Distances Correlation for Re-Ranking in Content-Based Image Retrieval

Daniel Carlos Guimarães Pedronette and Ricardo da S. Torres

RECOD Lab - Institute of Computing (IC)
University of Campinas (UNICAMP)
dcarlos@ic.unicamp.br, rtorres@ic.unicamp.br
Outline

• Distance Optimization Algorithm
  – The Algorithm
  – Clustering Approach

• Distances Correlation
  – Bidimensional Space
  – Distances Updating

• Evaluation
  – Shape, Color, Texture

• Conclusions
Distance Optimization Algorithm

• Basic Idea: Similarity of Ranked lists
  – If two images are similar, their ranked lists should be similar too.

• Distance Optimization Algorithm
  – Create clusters:
    • by exploring information of ranked lists
  – Update distances:
    • distances among images of a same cluster are decreased
Distance Optimization Algorithm

• Convergence
  – Process (make clusters and update distances) is repeated until the quality of clusters does not improve.

• Cohesion for measuring quality of ranked lists
  – Quantity of references among ranked lists of images on the same cluster (references in first positions of ranked lists have greater weights)
Distance Optimization Algorithm

Algorithm 1 Distance Optimization Algorithm [7]

Require: Distance matrix $W$
Ensure: Optimized distance matrix $W_o$

1: $lastCohesion \leftarrow 0$
2: $currentCohesion \leftarrow computeCohesion(W)$
3: while $curCohesion > lastCohesion$ do
4: $Cls \leftarrow createClusters(W)$
5: $W \leftarrow updateDistances(W,Cls)$
6: $lastCohesion \leftarrow currentCohesion$
7: $currentCohesion \leftarrow computeCohesion(W)$
8: end while
9: $W_o \leftarrow W$
Clustering Approach

- Graph-based clustering using ranked lists
  - Two images are assigned to the same cluster if they are cluster-similar
  - Basically, two images are cluster-similar if they refer to each other at the first positions of their ranked lists
Distances Updating

• Considering only clusters information:
  – If two images were assigned to the same cluster, the distance between them is decreased
    • Multiplied by a constant \( \lambda < 1 \)

• Ignoring other information encoded in the relations among images

• A new approach to update distances in an adaptative way
  – Distances Correlation
Distances Correlation

- **Bidimensional Space**
  - *Image space* $R^2$ defined by the image collection $C = \{\text{img}_1, \text{img}_2, \ldots, \text{img}_n\}$ and a distance function $\rho : C \times C \rightarrow R$, where $R$ denotes real numbers.
  - Given two reference images $\text{img}_i$ and $\text{img}_j$:
    - X axis represents the distances of collection images with regard to $\text{img}_i$.
    - Y axis represents the distances of collection images with regard to $\text{img}_j$. 
Distances Correlation

Bidimensional Representation - Dissimilar Images

Image i
apple-1.gif

Image j
fork-5.gif
Distances Correlation

• Statistical measures to characterize the images distribution:
  – Magnitude of a relationship among variables
  – Pearson’s Correlation Coefficient:

\[
r = \frac{\sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \overline{X})^2} \sqrt{\sum_{i=1}^{n} (Y_i - \overline{Y})^2}}
\]

  – KNNs of img\(_i\) and img\(_j\) for composition of X and Y
Distances Correlation

– Pearson’s Correlation Coefficient:
  • $r$ in the interval $[-1,1]$;
  • $\overline{r}$ in the interval $[0,1]$;
  1: Perfect correlation

– Central Idea:
  • Using cluster and correlation information for distances updating
Distances Updating

New Distances Computation: \( \hat{\rho}(i, j) \)

- \( Seg_1: \hat{\rho}(i, j) = \rho(i, j) \cdot \lambda \)
- \( Seg_2: \hat{\rho}(i, j) = \rho(i, j) \cdot (1 + [(1 - \lambda) \cdot (1 - \tau)]) \)
- \( Seg_3: \hat{\rho}(i, j) = \rho(i, j) \cdot [1 + (1 - \lambda)] \)
Experimental Analysis

• Impact of algorithm on Distances
  – (considering non-similar reference images)
Evaluation

• Shape

  – Shape Descriptors:
    • CFD[7], IDSC[14], BAS[16], SS[15]
  – MPEG-7 Dataset (70 shapes, 20 each class)

  – First row: retrieval results for the CFD Shape Descriptor[7] (first image as a query).
  – Second row: retrieval results for the same shape descriptor after distance optimization.
Evaluation

• Color
  – Color Descriptors:
    • ACC[17], BIC[18],
  – Soccer Dataset [22]
    • 7 soccer teams, containing 40 images per class
Evaluation

