illustrates how the *Reciprocal kNN Graph* algorithm iteratively propagates the similarity along the dataset structure considering the connectivity of the data set: (i) the red points in the left and; (ii) the blue points in the right.

The *Reciprocal kNN Graph* is mainly based on the information encoded in the top positions of the ranked lists. Given a query image, the ranked lists represent a relevant source of contextual information, since they define relationships not only between pairs of images (as distance functions), but also among all the

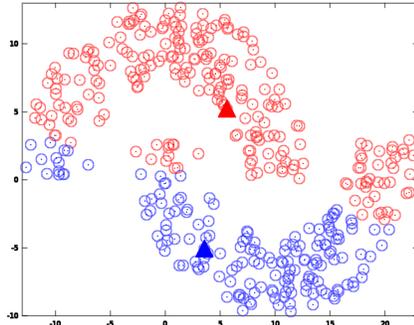


Figure 1: Euclidean distance.

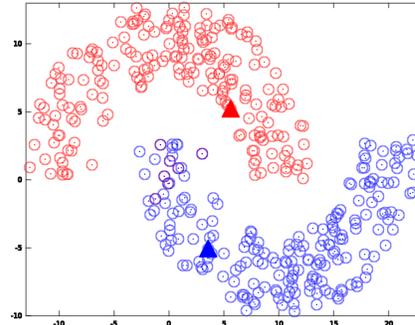


Figure 2: Ideal classification.

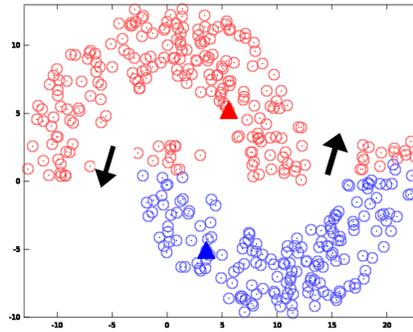


Figure 3: Reciprocal kNN Graph (one iteration).

images in the ranked list. The modelling of similarity information consists in the essential difference between the Reciprocal kNN Graph approach and existing diffusion-based algorithms: the Reciprocal kNN Graph is based only on the ranked lists, and therefore independent of any distance (or similarity) scores.

By analyzing the ranked lists, it is expected, for example, that similar images present reciprocal references at beginning of their ranked lists. It is also expected that images ranked at the top positions of ranked lists are similar to each other. In this way, aiming at redefining the distance between two images, the *Reciprocal kNN Graph* uses both the reciprocal nearest neighbor references and the graph structure considering all references among images at top positions of ranked lists. This approach represents the main contribution of our method, since it enables exploiting the maximum of contextual information available in the ranked lists with low computational efforts. Another contribution relies on the efficiency of the *Reciprocal kNN Graph* algorithm. Unlike other diffusion approaches based on matrices multiplication [45, 43, 3], which presents complexity of $O(n^3)$, our algorithm recomputes only the beginning of ranked lists with a constant size of elements, which presents computational and storage requirements of only $O(n)$,

where n represents the number of images in the collection.

We conducted a large evaluation protocol involving shape, color, and texture descriptors, different datasets and comparisons with other post-processing approaches. Experimental results demonstrate the effectiveness of our method. The re-ranking and rank aggregation algorithm yields better results in terms of effectiveness performance than various state-of-the-art algorithms.

This paper is organized as follows: Section 2 discusses related work; Section 3 discusses the definition of the image re-ranking problem; in Section 4, we present our *Reciprocal k NN Graph* algorithm. Section 5 presents the experimental evaluation and, finally, Section 6 draws on conclusions and presents future work.

2. Related Work

Defining an effective distance measures consists in a key role in many multimedia applications, including classification and retrieval tasks. For example, choosing a good distance measure is often critical for building a content-based image retrieval (CBIR) system. In general, aiming at retrieving the most similar images to a given query image, CBIR systems compute a predefined distance measure between the query image and each collection image. Traditional distance measures that consider only the pairwise similarity between two images, as Euclidean distance, are often adopted. These approaches fail to return correct results in many scenarios, mainly due to the well-known semantic gap problem [11].

Recently, there has been considerable research on improving the distance measures in CBIR systems [43, 44, 15, 13, 41, 45, 3, 26]. The main idea of various algorithms [45] is inspired by the success of Google PageRank [24] algorithm. Basically, the data manifold is represented as a graph with edge weights determined by similarity scores. Then, the similarities are propagated through weighted connections in the context of other dataset objects. In [43], a graph-based transductive learning algorithm is proposed for shape retrieval tasks. Inspired by semi-supervised label propagation algorithm, the shape retrieval was treated as an unsupervised problem. In [44], a locally constrained diffusion process is proposed where the influence of other shapes is propagated as a diffusion process on a graph formed by a given set of shapes. A shortest-path propagation algorithm was proposed [41] for explicitly finding the shortest path between them in the distance manifold of the dataset objects. However, a disadvantage of these methods is the large computational efforts required in the diffusion process, which is usually $O(n^3)$.

Other methods analyze the k -neighborhood relationships for learning new distance measures and performing re-ranking tasks. In [13], a contextual dissimilarity measure was introduced aiming at improving the symmetry of the k -neighborhood relationship by iteratively regularizing the average distance of each vector to its neighborhood. The reciprocal k NN relationships are considered in [29] to construct a close-set and a far-set, used for performing a re-ranking task. Another k NN re-ranking [34] uses the top- k retrieved object to

refine retrieval results by using information of ranked lists of each k neighbors. In [26], a recommendation process is simulated aiming at exploiting information encoded in ranked lists and collaboratively redefining pairwise distances.

The *Reciprocal kNN Graph* proposed in this paper combines characteristics of different methods. It uses the reciprocal k NN information, but also considers the graph structure constructed based on top positions of ranked lists. The distance among images are collaboratively redefined taking into account all references contained in top- k lists. Unlike other diffusion approaches [45, 43, 3], our algorithm redefines only the images placed at first positions of ranked lists. This procedure presents computational and storage requirements of only $O(n)$. In addition, the *Reciprocal kNN Graph* can be used for re-ranking and rank aggregation tasks, aiming at combining different descriptors.

3. Problem Formulation

Let $\mathcal{C}=\{img_1, img_2, \dots, img_n\}$ be an *image collection*, where n is the number of images in the collection. Let \mathcal{D} be an *image descriptor* which defines a distance function between two images img_i and img_j as $\rho(img_i, img_j)$. For simplicity and readability purposes, we use the notation $\rho(i, j)$ for denoting the distance between images img_i and img_j .

Based on the distance function ρ , a ranked list τ_q can be computed in response to a query image img_q . Although the ranked lists contain distance information from the entire collection, the top positions of ranked lists are expected to contain the most relevant images related to the query image. Therefore, it can be very desirable that the ranked list τ_q considers only a subset of the n_s most similar images, such that $n_s \ll n$ and n_s is a constant value. That is even more crucial specially for large collections, where n is very high, and therefore τ_q is very expensive to compute. In addition, various index structures [1, 31] that have been proposed to speed up similarity queries can be used for computing the ranked lists.

The ranked list $\tau_q=(img_1, img_2, \dots, img_{n_s})$ can be defined as a permutation of the subset $\mathcal{C}_s \subset \mathcal{C}$, which contains the most similar images to query image img_q , such that $|\mathcal{C}_s| = n_s$. A permutation τ_q is a bijection from the set \mathcal{C}_s onto the set $[n_s] = \{1, 2, \dots, n_s\}$. For a permutation τ_q , we interpret $\tau_q(i)$ as the position (or rank) of image img_i in the ranked list τ_q .

