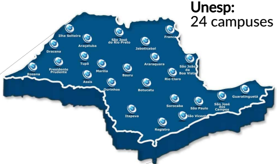
Rank-based Unsupervised Learning for Image Retrieval

Seminar at Polytechnique Montréal

Daniel Carlos Guimarães Pedronette
State University of São Paulo (UNESP)

About UNESP:



São Paulo
State University
unesp

THE WORLD UNIVERSITY RANKINGS

Young University Rankings 2022: 251–300

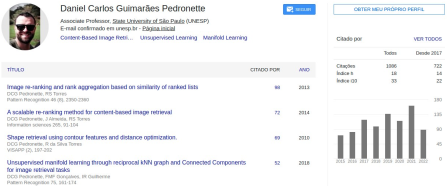
Brazil: 1st

About Me:

Daniel Carlos Guimarães Pedronette

Year	Degree	Institution
2012	PhD in Computer Science	Institute of Computing (IC) - UNICAMP
2008	Master's in Computer Science	Institute of Computing (IC) - UNICAMP
2005	Graduation in Computer Science	Inst. of Geo. and Exact Sciences (IGCE) - UNESP

- Assistant Professor - UNESP (2013)
- Associate Professor - UNESP (2019)
- Associate Editor - Pattern Reconition - Elsevier (2019)



Outline

Motivation

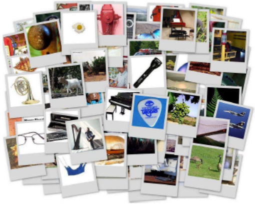
Huge growth of image collections:

- Evolution of image acquisition devices
- Reduction of storage costs
- Facilities and motivations for sharing

Motivation

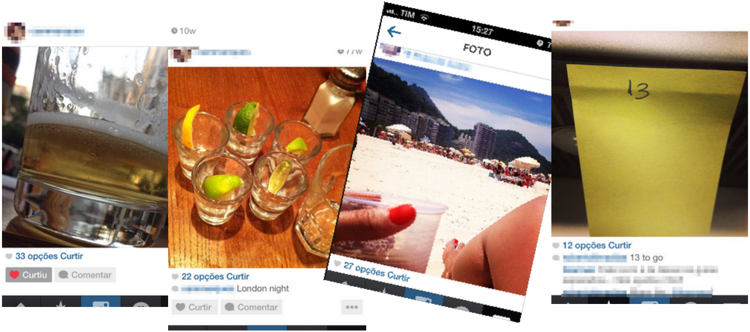
Huge growth of image collections:

- Not only a common/naive growth...
- **It is a change of behavior!**
- **People moved from consumers to producers of images.**



Motivation

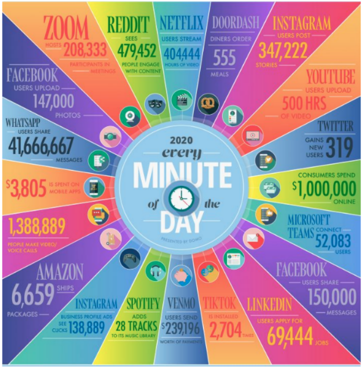
- **Change of behavior:**
  - A few figures demonstrate this:
  - Photos which probably would not exist before the digital era...



http://blogdetec.blogfolha.uol.com.br/2013/10/16/fotos-que-nao-tirariamos-se-tivessemos-que-revelar-o-filme/

Motivation

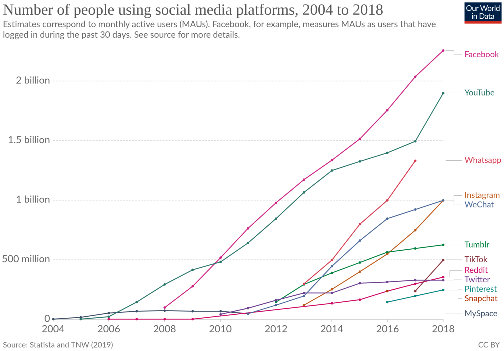
- **Huge growth of image collections:**
  - Some numbers:



https://www.domo.com/learn/data-never-sleeps-8/ (As of September/2020)

Motivation

- **Huge growth of image collections:**
  - Trends:



https://ourworldindata.org/rise-of-social-media (As of September/2020)

Motivation

- **Need for methods for indexing images:**
  - Image retrieval based on keywords and metadata
    - Ambiguous, facing serious challenges
    - Ignores the huge source of information: visual content!

Content-Based Image Retrieval

- Alternative solution?
  - **Content-Based Image Retrieval!**

Definition:

"Content-based image retrieval (CBIR), as we see it today, is **any technology that in principle helps to organize digital picture archives by their visual content**. By this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of CBIR." [15]

Content-Based Image Retrieval

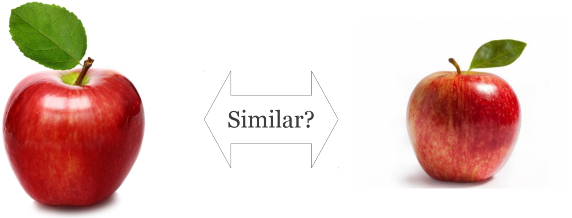
Content-Based Image Retrieval:

- **Input:**
  - Image collection
  - Query image
- **Objective:**
  - To retrieve similar images according to visual properties

Content-Based Image Retrieval

Content-Based Image Retrieval:

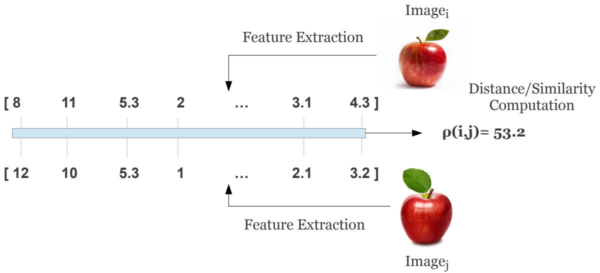
- How to measure the similarity between images?



Content-Based Image Retrieval

Comparing images:

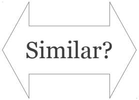
- Feature Extraction (shape, color, texture, learned-features)
- Images represented by a point in a high-dimensional space
- Distance Computation



Content-Based Image Retrieval

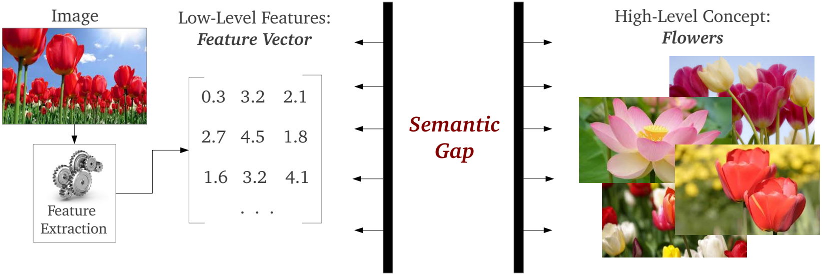
Content-Based Image Retrieval:

- How to measure the similarity between images?
- Features: shape, color, texture, learned-features?



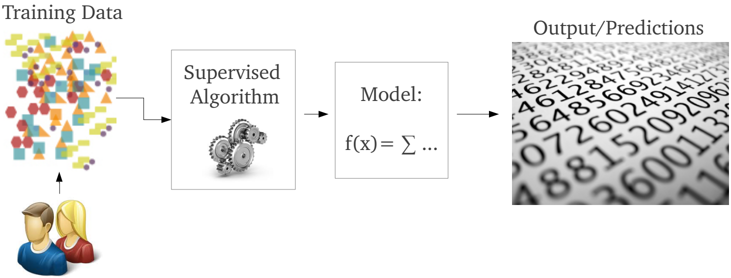
Content-Based Image Retrieval

- Limitations of CBIR Systems:
  - **"Semantic Gap"**:
  - Gap between low-level features and high-level concepts



Content-Based Image Retrieval

- Alternative Solution?:
  - Supervised Approaches
  - Relevance Feedback



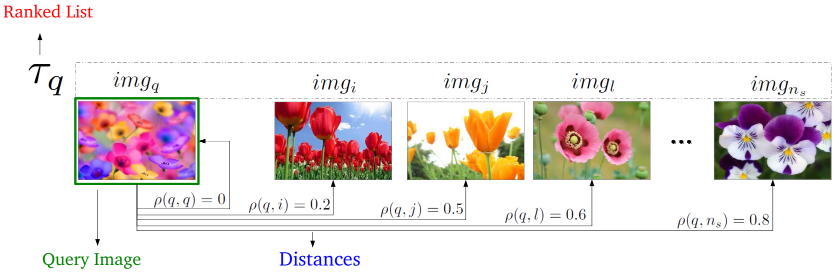
- Drawbacks:
  - Requires a lot of user intervention

Unsupervised Methods for Image Retrieval

- For decades, different visual features and distance measures have been proposed for image retrieval tasks.
- More recent research initiatives have focused on **other stages of the retrieval pipeline**, which are not directly related to feature extraction.
- **Post-processing** methods [89, 29, 90] have been proposed aiming at improving effectiveness of image retrieval tasks.
  - **Without the need of user intervention!**

Unsupervised Methods for Image Retrieval

- CBIR often performs only pairwise image analysis
  - Computes similarity (or distance) measures considering only pairs of images
  - Ignores rich information encoded in the relationships among images (**context**)



Unsupervised Distance Learning for Image Retrieval

More formally:

- Multimedia objects are often modeled as **high dimensional points in an Euclidean space**
- The distances between them often are measured by the Euclidean distance.
- Therefore, capturing and exploiting the intrinsic manifold structure becomes a central problem in the vision and learning community [28].
  - Even deep learning-based features faces similar challenges

Unsupervised Distance Learning for Image Retrieval

In general:

- Unsupervised distance learning methods propose:
  - **More general and global affinity measures** instead of strategies based on pairwise distance computations [90];
  - Capability of encoding the **geometry of dataset manifold** and structural similarity information.

Unsupervised Distance Learning for Image Retrieval

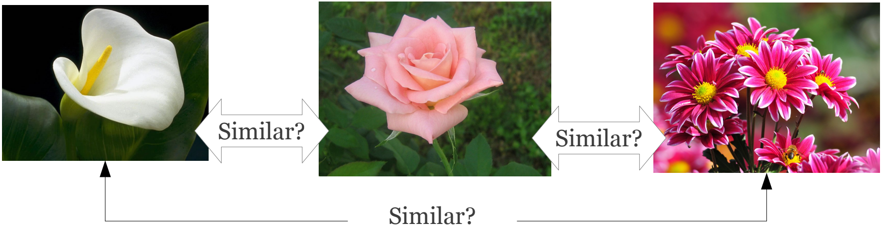
Unsupervised Distance Learning for Image Retrieval:

- **Goals**:
  - to improve the effectiveness of image retrieval tasks.
  - to reduce the Semantic Gap.
- **Strategies**:
  - Using global measures instead of pairwise distance computations
  - Considering the global dataset structure
  - Exploiting contextual information and relationship among images

Unsupervised Methods for Image Retrieval

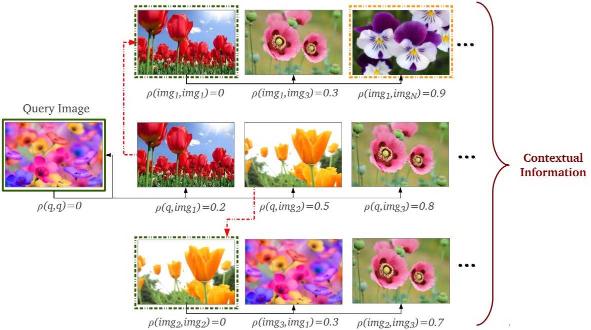
Unsupervised Methods for Image Retrieval:

- How to measure the similarity between images?
- **Answer:** in the context of other images.



Contextual Information

- **Contextual Information, encoded in ranked lists and distance among images, can be exploited to improve the effectiveness of image retrieval.**



Problem Definition

Image Descriptor:

Let D be an image descriptor. An image descriptor can be defined [13] as a tuple (ϵ, ρ):

- ϵ: $\hat{I} \rightarrow \mathbb{R}^n$ is a function, which extracts a feature vector $v_{\hat{I}}$ from an image $\hat{I}$;
- ρ: $\mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ is a distance function that computes the distance between two images according to the distance between their corresponding feature vectors.

The distance between two images img_i and img_j is given by the value of $\rho(\epsilon(img_i), \epsilon(img_j))$.
The notation $\rho(i, j)$ is used for readability purposes.

Problem Definition

Image Retrieval Model:

- Let $\mathcal{C}=\{img_1, img_2, \ldots, img_n\}$ be an image collection, where n is the size of the collection.
- Let $\rho(i,j)$ denotes the distance between two images img_i and img_j, according to a given image descriptor.
- Let $\tau_q=(img_1, img_2, \ldots, img_{n_s})$ be a ranked list, which can be defined as a permutation of the subset $\mathcal{C}_s \subset \mathcal{C}$.
  - The subset $\mathcal{C}_s$ contains the n_s most similar images to query image img_q, such that and $|\mathcal{C}_s| = n_s$.
- Taking every image $img_i \in \mathcal{C}$ as a query image img_q, a set of ranked lists $\mathcal{R} = \{\tau_1, \tau_2, \ldots, \tau_n\}$ can be computed.

Problem Definition

Unsupervised Distance Learning - Distances:

The objective is to define a function f_d which takes a distance matrix A as the input and computes a new and more effective distance matrix $\hat{A}$:

- $\hat{A} = f_d(A)$

Unsupervised Distance Learning - Ranked lists:

The 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 $\hat{\mathcal{R}}$:

- $\hat{\mathcal{R}} = f_r(\mathcal{R})$

Unsupervised Methods for Image Retrieval

Related Work:

- **Diverse Taxonomy according to the approach:**
  - Graph Transduction [88]
  - Diffusion Process [89, 86]
  - Affinity Learning [90]
  - Contextual Similarity/Dissimilarity Measures [27]
  - Context-Sensitive Similarity [87]
  - Unsupervised Metric Learning [28]
  - Re-Ranking and Rank Aggregation [23, 58, 57]
  - Unsupervised Manifold Learning [68, 61]

Unsupervised Methods for Image Retrieval

Related Work:

- **Two main categories:**
  - Diffusion Process
    - Use distance information for defining a graph
    - Spread the affinities through the graph
    - Effective, but require expensive matrix operations
  - **Rank-Based Algorithms**
    - Consider rank information, reducing computational costs
    - Excellent tradeoff effectiveness $\times$ efficiency

Rank-based Unsupervised Learning for Image Retrieval

- Various rank-based methods using different strategies: rank correlation measures, graphs, recommendation, Cartesian product, etc.
- Some representative methods:

Unsupervised Methods for Image Retrieval

- RL-Sim Algorithm [23, 58]
- Unsupervised Manifold Learning By Reciprocal kNN Graph [68]
- Unsupervised Manifold Learning By Correlation Graph [61]
- RL-Recommendation Algorithm [84]
- Hypergraph Manifold Ranking

RL-Sim Algortihm

- RL-Sim Algorithm: **R**anked **L**ists **Sim**ilarities [58]

Main Ideas:

- **Ranked lists** are a rich source of **contextual information**
  - They establishes a relationship among all collection images and not only pairs of images
- Based on the **similarity between ranked lists**, a more effective distance can be computed

RL-Sim Algortihm

Iterative Contextual Distance Measure:

- A distance measure is used to compute Ranked Lists ($\rho \rightarrow RLs$)
- Comparison between Ranked Lists can lead to more effective distance measures ($RLs \rightarrow \rho$)
- The process can be iteratively repeated
- A new contextual distance measure is **iteratively learned** in a unsupervised setting
- The measure is able to incorporate the contextual information, improving retrieval results.

