Discriminative Learning of Deep Convolutional Feature Point Descriptors

Edgar Simo-Serra*,1,5, Eduard Trulls*,2,5, Luis Ferraz2, Iasonas Kokkinos3, Pascal Fua2, Francesc Moreno-Noguer5
1 Waseda University, Tokyo, Japan  2 CVLab, Ecole Polytechnique Fédérale de Lausanne, Switzerland  3 Catchoom Technologies, Barcelona, Spain  4 CentraleSupelec and INRIA-Saclay, Chatenay-Malabry, France  5 Institut de Robótica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain  (*: First two authors contributed equally)

Objective
- Learn compact, discriminative representations of image patches with Convolutional Neural Networks.
- Optimize for comparisons with the L2 distance, i.e. no metric learning. Our descriptors work within existing pipelines.

Main features
- Drop-in replacement for SIFT: 128f, compare with the L2 norm.
- Consistent improvements over the state of the art.
- Trained in one dataset, but generalizes very well to scaling, rotation, deformation and illumination changes.
- Computational efficiency (on GPU: 0.76 ms; dense SIFT: 0.14 ms).

Code available: https://github.com/etrulls/deepdesc-release

Key observation
1. We train a Siamese architecture with pairs of patches. We want to bring matching pairs together and otherwise pull them apart.
2. Problem? Randomly sampled pairs are already easy to separate.
3. Solution: To train discriminative networks we use hard negative and positive mining. This proves essential for performance.

Model & Training

<table>
<thead>
<tr>
<th>Model</th>
<th>Training</th>
<th>Def.</th>
<th>Ill.</th>
<th>Def.+Ill.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>LY+YO</td>
<td>76.591</td>
<td>89.895</td>
<td>88.434</td>
</tr>
<tr>
<td>VGG</td>
<td>0.969</td>
<td>0.946</td>
<td>0.516</td>
<td>0.410</td>
</tr>
<tr>
<td>BinBoost-256</td>
<td>0.594</td>
<td>0.363</td>
<td>0.172</td>
<td>0.032</td>
</tr>
</tbody>
</table>

We minimize the hinge embedding loss. With 3D point indices $p_1, p_2$:
$$
\ell(x_1, x_2) = \begin{cases} 
\| D(x_1) - D(x_2) \|_2, & |p_1 - p_2| > 0.5 \\
0, & |p_1 - p_2| \leq 0.5 
\end{cases}
$$

This penalizes corresponding pairs that are placed far apart, and non-corresponding pairs that are less than $C$ units apart.

Methodology: Train over two sets and test over third ($\text{LuRe}$, $\text{INRIA}$, $\text{Y}$.), with cross-validation. Metric: precision-recall (PR). ‘Needle in a haystack’ setting: pick 10k unique points and generate one positive pair and 1k negative pairs for each, i.e. 10k positives vs. 10M negatives. Results summarized by ‘Area Under the Curve’ (AUC).

Effect of mining
(a) Forward-propagate positives $s_p \geq 128$ and negatives $s_n \geq 128$.
(b) Pick the 128 with the largest loss (for each) and back-propagate.

We take samples from [1], for illustration. Corresponding patches are shown with same color:
(a) Representation from t-SNE [3]. Distance encodes similarity.
(b) Random sampling: similar (close) positives and different (distant) negatives.
(c) We mine the samples to obtain dissimilar positives (+, long blue segments) and similar negatives (−, short red segments):
(d) Random sampling results in easy pairs.
(e) Mixed pairs with hard correspondences.

This allows us to train discriminative models with a small number of parameters (~45k), which also alleviates overfitting concerns.

References