Out with the Old?
Convolutional Neural Networks for Feature Matching and Visual Localization

Torsten Sattler
Department of Computer Science, ETH Zurich, Switzerland

Visual localization is the problem of estimating the position and orientation, *i.e.*, the camera pose, from which a novel image was taken with respect to some representation of a scene [6]. Visual localization is strongly related to Structure-from-Motion (SfM) and Simultaneous Localization and Mapping (SLAM) in the sense that all three aim to determine camera poses in a scene. Also, the point clouds generated by SfM and SLAM typically serve as the scene representation for visual localization, while localization is used to add additional images to a SfM model or to detect loop closures in SLAM. All three techniques typically use local image features such as SIFT [7] to establish correspondences. In turn, these matches are used to estimate the geometric relationship between images or an image and a 3D scene model.

Convolutional neural networks (CNNs) are a very popular class of deep learning techniques. They iteratively apply convolutions, followed by non-linear operators such as rectified linear units or the hyperbolic tangent function, to learn a desired function, *e.g.*, to classify objects [5], based on images as input. The power of CNNs lies in their ability to learn all parameters of the function from data rather than relying on hand-crafted features. As such, they learn features that are supported by evidence rather than using features based on human intuition (which might sometimes be wrong). As CNNs learn the desired function from data, there is no guarantee that the learned function provides meaningful results for inputs dissimilar to the training data. Thus, CNNs typically require a significant amount of training data to handle unseen data and thus generalize well.

In recent years, the availability of large-scale datasets such as ImageNet [10] and massive compute power in the form of GPUs has led to a renewed interest in CNNs. As a result of being able to train deep convolutional networks, deep learning in general and CNNs in particular have revolutionized many areas in Computer Vision: State-of-the-art techniques for object detection [9], object recognition [2], semantic segmentation [8], and single-view depth map prediction [19] all use CNNs and significantly outperform previous (hand-crafted) approaches.

In the light of the rise of deep learning and CNNs, a natural question is whether we should abandon our hand-crafted pipelines for visual localization (and in extension for SfM and SLAM) and replace them with learned alternatives. This extended abstract and the accompanying talk thus focus on this question and summarize our experience in this area: Sec. 1 considers the problem of learning visual localization in an end-to-end manner. Sec. 2 covers the problem of learning to detect local features via a CNN. Sec. 3 summarizes our findings when comparing learned feature descriptors with hand-crafted ones. This extended abstract is meant to give an overview over these three topics and highlight the main ideas and insights. For details, please see our original publications [13, 14, 17].

1. End-to-End Visual Localization

Traditionally, visual localization is solved in three stages [6]: In an offline stage, a 3D scene model is build and each 3D point in it is associated with the local image features it was triangulated from. During online operation, localization algorithms first extract local features from a given query image. Nearest neighbor search with their corresponding descriptors then establishes a set of 2D-3D correspondences between the features in the query image and the 3D points in the scene model. Finally, the pose of the query image is estimated by applying a perspective-n-point pose solver inside a RANSAC loop [1].

Differing from this classical pipeline, Kendall *et al.* were the first to approach the visual localization problem by training a CNN for camera pose regression [3, 4]. Their approach, termed PoseNet, essentially operates in two steps: The first part of the network learns an embedding from the space of input images into a 2048D space. This part is implemented using an existing neural network [15] pre-trained on the Places dataset [18][1]. The second part, trained from scratch, performs a linear regression from the 2048D space into the space of camera poses. The network is trained by

[1] It has been shown that the features on the more shallow layers of CNNs typically encode local appearance and geometry and thus are applicable to other tasks as well. As such, it is common to fine-tune pre-trained networks rather than training networks from scratch as this requires significantly less data.

either minimizing the difference between the predicted and the known ground truth pose [4] or by minimizing the reprojection error for a set of 2D-3D correspondences known for each training image [3].

We recently proposed a modification to PoseNet that significantly improves the accuracy of the predicted poses [17]. In addition, we were the first to perform a comparison against a state-of-the-art localization system based on handcrafted local features [11]. The results of that comparison are rather unfavorable for PoseNet and its variants: Even with our proposed modification, the camera poses predicted by PoseNet are still up to an order of magnitude less accurate compared with [11]. Results from another work [12] suggest that PoseNet does not necessarily outperform an even simpler pipeline: The baseline method uses image retrieval to identify the database image[2] most similar to the query. It then approximates the pose of the query image by the pose of the retrieved photo.

These results suggest that PoseNet is not particularly effective at learning to localize images. Our conjecture is that the mapping from image appearance to a 6DOF camera pose is too complex to be learned from the few hundreds to thousands training example typically available. As such, PoseNet seems unsuited if high camera pose accuracy is required, *e.g.*, in the context of Augmented Reality. However, we observed some promising results in scenes that cannot be handled by feature-based approaches [17]. For such scenarios, PoseNet could be a serviceable approach to obtain coarse camera pose predictions.

A natural question is whether one wants to learn the full visual localization pipeline. The problem of estimating a camera pose from a set of 2D-3D matches has a well-understood mathematical theory and there exist computationally efficient solutions. Similarly, efficient and effective algorithms for feature matching via nearest neighbor search are known. Thus, it might be sufficient to only learn the feature detection and description part of the pipeline. The next two sections thus cover our work on learned feature detectors [13] and descriptors [14].