We can say that, if img_i is ranked before img_j in the ranked list of img_q , that is, $\tau_q(i) < \tau_q(j)$, then $\rho(q, i) \leq \rho(q, j)$. Note that, if the position of img_i in the ranked list of img_q is higher than the constant n_s , then $\tau_q(i) = n_s$.

We also can take every image $img_i \in \mathcal{C}$ as a query image img_q , in order to obtain a set $\mathcal{R} = \{\tau_1, \tau_2, \dots, \tau_n\}$ of ranked lists for each image of the collection \mathcal{C} . The storage requirements for handling the set \mathcal{R} is $O(n)$, since the size of ranked lists is given by the constant n_s .

Our objective is to define a function f_r which takes a set of ranked lists \mathcal{R} as the input and computes a new and more effective set of ranked lists $\hat{\mathcal{R}}$ by taking into account contextual information available on ranked lists:

$$\hat{\mathcal{R}} = f_r(\mathcal{R}). \quad (1)$$

The *Reciprocal kNN Graph* algorithm, presented in next section, represents a definition of the function f_r . For rank aggregation tasks, which combines ranked lists computed by different descriptors, the input of function f_r is given by a set of sets $\{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_m\}$, where m is the number of descriptors used.

4. Reciprocal kNN Graph

In this section, we present the *Reciprocal kNN Graph* algorithm and its application in re-ranking and rank aggregation tasks. We also discuss convergence and efficiency aspects.

4.1. Motivation and Overview

Ranked lists represent a relevant source of contextual information, since they establish relationships not only between pairs of images (as distance functions), but also among all images in the ranked list. The objective of *Reciprocal kNN Graph* is to exploit all available contextual information in ranked lists using three main strategies:

- **Reciprocal Neighborhood:** the k -reciprocal nearest neighborhood mitigates the risk of false positives at top positions of ranked lists.
- **Collaborative Ranking:** a ranked list can provide useful information for improving effectiveness of other ranked lists. If two images appears at the top position of any ranked list, it indicates that they are probably similar to each other.
- **Authority of Ranked Lists:** we propose a score to estimate the authority of a given ranked list for collaborating with other ranked lists. The score is based on density of the graph that represents the reciprocal references among images at top positions of the ranked list.

The new distance score between two images is computed based on the reciprocal reference between their ranked lists and collaboratively by other ranked lists, considering their respective authority. The process is iteratively repeated until a convergence criterion is reached.

The main concepts of the algorithm are formally defined in terms of three scores: (i) *Reciprocal kNN Score*; (ii) *Authority Score* and; (iii) *Collaborative Score*. Based on these scores, a new distance measure (nominated *Reciprocal kNN Distance Measure*) is computed. Figure 4 illustrates the overall organization of the algorithm considering the proposed scores, which are discussed in details in the next sections.

4.2. Reciprocal kNN Score

Given a query image img_q , we can define a neighborhood set which contains the k most similar images to img_q as $\mathcal{N}(q, k)$. For the k -nearest neighbors query, we obviously have $|\mathcal{N}(q, k)| = k$. The nearest neighbor relationships are not symmetric [13, 29]. This means that $img_i \in \mathcal{N}(j, k)$ does not imply

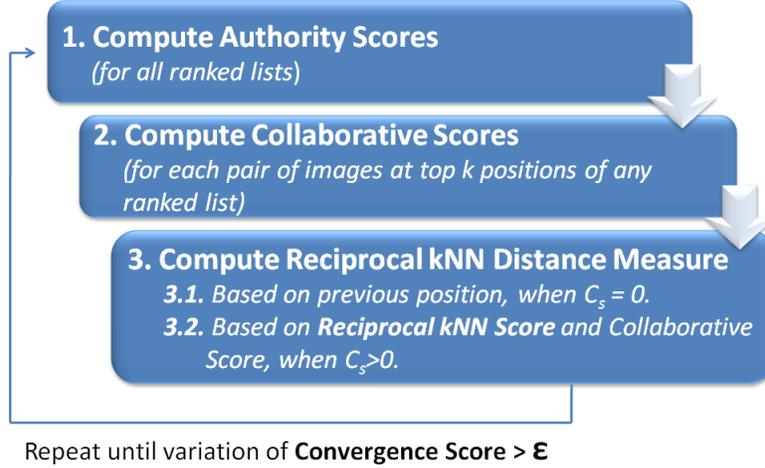


Figure 4: Main steps of the Reciprocal kNN Graph algorithm.

$img_j \in \mathcal{N}(i, k)$. The set of k -reciprocal nearest neighbors of image img_q can be defined [29] as:

$$\mathcal{N}_r(q, k) = \{img_i \in \mathcal{N}(q, k) \wedge img_q \in \mathcal{N}(i, k)\}. \quad (2)$$

We can easily verify if a given image img_i is a k -reciprocal nearest neighbor of an image img_q by checking if $img_i \in \mathcal{N}_r(q, k)$. However, beyond knowing if two images are reciprocal neighbors, we are also interested in the position from which on images became reciprocal neighbors. Therefore, we propose a *Reciprocal kNN Score* R_s that consider this position:

$$R_s(q, i) = \frac{\max(\tau_q(i), \tau_i(q))}{n_s}. \quad (3)$$

This score reduces the probability of false positives at top positions of ranked lists. Therefore, a low value represents a strong indication of similarity between images.

4.3. Reciprocal kNN Authority Score

We consider that a ranked list encodes contextual information that can be used for improving effectiveness of *other* ranked lists. In this scenario, it is important to estimate the authority of a given ranked list. Our approach is analogous to the PageRank [24] algorithm and inspired by the cohesion measure [26]. We consider that an accurate ranked list has their top images referencing to each other at the top positions of their ranked lists. This conjecture is somehow close to the cluster hypothesis [30], which states that “*closely associated items tend to be relevant to the same requests*”.

Another important role of the authority score in the proposed algorithm is to estimate the quality of a ranked list. In other words, the score is an unsupervised

predictor of effectiveness of a given ranked list. The computation of authority score is equivalent to measure the density of the graph that represents the reciprocal references among images at top- k positions. The *Reciprocal kNN Authority Score* of the ranked list τ_q is defined as follows:

$$A_s(q, k) = \frac{\sum_{i \in \mathcal{N}(q, k)} \sum_{j \in \mathcal{N}(i, k)} f_{in}(j, q)}{k^2}, \quad (4)$$

where f_{in} returns 1 if $img_j \in \mathcal{N}(q, k)$ and 0 otherwise.

The score A_s is defined in the interval $[0, 1]$. Note that, for a complete reciprocal graph (where all k images references each other at top- k positions) this score returns a perfect score $A_s(q, k) = 1$.

4.4. Reciprocal kNN Collaborative Score

Beyond the *Reciprocal kNN Score*, the distance between two images is defined using a collaborative score that considers information encoded in other ranked lists and their respective authority score. This means that, when an image img_i appears at top positions of a ranked list τ_q , the distance among img_i and neighbors of image img_q ($img_j \in \mathcal{N}(q, k)$) are reduced proportionally to the authority of τ_q .

