PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization

Abstract

We present a robust and real-time monocular six degree of freedom relocalization system. Our system trains a convolutional neural network to regress the 6-DOF camera pose from a single RGB image in an end-to-end manner with no need of additional engineering or graph optimisation. The algorithm can operate indoors and outdoors in real-time, taking 5ms per frame to compute. It obtains approximately 2m and 3◦ accuracy for large scale outdoor scenes and 0.5m and 5◦ accuracy indoors. This is achieved using an efficient 23 layer deep convnet, demonstrating that convnets can be used to solve complicated out of image plane regression problems. This was made possible by leveraging transfer learning from large scale classification data. We show that the PoseNet localizes from high level features and is robust to difficult lighting, motion blur and different camera intrinsics where point based SIFT registration fails. Furthermore we show how the pose feature that is produced generalizes to other scenes allowing us to regress pose with only a few dozen training examples.

Now we can abstract what the research group did and managed to do

- A relocalization system for Camera with 6 degrees, and the Accuracy is 0.5m and 5 degrees indoors, 2m and 3 degrees for large scale outdoor scenes, and real-time.
- Uses Convolutional Neural Network, input is a single RGB image, output is unknown, maybe the pose information.
- Prooves that convnets can be used to solve complicated out of image plane regression problems. (For I just get into the area, I don't quite sure about what's its meaning)
- It was made possible for **leveraging transfer learning from large scale classification data**.
- It can be generalizes to other scenes with only a few dozen training examples.

Some words and phrases that I am not familiar

- leveraging transfer learning from large scale classification data
- SIFT registration fails
-

1. Introction

Background

In the traditional computer vision field, many researches are based on the 3D point cloud method Structure from Motion (SfM), where a large number of training photos are collected in the offline phase and a two-by-two matching is performed to construct a 3D point cloud. In the localization phase, the user's query photos are input into the system. The system registers the query photos to the point cloud and finally infers the camera location. Although this method can obtain relatively high accuracy, it is limited by the large computational effort and long matching time.

Our Contribution

- **Main Contribution**

The main contribution is the deep CNN camera pose regressor.

2 novel techniques to achieve this:

I. Leverage transfer learning from recognition to relocalization **with very large scale classification datasets**

II. Structure from motion to automatically generate training labels (camera poses) **from a video of the scene**

- **Second Main Contribution**

Understanding the representation that this Convent generates.

2 good parts:

I. Compute the feature vectors which are easily mapped to pose

II. Easy to generalize

Appearance-based relocalization and SLAM are two important traditional solutions

Training CNN

The training process always depend on veay large labeled image datasets. **!!COSTLY!!**

- Solution
    - An auto method of labeling data using structure from motion to generate large regression datasets of camera pose
    - Transfer learning which trains a pose regressor

2. Related Work

- Approaches to localization
    1. Metric SLAM : Creating a sparse or dense map of the environment
    2. Appearance-based : SIFT features.
    3. CNN
- Our work
    1. Combines the strengths of these approaches
    2. Follows the Scene Coordinate Regression Forests for relocalization
    3. Using thesepre-learned representations allows convnets to be used on smaller datasets without overfitting.

3. Model for deep regression of camera pose

OUTPUT: A vector $\boldsymbol{p}$, given by a 3D camera position x and orientation represented by quaternion $\boldsymbol{q}$

$$\boldsymbol{p} = [\boldsymbol{x}, \boldsymbol{q}]$$

3.1 Simultaneously learning location and orientation (*Loss Function*)

$$loss(I) = ||\hat{x} - x||_2 + \beta||\hat{q} - \frac{q}{||q||}||_2$$

where β is a scale factor chosen to keep the expected value of position and orientation errors to be approximately equal. Founded by grid search. Indoors: 120~750. Outdoors: 250~2000

In the experiments, it was found that if the estimation of position and pose were decomposed into two networks for regression separately, the effect is not as good as combining the two together. Therefore, PoseNet specially designs the loss function for visual localization, and the loss function refers to both position error and orientation error.

3.2 Architecture

PoseNet uses GoogLeNet as the backbone network, GoogLeNet has 22 layers and contains 6 Inception structures. In my opinion, after years of development, there are more options for backbone deep neural network structure, so there is no need to stick to the original work. The modifications made by PoseNet on GoogLeNet are introduced here.