RL-Sim Algortihm

Contextual Distance Measure:

- Let $\mathcal{N}(i)$ be the *neighborhood set* of image img_i.
- Let $d(\tau_i, \tau_j, k)$ denote a given distance measure for comparing top k lists give by the neighborhood set.

A non-iterative contextual distance measure can be defined as:

$$\rho_c(img_i, img_j) = d(\tau_i, \tau_j, k) \quad (1)$$

RL-Sim Algortihm

Contextual Distance Measure:

- Let $\tau_i^{(t)}$ be the top k list for image img_i at iteration t.

We can define an iterative contextual measure as follows:

$$\rho_c^{(t+1)}(img_i, img_j) = d(\tau_i^{(t)}, \tau_j^{(t)}, k) \quad (2)$$

Once the effectiveness of the contextual distance measure improves, k can be increased:

$$\rho_c^{(t+1)}(img_i, img_j) = d(\tau_i^{(t)}, \tau_j^{(t)}, k + t) \quad (3)$$

After T iterations, a definitive new distance is computed.

RL-Sim Algortihm

Require: Original set of ranked lists $\mathcal{R}$ and parameters k_s, T, λ

while $t < T$ **do**
  for all $R_i \in \mathcal{R}^{(t)}$ **do**
    $counter \leftarrow 0$
    for all $img_j \in R_i$ **do**
      if $counter \leq \lambda$ **then**
        $A^{(t+1)}[i,j] \leftarrow d(\tau_i, \tau_j, k)$
      else
        $A^{(t+1)}[i,j] \leftarrow 1 + A^{(t)}[i,j]$
      end if
      $counter \leftarrow counter + 1$
    end for
  end for
  $\mathcal{R}^{(t+1)} \leftarrow perfomReRanking(A^{(t+1)})$
  $k \leftarrow k + 1$
  $t \leftarrow t + 1$
end while
$\hat{\mathcal{R}} \leftarrow \mathcal{R}^{(T)}$

RL-Sim Algortihm

Neighborhood Sets:

Different approaches can be used for computing the top k lists :

- k-Nearest Neighborhs
- Mutual k-Nearest Neighborhs
- (...)
- others

RL-Sim Algortihm

Comparing ranked lists:

Diverse rank correlation measures can be used [10]:

- Intersection Metric
- Kendall τ
- Spearman ρ
- Goodman
- Jaccard
- Rank Biased Overlap (RBO)
- $Jaccard_l$
- $Kendall\ \tau_w$

RL-Sim Algortihm

Intersection Metric:

- Distance between two top k lists τ_i and τ_i
- Proposed in [17] aiming at capturing the cumulative overlap at increasing depths (similarity measure):

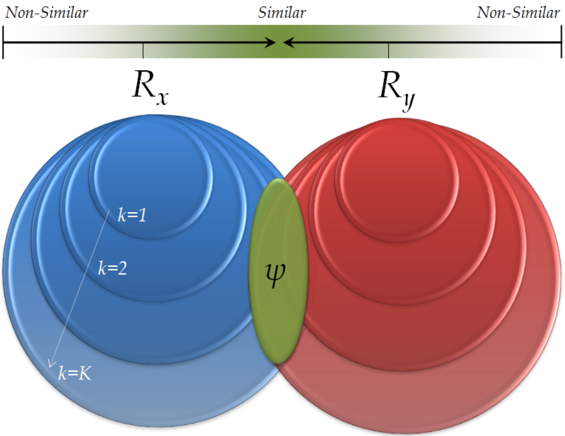
$$\psi(\tau_i, \tau_j, k) = \frac{\sum_{k_c=1}^{k} \mid \mathcal{N}(i, k_c) \cap \mathcal{N}(j, k_c) \mid}{k} \tag{4}$$

Since we are interested in a distance measure, we define d_ψ as follows:

$$d_\psi(\tau_i, \tau_j, k) = \frac{1}{1 + \psi(\tau_i, \tau_j, k)} \tag{5}$$

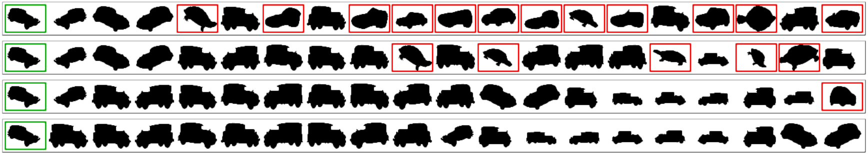
RL-Sim Algortihm

- **Intersection Metric:**



RL-Sim Algortihm

- **Iterative Visual Results:**
  - Query image (in green)
  - Wrong results (in red)



Experimental Evaluation - Intersection Metric

Shape Descriptors - MPEG-7 dataset

Positive gains ranging from +4.14% to +26.63%, considering MAP.

Descriptor	Score (MAP)	kNN	Gain	M-kNN	Gain
SS [14]	37.67%	43.06%	+14.31%	47.70%	+26.63%
BAS [4]	71.52%	74.57%	+4.25%	78.16%	+9.28%
IDSC [38]	81.70%	86.75%	+6.18%	87.67%	+7.31%
CFD [51]	80.71%	88.97%	+10.23%	90.78%	+12.48%
ASC [39]	85.28%	88.81%	+4.14%	90.88%	+6.57%
AIR [22]	89.39%	93.54%	+4.64%	93.52%	+4.62%

Experimental Evaluation - Intersection Metric

Color Descriptors - Soccer Dataset

Positive gains ranging from +4.40% to +20.28%, considering MAP scores.

Descriptor	**Score** (MAP)	**kNN**	**Gain**	**M-kNN**	**Gain**
GCH [77]	32.24%	33.66%	+4.40%	33.84%	+4.96%
ACC [24]	37.23%	43.54%	+16.95%	44.78%	+20.28%
BIC [75]	39.26%	43.45%	+10.67%	44.08%	+12.28%

Unsupervised Manifold Learning By Reciprocal kNN Graph

Main Ideas:

- The Reciprocal kNN Graph is mainly based on the information encoded in the top positions of the ranked lists.
- The algorithm uses:
  - The reciprocal nearest neighbor references (**Reciprocal kNN Score**);
  - The graph structure considering all references among images at top positions of ranked lists (**Authority Score** and **Collaborative Score**).

Unsupervised Manifold Learning By Reciprocal kNN Graph

Main Ideas:

- The Reciprocal kNN Graph is mainly based on the information encoded in the top positions of the ranked lists.
- The algorithm uses:
  - The reciprocal nearest neighbor references (**Reciprocal kNN Score**);
  - The graph structure considering all references among images at top positions of ranked lists (**Authority Score** and **Collaborative Score**).

Unsupervised Manifold Learning By Reciprocal kNN Graph

1. Compute Authority Scores
(for all ranked lists)

2. Compute Collaborative Scores
(for each pair of images at top k positions of any ranked list)

3. Compute Reciprocal kNN Distance Measure
3.1. *Based on previous position, when $C_s = 0$.*
3.2. *Based on* ***Reciprocal kNN Score*** *and Collaborative Score, when $C_s > 0$.*

Repeat until variation of **Convergence Score > ε**

Unsupervised Manifold Learning By Reciprocal kNN Graph

Authority Score:

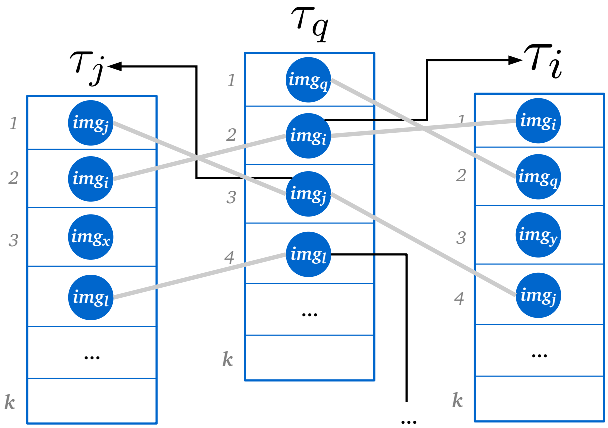
- **Motivation:** to estimate the quality of a ranked list.
- An accurate ranked list has their top images referencing to each other at the top positions of their ranked lists.

$$A_s(q,k) = \frac{\sum_{i \in \mathcal{N}(q,k)} \sum_{j \in \mathcal{N}(i,k)} f_{in}(j,q)}{k^2}, \tag{6}$$

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

Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Authority Score:**



Unsupervised Manifold Learning By Reciprocal kNN Graph

Collaborative Score:

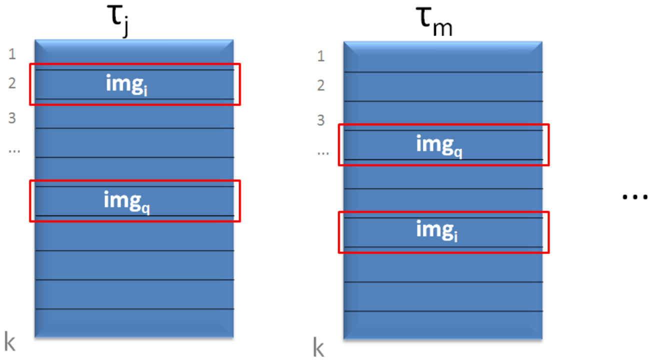
- **Motivation:** to exploit contextual information encoded in all ranked lists, according to its authority.
- If two images appears at top positions of a ranked list with high authority, they are probably similar.

$$C_s(q, i, k) = \sum_{c=1}^{k} \sum_{j \in \mathcal{C}} A_s(j, c)^2 \times f_{in}(q, i, j), \tag{7}$$

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

Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Collaborative Score:**



$$\sum \quad A_s(j)^2 \quad A_s(m)^2 \quad \ldots$$

Unsupervised Methods for Image Retrieval

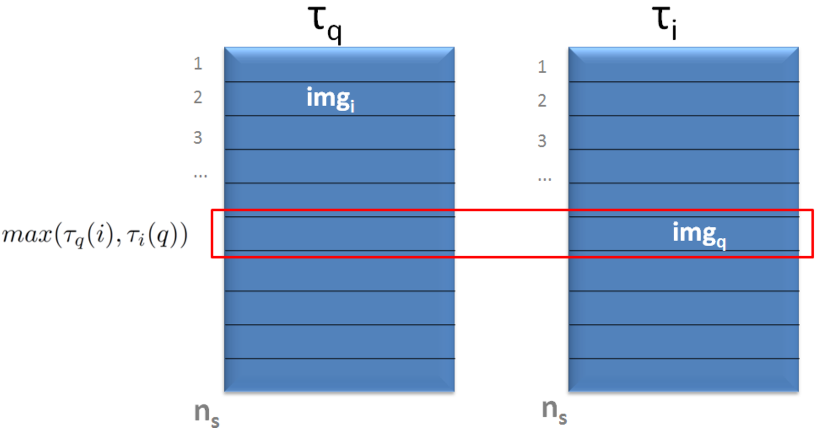
Reciprocal kNN Score:

- **Motivation:** to exploit the reciprocal neighborhood as a stronger indication of similarity.
- Give the position from which images became reciprocal neighbors.

$$R_s(q, i) = \frac{max(\tau_q(i), \tau_i(q))}{n_s}. \tag{8}$$

Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Reciprocal kNN Score:**



Unsupervised Methods for Image Retrieval

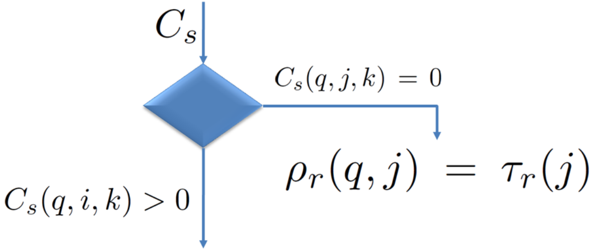
Reciprocal kNN Distance:

- The Reciprocal kNN Distance uses the Collaborative (global) and the Reciprocal kNN (local) scores for computing the new distance.
- The images with zero collaborative score keep the distance between them as their current ranking.

$$\rho_r(q, i) = \frac{R_s(q, i)}{1 + C_s(q, i, k)}. \tag{9}$$

Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Reciprocal kNN Distance:**



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

- Collaborative Score ($C_s > 0$): img_q is probably similar to image img_i.