In this way, given two images img_q and img_i , we define a *Reciprocal kNN Collaborative Score*, that accumulates the authority scores of all ranked lists in which images img_q and img_i appear. The collaborative score considers different values of k (varying from $1, 2, \dots, k$) with the purpose of giving greater weights to references at top positions. The collaborative score C_s between two images img_q and img_i is defined as follows:

$$C_s(q, i, k) = \sum_{c=1}^k \sum_{j \in \mathcal{C}} A_s(j, c)^2 \times f_{in}(q, i, j), \quad (5)$$

where f_{in} returns 1 if $img_q, img_i \in \mathcal{N}(j, k)$ and 0 otherwise. A squared value is used for the authority score aiming at penalizing low scores. Notice that, although by definition j iterates for each $j \in \mathcal{C}$ (what requires linear computation efforts on the size of the collection), the value $A_s(j, c)^2$ is added only when $img_q, img_i \in \mathcal{N}(j, k)$, that is, considering only top- k positions. Therefore, the collaborative score $C_s(q, i, k)$ can be computed requiring only $O(k^2)$.

4.5. Reciprocal kNN Distance Measure

In this section, we define an iteratively distance measure which is the basis of the proposed re-ranking and rank aggregation algorithms. Using this distance, a new set of ranked lists can be computed.

Based on *Reciprocal kNN Score* (R_s) and *Reciprocal kNN Collaborative Score* (C_s), we define the distance measure ρ_r . All images $img_q, img_i \in \mathcal{C}$ that present collaborative score $C_s(q, i, k) > 0$ have the distance between them updated as:

$$\rho_r(q, i) = \frac{R_s(q, i)}{1 + C_s(q, i, k)}. \quad (6)$$

The remaining images with zero collaborative score ($C_s(q, j, k) = 0$) keep the distance between them as their current ranking, i. e., $\rho_r(q, j) = \tau_r(j)$. No more computation efforts are required for those images. The main motivation of Equation 6 is to consider information from both: (i) reciprocal reference between images being compared (dividend) and; (ii) information given by ranked list of other images, according to their authority (divisor). While the *Reciprocal kNN Score* (R_s) avoids false positives at top positions of ranked lists, the *Reciprocal kNN Collaborative Score* (C_s) propagates the similarity among different ranked lists, considering their quality estimation given by authority score.

Based on the distance ρ_r , the set of ranked lists \mathcal{R} is updated, for ensuring that, if $\tau_q(i) < \tau_q(j)$, then $\rho(q, i) \leq \rho(q, j)$. Finally, the process can be iteratively repeated. Let the superscript (t) denotes the iteration and let $\rho_r^{(0)}$ be the distance at first iteration, we can define an iterative distance measure as follows:

$$\rho_r^{(t+1)}(q, i) = \frac{R_s^{(t)}(q, i)}{1 + C_s^{(t)}(q, i, k + t)}. \quad (7)$$

Note the value of k grows along iterations. It is expected that non-relevant images are moved out from the first positions of the ranked lists and, therefore, k can be increased for considering more images.

4.6. Discussion

This section presents a detailed discussion on how the method works as well as its main equations. As previously mentioned, the ranked lists define relationships not only between pairs of images (as distance functions), but also among all the images in the ranked list. In this sense, if an image is well ranked for a given query, other images, similar to this image, are also expected to be well ranked for the same query. This observation is consistent with the “*cluster hypothesis*” [30], which states that closely related items tend to be relevant to the same requests.

The Reciprocal kNN Graph aims at exploiting the cluster hypothesis by analyzing the reciprocal references among ranked lists at their top positions. The three scores which define the algorithm are based on this principle: (i) Reciprocal kNN Score; (ii) Reciprocal kNN Authority Score; (iii) Reciprocal kNN Collaborative Score. We discuss the cluster hypothesis for each score in the following.

Given two similar images, a content-based descriptor is expected to produce ranked lists which present reciprocal references at the beginning of their ranked lists. When an image does not refer to the other image at the top positions of its ranked list, this behavior indicates a low confidence in the similarity between them. The Reciprocal kNN Score (Equation 3) represents the position in which this confidence is reached. Therefore, this score can be used to provide a more accurate distance measure than that used to compute the initial ranked lists.

However, the Reciprocal kNN Score considers the reciprocal references only between pairs of images. In addition, it is also expected that images ranked

at the top positions of ranked lists present reciprocal references among each other (similar images tend to be relevant to the same queries). Therefore, the Reciprocal kNN Authority Score (Equation 4) measures the amount of reciprocal references by computing the density of the graph that represents the references among images at top-k positions of a given ranked list. Therefore, an effective ranked list which presents similar images at its top positions will also present a high authority score. In this way, this score can be used to estimate the effectiveness of a ranked list. Figure 5 illustrates the computation of authority score, which is proportional to the number of edges on the Reciprocal kNN Graph.

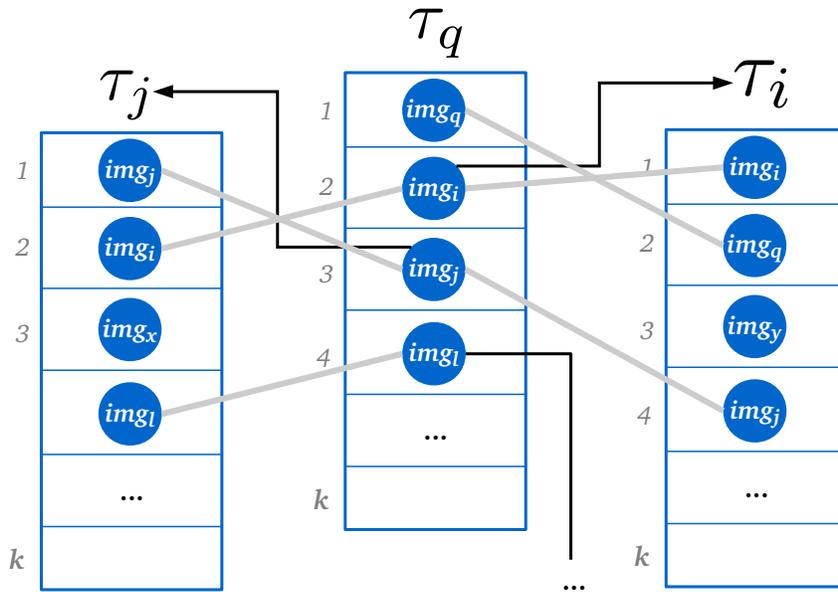


Figure 5: Reciprocal kNN Authority Score computation.

Once we have an estimation of the effectiveness of ranked lists given by the authority score, images which appear at top positions of effective ranked lists are very likely to be similar. This assumption consists in the basis of the Reciprocal kNN Collaborative Score. The collaborative score between two images is given by the sum of authority scores of ranked lists in which these two images appears at top positions.

Finally, the Reciprocal kNN Distance Measure is computed by combining the Reciprocal kNN Score (that considers reciprocal references between two images), and the Reciprocal kNN Collaborative Score (that considers the top-k positions of all ranked lists using the Reciprocal kNN Authority Score).

4.7. Convergence

Basically, an iterative method is said to converge, if the difference between results obtained along iterations decreases, tending to reach an ultimate result. In our case, a new distance measure should be iteratively executed while the quality of ranked lists is improved. We define a convergence criterion that determines the number of iterations using the proposed authority score. While the authority of ranked lists is increasing more than a threshold ϵ per iteration, the algorithm should continue executing. Given that the authority score estimates the quality of ranked lists, the proposed criterion is equivalent to keep executing the algorithm while the quality of ranked lists are improving.