GoogLeNet includes 3 classifiers, while the purpose of PoseNet is not to classify, its implementation can be described by a regression problem, so the classifiers of GoogLeNet are all replaced with regressors, specifically including

- Removing the softmax activation function
- setting 7 output neurons (corresponding to 3 position components and 4 quaternion direction components)
- A fully connected layer with 2048 neurons was added before the final output layer to collate the final output

In addition, the direction vectors in quaternion form need to perform a normalization operation when tested.

Input: 224x224

Output: $\boldsymbol{p} = [\boldsymbol{x}, \boldsymbol{q}]$

4. Dataset

Cambridge Landmarks, with 5scenes, *Search Posenet in* https://www.repository.cam.ac.uk/handle/1810/251342, or view https://github.com/SummerHuiZhang/PoseNet_Cambridge

The dataset was generated using structure from motion techniques which we use as ground truth measurementsfor this paper.

To test on indoor scenes we use the publically available 7Scenes dataset, https://www.microsoft.com/en-us/research/project/rgb-d-dataset-7-scenes/

5. Experiments

Figure 4: **Map of dataset** showing training frames (green), testing frames (blue) and their predicted camera pose (red). The testing sequences are distinct trajectories from the training sequences and each scene covers a very large spatial extent.

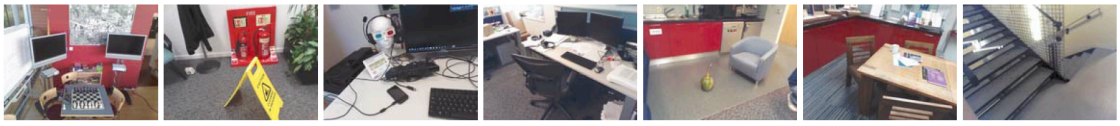
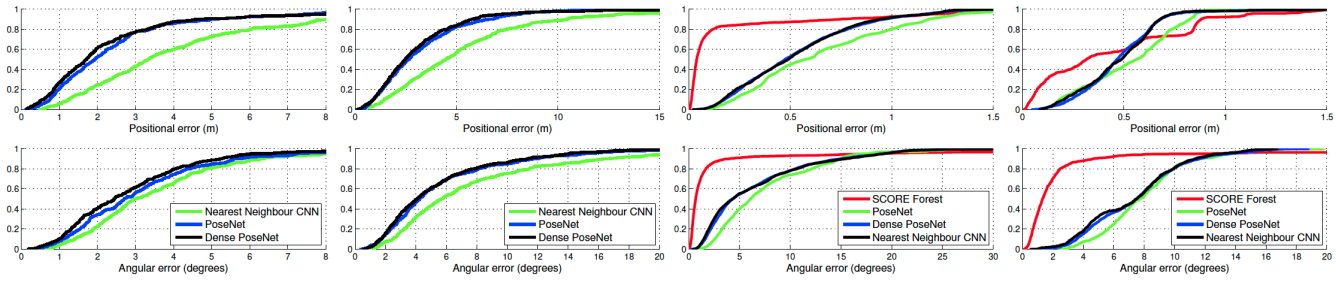


Figure 5: **7 Scenes dataset** example images from left to right; Chess, Fire, Heads, Office, Pumpkin, Red Kitchen and Stairs.

Scene	# Frames Train Test	Spatial Extent (m)	SCoRe Forest (Uses RGB-D)	Dist. to Conv. Nearest Neighbour	PoseNet	Dense PoseNet	
King's College	1220	343	140 x 40m	N/A	3.34m, 2.96◦	1.92m, 2.70◦	1.66m, 2.43◦
Street	3015	2923	500 x 100m	N/A	1.95m, 4.51◦	3.67m, 3.25◦	2.96m, 3.00◦
Old Hospital	895	182	50 x 40m	N/A	5.38m, 4.51◦	2.31m, 2.69◦	2.62m, 2.45◦
Shop Fac¸ade	231	103	35 x 25m	N/A	2.10m, 5.20◦	1.46m, 4.04◦	1.41m, 3.59◦
St Mary's Church	1487	530	80 x 60m	N/A	4.48m, 5.65◦	2.65m, 4.24◦	2.45m, 3.98◦
Chess	4000	2000	3 x 2 x 1m	0.03m, 0.66◦	0.41m, 5.60◦	0.32m, 4.06◦	0.32m, 3.30◦
Fire	2000	2000	2.5 x 1 x 1m	0.05m, 1.50◦	0.54m, 7.77◦	0.47m, 7.33◦	0.47m, 7.02◦
Heads	1000	1000	2 x 0.5 x 1m	0.06m, 5.50◦	0.28m, 7.00◦	0.29m, 6.00◦	0.30m, 6.09◦
Office	6000	4000	2.5 x 2 x 1.5m	0.04m, 0.78◦	0.49m, 6.02◦	0.48m, 3.84◦	0.48m, 3.62◦
Pumpkin	4000	2000	2.5 x 2 x 1m	0.04m, 0.68◦	0.58m, 6.08◦	0.47m, 4.21◦	0.49m, 4.06◦
Red Kitchen	7000	5000	4 x 3 x 1.5m	0.04m, 0.76◦	0.58m, 5.65◦	0.59m, 4.32◦	0.58m, 4.17◦
Stairs	2000	1000	2.5 x 2 x 1.5m	0.32m, 1.32◦	0.56m, 7.71◦	0.47m, 6.93◦	0.48m, 6.54◦