Unsupervised Methods for Image Retrieval

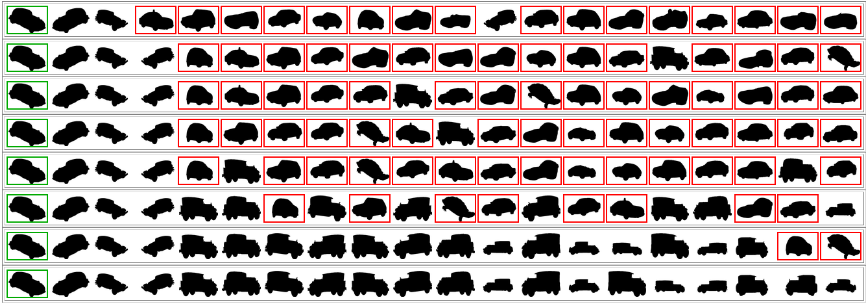
Iterative Reciprocal kNN Distance:

- Based on the distance ρ_r, the set of ranked lists $\mathcal{R}$ is updated
- The process can be iteratively repeated. We can define an iterative distance measure as follows:

$$\rho_r^{(t+1)}(q, i) = \frac{R_s^{(t)}(q, i)}{1 + C_s^{(t)}(q, i, k + t)}. \tag{10}$$

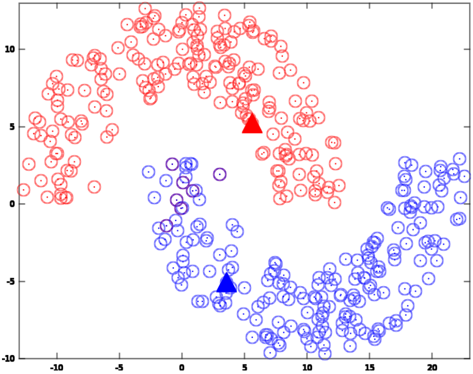
Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Iterative Visual Results:**
  - Query image (in green)
  - Wrong results (in red)



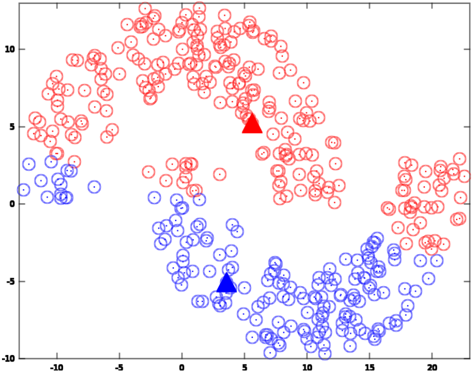
Unsupervised Manifold Learning By Reciprocal kNN Graph

- Two moons dataset: **Ideal distance**



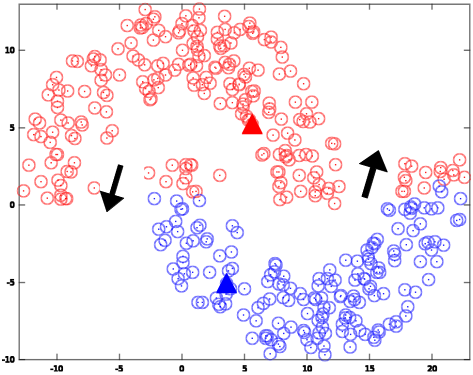
Unsupervised Manifold Learning By Reciprocal kNN Graph

- Two moons dataset: **Euclidean Distance**



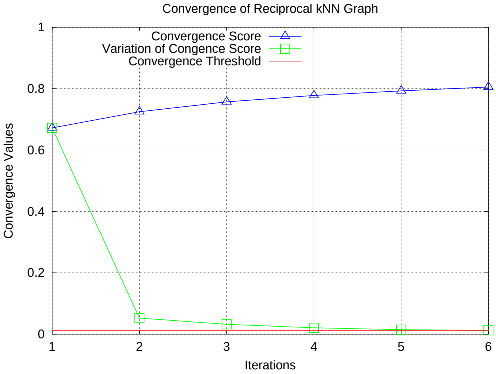
Unsupervised Manifold Learning By Reciprocal kNN Graph

- Two moons dataset: **Reciprocal kNN Distance**
  - Impact on distances after 1 iteration



Unsupervised Manifold Learning By Reciprocal kNN Graph

- **Convergence Analysis**
  - Compute while the quality of ranked lists is improved
  - **Convergence Score:** average Authority Score



Experimental Evaluation - Shape

Shape Descriptors

Positive gains ranging from +8.54% to +37.99%, considering MAP.

Descriptor	**Type**	**Dataset**	**Score** (MAP)	**Reciprocal kNN Graph**	**Gain**
SS [14]	Shape	MPEG-7	37.67%	51.98%	**+37.99%**
BAS [4]	Shape	MPEG-7	71.52%	82.01%	+14.67%
IDSC [38]	Shape	MPEG-7	81.70%	91.16%	+11.58%
ASC [39]	Shape	MPEG-7	85.28%	93.15%	+9.23%
CFD [51]	Shape	MPEG-7	80.71%	94.12%	+16.62%
AIR [22]	Shape	MPEG-7	89.39%	97.02%	+8.54%

Experimental Evaluation - Color

Color Descriptors

Positive gains ranging from +4.50% to +15.33%, considering MAP scores.

Descriptor	**Type**	**Dataset**	**Score** (MAP)	**Reciprocal kNN Graph**	**Gain**
GCH [77]	Color	Soccer	32.24%	33.69%	+4.50%
ACC [24]	Color	Soccer	37.23%	42.11%	+13.11%
BIC [75]	Color	Soccer	39.26%	45.28%	+15.33%

Experimental Evaluation - Texture

Texture Descriptors

Positive gains ranging from +3.85% to +15.16%, considering MAP scores.

Descriptor	**Type**	**Dataset**	**Score** (MAP)	**Reciprocal kNN Graph**	**Gain**
LBP [45]	Texture	Brodatz	48.40%	51.05%	+5.48%
CCOM [30]	Texture	Brodatz	57.57%	66.30%	+15.16%
LAS [78]	Texture	Brodatz	75.15%	78.04%	+3.85%

Unsupervised Manifold Learning by Correlation Graph and Strongly Connected Components

Main Ideas:

- The edges defined by the Correlation Graph provide a very **strong indication of similarity** among images (specially for high correlation thresholds).
- However, although very precise, the edges include a very small neighborhood.
- We aim at **expanding the similarity neighborhood**, but still considering the **geometry of the dataset** manifold, by using SCCs .

Unsupervised Manifold Learning by Correlation Graph and Strongly Connected Components

Contribution:

A novel Unsupervised Manifold Learning Algorithm based on the Correlation Graph and Strongly Connected Components (SCCs).

- The proposed algorithm computes a new distance which takes into account the **intrinsic geometry of the dataset manifold**.

Correlation Graph Motivation

Main Steps:

1. Constructing a graph representation of the dataset by exploiting the distance correlation between kNN constrained by a correlation threshold
2. Strongly Connected Components (SCCs) of the graph are analyzed with the aim of discovering the *intrinsic geometry of the dataset manifold*;
3. A similarity score combines information from the Correlation Graph Adjacency and Strongly Connected Components;
4. A new **Correlation Graph Distance** is computed based on the similarity score.

Correlation Graph

Graph Definition:

- Given a directed graph $G = (V, E)$, the set of vertices V is defined by the image collection $\mathcal{C}$, such that each image is represented by a node and $V = \mathcal{C}$.
- The edge set E is defined considering the distances correlation among images at the top n_s positions of each ranked list:
  - $E = \{(img_q, img_j) \mid \tau_q(j) \leqslant n_s \wedge cor(q, j) \geqslant t_c\}$,
  - $cor(q, j)$ is the correlation score between img_q and img_j
  - t_c is the correlation threshold considered.

Correlation Graph

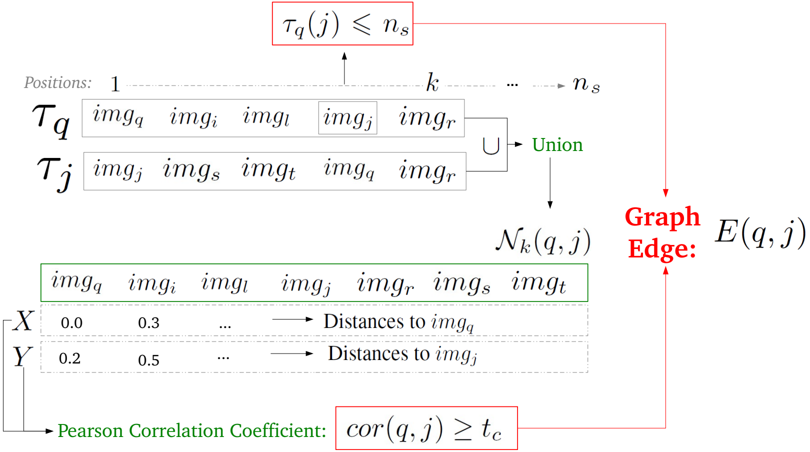
Correlation Score:

The correlation score $cor(q, j)$ is computed by the Pearson's Correlation Coefficient, considering the distances to the kNN.

- Let $\mathcal{N}_k(q)$ be the set containing the k-nearest neighbors to given image img_q and $\mathcal{N}_k(q, j) = \mathcal{N}_k(q) \cup \mathcal{N}_k(j)$.
- Vectors X and Y contain the distances from images img_q, img_j to $img_i \in \mathcal{N}_k(q, j)$:
  - $X_i = \rho(q, i)$ and $Y_i = \rho(j, i)$

$$cor(q, j) = \frac{\sum_{i=1}^{k_u}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{k_u}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{k_u}(Y_i - \overline{Y})^2}}. \quad (11)$$

Correlation Graph



Correlation Graph

Strongly Connected Components (SCCs)

The Strongly Connected Components of a directed graph are defined by subgraphs that are themselves strongly connected.

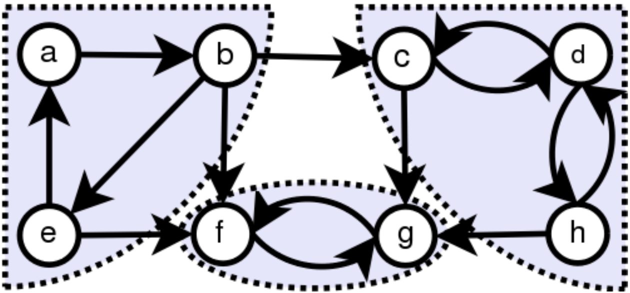
- Every vertex is reachable from every other vertex.

SCCs Computation

- Each SCC is defined as a set of images $\mathcal{S}_i$, computed using Tarjan's [79] Algorithm.
- The overall output of the algorithm is a set of SCCs $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_m\}$

Correlation Graph

- Strongly Connected Components (SCCs):
  - Sets of similar images



Correlation Graph Distance - Algorithm

Require: Correlation Graph $G = (V, E)$, Set of SCCs $\mathcal{S}$
Ensure: Correlation Graph Similarity Score $W_{i,j}$

```
1:  t_c ← t_start
2:  while t_c ⩽ 1 do
3:     { Correlation Graph Adjacency }
4:     for all img_q ∈ V  do
5:        for all img_i, img_j ∈ E(q)  do
6:           W_{i,j} ← W_{i,j} + t_c
7:        end for
8:     end for
9:     { Strongly Connected Components }
10:    for all S_c ∈ S  do
11:       for all img_i, img_j ∈ S_c  do
12:          W_{i,j} ← W_{i,j} + t_c
13:       end for
14:    end for
15:    t_c ← t_c + t_inc
16: end while
```

Correlation Graph Distance

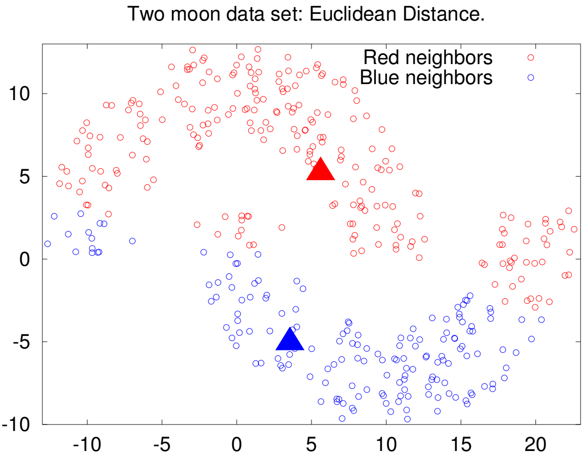
The similarity score $W_{i,j}$ uses information from both Correlation Graph Adjacency and Strongly Connected Components (SCCs).

Correlation Graph Distance

Based on the similarity score $W_{i,j}$, the *Correlation Graph Distance* $\rho_c(i,j)$ is computed:

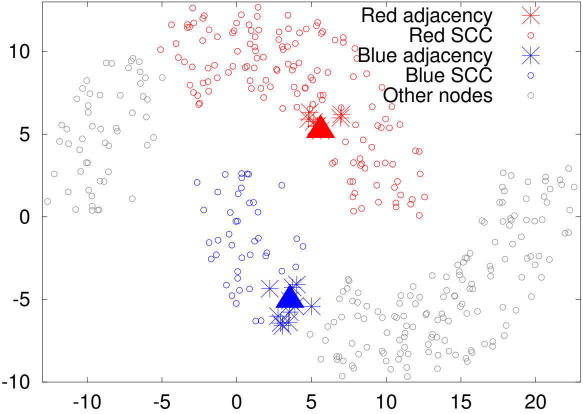
$$\rho_c(i,j) = \frac{1}{1 + W_{i,j}}. \tag{12}$$

Example: Euclidean Distance

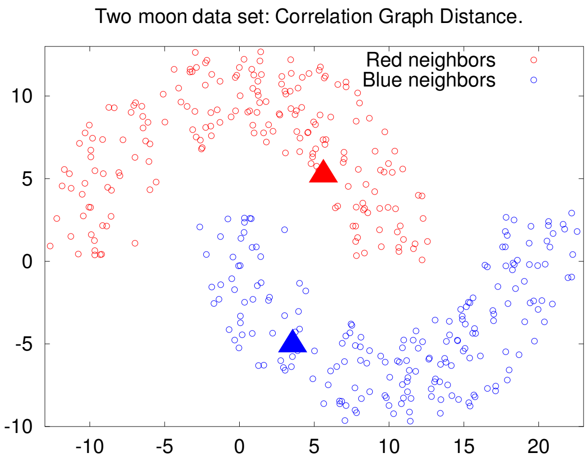


Example: Intermediary Correlation Graph Structures

Two moons data set: Correlation Graph at an intermediary threshold.



Example: Correlation Graph Distance



Experimental Evaluation

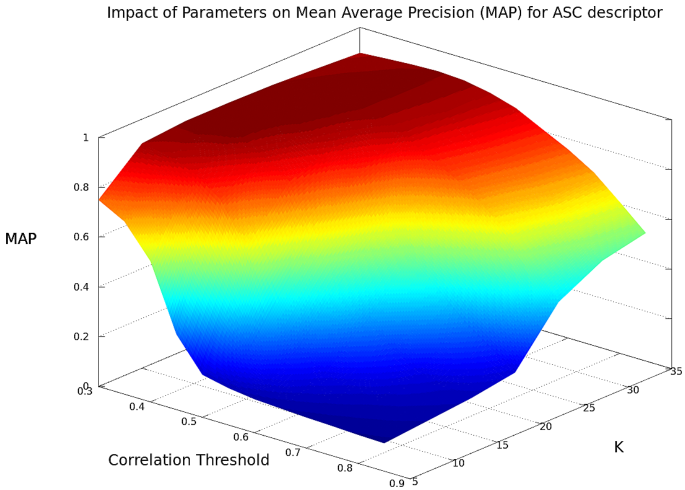
Experimental Evaluation

- Evaluation of impact of parameters
- 4 different datasets
- 13 CBIR descriptors
  - shape, color, and texture
- Statistical tests (t-tests)
- Comparison with state-of-the-art approaches

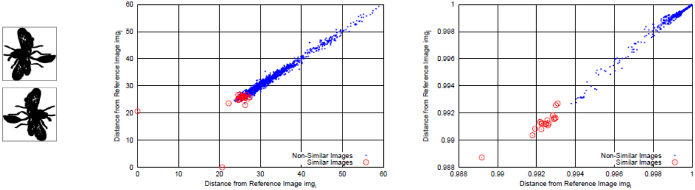
Results

- Effectiveness gains up to +34.54%.