Aiming at verifying the convergence criterion, an average authority score between all collection images is computed as follows:

$$G_s = \frac{\sum_{c=1}^k \sum_{j \in \mathcal{C}} A_s(j, c)}{k \times n} \quad (8)$$

In the following, the difference between iterations is compared with the threshold ϵ for defining the convergence criterion. The re-ranking is executed while $(G_s^{(t+1)} - G_s^{(t)}) > \epsilon$.

4.8. Rank Aggregation

In general, different types of measures may focus on different aspects of the images and are often complementary to each other [3]. Our goal is to use the *Reciprocal kNN Graph* algorithm to fuse different distance measures in rank aggregation tasks. We propose a multiplicative approach inspired by [26] for combining the *Reciprocal kNN Graph Distance Measures* of different descriptors. The combination is computed only at the first iteration and subsequent iterations are computed over the already combined ranked lists. Let $\rho_{r_d}^{(1)}$ be the *kNN* reciprocal distance measure at first iteration of a given descriptor d , with d defined in the interval $[1, m]$, where m is the number of descriptors considered. The fused distance measure can be defined as:

$$\rho_r^{(1)}(q, i) = \prod_{d=1}^m \rho_{r_d}^{(1)}(q, i) \quad (9)$$

By multiplying the distance measures between the same images considering different descriptors, high distances obtained by one descriptor will be propagated to the others, leading to high aggregate values. Note that our algorithm multiplies distance measures computed by the *Reciprocal kNN Graph* algorithm and it does not depend on the original descriptors distance measures. It only considers the ranked lists. Therefore, our approach does not present a common problem in late fusion tasks of combining heterogeneous scores at different numeric scales.

4.9. Aspects of Efficiency

This section briefly discusses some aspects of efficiency, computational complexity and storage requirements. The proposed algorithm takes as input only the beginning of ranked lists (with a constant size n_s of elements), which becomes the storage requirements of $O(n)$. It represents a significant advantage in comparison with other methods that requires the complete similarity/distance matrix [45, 26].

The asymptotic computational complexity of the algorithm is also $O(n)$, since only the n_s top positions of ranked lists are redefined, independent on the size n of the dataset. The computation of the authority and collaborative scores, which presents the high computational effort needed, is proportional to (nTk^2) , where T denotes the number of iterations and k the number of neighbors considered when algorithm starts. Other steps of the algorithm have diverse computation cost, but all limited to the asymptotic cost of $O(n)$. State-of-the-art methods based on random walks on graphs [45, 43, 3] uses matrices multiplication which presents complexity of $O(n^3)$. Beyond that, the *Reciprocal kNN Graph* algorithm has potential to be massively parallelized.

5. Experimental Evaluation

This section demonstrates the effectiveness of the proposed re-ranking and rank aggregation methods in image retrieval tasks. A large set of experiments was conducted considering four datasets and nineteen CBIR descriptors, aiming at analyzing and comparing our method under several aspects.

5.1. Experiment 1: Impact of Parameters

The computation of *Reciprocal kNN Graph* algorithm considers only two parameters: (i) k : number of neighbors considered when algorithm starts; and (ii) ϵ : convergence threshold parameter.

To evaluate the influence of different parameter settings on the retrieval scores and for determining the best parameters values, we conducted a set of experiments considering the MPEG-7 [17] dataset. The MPEG-7 [17] dataset is a well-known shape dataset, composed of 1400 shapes divided in 70 classes. For evaluation, the so-called bull’s eye score was considered, which counts all matching objects within the 40 most similar candidates. Since each class consists of 20 objects, the retrieved score is normalized with the highest possible number of hits. For distance computation, we used the Contour Features Descriptor (CFD) [25] shape descriptor.

We varied the parameter k in the interval $[1, 20]$ and iterations in the interval $[1, 15]$. Figure 6 illustrates the results of precision scores for different values of k and iterations.

We observed that best retrieval scores increased along iterations yielding the best precision score (96.49%) for $k = 15$ and final iteration $T = 7$. At this iteration, the difference value of convergence score to previous iteration is 0.0125. Therefore, we set the parameter values as $k = 15$ and $\epsilon = 0.0125$.

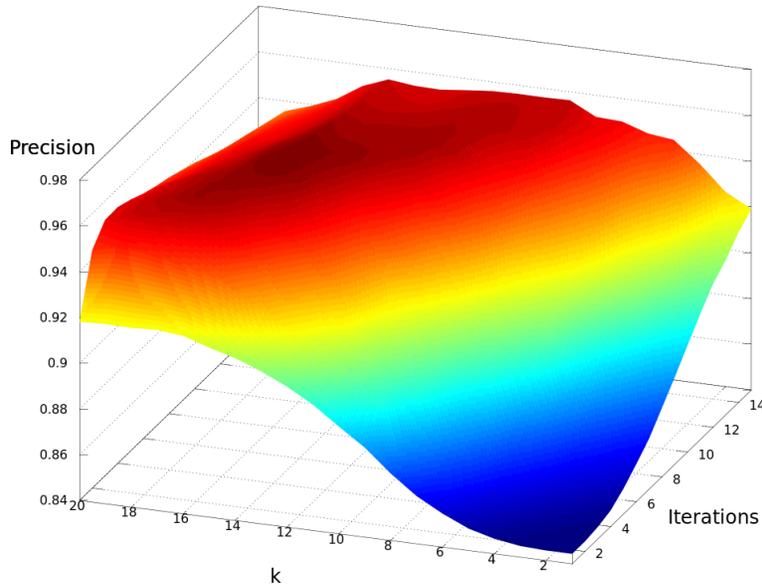


Figure 6: Impact of k and the number of iterations on precision.

Figure 7 illustrates the evolution of convergence score along iterations. As we can observe, the algorithm converges very quickly, requiring a very small number of iterations for reaching the threshold.

As previous discussed, our algorithm considers only a subset of ranked lists. The size of ranked lists considered in experiments is $n_s = 200$. These values are used in all experiments, both for re-ranking and rank aggregation tasks, considering different descriptors and datasets, what demonstrates the robustness of our method.

5.2. Experiment 2: Re-Ranking Evaluation

In this section, we present the set of conducted experiments for evaluating our method in the task of re-ranking images considering shape, color, and texture descriptors.

5.2.1. Shape Descriptors

We evaluate the use of our method with six shape descriptors: Segment Saliences (SS) [32], Beam Angle Statistics (BAS) [2], Inner Distance Shape Context (IDSC) [18], Contour Features Descriptor (CFD) [25], Aspect Shape Context (ASC) [19], and Articulation-Invariant Representation (AIR) [10]. We consider the MPEG-7 [17] dataset, described in Section 5.1.