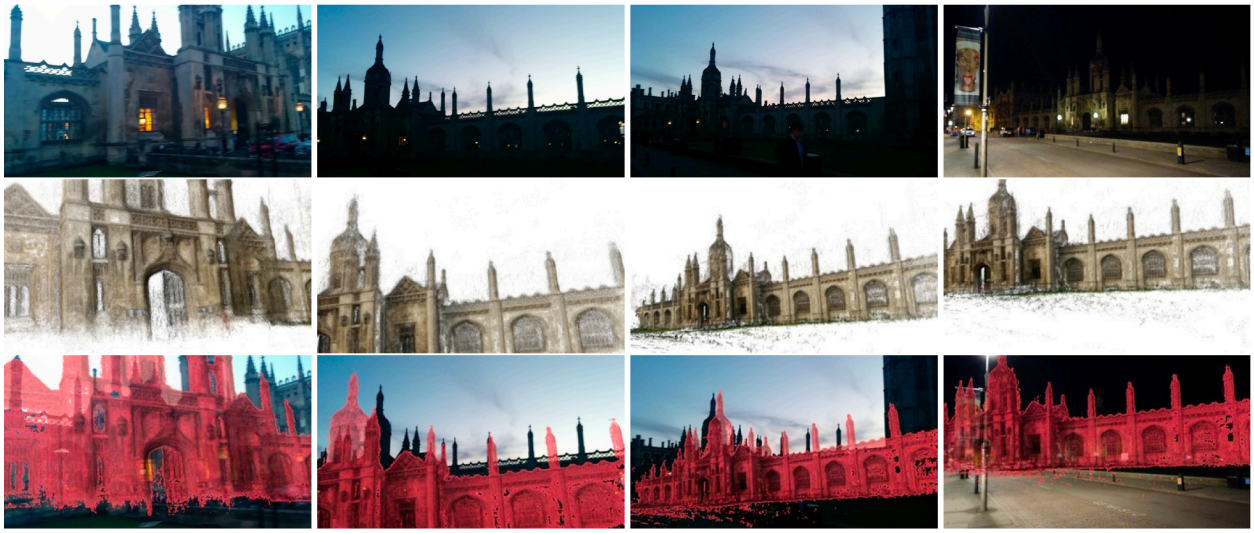
Figure 6: **Dataset details and results.** We show median performance for PoseNet on all scenes, evaluated on a single 224x224 center crop and 128 uniformly separated dense crops. For comparison we plot the results from SCoRe Forest [20] which uses depth, therefore fails on outdoor scenes. This system regresses pixel-wise world coordinates of the input image at much larger resolution. This requires a dense depth map for training and an extra RANSAC step to determine the camera's pose. Additionally, we compare to matching the nearest neighbour feature vector representation from PoseNet. This demonstrates our regression PoseNet performs better than a classifier.



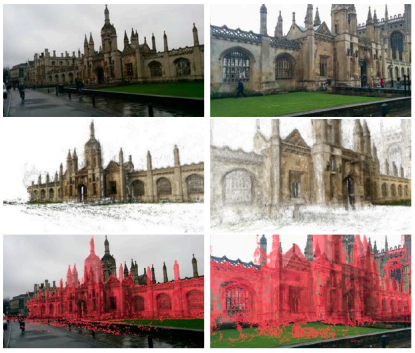
(a) King's College (b) St Mary's Church (c) Pumpkin (d) Stairs

Figure 7: **Localization performance.** These figures show our localization accuracy for both position and orientation as a cumulative histogram of errors for the entire testing set. The regression convnet outperforms the nearest neighbour feature matching which demonstrates we regress finer resolution results than given by training. Comparing to the RGB-D SCoRe Forest approach shows that our method is competitive, but outperformed by a more expensive depth approach. Our method does perform better on the hardest few frames, above the 95th percentile, with our worst error lower than the worst error from the SCoRe approach.