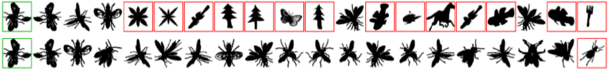
Impact of Parameter on Effectiveness



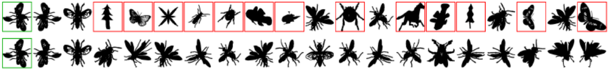
Impact of Algorithm on Distances - Similar Images



Impact of the algorithm on distances distribution for similar reference images: (a) Similar Reference Images (fly-2.gif and fly-3.gif) from the MPEG-7 [25] dataset; (b) Original distances distribution; (c) Distances distribution after the algorithm.

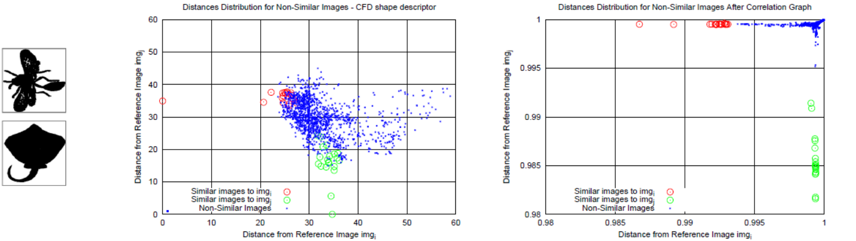


Visual example of the effectiveness gain. Retrieval results before (first row) and after the use of the algorithm (second row). Query image (fly-2.gif) from the MPEG-7 [25] dataset with green border and wrong images with red borders.

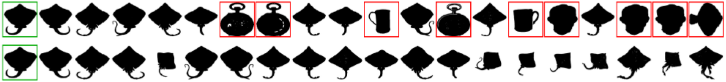


Visual example analogous to Figure 10, considering other similar query image (fly-3.gif) from the MPEG-7 [25] dataset.

Impact of Algorithm on Distances - Non-Similar Images



Impact of the algorithm on distances distribution for non-similar images: (a) Non-similar reference images (fly-2.gif and ray-16.gif) from the MPEG-7 [25] dataset; (b) Original distances distribution; (c) Distances distribution after the algorithm.



Visual examples of retrieval results before and after the algorithm, considering the query image ray-16.gif.

Experimental Evaluation - Shape

Shape Descriptors

Positive gains ranging from +7.25% to +34.54%, considering MAP scores.

Descriptor	Dataset	Score (MAP)	Correlation Graph Distance	Gain	Statistical Significance 99%
SS [14]	MPEG-7 [31]	37.67%	50.68%	+34.54%	•
BAS [4]	MPEG-7 [31]	71.52%	81.97%	+14.61%	•
IDSC [38]	MPEG-7 [31]	81.70%	89.39%	+9.41%	•
CFD [51]	MPEG-7 [31]	80.71%	91.93%	+13.90%	•
ASC [39]	MPEG-7 [31]	85.28%	92.53%	+7.25%	•
AIR [22]	MPEG-7 [31]	89.39%	97.98%	+9.61%	•

Experimental Evaluation - Shape

Shape Descriptors

Positive gains ranging from +6.90% to +29.28%, considering Bull's Eye Score (Recall@40).

Shape Descriptor	**Bull's Eye Score**	**Correlation Graph Distance**	**Gain**
SS [14]	43.99%	56.88%	+29.28%
BAS [4]	75.20%	86.52%	+15.05%
IDSC [38]	85.40%	92.20%	+7.80%
CFD [51]	84.43%	94.27%	+11.65%
ASC [39]	88.39%	95.22%	+7.73%
AIR [22]	93.67%	100%	+6.90%

Experimental Evaluation - Color

Color Descriptors

Positive gains ranging from +7.29% to +21.51%, considering MAP scores.

Descriptor	**Dataset**	**Score (MAP)**	**Correlation Graph Distance**	**Gain**	**Statistical Significance 99%**
GCH [77]	Soccer [85]	32.24%	34.59%	+7.29%	•
ACC [24]	Soccer [85]	37.23%	45.24%	+21.51%	•
BIC [75]	Soccer [85]	39.26%	47.37%	+20.65%	•

Experimental Evaluation - Texture

Texture Descriptors

Positive gains ranging from +6.28% to +12.44%, considering MAP scores.

Descriptor	**Dataset**	**Score (MAP)**	**Correlation Graph Distance**	**Gain**	**Statistical Significance 99%**
LBP [45]	Brodatz [8]	48.40%	50.12%	+3.55%	•
CCOM [30]	Brodatz [8]	57.57%	64.73%	+12.44%	•
LAS [78]	Brodatz [8]	75.15%	79.87%	+6.28%	•

Experimental Evaluation - Object Retrieval

Object Retrieval - Color Descriptors

Positive gains ranging from +4.39% to +18.10%, considering MAP scores.

Descriptor	Dataset	Score (MAP)	Correlation Graph Distance	Gain
BIC [75]	ETH-80 [32]	49.72%	54.20%	+9.01%
ACC [24]	ETH-80 [32]	48.50%	50.63%	+4.39%
CSD [43]	ETH-80 [32]	48.46%	57.23%	+18.10%
GCH [77]	ETH-80 [32]	41.62%	45.07%	+8.29%

Comparison to State-of-the-Art

Algorithm	Descriptor(s)	Bull's Eye Score
LCDP [89]	IDSC [38]	93.32%
Shortest Path Propagation [87]	IDSC [38]	93.35%
Mutual kNN Graph [29]	IDSC [38]	93.40%
Pairwise Recommendation [57]	ASC [39]	94.66%
RL-Sim [58]	ASC [39]	94.69%
Correlation Graph Distance	**ASC [39]**	**95.22%**
LCDP [89]	ASC [39]	95.96%
Tensor Product Graph [90]	ASC [39]	96.47%
Self-Smoothing Operator [28]	SC [6] +IDSC [38]	97.64%
Pairwise Recommendation [57]	CFD [51]+IDSC [38]	99.52%
RL-Sim [58]	AIR [22]	99.94%
Tensor Product Graph [90]	AIR [22]	99.99%
Correlation Graph Distance	**AIR [22]**	**100%**

RL-Recommendation Algorithm: Motivation

RL-Recommendation: Motivation

- Various methods have demonstrated the high potential for producing relevant *effectiveness* gains.
- Most of approaches consider only effectiveness.
- However, for real-word applications, the three aspects should be considered:
  - **Effectiveness:** quality of the retrieval process,
  - **Efficiency:** the time spent to obtain the results
  - **Scalability:** the capability of handling growing image collections

RL-Recommendation Algorithm

Contribution:

- A novel a novel unsupervised distance learning method for improving the **effectiveness** of image retrieval tasks.
- The proposed method is **scalable** and **efficient** as it exploits parallel and heterogeneous computing on CPU and GPU devices.

RL-Recommendation Algorithm

Main Steps:

1. **Computing the Sparse Distance Matrix:**
   - The input of the algorithm is a set of ranked lists
   - The recommendation are performed based on distance scores
   - Ranked lists are used for computing a sparse distance matrix, maintaining scalability properties
2. **Computing the Cohesion Measure:**
   - Density of references among images at top positions
   - Unsupervised estimation of effectiveness of ranked lists
   - Also used as convergence criterion

RL-Recommendation Algorithm

Main Steps:

3. **Performing Unsupervised Recommendations:**
   - Top positions of ranked lists results with higher accuracy
   - Two images at top position of a ranked lists are recommended to each other
   - Recommendations indicated decrease of distance between images
4. **Sorting Ranked Lists:**
   - Recommendations update distance among images
   - Ranked lists must reflect the updates
   - Sorting of ranked lists according to new distances

Computing Sparse Distance Matrix

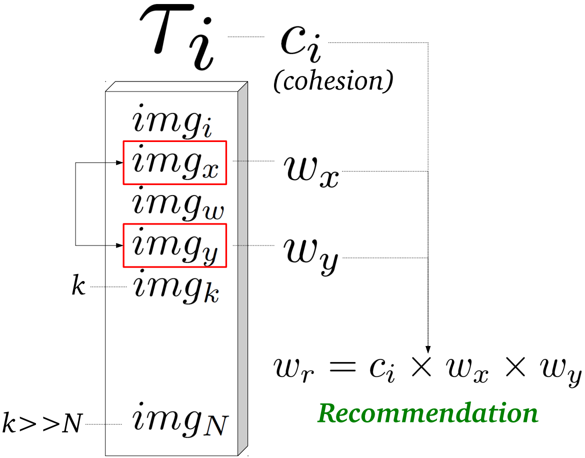
- Computing distances from ranking information:
  - Distance is computed based on the **sum of the reciprocal references** at their ranked lists.
  - $\rho(q,i) = \tau_q(i) + \tau_i(q)$
- Only images at top-L positions have their distances computed
  - Sparse distance matrix
  - Scalability purposes
- An algorithm was proposed for non-symmetric references between ranked lists

Computing Sparse Distance Matrix

Require: Blank matrix A and set of ranked lists $\mathcal{R}$
Ensure: Processed sparse distance matrix A

1: **for all** $img_q \in \mathcal{C}$ **do**
2:     **for all** $img_i \in \tau_q$ **do**
3:         $A_{qi} \leftarrow 2 \times L$
4:         $A_{iq} \leftarrow 2 \times L$
5:     **end for**
6: **end for**
7: **for all** $img_q \in \mathcal{C}$ **do**
8:     **for all** $img_i \in \tau_q$ **do**
9:         $A_{qi} \leftarrow A_{qi} + \tau_q(i) - L$
10:     **end for**
11: **end for**

Unsupervised Recommendations



Cohesion Measure

Cohesion Measure

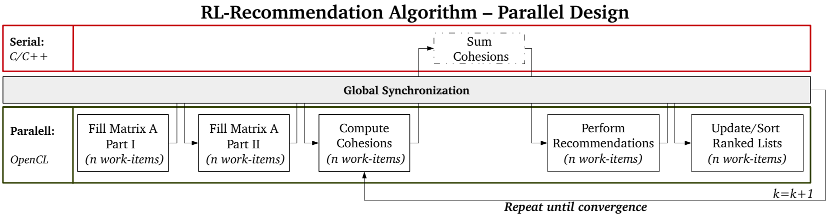
- Unsupervised estimation of the effectiveness of ranked lists
- High cohesion scores indicate that ranked lists have more **authority** to recommend

- **Density** of references among images at top positions of a given ranked list
- Convergence criterion:
  - The algorithm is iteratively computed while cohesion is increasing ($\geq \epsilon$)

Parallel Design

Parallel Design and Heterogeneous Computing

- **OpenCL:**
  - Standard for parallel and heterogeneous computing
  - Evaluation on CPU and GPU devices



Experimental Evaluation

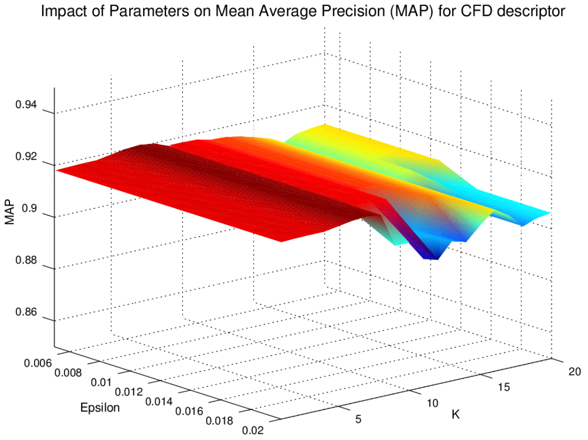
Experimental Evaluation

- Impact of parameters
- Five public datasets
  - Ranging from 280 to 70,000 images
- Effectiveness evaluation
- Efficiency evaluation
  - Serial and Parallel, CPU and GPU
- Scalability evaluation
- Comparison with state-of-the-art approaches

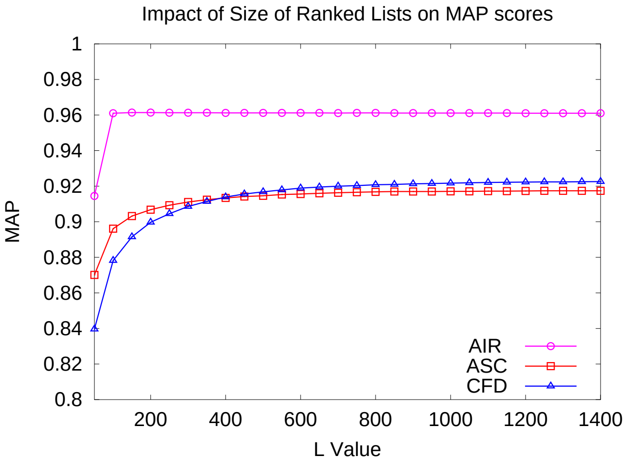
Results

- Effectiveness gains up to $+29\%$.

Impact of Parameter on Effectiveness: $k \times \epsilon$



Impact of Parameter on Effectiveness: L



Effectiveness Evaluation - Shape

Shape Descriptors -Shape Dataset (1,400 images)

Positive gains ranging from +7.11% to +29.22%, considering MAP scores.

Descriptor	Dataset	Original MAP	Pairwise Recom. [57]	RL-Recom. Serial	RL-Recom. Parallel GPU	Gain
SS [14]	MPEG-7	37.67%	39.90%	48.68%	48.64% $\pm$ 0.0062	+29.22%
BAS [4]	MPEG-7	71.52%	77.65%	79.58%	79.57% $\pm$ 0.0047	+11.27%
IDSC [38]	MPEG-7	81.70%	86.83%	88.80%	88.78% $\pm$ 0.0067	+11.86%
CFD [51]	MPEG-7	80.71%	91.38%	91.39%	91.37% $\pm$ 0.0055	+13.23%
ASC [39]	MPEG-7	85.28%	91.80%	91.34%	91.32% $\pm$ 0.0050	+7.11%
AIR [22]	MPEG-7	89.39%	95.50%	96.12%	96.12% $\pm$ 0.0071	+7.53%

Effectiveness Evaluation - Color

Color Descriptors - Soccer Dataset (280 images)

Positive gains ranging from +6.64% to +15.00%, considering MAP scores.

Descriptor	Dataset	Original	Pairwise	RL-Recom.	RL-Recom.	Gain
GCH [77]	Soccer	32.24%	32.35%	34.38%	34.44% $\pm$ 0.0340	+6.64%
ACC [24]	Soccer	37.23%	40.31%	41.23%	41.20% $\pm$ 0.0239	+10.74%
BIC [75]	Soccer	39.26%	42.64%	45.15%	45.17% $\pm$ 0.0693	+15.00%

Effectiveness Evaluation - Texture

Texture Descriptors - Brodatz Dataset (1,776 images)

Positive gains ranging from +5.91% to +11.76%, considering MAP scores.

