LEARNING FINE-GRAINED IMAGE SIMILARITY WITH DEEP RANKING

Jiang Wang (Northwestern)  Yang Song (Google)  Thomas Leung (Google)  Chuck Rosenberg (Google)  Jingbin Wang
James Philbin (Google)  Bo Chen (Caltech)  Ying Wu (Northwestern)

PROBLEM
Fine-grained image similarity, for images with the same category. It is for image-search application, defined by triplets.

- image similarities are defined subtle difference.
- it is more difficult to obtain triplet training data.
- we would like to train a model directly from images instead of rely on the hand-crafted features.

RELATED WORK
- category-level image similarity: the similarities are purely defined by labels.
- classification deep learning models.
- pairwise ranking model.

FORMULATION
The similarity of two images $P$ and $Q$ can be defined according to their squared Euclidean distance in the image embedding space:

$$D(f(P), f(Q)) = \|f(P) - f(Q)\|^2$$  
(1)

Triplet-based Objective: $r_{i,j} = r(p_i, p_j)$ is pairwise relevance score.

$$D(f(p_i), f(p_j^+)) < D(f(p_i), f(p_j^-)), \forall p_i, p_j^+, p_j^- \text{ such that } r(p_i, p_j^+) > r(p_i, p_j^-)$$  
(2)

Triplet $t_i = (p_i, p_i^+, p_i^-)$ a triplet. The hinge loss is:

$$l(p_i, p_i^+, p_i^-) = \max(0, D(f(p_i), f(p_i^+)) - D(f(p_i), f(p_i^-)))$$  
(3)

ARCHITECTURE
- a novel deep learning that can learn fine-grained image similarity model directly from images.
- a multi-scale network structure.
- a computationally efficient online triplet sampling algorithm.
- high quality triplet evaluation dataset.

OBTIMIZATION
- Asynchronized stochastic gradient algorithm.
- Momentum algorithm.
- Dropout to avoid overfitting

Challenges:
- Cannot enumerate all the triplets, need to sample important triplets.
- Cannot load all the images into memory, need to generate triplets online.

TRIPLET SAMPLING
Sampling criteria: we sample more highly relevant images.

Total relevance score $r_i$:

$$r_i = \sum_{j \in c_i \cap \{j|j \neq i\}} r_{i,j}$$  
(4)

- For query image: according to total relevance score.
- For positive image: sample images with the same label as the query image, sampling probability is $P(p_i^+) = \frac{\min(T, r_{i,i}^+)}{z_i}$. $T_i$.
- For negative image, we have two types of samples:
  1. in-class negative: we draw in-class negative samples $p_i^-$ with the same distribution as the positive image. We also require that the margin between the relevance score $r_{i,i}$ and $r_{i,i-}$ should be larger than $T_i$.
  2. out-of-class negative: drawn uniformly from all the images in different categories.

MULTI-SCALE ARCHITECTURE

DATA
High quality image triplet evaluation dataset:
Available at https://sites.google.com/siteimagesimilaritydata/

ACKNOWLEDGMENT
The work was done when the first author is working as an intern at Google.

EXPERIMENTS

Comparison with hand-crafted features:

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Score @30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wavelet</td>
<td>62.2%</td>
<td>2735</td>
</tr>
<tr>
<td>Color</td>
<td>62.3%</td>
<td>2935</td>
</tr>
<tr>
<td>SIFT-like</td>
<td>65.5%</td>
<td>2863</td>
</tr>
<tr>
<td>Fisher</td>
<td>67.2%</td>
<td>3064</td>
</tr>
<tr>
<td>HOG</td>
<td>68.4%</td>
<td>3099</td>
</tr>
<tr>
<td>SPMKtexton1024max</td>
<td>66.5%</td>
<td>3556</td>
</tr>
<tr>
<td>L1HashKPCA</td>
<td>76.2%</td>
<td>6156</td>
</tr>
<tr>
<td>Golden Features</td>
<td>80.3%</td>
<td>7165</td>
</tr>
<tr>
<td>DeepRanking</td>
<td>85.7%</td>
<td>7004</td>
</tr>
</tbody>
</table>

Comparison of different architectures:

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Score @30</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvNet</td>
<td>82.8%</td>
<td>5772</td>
</tr>
<tr>
<td>Single-scale Ranking</td>
<td>84.6%</td>
<td>6245</td>
</tr>
<tr>
<td>OASIS on Single-scale Ranking</td>
<td>82.5%</td>
<td>6263</td>
</tr>
<tr>
<td>Single Scale &amp; visual Feature</td>
<td>84.1%</td>
<td>6766</td>
</tr>
<tr>
<td>DeepRanking</td>
<td>85.7%</td>
<td>7004</td>
</tr>
</tbody>
</table>

Comparison of different sampling methods:

TRAINING DATA
- ImageNet for pre-training, category-level information.
- Relevance training data. Fine-grained visual information.
  - Golden Feature, good for visual similarity but not so good for semantic similarity, and it is expensive to compute,
  - Buffer for query: first buffer of the query
  - Buffer for positive
  - Buffer for negative
  - Positive
  - Negative

RANKING EXAMPLES