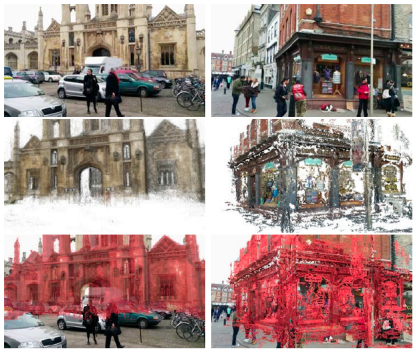
(a) Relocalization with increasing levels of motion blur. The system is able to recognize the pose as high level features such as the contour outline still exist. Blurring the landmark increases apparent contour size and the system believes it is closer.



(b) Relocalization under difficult dusk and night lighting conditions. In the dusk sequences, the landmark is silhouetted against the backdrop however again the convnet seems to recognize the contours and estimate pose.



(c) Relocalization with different weather conditions. PoseNet is able to effectively estimate pose in fog and rain.



(d) Relocalization with significant people, vehicles and other dynamic objects.

(e) Relocalization with unknown camera intrinsics: SLR with focal length 45mm (left), and iPhone 4S with focal length 35mm (right) compared to the dataset's camera which had a focal length of 30mm.

Figure 8: **Robustness to challenging real life situations.** Registration with point based techniques such as SIFT fails in examples (a-c), therefore ground truth measurements are not available. None of these types of challenges were seen during training. As convnets are able to understand objects and contours they are still successful at estimating pose from the building's contour in the silhouetted examples (b) or even under extreme motion blur (a). Many of these quasi invariances were enhanced by pretraining from the scenes dataset.

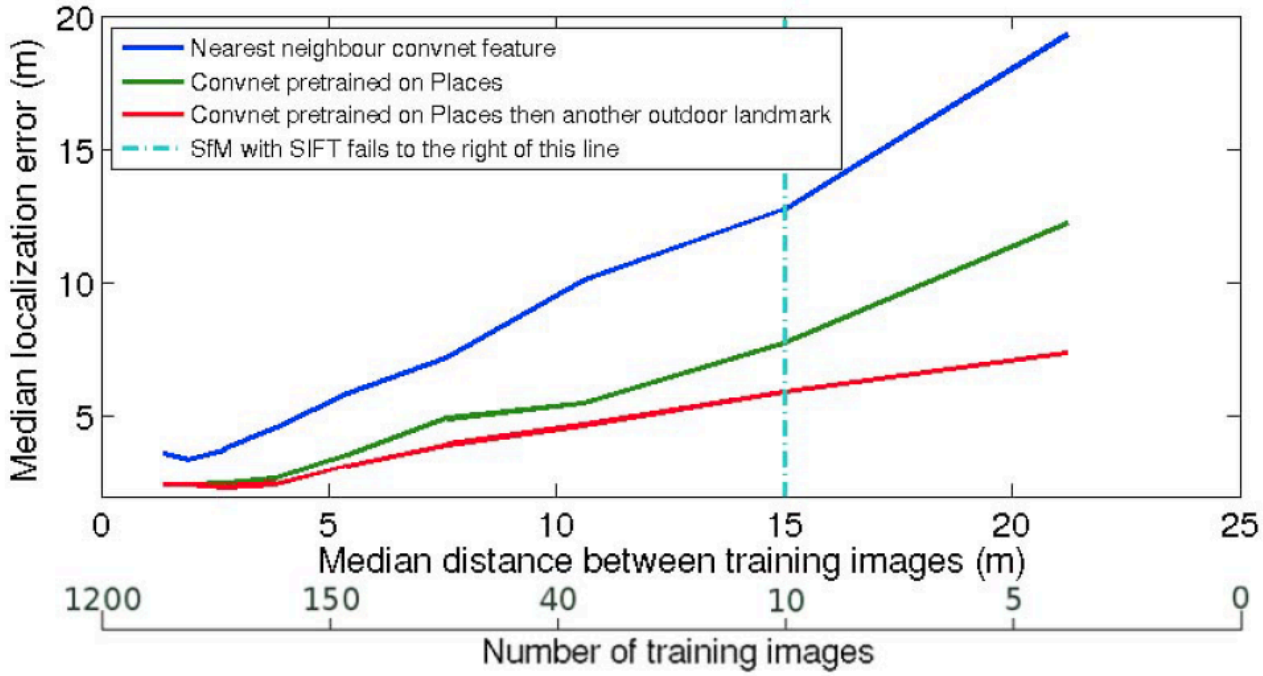


Figure 9: **Robustness to a decreasing training baseline** for the King's College scene. Our system exhibits graceful decline in performance as fewer training samples are used.