Descriptor	Dataset	Original	Pairwise	RL-Recom.	RL-Recom.	Gain
LBP [45]	Brodatz	48.40%	51.92%	51.26%	51.24% $\pm$ 0.0047	+5.91%
CCOM [30]	Brodatz	57.57%	66.46%	64.34%	64.32% $\pm$ 0.0059	+11.76%
LAS [78]	Brodatz	75.15%	80.73%	79.71%	79.71% $\pm$ 0.0031	+6.07%

Effectiveness Evaluation - Natural Image Retrieval

Natural Image Retrieval - N-S Dataset (10,200 images)

Positive gains ranging from +3.23% to +13.39%, considering MAP scores.

Descriptor	Type	Original Score	RL-Recom.	Gain
ACC [24]	Color	3.36	3.53	+5.06%
BIC [75]	Color	3.04	3.15	+3.62%
CEED [11]	Color/Text.	2.61	2.72	+4.21%
FCTH [12]	Color/Text.	2.73	2.80	+2.56%
JCD [91]	Color/Text.	2.79	2.88	+3.23%
SIFT [41]	Local	2.54	2.88	+13.39%

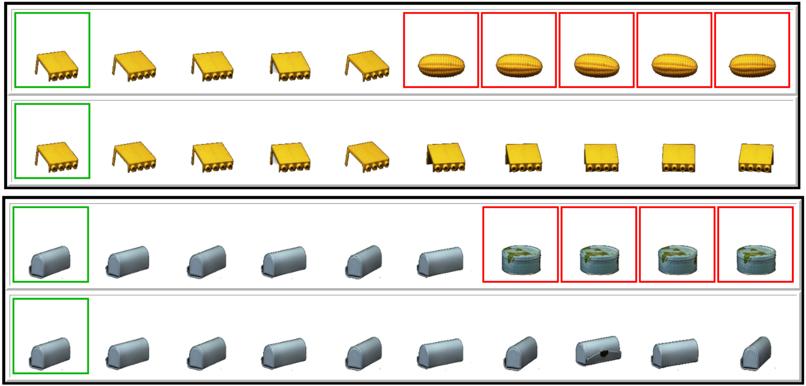
Effectiveness Evaluation - Object Retrieval

Natural Image Retrieval - ALOI Dataset (70,000 images)

Positive gains ranging from +11.67% to +23.42%, considering MAP scores.

Descriptor	**Original MAP**	**Baseline:** RL-Sim [23]	**RL-Recom.**	**Gain**
ACC [24]	44.15%	46.12%	50.11%	+13.50%
BIC [75]	71.95%	78.84%	80.35%	+11.67%
CCV [47]	47.77%	50.96%	53.52%	+12.04%
GCH [77]	50.87%	53.14%	55.81%	+9.71%
LCH [42]	58.85%	66.03%	72.63%	+23.42%

Effectiveness Evaluation: Visual Results



Efficiency Evaluation - Various Datasets

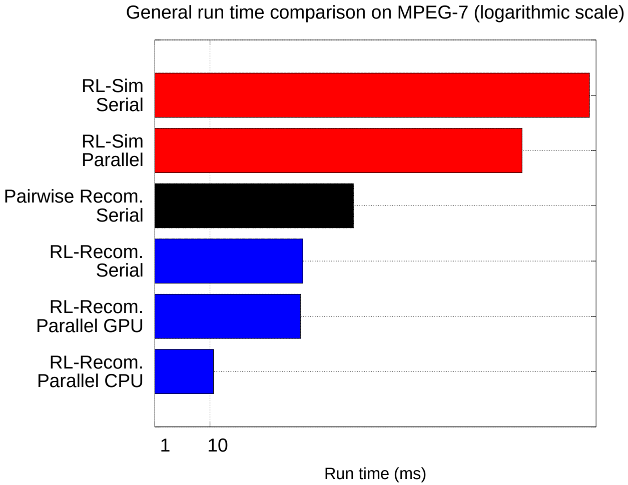
Efficiency Evaluation

Serial, Parallel CPU and GPU

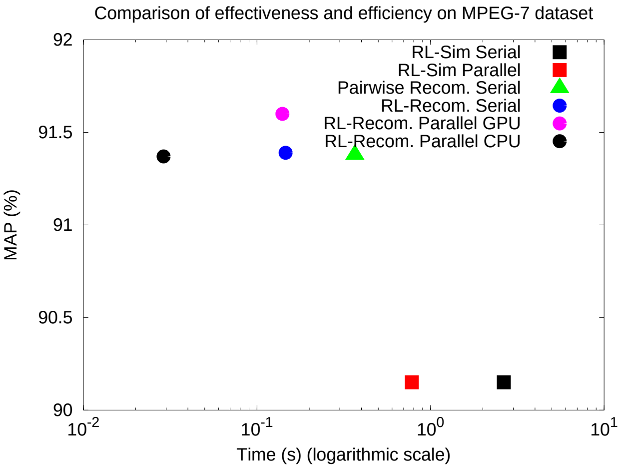
Algorithm	Exec.	Device	Soccer [85]	MPEG-7 [31]	Brodatz [8]	N-S Dataset [44]
Pairwise Recom. [57]	Serial	CPU	0.1149 $\pm$ 0.00018	0.3663 $\pm$ 0.00094	0.6672 $\pm$ 0.00140	14.802 $\pm$ 0.11059
RL-Recommendation	Serial	CPU	0.0607 $\pm$ 0.00000	0.1462 $\pm$ 0.00021	0.1108 $\pm$ 0.00102	0.1868 $\pm$ 0.00018
RL-Recommendation	Parallel	GPU[1]	0.1380 $\pm$ 0.00642	0.1401 $\pm$ 0.00250	0.1004 $\pm$ 0.00412	**0.0582 $\pm$ 0.00633**
RL-Recommendation	Parallel	GPU[2]	0.1538 $\pm$ 0.01056	0.2438 $\pm$ 0.00371	0.2376 $\pm$ 0.00326	0.3754 $\pm$ 0.00604
RL-Recommendation	Parallel	CPU[1]	0.0131 $\pm$ 0.00100	0.0319 $\pm$ 0.00043	0.0299 $\pm$ 0.00129	0.1166 $\pm$ 0.00085
RL-Recommendation	Parallel	CPU[2]	**0.0128 $\pm$ 0.00104**	**0.0290 $\pm$ 0.00075**	**0.0284 $\pm$ 0.00114**	0.1149 $\pm$ 0.00055

Memory Transfer Model: [1]Write Buffer; [2]Map Buffer.

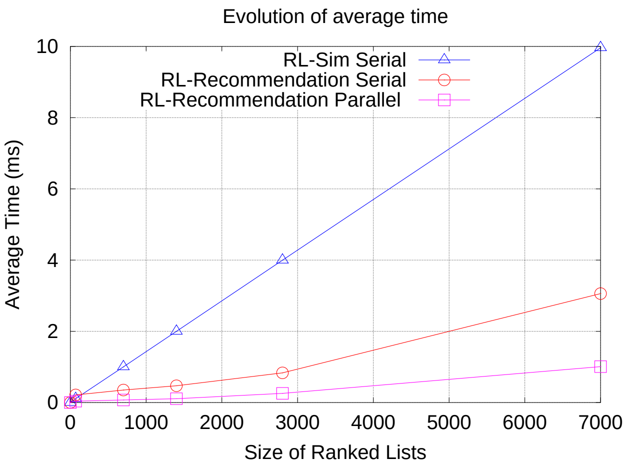
Efficiency Evaluation: different executions



Effectiveness and Efficiency Evaluation: MPEG-7 dataset



Scalability Evaluation: ALOI dataset



Comparison to State-of-the-Art

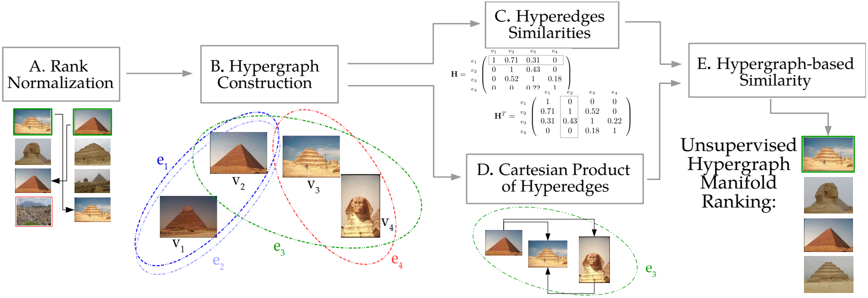
Shape Descriptors		
DDGM [81]	-	80.03%
CFD [51]	-	84.43%
IDSC [38]	-	85.40%
SC [6]	-	86.80%
ASC [39]	-	88.39%
AIR [22]	-	93.67%
Post-Processing Methods		
Algorithm	**Descriptor(s)**	**Score**
Locally C. Diffusion Process [89]	IDSC	93.32%
Shortest Path Propagation [87]	IDSC	93.35%
Mutual kNN Graph [29]	IDSC	93.40%
RL-Sim [58]	CFD	94.13%
RL-Recommendation	**CFD**	**94.38%**
RL-Recommendation	**ASC**	**94.40%**
Locally C. Diffusion Process [89]	ASC	95.96%
Self-Smoothing Operator [28]	SC+IDSC	97.64%
Co-Transduction [5]	SC+IDSC	97.72%
Self-Smoothing Operator [28]	SC+IDSC+DDGM	99.20%
Pairwise Recommendation [57]	CFD+IDSC	99.52%
RL-Recommendation	**AIR**	**99.78%**
Tensor Product Graph [90]	AIR	99.99%

Hypergraph Manifold Ranking

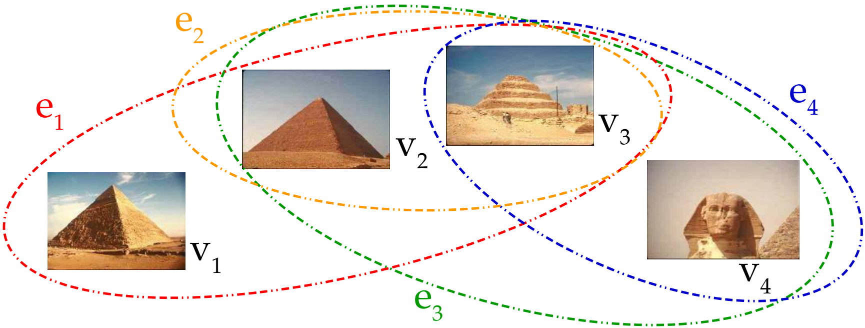
Log-based Hypergraph of Ranking References:

- Hypergraphs are a generalization of graphs
- Graphs often model pairwise relationships
- Many relationships among objects are more complex than pairwise
- **Main ideas:**
  - Each query defines a hyperedge
  - Similarity between images is given by similarity between hyperedges
  - Cartesian product among elements in a hyperedge

Hypergraph Manifold Ranking



Hypergraph Manifold Ranking



Hypergraph Manifold Ranking

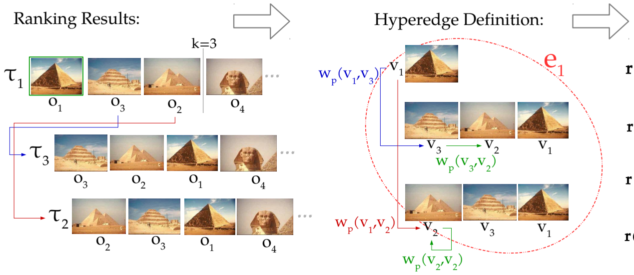
Log-based Hypergraph of Ranking References:

$$h(e_i, v_j) = \begin{cases} r(e_i, v_j), & \text{if } v_j \in e_i, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

$$r(e_i, v_j) = \sum_{o_x \in \mathcal{N}(i,k) \wedge o_j \in \mathcal{N}(x,k)} w_p(i, x) \times w_p(x, j), \quad (14)$$

$$w_p(i, x) = 1 - \log_k \tau_i(x). \quad (15)$$

Hypergraph Manifold Algorithm



Hyperedge *vs* Vertice

$$r(e_1,v_2) = w_p(o_1,o_3) \times w_p(o_3,o_2) + w_p(o_1,o_2) \times w_p(o_2,o_2)$$

$$r(e_1,v_2) = (1-\log_3 2) \times (1-\log_3 2) + (1-\log_3 3) \times (1-\log_3 1)$$

$$r(e_1,v_2) = (0.63) \times (0.63) + (0) \times (1)$$

$$r(e_1,v_2) = 0.40$$

Hypergraph Manifold Ranking

Hyperedge Weights

$$\mathcal{N}_h(q,k) = \{\mathcal{S} \subseteq e_q, |\mathcal{S}| = k \land \forall o_i \in \mathcal{S}, o_j \in e_q - \mathcal{S} : h(q,i) > h(q,j)\}. \tag{16}$$

$$w(e_i) = \sum_{j \in \mathcal{N}_h(i,k)} h(i,j). \tag{17}$$

Hypergraph Manifold Ranking

Hyperedge Similarities

$$S_h = HH^T \tag{18}$$

$$S_v = H^T H \tag{19}$$

$$S = S_h \circ S_v \tag{20}$$

Hypergraph Manifold Ranking

Cartesian Product of Hyperedge Elements

$$e_q \times e_i = \{(v_x, v_y) : \; v_x \in e_q \wedge v_y \in e_i\}. \tag{21}$$

$$p(e_q, v_i, v_j) = w(e_q) \times h(e_q, v_i) \times h(e_q, v_j). \tag{22}$$

$$c(i,j) = \sum_{e_q \in E \wedge (v_i, v_j) \in {e_q}^2} p(v_i, v_j) \tag{23}$$

Hypergraph Manifold Ranking

Hypergraph-Based Similarity

$$W = C \circ S \tag{24}$$

Hypergraph Manifold Ranking

- Admits an efficient algorithm solution
- Can be used for rank aggregation

Hypergraph Manifold Ranking

Experimental Results

Table: Comparison with state-of-the-art on the Holidays [26] dataset (MAP score).

MAP scores for state-of-the-art methods.

Tolias *et al.* [80]	Paulin *et al.* [48]	Qin *et al.* [73]	Zheng *et al.* [94]	Sun *et al.* [76]
82.20%	82.90%	84.40%	85.20%	85.50%

Zheng *et al.* [93]	Pedronette et al. [66]	Iscen et al. [25]	Li *et al.* [37]	Liu et al. [40]
85.80%	86.19%	87.5%	89.20%	90.89 %

Hypergraph Manifold Ranking

Experimental Results

Table: Comparison with state-of-the-art on the Holidays [26] dataset (MAP score).