2. Learning Feature Detectors

Feature detection is the problem of determining which local structures in an image can be detected repeatably under changes in viewing conditions while being ”interesting” enough to produce good descriptors for matching. We model this task as a ranking problem [13]: We want to learn a ranking function H that consistently ranks ”interesting” structures, *i.e.*, local image patches, higher or lower than ”uninteresting” ones. A set of feature detections can then be obtained by evaluating the ranking function for the patches around each pixel in an image, performing non-extrema suppression, and taking the top and bottom quantiles (according to H) of the remaining pixels.

We model the ranking function via a CNN and train it using quadruplets $(p_1, p_2, T(p_1), T(p_2))$ of patches. Here, p_1 and p_2 are different patches from the same image while $T(p_1)$ and $T(p_2)$ are versions of these patches undergoing the same transformation (*e.g.*, extracted from an image taken from a different viewpoint). The network is then asked to ensure that the relative ranking of the patches is similar, *i.e.*, that $\text{sign}(H(p_1) - H(p_2)) = \text{sign}(H(T(p_1)) - H(T(p_2)))$.

One advantage of our approach is that is does not require any pre-defined notion of what constitutes an ”interesting” patch but learns a definition of ”interestingness” from data. As such, our approach can be trained from data generated in an unsupervised manner. Our results show that our method outperforms the classical Difference-of-Gaussians detector used by SIFT, both in terms of repeatability and matchability. In addition, we show that our approach can also be applied for multi-modal data. For the latter, it is often hard to develop an intuition about which structures can be correlated between different sensor modalities. Thus, learning this correlation from data via our approach can be used to potentially solve this problem.

3. Learning Feature Descriptors

The goal of feature descriptor learning is to learn an embedding $D : \mathbb{R}^{w \times w} \rightarrow \mathbb{R}^n$ of $w \times w$ pixel patches into a n-dimensional descriptor space. This embedding should be discriminative: Patches depicting the same physical structure should have very similar descriptors while patches depicting different structures should results in very dissimilar descriptors. This behavior can be modelled mathematically both based on tuples and triplets of patches [16], resulting in (slightly) different performance. Training data is typically obtained from SfM: The transitivity of feature matching is used to obtain pairs of correlated patches that cannot be matched with existing techniques and can thus be used to learn better descriptors.

Existing CNN-based approaches for descriptor learning are typically evaluated on a patch retrieval task, where the goal is to determine whether two patches depict the same physical structure or not based on the learned descriptors. The evaluation criterion is the false positive classification rate at a recall of 95% for related pairs. Work on descriptor learning shows that the learned descriptors outperform SIFT under this metric.

Unfortunately, the false positive rate at 95% recall is not necessarily a meaningful measure in the context of visual localization or SfM, where one typically prefers a high precision of matches over a high recall. We thus evaluated whether the good classification performance of learned de-

[2] Database images are those images used to reconstruct the 3D scene model.

scriptors translates to a good performance in the context of SfM [14].

Our results show that learned descriptors, while outperforming SIFT, do not necessarily perform better than advanced (but still hand-crafted) SIFT variants. In addition, we noticed that learned descriptors exhibit a stronger variation in performance compared to hand-crafted features. We attribute this performance to two potential factors: The loss functions used for learning the descriptors might not be suited for the matching algorithms used by SfM pipelines and / or the learning process might not use enough training data to sufficiently cover the space of patch appearance. In any case, our results suggest that current learned descriptors should not automatically replace hand-crafted ones.

4. Conclusion

Returning to the original question about replacing hand-crafted localization (and SfM) systems with learned alternatives, the answer is clearly in favor of retaining parts of existing methods. More precisely, the stage that estimates the geometric relationships still seems to be required to produce accurate pose predictions. Based on our results so far, learning this stage from scratch seems to require a massive amount of data that is typically not available. However, it makes sense to replace parts of the feature detection and description pipeline. For the first part, feature detections, we have clearly demonstrated the promise of using learning. For the second part, replacing feature descriptors, our results suggest that hand-crafted descriptors still perform better in general. However, this might change with more training data becoming available.

Overall, we observe that deep learning and CNNs seem to (currently) be less dominating in the areas of localization and SfM compared to other research field in Computer Vision.

References

[1] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM (CACM)*, 1981. 1

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[3] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2

[4] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 1, 2

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Conference on Neural Information Processing Systems (NIPS)*, 2012. 1

[6] Y. Li, N. Snavely, and D. P. Huttenlocher. Location Recognition Using Prioritized Feature Matching. In *European Conference on Computer Vision (ECCV)*, 2010. 1

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 2004. 1

[8] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-Resolution Residual Networks for Semantic Segmentation in Street Scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[9] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Conference on Neural Information Processing Systems (NIPS)*, 2015. 1

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 2015. 1

[11] T. Sattler, B. Leibe, and L. Kobbelt. Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2016 (to appear). 2

[12] T. Sattler, A. Torii, J. Sivic, M. Pollefeys, H. Taira, M. Okutomi, and T. Pajdla. Are Large-Scale 3D Models Really Necessary for Accurate Visual Localization? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[13] N. Savinov, A. Seki, L. Ladicky, T. Sattler, and M. Pollefeys. Quad-Networks: Unsupervised Learning to Rank for Interest Point Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

[14] J. L. Schönberger, H. Hardmeier, T. Sattler, and M. Pollefeys. Comparative Evaluation of Hand-Crafted and Learned Local Features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[16] D. P. Vassileios Balntas, Edgar Riba and K. Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2016. 2

[17] F. Walch, C. Hazirbas, L. Leal-Taixé, T. Sattler, S. Hilsenbeck, and D. Cremers. Image-based localization using LSTMs for structured feature correlation. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2

[18] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Conference on Neural Information Processing Systems (NIPS)*, 2014. 1

[19] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1