• Texture
  – Texture Descriptors
    • CCOM[19], LAS[20]
  – Brodatz Dataset [21]
    • 111 different texture classes
Experimental Results

- Example of Precision x Recall for Texture Descriptors:
Experimental Results

- General CBIR tasks:

<table>
<thead>
<tr>
<th>Image Descriptor</th>
<th>Type</th>
<th>Dataset</th>
<th>Score [%] (MAP)</th>
<th>Distance Optimization + Update Correlation</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS [15]</td>
<td>Shape Descriptor</td>
<td>MPEG-7</td>
<td>37.67%</td>
<td>46.53%</td>
<td>+23.52%</td>
</tr>
<tr>
<td>BAS [16]</td>
<td>Shape Descriptor</td>
<td>MPEG-7</td>
<td>71.52%</td>
<td>81.05%</td>
<td>+13.32%</td>
</tr>
<tr>
<td>IDSC+DP [14]</td>
<td>Shape Descriptor</td>
<td>MPEG-7</td>
<td>81.70%</td>
<td>86.94%</td>
<td>+6.41%</td>
</tr>
<tr>
<td>CFD [7]</td>
<td>Shape Descriptor</td>
<td>MPEG-7</td>
<td>80.71%</td>
<td>91.79%</td>
<td>+13.73%</td>
</tr>
<tr>
<td>ACC [17]</td>
<td>Color Descriptor</td>
<td>Soccer Dataset</td>
<td>37.23%</td>
<td>42.46%</td>
<td>+14.05%</td>
</tr>
<tr>
<td>BIC [18]</td>
<td>Color Descriptor</td>
<td>Soccer Dataset</td>
<td>39.26%</td>
<td>38.16%</td>
<td>-2.80%</td>
</tr>
<tr>
<td>CCOM [19]</td>
<td>Texture Descriptor</td>
<td>Brodatz</td>
<td>57.57%</td>
<td>59.27%</td>
<td>+2.95%</td>
</tr>
<tr>
<td>LAS [20]</td>
<td>Texture Descriptor</td>
<td>Brodatz</td>
<td>75.15%</td>
<td>80.36%</td>
<td>+6.93%</td>
</tr>
</tbody>
</table>
Experimental Results

- Post-processing methods comparison on MPEG-7 (Recall@40)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Descriptor</th>
<th>Score</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFD [7]</td>
<td>-</td>
<td>84.43%</td>
<td>-</td>
</tr>
<tr>
<td>IDSC+DP [14]</td>
<td>-</td>
<td>85.40%</td>
<td>-</td>
</tr>
<tr>
<td>Graph Transduction [6]</td>
<td>IDSC+DP</td>
<td>91.00%</td>
<td>+6.56%</td>
</tr>
<tr>
<td>Distance Optimization [7]</td>
<td>CFD</td>
<td>92.56%</td>
<td>+9.63%</td>
</tr>
<tr>
<td>Constrained Diffusion Process [5]</td>
<td>IDSC+DP</td>
<td>93.32%</td>
<td>+9.27%</td>
</tr>
<tr>
<td>Mutual kNN Graph [4]</td>
<td>IDSC+DP</td>
<td>93.40%</td>
<td>+9.37%</td>
</tr>
<tr>
<td>DistOpt+UpCor</td>
<td>CFD</td>
<td>93.62%</td>
<td>+10.88%</td>
</tr>
</tbody>
</table>
Conclusions

• New concept of ‘Distances Correlation’
• New approach for a Re-Ranking method using this concept
• Experimental Evaluation
  – Shape, Color, Texture
  – Comparison to other post-processing methods
• Future Work
  – Application of method to other information retrieval tasks
Distances Correlation for Re-Ranking in Content-Based Image Retrieval

Daniel Carlos Guimarães Pedronette and Ricardo da S. Torres

RECOD Lab - Institute of Computing (IC)
University of Campinas (UNICAMP)
dcarlos@ic.unicamp.br, rtorres@ic.unicamp.br
References


Cohesion Computation

- Let $C = \{\text{img}_1, \text{img}_2, \ldots, \text{img}_n\}$ be a collection (or a cluster) of images, cohesion is defined as follows:

$$\text{cohesion}(C) = \frac{\sum_{j=0}^{\text{size}} \sum_{i=0}^{\text{top}_n} (\text{top}_n - i) \times (\text{top}_n / \text{size}) \times S(i)}{\text{size}^2}$$

where $S$ is a function $S: i \rightarrow \{0, 1\}$, that assumes value 1 if $C$ contains the image at position $i$ of ranked list and assumes value 0, otherwise.