Table 1 presents results considering the bull’s eye score ($Recall@40$) and accuracy ($Precision@20$) for shape descriptors on the MPEG-7 [17] dataset. We can observe very significant gains in relation to the results observed for each

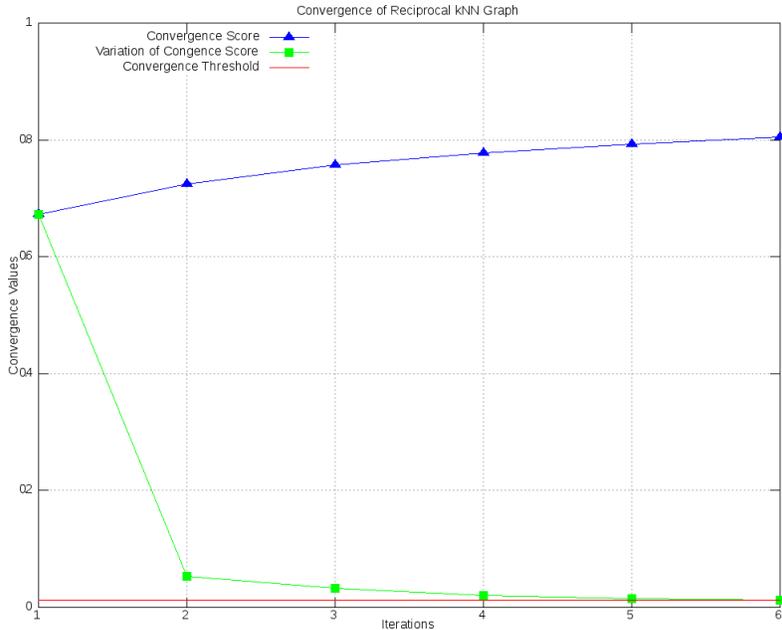


Figure 7: Convergence of Reciprocal kNN Graph algorithm.

descriptor initially, ranging from +6.69% to +29.48% for the bull’s eye score and ranging from +6.10% to +40.50% for the accuracy measure.

The iterative behavior of the *Reciprocal kNN Graph* algorithm can be observed in the results shown in Figure 8. The figure shows the evolution of rankings along iterations. The first row presents a query image (first column with green border) and 20 image results, according to the CFD [25] shape descriptor. Note that wrong results (with red border) contain images from different classes. The remaining rows present the results for each iteration of *Reciprocal kNN Graph* algorithm. We can observe the significant improvement in terms of precision, ranging from 20% (on the ranking computed by the CFD [25] descriptor) to 100% at the 7th iteration of the proposed re-ranking algorithm.

Table 1: **Reciprocal kNN Graph** for various shape descriptors on the MPEG-7 dataset.

Shape Descriptor	Bull’s Eye Score	Reciprocal kNN Graph	Gain	Accuracy	Reciprocal kNN Graph	Gain
SS [32]	43.99%	56.96%	+29.48%	35.50%	49.88%	+40.50%
BAS [2]	75.20%	86.50%	+15.03%	67.33%	78.25%	+16.22%
IDSC [18]	85.40%	93.38%	+9.34%	77.21%	88.91%	+15.15%
CFD [25]	84.43%	96.49%	+14.28%	75.59%	91.80%	+21.44%
ASC [19]	88.39%	95.27%	+7.78%	80.66%	90.60%	+12.32%
AIR [10]	93.67%	99.94%	+6.69%	88.17%	93.55%	+6.10%

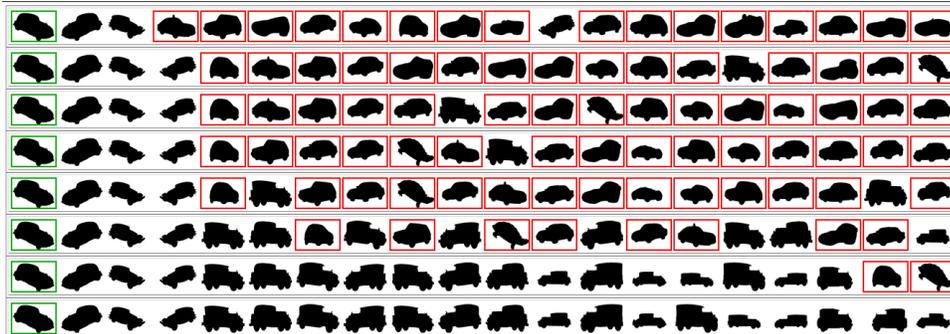


Figure 8: Evolution of rankings along iterations on the MPEG-7 [17] dataset. The first column (green border) contains the query image. The first row presents the results of CFD [25] shape descriptor (wrong results with red borders). The remaining rows present the results of **Reciprocal kNN Graph** algorithm for each iteration (iterations grow from top to bottom).

Results for shape descriptor considering the MAP (*Mean Average Precision*) score are presented in Table 2. We can observe positive gains for all shape descriptors ranging from +8.54% to +37.99%.

5.2.2. Color Descriptors

We evaluate our method with three color descriptors: Border/Interior Pixel Classification (BIC) [35], Auto Color Correlograms (ACC) [12], and Global Color Histogram (GCH) [36]. The experiments were conducted on a dataset used in [42] and composed of images from 7 soccer teams, containing 40 images per class. The size of images range from (198×148) to (537×672) pixels.

Table 2 presents the experimental results considering MAP as score. We can observe a positive gain for all color descriptors ranging from +4.50% to +15.33%.

5.2.3. Texture Descriptors

The experiments consider three well-known texture descriptors: Local Binary Patterns (LBP) [22], Color Co-Occurrence Matrix (CCOM) [16], and Local Activity Spectrum (LAS) [38]. We used the Brodatz [6] dataset, a popular dataset for texture descriptors evaluation. The Brodatz dataset is composed of 111 different textures of size (512×512) pixels. Each texture is divided into 16 blocks (128×128) pixels of non-overlapping sub images, such that 1776 images are considered.

Table 2 presents the experimental results considering MAP as score. We can observe a positive gain for all texture descriptors ranging from +3.85% to +15.16%.

5.3. Experiment 3: Rank Aggregation Evaluation

We evaluate the use of *Reciprocal kNN Graph* method to combine different CBIR descriptors.

Table 2: MAP scores for **Reciprocal kNN Graph** in different re-ranking tasks.

Descriptor	Type	Dataset	Score (MAP)	Reciprocal kNN Graph	Gain
SS [32]	Shape	MPEG-7	37.67%	51.98%	+37.99%
BAS [2]	Shape	MPEG-7	71.52%	82.01%	+14.67%
IDSC [18]	Shape	MPEG-7	81.70%	91.16%	+11.58%
ASC [19]	Shape	MPEG-7	85.28%	93.15%	+9.23%
CFD [25]	Shape	MPEG-7	80.71%	94.12%	+16.62%
AIR [10]	Shape	MPEG-7	89.39%	97.02%	+8.54%
GCH [36]	Color	Soccer	32.24%	33.69%	+4.50%
ACC [12]	Color	Soccer	37.23%	42.11%	+13.11%
BIC [35]	Color	Soccer	39.26%	45.28%	+15.33%
LBP [22]	Texture	Brodatz	48.40%	51.05%	+5.48%
CCOM [16]	Texture	Brodatz	57.57%	66.30%	+15.16%
LAS [38]	Texture	Brodatz	75.15%	78.04%	+3.85%

Table 3: Bull’s eye score, MAP, and Accuracy retrieval scores for **Reciprocal kNN Graph** in rank aggregation tasks considering Shape Descriptors on the MPEG-7 dataset.