MAP scores for the proposed method		
Descriptor	**Baseline:** Graph Fusion [92]	**Proposed: LHRR**
ACC	66.42%	71.61%
CNN-Caffe	66.79%	70.81%
CNN-Overfeat	83.79%	85.54%
CNN-OLDFP	89.00 %	89.15%
ACC+CNN-Caffe	71.02%	81.84%
ACC+CNN-Overfeat	76.55%	86.35%
ACC+CNN-Caffe+CNN-Overfeat	80.06%	87.62%
CNN-OLDFP+CNN-Overfeat	79.36%	**90.94%**

Unsupervised Distance Learning Framework (UDLF)

Contribution:

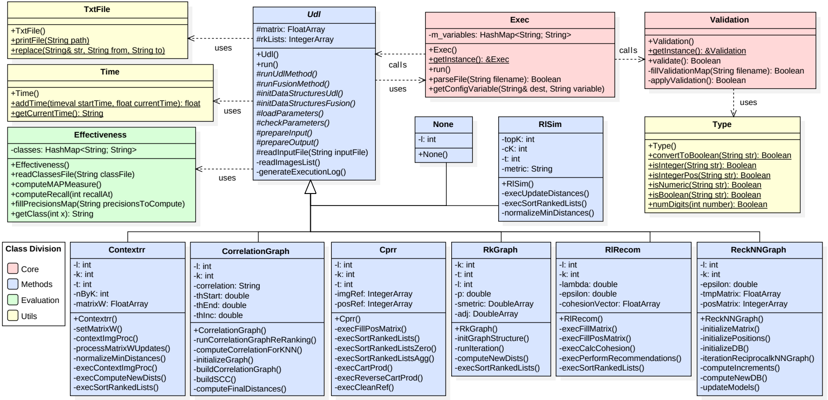
A common software environment to easily implement, use, and evaluate unsupervised learning methods

- The framework defines a general model, allowing the **implementation of different methods**
- Easy tool to **execute, evaluate and compare** unsupervised methods
- The retrieval results can be represented by distance measures or ranked lists

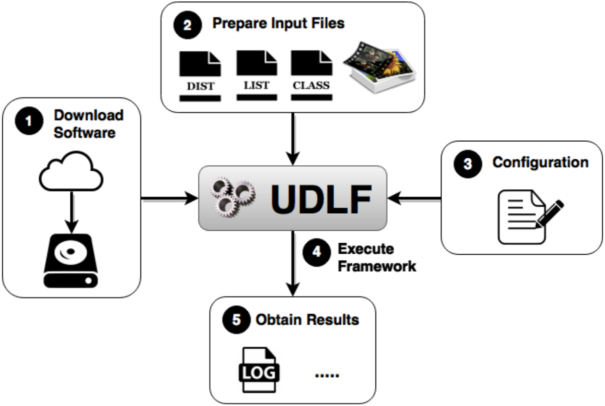
Unsupervise Distance Learning Framework (UDLF)

- UDLF is implemented on C++ trough an object-oriented paradigm
- The framework is **independent of external libraries** and portable among different operation systems
- No installation is required. Both **source code and binary releases** are available
- Different executions can be done just by changing a **configuration file**
- The framwork includes evaluation aspects, computing effectiveness measures (Precision, Recall, MAP)

Overall Organization



ULDF Execution Workflow



UDLF Configuration

Configuration:

- The framework can be configured by using a single file
- The file is validated by the framework before execution
- The configurations are divided in 5 categories:
  - **Category 1:** General Configurations
  - **Category 2:** Input File Settings
  - **Category 3:** Output File Settings
  - **Category 4:** Evaluation Settings
  - **Category 5:** Method Parameters

UDLF Configuration

- C1. General Configurations:

```
#The comments follow the structure:
#PARAMETER = VALUE #(regular expression): Explanation about the parameter
#If a regular expression is not specified, any input string can be used
#To simplify the expressions, we adopt:
#TBool  = (TRUE|FALSE)
#TUInt  = (0-9)*
#TFloat = ["+"|"-"] [0-9]* ["."] [0-9]+

#CATEGORY 1. GENERAL CONFIGURATION
UDL_TASK = UDL #(UDL|FUSION): Selection of task to be executed
UDL_METHOD = CPRR #(NONE|CPRR|RLRECOM|RLSIM|CONTEXTRR|RECKNNGRAPH|RKGRAPH|
CORGRAPH): Selection of method to be executed
```

- C2. Input file settings:

```
#CATEGORY 2. INPUT FILE SETTINGS
SIZE_DATASET = 1400 #(TUint): Number of images in the dataset
INPUT_FILE_FORMAT = MATRIX #(MATRIX|RK): Format of input file
INPUT_MATRIX_TYPE = DIST #(DIST|SIM): Type of matrix file
INPUT_RK_FORMAT = NUM #(NUM|STR): Format of ranked list file
MATRIX_TO_RK_SORTING = HEAP #(HEAP|INSERTION): Convert matrix to rks
NUM_INPUT_FUSION_FILES = 2 #(TUint): Number of files for FUSION tasks
INPUT_FILES_FUSION_1 = input1.txt #Path of the first input file
INPUT_FILES_FUSION_2 = input2.txt #Path of the second input file
#INPUT_FILES_FUSION_* = input*.txt #Path of the *th input file
INPUT_FILE = input.txt #Path of the main input file (matrix/rks)
INPUT_FILE_LIST = list.txt #Path of the list file
INPUT_FILE_CLASSES = classes.txt #Path of the classes file
INPUT_IMAGES_PATH = images/ #Dataset images path
```

UDLF Configuration

- C3. Output File Settings:

```
#CATEGORY 3. OUTPUT FILE SETTINGS
OUTPUT_FILE = TRUE #(TBool): Generate output file(s)
OUTPUT_FILE_FORMAT = MATRIX #(RK|MATRIX): Format of output file
OUTPUT_MATRIX_TYPE = DIST #(DIST|SIM): Type of matrix file to output
OUTPUT_RK_FORMAT = ALL #(NUM|STR|HTML|ALL): Output format for rks
OUTPUT_FILE_PATH = output #Path of the output file(s)
OUTPUT_HTML_RK_PER_FILE = 1 #(TUint): Number of rks for each html file
OUTPUT_HTML_RK_SIZE = 20 #(TUint): Number of images per ranked list
OUTPUT_HTML_RK_COLORS = TRUE #(TBool): Color borders around images
OUTPUT_HTML_RK_BEFORE_AFTER = TRUE #(TBool): Comparison of rks
```

- C4. Evaluation Settings:

```
#CATEGORY 4. EVALUATION SETTINGS
EFFICIENCY_EVAL = TRUE #(TBool): Enable efficiency evaluation
EFFECTIVENESS_EVAL = TRUE #(TBool): Enable effectiveness evaluation
EFFECTIVENESS_COMPUTE_PRECISIONS = TRUE #(TBool): Compute precisions
EFFECTIVENESS_COMPUTE_MAP = TRUE #(TBool): Compute MAP
EFFECTIVENESS_COMPUTE_RECALL = TRUE #(TBool): Compute recall
EFFECTIVENESS_RECALL_AT = 40 #(TUint): Position to compute recall
EFFECTIVENESS_PRECISIONS_TO_COMPUTE = 5, 20 #(TUint [","  TUint]*):
Precisions to be computed (unsigned integers separated by commas)
```

UDLF Configuration and Execution

- C5. Method Parameters:

```
#CATEGORY 5. METHOD PARAMETERS
PARAM_CPRR_L = 400 #(TUint): Size of ranked lists to consider
PARAM_CPRR_K = 20  #(TUint): Number of nearest neighbors
PARAM_CPRR_T = 2   #(TUint): Number of iterations
```

Framework execution:

- **./udlf** [config.ini]
  - Different config files can be used for distinct executions

Input/Output Data

Input/Output Data:

- Simplicity and Flexibility
  - Text files (or HTML output)
  - Similarity information defined by configuration
    - Ranked Lists
    - Distance/Similarity Matrix
- Make it part of the Retrieval Pipeline
  - Input and output files use the same format

Input Data

List File:

- List of images in the dataset
- Also used to assign an identifier to each multimedia object (line number)

List file

```
0 apple1.png
1 apple2.png
2 bird1.png
3 bird2.png
4 bat1.png
5 bat2.png
```

Input Data

Distances or Ranked Lists file:

- **Main input file:** represents the retrieval results (distances or ranked lists)
- Ranked List (string file name or numeric id)
- Distance/Similarity matrix (float separated by spaces)

Ranked list file example - string format

```
apple1.png apple2.png bird1.png bat1.png bat2.png bird2.png
apple2.png apple1.png bird2.png bird1.png bat1.png bat2.png
bird1.png bird2.png bat2.png apple2.png apple1.png bat1.png
bird2.png bird1.png bat2.png apple1.png apple2.png bat1.png
bat1.png bat2.png apple1.png apple2.png bird2.png bird1.png
bat2.png apple1.png apple2.png bat1.png bird2.png bird1.png
```

Input Data

Class File:

- Indicates the class of each multimedia object
- Used only for computing effectieveness measures
  - Precision, Recall, MAP

Classes file

```
0 apple1.png:apple
1 apple2.png:apple
2 bird1.png:bird
3 bird2.png:bird
4 bat1.png:bat
5 bat2.png:bat
```

Output Data

Log File

- General information about the execution
- Method and parameters used

```
- GENERAL INFORMATION -
Task:              UDL
Method:            CPRR
Dataset Size:      1400
Image List File:   desc/lists/mpeg7.txt
Image Class File:  desc/classes/mpeg7.txt
Input File:        desc/matrices/mpeg7/cfd.txt
Input Format:      MATRIX DIST
Output File:       output/output
Output Format:     RK ALL
--------------------------------------
- METHOD PARAMETERS -
PARAM_CPRR_K = 20
PARAM_CPRR_L = 400
PARAM_CPRR_T = 2
--------------------------------------
```

Output Data

Log File

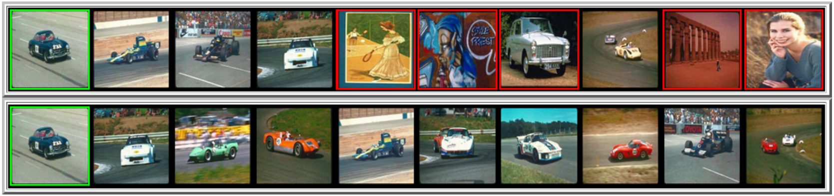
- Evaluation Results
  - Effectiveness (configured measures)
  - Efficiency (time)

```
-------------------------------------
- EVALUATION RESULTS -
* Efficiency: Total Time of the Algorithm Execution: 0.0438 s
* Effectiveness:
Before:
    P@20        0.7559
    Recall@40   0.8444
    MAP         0.8064
After:
    P@20        0.8979
    Recall@40   0.9477
    MAP         0.9215
Relative Gains:
    P@20        +18.7866%
    Recall@40   +12.2404%
    MAP         +14.2707%
-------------------------------------
Log generated at 2017/1/26 16:37:24
```

Output Data

Various formats available:

- Ranked lists (string or numeric id)
- Distance/Similrity Matrix (float speárated by space)
- HTML:



Output Data

Other visual examples

- HTML:

pyUDLF

pyUDLF

- Wrapper for acessing ULDF methods in Python

```python
from pyUDLF import run_calls as udlf
from pyUDLF import inputType

# 1) Defining the paths to the binary and configuration file
udlf.setBinaryPath("/home/usr/Desktop/UDLF/UDLF/bin/udlf")
udlf.setConfigPath("/home/usr/Desktop/UDLF/UDLF/bin/minha_config.ini")

# 2) Set functions examples
files_path = "../Soccer/matrices/distance/acc.txt"
classes_path = "/home/gustavo/Desktop/UDLF/UDLF/Soccer/classes.txt"

input_data = inputType.InputType()
input_data.set_param("UDL_TASK", "UDL")
input_data.set_param("PARAM_NONE_L", 1400)
input_data.set_method_name("CPRR")
input_data.set_param("PARAM_CPRR_L", 280)
```

https://github.com/UDLF/pyUDLF

Discussion

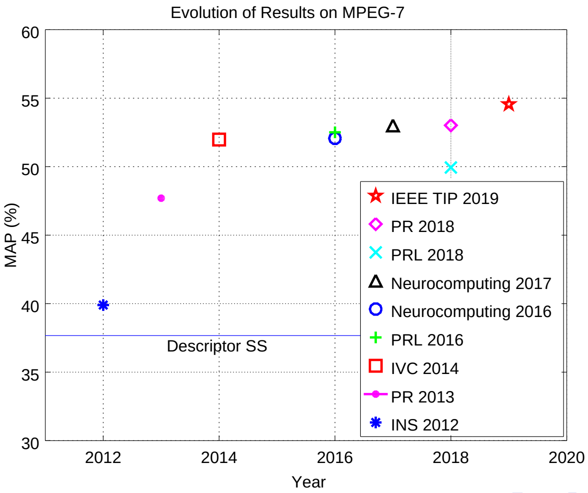
In general:

- Unsupervised Distance Learning methods can achieve **significant effectiveness gains** for image retrieval tasks
  - Without the need of user intervention
  - Capacity of considering the **intrinsic dataset geometry**
- Recent advances in **effectiveness** and **scalability**, enabling real-world applications

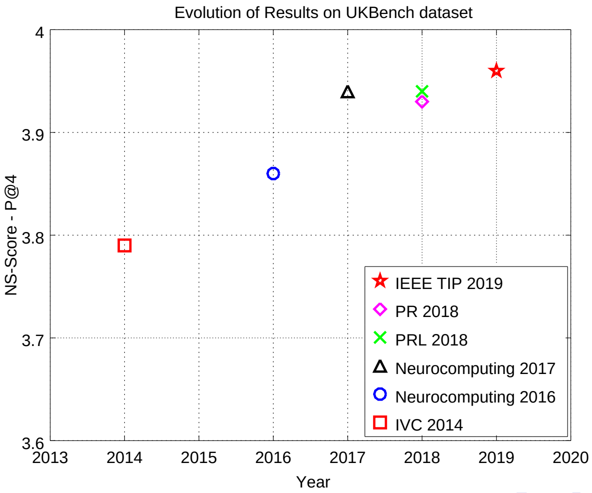
Discussion

- **What these methods have in common?**
  - All the methods are rank-based approaches
  - The use of the ranked lists information represents an important advantage: **the top positions encodes the most relevant information**, reducing the computations costs.
  - The complexity of some rank-based algorithms is defined by the re-sorting procedures at top L positions, therefore $O(n \times L \ log \ L)$ or $O(n)$ if $L \ll N$.
  - Various related approaches, which uses distance information and diffusion processes, have typically computational complexity of $O(n^3)$

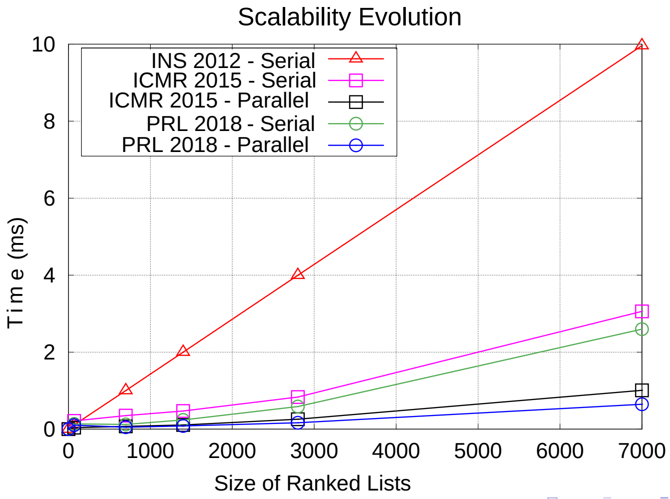
Evolution of the methods:



Evolution of the methods:



Evolution of the methods:



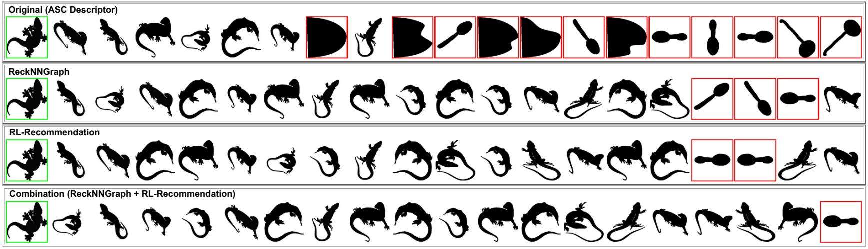
Discussion

- Why to have various methods? Which method to use?
- Each approach encodes the contextual information in a specific way, producing different results in different situations.
- One method can be more adequate for a descriptor or dataset in particular
- Different methods generate **complementary information, and therefore can be combined** [55].