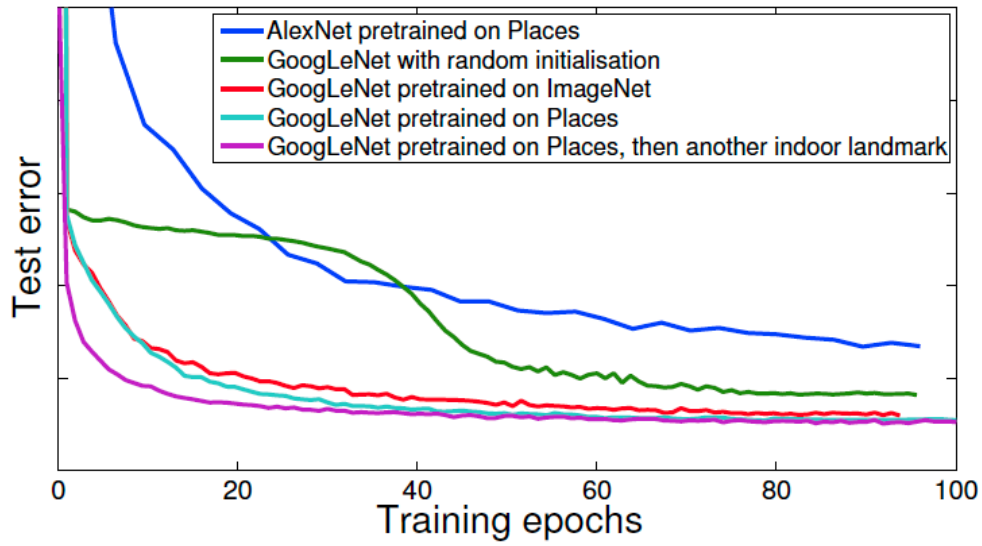


Figure 10: **Importance of transfer learning.** Shows how pre-training on large datasets gives an increase in both performance and training speed.

Above is the main experiment part of the research. To conclude, they show a state-of-the-art results. On the outdoor dataset, the localization results produced by center cropping produce an average of 2 to 3 meters and 3 to 4 degrees of localization error, with dense cropping producing slightly less error than center cropping; on the indoor dataset and, on the other hand, it can basically produce localization error within 0.5 meters.

In addition, PoseNet is more resistant to special situations such as light transformation (e.g., night) and unknown internal parameters, and can produce acceptable and more robust results. In general, this is an advantage of deep learning-based methods, while traditional geometric methods often produce less than satisfactory results in these special cases.

What's more, the direction vectors in quaternion form need to perform a normalization operation when tested.

6. Conclusions

We present, to our knowledge, the first application of deep convolutional neural networks to end-to-end 6-DOF camera pose localization. We have demonstrated that one can sidestep the need for millions of training images by use of transfer learning from networks trained as classifiers. We showed that such networks preserve ample pose informa- tion in their feature vectors, despite being trained to produce pose-invariant outputs. Our method tolerates large baselines that cause SIFT-based localizers to fail sharply.

In future work, we aim to pursue further uses of mul- tiview geometry as a source of training data for deep pose regressors, and explore probabilistic extensions to this algo- rithm [12]. It is obvious that a finite neural network has an upper bound on the physical area that it can learn to localize within. We leave finding this limit to future work.

My comments

The PoseNet, as an early work using deep learning methods for visual localization, has been a profound inspiration for all subsequent solutions using deep learning methods for visual localization.

The opening work of relocalization

As the title points out, the main contribution of the paper is a real-time relocation method. Although its main test site is outdoors, it can also be used in indoor environments. However, it is important to note that the main contribution of this paper is to learn the relationship between the photo and the corresponding camera pose through deep learning.

Since the data calibration of this paper is based on SfM, its accuracy is also limited by the accuracy of SfM in real situations. For example, in an outdoor environment, intense lighting and strong shadows may lead to local features that are not obvious, and it is difficult to construct a better point cloud model based on SfM. In this case, it leads to poor photo calibration accuracy, and it is difficult for PoseNet to achieve high accuracy too.

Welcome to Share your idea and comments!