Descriptor	Bull’s eye score	MAP	Accuracy
CFD [25]	84.43%	80.71%	75.59%
ASC [19]	88.39%	85.28%	80.66%
AIR [10]	93.67%	89.39%	88.17%
CFD+ASC	99.83%	99.24%	98.61%
CFD+AIR	100.00%	100.00%	100.00%
ASC+AIR	99.98%	99.93%	99.88%

We selected three shape descriptors with highest retrieval scores in re-ranking tasks and evaluated the different combinations between them. Table 3 presents the rank aggregation results. Besides MAP scores, we also present the accuracy and the bull’s eye score on the MPEG-7 dataset. Notice that the combination of CFD [25]+AIR [10] presents retrieval scores of 100% for the three considered measures, which means *perfect retrieval results*.

We also selected two color and texture descriptors, with the highest MAP scores in re-ranking tasks. Table 4 presents results of MAP score of these descriptors. We can observe that significant gains are obtained when compared with the results of descriptors in isolation.

Figure 9 illustrates the *Precision × Recall* curves of two shape descriptors in different situations: before and after applying the Reciprocal kNN Graph algorithm, and after using it for rank aggregation. We can observe that significant gains in terms of precision have been achieved, in special for the rank aggregation task.

Table 4: MAP scores for **Reciprocal kNN Graph** in rank aggregation tasks for color and texture descriptors.

Descriptor	Type	Dataset	Score (MAP)
ACC [12]	Color	Soccer	37.23%
BIC [35]	Color	Soccer	39.26%
BIC+ACC	Color	Soccer	43.24%
CCOM [16]	Texture	Brodatz	57.57%
LAS [38]	Texture	Brodatz	75.15%
LAS+CCOM	Texture	Brodatz	83.71%

5.4. Experiment 4: Comparison with Other Approaches

Finally, we also evaluate our method in comparison with other state-of-the-art post-processing methods. We use the MPEG-7 [17] dataset, with the bull’s eye score, commonly used for post-processing methods evaluation and comparison. Table 5 presents results of the proposed *Reciprocal kNN Graph* algorithm in comparison with several other post-processing methods recently proposed in the literature. Note that the results of *Reciprocal kNN Graph* algorithm presents better effectiveness performance when compared to various recently proposed methods in re-ranking tasks. We report the results of CFD [25] and AIR [10] descriptors, respectively the descriptors that presented the highest gain and the highest bull’s eyes score in re-ranking tasks.

We also present results of Reciprocal kNN Graph in rank aggregation tasks in comparison to other recently proposed methods. Note that the Reciprocal kNN Graph applied to the combination of only two descriptors CFD [25]+AIR [10] reached a perfect retrieval scores (considering MAP, accuracy and the bull’s eye score), obtained by other state-of-the-art method only combining three descriptors.

5.5. Experiment 5: Natural Image Retrieval

The University of Kentucky Recognition Benchmark [21] (“UKBench”), also referred as Nister and Stewenius (N-S) Dataset, has a total of 10,200 images. The dataset is composed of 2,550 objects or scenes, where each object/scene is captured 4 times from different viewpoints, distances and illumination conditions. Thus, it consists of 2,550 image classes, and each class has only 4 images. For evaluation purposes, each image is used as query and the N-S retrieval score between 1 and 4 is computed. The score corresponds to the number of relevant images among the first four image returned (the highest achievable score is 4, indicating that all similar images are retrieved at top positions). Due to the small number of images in each class, the N-S dataset is a very challenging dataset especially for unsupervised learning approaches.

In the following, we describe several experiments conducted aiming at evaluating the Reciprocal kNN Graph on N-S Dataset. For all experiments, we use

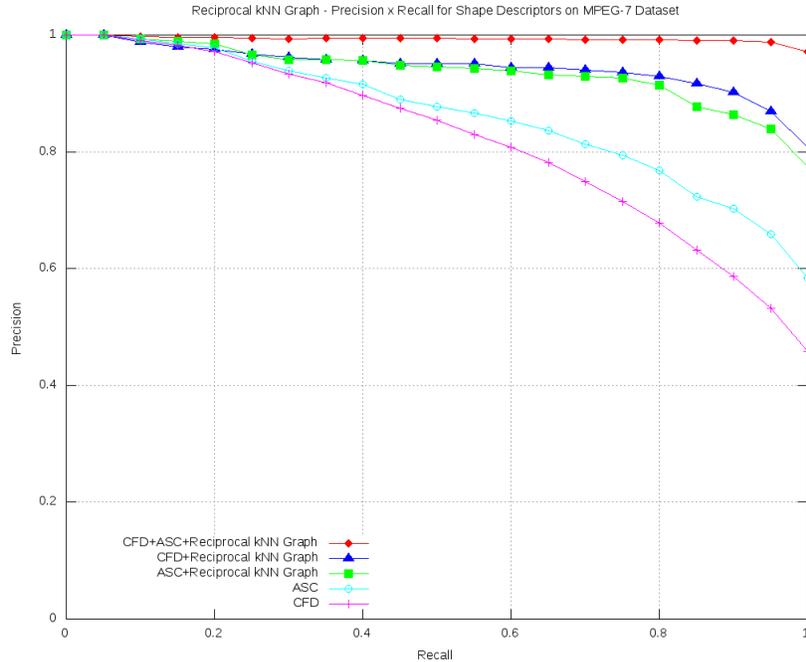


Figure 9: Impact of Reciprocal kNN Graph algorithm for re-ranking and rank aggregation tasks on MPEG-7 dataset.

the same parameters used for other datasets, except for $k = 5$, due to small number of similar images. The experiments consider nine descriptors¹, exploiting several features as color, texture and local features. In the following, we briefly describe the descriptors based on considered features:

- **Color:** Auto Color Correlograms (ACC) [12], Border/Interior Pixel Classification (BIC) [35], Local Color Histogram (LCH) [37].
- **Color and Texture:** Color and Edge Directivity Descriptor (CEED) [7], Fuzzy Color and Texture Histogram (FCTH) [8], Joint Composite Descriptor (JCD) [46].
- **Local:** Scale-Invariant Feature Transform (SIFT) [20], considering the number of matches between images as a similarity score².
- **Bag of visual words:** Several configurations of descriptors based on

¹We used the LIRE implementation (<http://www.semanticmetadata.net/lire/>) of descriptors ACC [12], CEED [7], FCTH [8], and JCD [46].

²We considered resized dataset images of 160×120 pixels and a public available implementation at <http://www.cs.ubc.ca/~lowe/keypoints>.

Table 5: Post-processing methods comparison on the MPEG-7 dataset (*bull's eye score*).

Algorithm	Shape descriptor	Bull's eye score
Shape Descriptors		
DDGM [39]	-	80.03%
CFD [25]	-	84.43%
IDSC [18]	-	85.40%
SC [4]	-	86.80%
ASC [19]	-	88.39%
AIR [10]	-	93.67%
Post-Processing Methods		
Graph Transduction [43]	IDSC	91.00%
Locally Constrained Diffusion Process [44]	IDSC	93.32%
Shortest Path Propagation [41]	IDSC	93.35%
Mutual kNN Graph [15]	IDSC	93.40%
Locally Constrained Diffusion Process [44]	ASC	95.96%
Pairwise Recommendation [26]	CFD	96.15%
Tensor Product Graph [45]	ASC	96.47%
Reciprocal kNN Graph	CFD	96.49%
Co-Transduction [3]	IDSC+DDGM	97.31%
Co-Transduction [3]	SC+DDGM	97.45%
Self-Smoothing Operator [14]	SC+IDSC	97.64%
Co-Transduction [3]	SC+IDSC	97.72%
Self-Smoothing Operator [14]	SC+IDSC+DDGM	99.20%
Pairwise Recommendation [26]	CFD+IDSC	99.52%
Reciprocal kNN Graph	AIR	99.94%
Tensor Product Graph [45]	AIR	99.99%
Cross Diffusion Process [40]	SC+IDSC+DDGM	100%
Reciprocal kNN Graph	AIR+CFD	100%

visual words were evaluated³. The results presented in this section consider only the best configurations, which are: sparse sampling (Harris-Laplace detector), OpponentSIFT [33], codebook of 1,000 visual words, soft assignment ($\sigma = 150$) and spatial pooling WSA (Word Spatial Arrangement) [27, 28], using threshold for soft assignment as 0.01 and the distance function presented in [27].