Combination of Methods

Selection and Combination

Based on effectiveness and correlation



Key Challenge in Machine Learning

Data hungry algorithms:

- Excelent results!
- Need for huge sets of training data

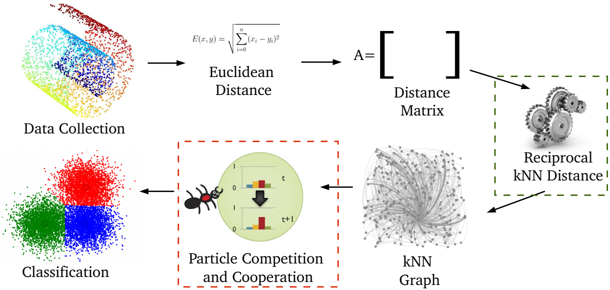
- **But how to train with less labeled data?**
- **Or with no labeled data?**

Applications in Machine Learning

- Several machine learning algorithms are based on similarity between data elements
- The use of **unsupervised distance learning** algorithms can provide:
  - contextual information
  - more effective similarity measures
- More effective similarity measures can lead to:
  - **train better with less labeled data**

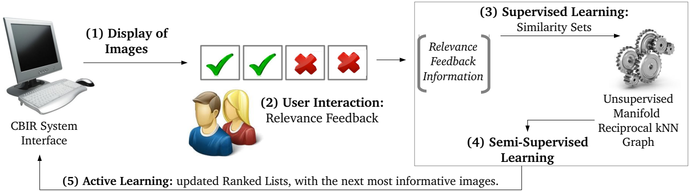
Applications in Machine Learning

- Similarity pre-processing by unsupervised distance learning for graph-based classification methods
- Classification based in semi-supervised learning [7]
- Classification based in supervised learning [1]



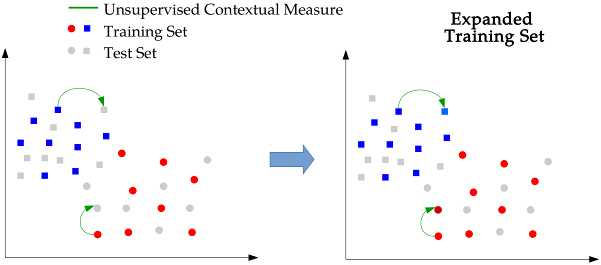
Applications in Machine Learning

- Interative image retrieval based on semi-supervised learning [69]
- Unsupervised manifold learning as a step



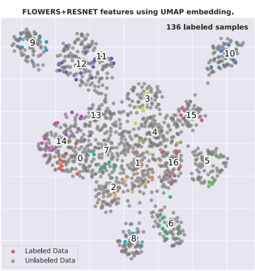
Applications in Machine Learning

- Weakly supervised learning based on ranking information [72]
- Label expansion based on rank correlation measures

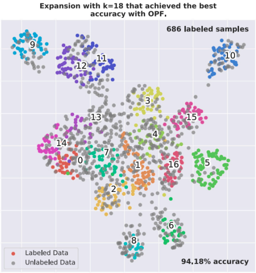


Applications in Machine Learning

- Weakly supervised learning based on a rank-based hypergraph [71]
- Label expansion based on hypergraph measures



(a) **Initial Training Set**



(b) **Expanded Training Set**

Applications in Machine Learning

- Similarity pre-processing for improved clustersing tasks [74]

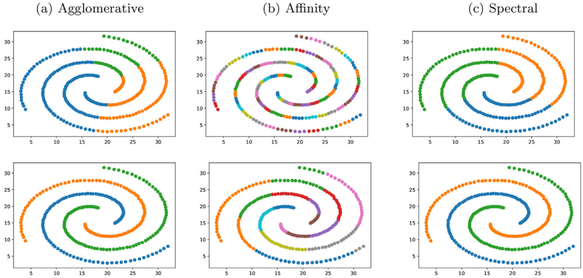
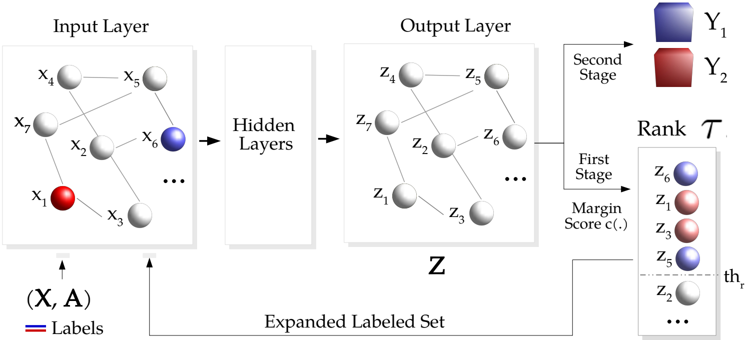


Fig. 12. Spiral dataset: original clustering results (first line) and associated with manifold learning (second line).

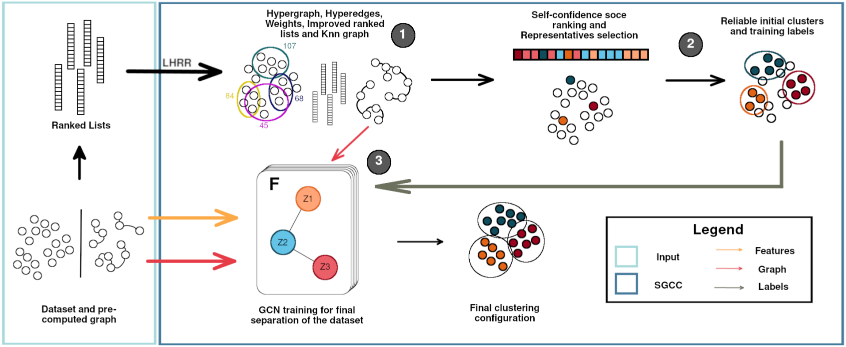
Applications in Machine Learning

- Rank-based Self-Training for Classification on Graph Convolutional Networks [67]



Applications in Machine Learning

- Self-Supervised Clustering based on Graph Convolutional Networks (Accepted on WACV 2023)



Applications in Machine Learning

- Impact of Self-Supervised Clustering on Data Distribution

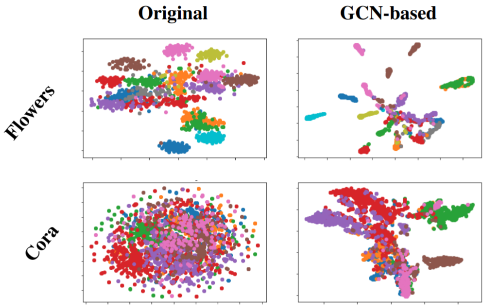


Figure 3. Visual analysis: the impact of the GCN embeddings.

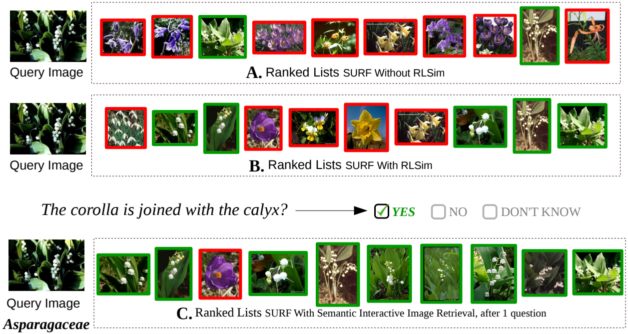
Applications in Other Domains

Different Tasks and Domains:

- Time Series Retrieval [2]
- Video Retrieval [3]
- Image Multimodal Retrieval [60]
- Multimedia Geocoding [35, 36, 33, 34]
- Classifier Selection for Remote Sensing [18]
- Image Segmentation [70]
- Speaker Recognition [9]
- Diagnostic Support in Medical Domain [46]

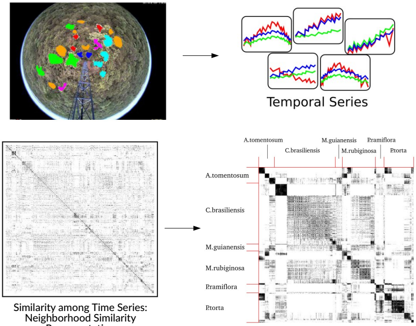
Applications in Other Domains

- Plant species identification based on semantic user interaction [20]



Applications in Other Domains

- Plant species identification based times series [2]



Similarity among Time Series: Neighborhood Similarity Representation

Contextual Similarity

Applications in Other Domains

- Person Re-Identification

Applications in Other Domains: Medical Image Retrieval

- Use of **Rank-based Contextual Similarity** for improving effectiveness of medical image retrieval
- Fusion of different features extracted from **brain MRI images**
- Ranked lists before and after contextual similarity learning:

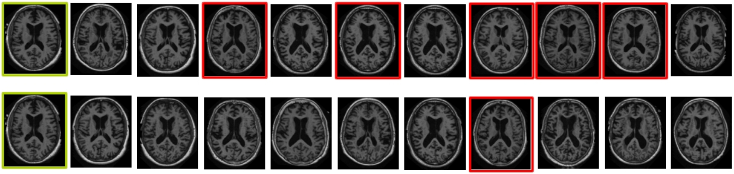


Figure: Diagnostic Support for Alzheimers Disease through Feature-Based Brain MRI Retrieval and Unsupervised Distance Learning [46]

Other Related Works

Other Related Tasks and Works:

- Rank-based Semi-Supervised Classifier [83]
- Graph Embedding [16]
- Feature Selecton [82]
- Re-Ranking for Regression [21]

- **Related References:**
  - Parallel Design of Unsupervised Methods [62, 64, 63, 19]
  - Unsupervised Effectiveness Estimation [59]
  - Other Re-Ranking and Rank Aggregation Algorithms [56, 65, 53, 50, 49, 51, 52, 54]

Questions?

Thank you for your attention!
Questions?

- Available at: `http://www.ic.unicamp.br/~dcarlos`

References I

[1] L. C. S. Afonso, D. C. G. Pedronette, A. N. de Souza, and J. P. Papa.
Improving optimum- path forest classification using unsupervised manifold learning.
In *24th International Conference on Pattern Recognition, ICPR*, pages 560–565, 2018.

[2] J. Almeida, D. C. G. Pedronette, B. C. Alberton, L. P. C. Morellato, and R. d. S. Torres.
Unsupervised distance learning for plant species identification.
IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, pages 1–14, 2016.

[3] J. Almeida, D. C. G. Pedronette, and O. A. B. Penatti.
Unsupervised manifold learning for video genre retrieval.
In *Iberoamerican Congress on Pattern Recognition (CIARP'2014)*, 2014.

[4] N. Arica and F. T. Y. Vural.
BAS: a perceptual shape descriptor based on the beam angle statistics.
Pattern Recognition Letters, 24(9-10):1627–1639, 2003.

[5] X. Bai, B. Wang, X. Wang, W. Liu, and Z. Tu.
Co-transduction for shape retrieval.
In *ECCV*, volume 3, pages 328–341, 2010.

[6] S. Belongie, J. Malik, and J. Puzicha.
Shape matching and object recognition using shape contexts.
PAMI, 24(4):509–522, 2002.

References II

[7] F. A. Breve and D. C. G. Pedronette.
Combined unsupervised and semi-supervised learning for data classification.
In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2016.

[8] P. Brodatz.
Textures: A Photographic Album for Artists and Designers.
Dover, 1966.

[9] V. d. A. Campos and D. C. G. a. Pedronette.
Effective speaker retrieval and recognition through vector quantization and unsupervised distance learning.
In *International Workshop on Multimedia Analysis and Retrieval for Multimodal Interaction*, MARMI '16, pages 27–32, 2016.

[10] D. C. G. P. Cesar Yugo Okada and R. d. S. Torres.
Unsupervised distance learning by rank correlation measures for image retrieval.
ICMR, 2015.

[11] S. A. Chatzichristofis and Y. S. Boutalis.
Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval.
In *ICVS*, pages 312–322, 2008.

[12] S. A. Chatzichristofis and Y. S. Boutalis.
Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval.
In *WIAMIS*, pages 191–196, 2008.

References III

[13] R. da S. Torres and A. X. Falcão.
Content-Based Image Retrieval: Theory and Applications.
Revista de Informática Teórica e Aplicada, 13(2):161–185, 2006.

[14] R. da S. Torres and A. X. Falcão.
Contour Salience Descriptors for Effective Image Retrieval and Analysis.
Image and Vision Computing, 25(1):3–13, 2007.

[15] R. Datta, D. Joshi, J. Li, and J. Z. Wang.
Image retrieval: Ideas, influences, and trends of the new age.
ACM Computing Surveys, 40(2):5:1–5:60, 2008.

[16] F. A. de Fernando, D. C. G. Pedronette, G. J. de Sousa, L. P. Valem, and I. R. Guilherme.
Rade: A rank-based graph embedding approach.
In *VISAPP 2020*, pages 142–152, 2020.

[17] R. Fagin, R. Kumar, and D. Sivakumar.
Comparing top k lists.
In *ACM-SIAM Symposium on Discrete algorithms (SODA'03)*, pages 28–36, 2003.

[18] F. Faria, D. Pedronette, J. dos Santos, A. Rocha, and R. Torres.
Rank aggregation for pattern classifier selection in remote sensing images.
Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of, 7(4):1103–1115, April 2014.

References IV

[19] D. C. G. P. Flávia Pisani, R. da S. Torres, and E. Borin.
Contextual spaces algorithm acceleration on APUs.
In *Escola Regional de Alto Desempenho - São Paulo (ERAD'2013)*, 2013.

