Learning to Detect Scene Landmarks for Camera Localization

Tien Do\textsuperscript{1} \quad Ondrej Miksik\textsuperscript{2} \quad Joseph DeGol\textsuperscript{2} \quad Hyun Soo Park\textsuperscript{1} \quad Sudipta N. Sinha\textsuperscript{2}

\textsuperscript{1}University of Minnesota \quad \textsuperscript{2}Microsoft

CVPR 2022
Camera localization problem

Given a query image, compute the 3D position and 3D orientation of the camera within a precomputed 3D map of the scene.
Related work

• Vast literature

• Retrieval-based methods
  • Hierarchical Localization (Hloc)
    Learning Feature Matching with Graph Neural Networks [Sarlin et al. 2020]

• Learned methods
  • Absolute pose regression (APR)
    PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization [Kendall et al. 2015]

  • Dense scene coordinate regression (SCR)
    Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images [Shotton et al. 2013]

    Visual Camera Re-Localization from RGB and RGB-D Images Using DSAC [Brachmann and Rother 2021]
Retrieval-based methods

- Accurate
- High storage requirements
- Not privacy preserving
  - Image can be reconstructed from stored feature descriptors

Revealing scenes by inverting structure from motion reconstructions. [Pittaluga et al. 2019]
Learned methods (low storage)

Abs. Pose Regression (PoseNet)

- Query Image
  - CNN
  - $\mathbf{R}, \mathbf{t}$

Scene Coordinate Regression (DSAC*)

- Query Image
  - CNN
  - Dense scene coordinates
  - Pose solver
  - $\mathbf{R}, \mathbf{t}$
Main Idea

• Designate a few scene landmarks (3D points).
• Learn a detector to localize those scene landmarks in a query image.
• Estimate camera pose from the 2D-3D scene landmark correspondences.
Learned methods (low storage)

Abs. Pose Regression (PoseNet)

```
Query Image
  ↓
  CNN
  ↓
(R, t)
```

Scene Coordinate Regression (DSAC*, ...)

```
Query Image
  ↓
  CNN
  ↓
Dense scene coordinates
  ↓
Pose solver
  ↓
(R, t)
```

Scene Landmark Detection (ours)

```
Query Image
  ↓
  CNN
  ↓
Sparse 2D landmark detection
  ↓
Pose solver
  ↓
(R, t)
```

Scene landmarks (3D points)

- $X_1 Y_1 Z_1$
- $X_2 Y_2 Z_2$
- $X_3 Y_3 Z_3$
Learned methods (low storage)

**APR** (PoseNet)
- **Query Image**
- **CNN**
- **(R, t)**

**SCR (DSAC*)**
- **Query Image**
- **CNN**
- **Dense scene coordinates**
- **Pose solver**
- **(R, t)**

**3D Scene landmarks**
- **Sparse 2D landmark detection**
- **Pose solver**
- **(R, t)**
Scene Landmark Detector (SLD) Model

- Output heatmap for each landmark
- Dilated convolution architecture
- Mean Sq. Error (MSE) pixel-wise loss
- Homography and intensity data augmentation
Example: training data
Neural Bearing Estimator (NBE)

- From image appearance, directly predict landmark bearing vector (3D)
- Can do it for visible as well as invisible landmarks
Indoor-6 Dataset

- Multiple captures (different day and time) of the same scene
- SfM reconstructions
Results

- NBE+SLD (ours) achieves the best performance among learned (low storage) methods.
- NBE+SLD(E) outperforms SOTA DSAC* using similar network capacity.
- NBE+SLD (ours) outperforms Hloc_{1000} that uses 3x more landmarks.

<table>
<thead>
<tr>
<th></th>
<th>Storage (MB)</th>
<th>scene1</th>
<th>scene2</th>
<th>scene3</th>
<th>scene4</th>
<th>scene5</th>
<th>scene6</th>
</tr>
</thead>
<tbody>
<tr>
<td>PoseNet</td>
<td>12</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>DSAC*</td>
<td>27</td>
<td>18.7</td>
<td>12.3</td>
<td>19.7</td>
<td>44.9</td>
<td>10.6</td>
<td>44.3</td>
</tr>
<tr>
<td>NBE+SLD(E)_{300}</td>
<td>29</td>
<td>28.4</td>
<td>26.1</td>
<td>43.5</td>
<td>48.9</td>
<td>37.5</td>
<td>44.6</td>
</tr>
<tr>
<td>NBE+SLD_{300}</td>
<td>132</td>
<td>38.4</td>
<td>37.0</td>
<td>53.0</td>
<td>62.5</td>
<td>40.0</td>
<td>50.5</td>
</tr>
</tbody>
</table>

Recall (%) @ (5cm, 5°)

- NBE+SLD (ours) achieves the best performance among learned (low storage) methods.
- NBE+SLD(E) outperforms SOTA DSAC* using similar network capacity.
- NBE+SLD (ours) outperforms Hloc_{1000} that uses 3x more landmarks.
Conclusion

• New learned camera localization method that predicts pre-determined scene landmarks in images.
• Leverages mature heatmap-based keypoint detection architectures.
• Low storage, privacy preserving, and high accuracy
• Code & Dataset: github.com/microsoft/SceneLandmarkLocalization
Learning to Detect Scene Landmarks for Camera Localization

Tien Do¹  Ondrej Miksik²  Joseph DeGol²  Hyun Soo Park¹  Sudipta N. Sinha²

¹University of Minnesota  ²Microsoft

CVPR 2022