- **Holistic:** GIST [23], a scene recognition descriptor based on the Spatial Envelope low dimensional representation⁴.

Table 6 presents the experimental results considering the N-S score for all considered descriptors on re-ranking tasks. We can observe positive gains for all descriptors ranging from +3.30% to +15.35%. Table 6 also presents the experimental results for Reciprocal kNN Graph considering rank aggregation tasks. We consider for the combination the three descriptors which have obtained the best scores in the re-ranking tasks (ACC [12], BIC [35], SIFT [20]). We compute

³We varied the low-level feature extraction using sparse and dense sampling, and using SIFT [20] and OpponentSIFT [33]. Different codebook sizes were evaluated, varying from 100 to 50,000 visual words. Hard and soft assignment [9] were also used, and we tested average and max pooling [5] and a spatial pooling approach called WSA (word spatial arrangement) [27].

⁴We considered resized dataset images of 256×256 pixels and a public available C implementation at <http://lear.inrialpes.fr/software>.

Table 6: **Reciprocal kNN Graph** evaluation for various descriptors on the N-S Dataset.

Descriptor	N-S Score	Reciprocal kNN Graph	Gain
GIST [23]	2.18	2.28	+4.59%
LCH [37]	2.25	2.41	+7.11%
WSA [27]	2.28	2.59	+13.60%
CEED [7]	2.61	2.74	+4.98%
FCTH [8]	2.73	2.82	+3.30%
JCD [46]	2.79	2.91	+4.30%
SIFT [20]	2.54	2.93	+15.35%
BIC [35]	3.04	3.20	+5.26%
ACC [12]	3.36	3.58	+6.55%
ACC [12]+BIC [35]	-	3.55	+5.65%
ACC [12]+SIFT [20]	-	3.79	+12.80%
BIC [35]+SIFT [20]	-	3.60	+18.42%

Table 7: Comparison of N-S scores on the N-S dataset considering the best scores obtained by recent retrieval methods and fusion approaches.

Jégou <i>et al.</i> [13]	Qin <i>et al.</i> [29]	Wang <i>et al.</i> [40]	Zhang <i>et al.</i> [47]	Reciprocal kNN Graph
3.68	3.67	3.68	3.77	3.79

the gain considering the highest score obtained by descriptors considered in the combination. We can observe that the combination of ACC [12]+SIFT [20] present a very high score (N-S=3.79) and BIC [35]+SIFT [20] a very high gain (+18.42%).

We also compare the Reciprocal kNN Graph with state-of-the-art methods recently proposed on the N-S dataset. Table 7 presents the best N-S scores obtained by Reciprocal kNN Graph and reported by the recent retrieval methods and fusion approaches. We can observe that the Reciprocal kNN Graph presents the highest N-S score when compared with other approaches.

5.6. Experiment 6: Efficiency Analysis

In this Section, we present the run time for the Reciprocal kNN Graph on the N-S dataset, considering the four descriptors which have presented the highest retrieval scores (ACC [12], BIC [35], SIFT [20], JCD [46]). We use a serial C implementation running over Linux on an Intel Xeon 2.40 GHz processor. Figure 10 presents the total run time and the run time per iteration for the four considered descriptors. The ACC [12] descriptor, for example, converged after only 3 iterations, with a total run time of 8.3 s. The average time required for the re-ranking of each image in this case is only 8.1 ms.

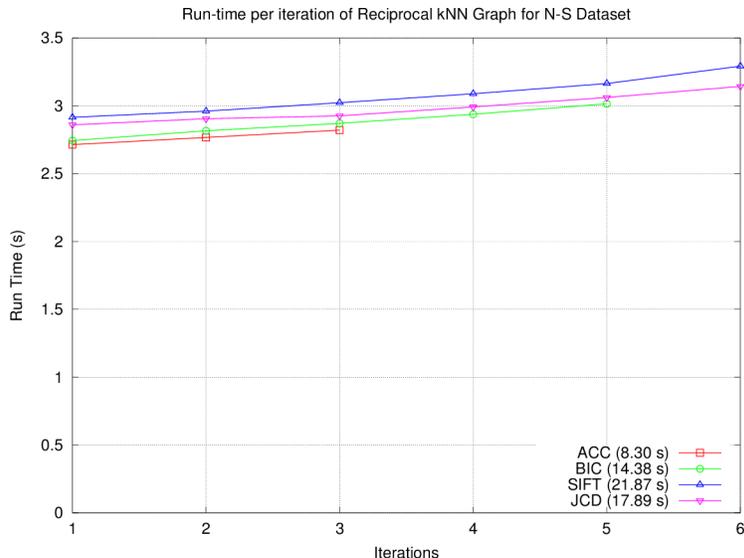


Figure 10: Run time of Reciprocal kNN Graph for the N-S Dataset.

6. Conclusions

In this work, we have presented a novel re-ranking and rank aggregation approach that exploits the *Reciprocal kNN Graph* for improving image retrieval tasks. The main idea consists in analyzing the reciprocal references at top positions of ranked lists for performing re-ranking and rank aggregation tasks. The *Reciprocal kNN Graph* algorithm iteratively propagates the similarity along the dataset structure by taking into account intrinsic geometry of the dataset manifold.

We conducted a large set of experiments considering different descriptors and datasets. Experimental results demonstrated the applicability of our method to several image retrieval tasks based on shape, color, and texture descriptors. In re-ranking tasks, for example, the *Reciprocal kNN Graph* algorithm achieves gains up to +37.99% considering MAP scores. Our proposed approach also achieves very high effectiveness performance when compared with recent state-of-the-art methods on well-known datasets.

Future work focuses on optimizing the proposed re-ranking and rank aggregation methods by considering parallel architectures.

7. Acknowledgments

Authors thank AMD, FAEPEX, CAPES, FAPESP, and CNPq for financial support.