[20] F. M. F. Gonçalves, I. R. Guilherme, and D. C. G. Pedronette.
Semantic guided interactive image retrieval for plant identification.
Expert Syst. Appl., 91:12–26, 2018.

[21] F. M. F. Gonçalves, D. C. G. Pedronette, and R. da Silva Torres.
Regression by re-ranking.
Pattern Recognition, 140:109577, 2023.

[22] R. Gopalan, P. Turaga, and R. Chellappa.
Articulation-invariant representation of non-planar shapes.
In *ECCV*, volume 3, pages 286–299, 2010.

[23] D. C. Guimarães Pedronette, J. Almeida, and R. Da S. Torres.
A scalable re-ranking method for content-based image retrieval.
Information Sciences, 265:91–104, May 2014.

[24] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih.
Image indexing using color correlograms.
In *CVPR*, pages 762–768, 1997.

[25] A. Iscen, G. Tolias, Y. S. Avrithis, and O. Chum.
Mining on Manifolds: Metric learning without labels.
In *IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR2018)*, 2018.

References V

[26] H. Jegou, M. Douze, and C. Schmid.
Hamming embedding and weak geometric consistency for large scale image search.
In *European Conference on Computer Vision*, ECCV '08, pages 304–317, 2008.

[27] H. Jegou, C. Schmid, H. Harzallah, and J. Verbeek.
Accurate image search using the contextual dissimilarity measure.
PAMI, 32(1):2–11, 2010.

[28] J. Jiang, B. Wang, and Z. Tu.
Unsupervised metric learning by self-smoothing operator.
In *ICCV*, pages 794–801, 2011.

[29] P. Kontschieder, M. Donoser, and H. Bischof.
Beyond pairwise shape similarity analysis.
In *ACCV*, pages 655–666, 2009.

[30] V. Kovalev and S. Volmer.
Color co-occurence descriptors for querying-by-example.
In *ICMM*, page 32, 1998.

[31] L. J. Latecki, R. Lakmper, and U. Eckhardt.
Shape descriptors for non-rigid shapes with a single closed contour.
In *CVPR*, pages 424–429, 2000.

[32] B. Leibe and B. Schiele.
Analyzing appearance and contour based methods for object categorization.
In *CVPR*, volume 2, pages II–409–15 vol.2, 2003.

References VI

[33] L. T. Li, J. Almeida, D. C. G. Pedronette, O. A. B. Penatti, and R. da Silva Torres.
A multimodal approach for video geocoding.
In *MediaEval'2012*, 2012.

[34] L. T. Li, J. Almeida, R. T. C. D. C. G. Pedronette, O. A. B. Penatti, and R. da Silva Torres.
Multimodal image geocoding: The 2013 recod's approach.
In *MediaEval'2013*, 2013.

[35] L. T. Li, D. C. G. Pedronette, J. Almeida, O. A. B. Penatti, R. T. Calumby, and R. da Silva Torres.
Multimedia multimodal geocoding.
In *SIGSPATIAL/GIS'2012*, pages 474–477, 2012.

[36] L. T. Li, D. C. G. Pedronette, J. Almeida, O. A. B. Penatti, R. T. Calumby, and R. da Silva Torres.
A rank aggregation framework for video multimodal geocoding.
Multimedia Tools and Applications, 2013.
On-Line.

[37] X. Li, M. Larson, and A. Hanjalic.
Pairwise geometric matching for large-scale object retrieval.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2015)*, pages 5153–5161, June 2015.

[38] H. Ling and D. W. Jacobs.
Shape classification using the inner-distance.
PAMI, 29(2):286–299, 2007.

References VII

[39] H. Ling, X. Yang, and L. J. Latecki.
Balancing deformability and discriminability for shape matching.
In *ECCV*, volume 3, pages 411–424, 2010.

[40] Z. Liu, S. Wang, L. Zheng, and Q. Tian.
Robust imagegraph: Rank-level feature fusion for image search.
IEEE Transactions on Image Processing, 26(7):3128–3141, 2017.

[41] D. Lowe.
Object recognition from local scale-invariant features.
In *ICCV*, pages 1150–1157, 1999.

[42] H. Lu, B. Ooi, and K. Tan.
Efficient image retrieval by color contents.
In *ADB*, pages 95–108, 1994.

[43] B. Manjunath, J.-R. Ohm, V. Vasudevan, and A. Yamada.
Color and texture descriptors.
IEEE Transactions on Circuits and Systems for Video Technology, 11(6):703–715, 2001.

[44] D. Nistér and H. Stewénius.
Scalable recognition with a vocabulary tree.
In *CVPR*, volume 2, pages 2161–2168, 2006.

[45] T. Ojala, M. Pietikäinen, and T. Mäenpää.
Multiresolution gray-scale and rotation invariant texture classification with local binary patterns.
PAMI, 24(7):971–987, 2002.

References VIII

[46] B. T. Padovese, D. H. P. Salvadeo, and D. C. G. Pedronette.
Diagnostic support for alzheimers disease through feature-based brain MRI retrieval and unsupervised distance learning.
In *16th IEEE International Conference on Bioinformatics and Bioengineering, BIBE 2016, Taichung, Taiwan, October 31 - November 2, 2016*, pages 242–249. IEEE Computer Society, 2016.

[47] G. Pass, R. Zabih, and J. Miller.
Comparing images using color coherence vectors.
In *ACM-MM*, pages 65–73, 1996.

[48] M. Paulin, J. Mairal, M. Douze, Z. Harchaoui, F. Perronnin, and C. Schmid.
Convolutional patch representations for image retrieval: An unsupervised approach.
Int. Journal of Computer Vision, 2017.

[49] D. C. G. Pedronette and R. da S. Torres.
Distances correlation for re-ranking in content-based image retrieval.
In *Conference on Graphics, Patterns and Images (SIBGRAPI'2010)*, volume 1, pages 1 – 8, 2010.

[50] D. C. G. Pedronette and R. da S. Torres.
Exploiting contextual information for image re-ranking.
In *Iberoamerican Congress on Pattern Recognition (CIARP'2010)*, pages 541–548, 2010.

[51] D. C. G. Pedronette and R. da S. Torres.
Shape retrieval using contour features and distance optmization.
In *VISAPP*, volume 1, pages 197 – 202, 2010.

References IX

[52] D. C. G. Pedronette and R. da S. Torres.
Exploiting clustering approaches for image re-ranking.
Journal of Visual Languages and Computing, 22(6):453–466, 2011.

[53] D. C. G. Pedronette and R. da S. Torres.
Exploiting contextual information for rank aggregation.
In *International Conference on Image Processing (ICIP'2011)*, pages 97–100, 2011.

[54] D. C. G. Pedronette and R. da S. Torres.
Exploiting contextual spaces for image re-ranking and rank aggregation.
In *ACM International Conference on Multimedia Retrieval (ICMR'11)*, pages 13:1–13:8, 2011.

[55] D. C. G. Pedronette and R. da S. Torres.
Combining re-ranking and rank aggregation methods.
In *Iberoamerican Congress on Pattern Recognition (CIARP'2012)*, 2012.

[56] D. C. G. Pedronette and R. da S. Torres.
Exploiting contextual information for image re-ranking and rank aggregation.
International Journal of Multimedia Information Retrieval, 1(2):115–128, 2012.

[57] D. C. G. Pedronette and R. da S. Torres.
Exploiting pairwise recommendation and clustering strategies for image re-ranking.
Information Sciences, 207:19–34, 2012.

[58] D. C. G. Pedronette and R. da S. Torres.
Image re-ranking and rank aggregation based on similarity of ranked lists.
Pattern Recognition, 46(8):2350–2360, 2013.

References X

[59] D. C. G. Pedronette and R. da S. Torres.
Unsupervised measures for estimating the effectiveness of image retrieval systems.
In *Conference on Graphics, Patterns and Images (SIBGRAPI'2013)*, 2013.

[60] D. C. G. Pedronette and R. da S. Torres.
Unsupervised distance learning by reciprocal knn distance for image retrieval.
In *International Conference on Multimedia Retrieval (ICMR'14)*, 2014.

[61] D. C. G. Pedronette and R. da S. Torres.
Unsupervised manifold learning by correlation graph and strongly connected components for image retrieval.
In *International Conference on Image Processing (ICIP'2014)*, 2014.

[62] D. C. G. Pedronette, R. da S. Torres, E. Borin, and M. Breternitz.
Efficient image re-ranking computation on GPUs.
In *ISPA*, 2012.

[63] D. C. G. Pedronette, R. da S. Torres, E. Borin, and M. Breternitz.
Efficient image re-ranking computation using GPUs.
In *Escola Regional de Alto Desempenho - São Paulo (ERAD'2012)*, 2012.

[64] D. C. G. Pedronette, R. da S. Torres, E. Borin, and M. Breternitz.
Rl-sim algorithm acceleration on GPUs.
In *SBAC*, 2013.

References XI

[65] D. C. G. Pedronette, R. da S. Torres, and R. C. Tripodi.
Using contextual spaces for image re-ranking and rank aggregation.
Multimedia Tools and Applications.
Accepted.

[66] D. C. G. Pedronette, F. M. F. Gonçalves, and I. R. Guilherme.
Unsupervised manifold learning through reciprocal kNN graph and Connected Components for image retrieval tasks.
Pattern Recognition, 75:161 – 174, 2018.

[67] D. C. G. Pedronette and L. J. Latecki.
Rank-based self-training for graph convolutional networks.
Inf. Process. Manag., 58(2):102443, 2021.

[68] D. C. G. Pedronette, O. A. Penatti, and R. da S. Torres.
Unsupervised manifold learning using reciprocal knn graphs in image re-ranking and rank aggregation tasks.
Image and Vision Computing, 32(2):120 – 130, 2014.

[69] D. C. G. Pedronette, Y. Weng, A. Baldassin, and C. Hou.
Semi-supervised and active learning through manifold reciprocal knn graph for image retrieval.
Neurocomputing, 340:19–31, 2019.

[70] T. Pinto, M. de Carvalho, D. Pedronette, and P. Martins.
Image segmentation through combined methods: Watershed transform, unsupervised distance learning and normalized cut.
In *Image Analysis and Interpretation (SSIAI), 2014 IEEE Southwest Symposium on*, pages 153–156, April 2014.

References XII

[71] J. G. C. Presotto, S. F. dos Santos, L. P. Valem, F. A. Faria, J. P. Papa, J. Almeida, and D. C. G. Pedronette.
Weakly supervised learning based on hypergraph manifold ranking.
Journal of Visual Communication and Image Representation, 89:103666, 2022.

[72] J. G. C. Presotto, L. P. Valem, N. G. de Sá, D. C. G. Pedronette, and J. P. Papa.
Weakly supervised learning through rank-based contextual measures.
In *25th International Conference on Pattern Recognition, ICPR*, pages 5752–5759. IEEE, 2020.

[73] D. Qin, C. Wengert, and L. V. Gool.
Query adaptive similarity for large scale object retrieval.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2013)*, pages 1610–1617, June 2013.

[74] B. Rozin, V. H. Pereira-Ferrero, L. T. Lopes, and D. C. G. Pedronette.
A rank-based framework through manifold learning for improved clustering tasks.
Inf. Sci., 580:202–220, 2021.

[75] R. O. Stehling, M. A. Nascimento, and A. X. Falcão.
A compact and efficient image retrieval approach based on border/interior pixel classification.
In *CIKM*, pages 102–109, 2002.

[76] S. Sun, Y. Li, W. Zhou, Q. Tian, and H. Li.
Local residual similarity for image re-ranking.
Information Sciences, 417(Sup. C):143 – 153, 2017.

References XIII

[77] M. J. Swain and D. H. Ballard.
Color indexing.
International Journal on Computer Vision, 7(1):11–32, 1991.

[78] B. Tao and B. W. Dickinson.
Texture recognition and image retrieval using gradient indexing.
JVCIR, 11(3):327–342, 2000.

[79] R. Tarjan.
Depth first search and linear graph algorithms.
SIAM Journal on Computing, 1972.

[80] G. Tolias, Y. Avrithis, and H. Jégou.
To aggregate or not to aggregate: Selective match kernels for image search.
In *IEEE International Conference on Computer Vision (ICCV'2013)*, pages 1401–1408, Dec 2013.

[81] Z. Tu and A. L. Yuille.
Shape matching and recognition - using generative models and informative features.
In *ECCV*, pages 195–209, 2004.

[82] L. P. Valem and D. C. G. Pedronette.
Graph-based selective rank fusion for unsupervised image retrieval.
Pattern Recognit. Lett., 135:82–89, 2020.

References XIV

[83] L. P. Valem, D. C. G. Pedronette, F. A. Breve, and I. R. Guilherme.
Manifold correlation graph for semi-supervised learning.
In *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*, pages 1–7, 2018.

[84] L. P. Valem, D. C. G. Pedronette, R. d. S. Torres, E. Borin, and J. Almeida.
Effective, efficient, and scalable unsupervised distance learning in image retrieval tasks.
ICMR, 2015.

[85] J. van de Weijer and C. Schmid.
Coloring local feature extraction.
In *ECCV*.

[86] B. Wang, J. Jiang, W. Wang, Z.-H. Zhou, and Z. Tu.
Unsupervised metric fusion by cross diffusion.
In *CVPR*, pages 3013 –3020, 2012.

[87] J. Wang, Y. Li, X. Bai, Y. Zhang, C. Wang, and N. Tang.
Learning context-sensitive similarity by shortest path propagation.
Pattern Recognition, 44(10-11):2367–2374, 2011.

[88] X. Yang, X. Bai, L. J. Latecki, and Z. Tu.
Improving shape retrieval by learning graph transduction.
In *ECCV*, volume 4, pages 788–801, 2008.

References XV

[89] X. Yang, S. Koknar-Tezel, and L. J. Latecki.
Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval.
In *CVPR*, pages 357–364, 2009.

[90] X. Yang, L. Prasad, and L. Latecki.
Affinity learning with diffusion on tensor product graph.
IEEE Transactions on Pattern Analysis and Machine Intelligence,, 35(1):28–38, 2013.

[91] K. Zagoris, S. Chatzichristofis, N. Papamarkos, and Y. Boutalis.
Automatic image annotation and retrieval using the joint composite descriptor.
In *PCI*, pages 143–147, 2010.

[92] S. Zhang, M. Yang, T. Cour, K. Yu, and D. Metaxas.
Query specific rank fusion for image retrieval.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(4):803–815, April 2015.

[93] L. Zheng, S. Wang, Z. Liu, and Q. Tian.
Packing and padding: Coupled multi-index for accurate image retrieval.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2014)*, pages 1947–1954, June 2014.

[94] L. Zheng, S. Wang, and Q. Tian.
Coupled binary embedding for large-scale image retrieval.
IEEE Transactions on Image Processing (TIP), 23(8):3368–3380, 2014.