References

- [1] Almeida, J., da S. Torres, R., Leite, N.J.. BP-tree: An efficient index for similarity search in high-dimensional metric spaces. In: ACM Int. Conf. Information and Knowledge Management (CIKM'10). 2010. p. 1365–1368.
- [2] Arica, N., Vural, F.T.Y.. BAS: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters* 2003;24(9-10):1627–1639.
- [3] Bai, X., Wang, B., Wang, X., Liu, W., Tu, Z.. Co-transduction for shape retrieval. In: European Conference on Computer Vision (ECCV'2010). volume 3; 2010. p. 328–341.
- [4] Belongie, S., Malik, J., Puzicha, J.. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2002;24(4):509–522.
- [5] Boureau, Y.L., Bach, F., LeCun, Y., Ponce, J.. Learning mid-level features for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2010. p. 2559–2566.
- [6] Brodatz, P.. *Textures: A Photographic Album for Artists and Designers*. Dover, 1966.
- [7] Chatzichristofis, S.A., Boutalis, Y.S.. Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. In: Proceedings of the 6th international conference on Computer vision systems. ICVS'08; 2008. p. 312–322.
- [8] Chatzichristofis, S.A., Boutalis, Y.S.. Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval. In: Ninth International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS '08). 2008. p. 191–196.
- [9] van Gemert, J.C., Veenman, C.J., Smeulders, A.W.M., Geusebroek, J.M.. Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2010;32(7):1271–1283.
- [10] Gopalan, R., Turaga, P., Chellappa, R.. Articulation-invariant representation of non-planar shapes. In: 11th European Conference on Computer Vision (ECCV'2010). volume 3; 2010. p. 286–299.
- [11] Hoi, S.C., Liu, W., Chang, S.F.. Semi-supervised distance metric learning for collaborative image retrieval and clustering. *ACM Transactions on Multimedia Computing and Communication Applications* 2010;6(3):18:1–18:26.
- [12] Huang, J., Kumar, S.R., Mitra, M., Zhu, W.J., Zabih, R.. Image indexing using color correlograms. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97). 1997. p. 762–768.

- [13] Jegou, H., Schmid, C., Harzallah, H., Verbeek, J.. Accurate image search using the contextual dissimilarity measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2010;32(1):2–11.
- [14] Jiang, J., Wang, B., Tu, Z.. Unsupervised metric learning by self-smoothing operator. In: *International Conference on Computer Vision (ICCV'2011)*. 2011. p. 794–801.
- [15] Kotschieder, P., Donoser, M., Bischof, H.. Beyond pairwise shape similarity analysis. In: *Asian Conference on Computer Vision*. 2009. p. 655–666.
- [16] Kovalev, V., Volmer, S.. Color co-occurrence descriptors for querying-by-example. In: *International Conference on Multimedia Modeling*. 1998. p. 32.
- [17] Latecki, L.J., Lakmper, R., Eckhardt, U.. Shape descriptors for non-rigid shapes with a single closed contour. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2000)*. 2000. p. 424–429.
- [18] Ling, H., Jacobs, D.W.. Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2007;29(2):286–299.
- [19] Ling, H., Yang, X., Latecki, L.J.. Balancing deformability and discriminability for shape matching. In: *European Conference on Computer Vision (ECCV'2010)*. volume 3; 2010. p. 411–424.
- [20] Lowe, D.. Object recognition from local scale-invariant features. In: *IEEE International Conference on Computer Vision*. 1999. p. 1150–1157.
- [21] Nistér, D., Stewénius, H.. Scalable recognition with a vocabulary tree. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. volume 2; 2006. p. 2161–2168.
- [22] Ojala, T., Pietikäinen, M., Mäenpää, T.. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2002;24(7):971–987.
- [23] Oliva, A., Torralba, A.. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision* 2001;42(3):145–175.
- [24] Page, L., Brin, S., Motwani, R., Winograd, T.. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66; Stanford InfoLab; 1999. Previous number = SIDL-WP-1999-0120.

- [25] Pedronette, D.C.G., da S. Torres, R.. Shape retrieval using contour features and distance optimization. In: International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP'2010). volume 1; 2010. p. 197 – 202.
- [26] Pedronette, D.C.G., da S. Torres, R.. Exploiting pairwise recommendation and clustering strategies for image re-ranking. Information Sciences 2012;207:19–34.
- [27] Penatti, O.A.B., Silva, F.B., Valle, E., Gouet-Brunet, V., Torres, R.d.S.. Visual word spatial arrangement for image retrieval and classification. Pattern Recognition ;
- [28] Penatti, O.A.B., Valle, E., da S. Torres, R.. Encoding spatial arrangement of visual words. In: Proceedings of the 16th Iberoamerican Congress conference on Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP'11; 2011. p. 240–247.
- [29] Qin, D., Gammeter, S., Bossard, L., Quack, T., van Gool, L.. Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2011). 2011. p. 777 –784.
- [30] van Rijsbergen, C.J.. Information Retrieval. London: Butterworth-Heinemann, 1979.
- [31] Rocha, A., Almeida, J., Nascimento, M.A., Torres, R., Goldenstein, S.. Efficient and flexible cluster-and-search approach for cbir. In: Int. Conf. Advanced Concepts for Intelligent Vision Systems (ACIVS'08). volume 5259; 2008. p. 77–88.
- [32] da S. Torres, R., Falcão, A.X.. Contour Saliency Descriptors for Effective Image Retrieval and Analysis. Image and Vision Computing 2007;25(1):3–13.
- [33] van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.. Evaluating color descriptors for object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 2010;32(9):1582–1596.
- [34] Shen, X., Lin, Z., Brandt, J., Avidan, S., Wu, Y.. Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2012). 2012. p. 3013 –3020.
- [35] Stehling, R.O., Nascimento, M.A., Falcão, A.X.. A compact and efficient image retrieval approach based on border/interior pixel classification. In: ACM Conference on Information and Knowledge Management (CIKM'2002). 2002. p. 102–109.

- [36] Swain, M.J., Ballard, D.H.. Color indexing. *International Journal on Computer Vision* 1991;7(1):11–32.
- [37] Swain, M.J., Ballard, D.H.. Color indexing. *International Journal of Computer Vision* 1991;7(1):11–32.
- [38] Tao, B., Dickinson, B.W.. Texture recognition and image retrieval using gradient indexing. *Journal of Visual Communication and Image Representation* 2000;11(3):327–342.
- [39] Tu, Z., Yuille, A.L.. Shape matching and recognition - using generative models and informative features. In: *European Conference on Computer Vision (ECCV'2004)*. 2004. p. 195–209.
- [40] Wang, B., Jiang, J., Wang, W., Zhou, Z.H., Tu, Z.. Unsupervised metric fusion by cross diffusion. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2012)*. 2012. p. 3013 –3020.
- [41] Wang, J., Li, Y., Bai, X., Zhang, Y., Wang, C., Tang, N.. Learning context-sensitive similarity by shortest path propagation. *Pattern Recognition* 2011;44(10-11):2367–2374.
- [42] van de Weijer, J., Schmid, C.. Coloring local feature extraction. In: *European Conference on Computer Vision (ECCV'2006)*. .
- [43] Yang, X., Bai, X., Latecki, L.J., Tu, Z.. Improving shape retrieval by learning graph transduction. In: *European Conference on Computer Vision (ECCV'2008)*. volume 4; 2008. p. 788–801.
- [44] Yang, X., Koknar-Tezel, S., Latecki, L.J.. Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2009)*. 2009. p. 357–364.
- [45] Yang, X., Latecki, L.J.. Affinity learning on a tensor product graph with applications to shape and image retrieval. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2011)*. 2011. p. 2369–2376.
- [46] Zagoris, K., Chatzichristofis, S., Papamarkos, N., Boutalis, Y.. Automatic image annotation and retrieval using the joint composite descriptor. In: *14th Panhellenic Conference on Informatics (PCI)*. 2010. p. 143–147.
- [47] Zhang, S., Yang, M., Cour, T., Yu, K., Metaxas, D.N.. Query specific fusion for image retrieval. In: *European conference on Computer Vision (ECCV'12)*. 2012. p. 660–673.