Camera Pose Auto-Encoders for Improving Pose Regression: Supplementary Materials

Yoli Shavit and Yosi Keller

Bar-Ilan University, Ramat Gan, Israel
{yolisha, yosi.keller}@gmail.com

1 Appendix

1.1 Memory Requirements of APRs and RPRs

A key motivation for PAEs is to reduce the memory burden associated with RPRs, which require train images or their encoding to be available at inference time. Table 1 shows the memory requirements for RPRs, single and multi-scene APRs with and without PAEs.

Table 1. Order of magnitude of the storage required for different APRs and RPRs (considering 10 scenes). For RPRs, we assume that a single encoding weighs 5Kb and each scene contains 2000 images.

Method	Storage
RPR ([1])	Gb
Single Scene APR [7]	Mb
Multi Scene APR	Mb
Multi Scene APR + PAE	Mb

1.2 Data Augmentation and Training

When training our camera pose auto-encoder and during test-time optimization, we follow the same test-time data pre-processing used by [7]. Specifically, images are first resized, where the smaller edge is resized to 256 pixels, and then a 224×224 center crop is taken. When training teacher APRs, we follow a similar procedure but additionally apply random jitter to the brightness, contrast, and saturation and take a random crop (rather than the center one). To train our decoder, we used $64x64$ crops (rescaling is done to maintain the original ratio between scaling and resizing). In order to support easy reproduction of the results reported by other researchers, we provide training and evaluation code, pretrained models, dedicated configuration files, and examples to perform each experiment.

1.3 Additional Ablations

Ablation of Fourier Features We carry additional ablation on the number of periodic encoding functions L (see Section 3.2 in the main text) for our Fourier Features. Table 2 shows the results for a PAE with a 4-layers MLP trained without Fourier Features and for a PAE with a 4-layers MLP trained with Fourier Features with $L = 3$ and $L = 6$. The latter configuration, which yields the lowest position error, is selected for our PAE architecture.

Table 2. Ablations of Fourier Features for the PAE architecture . We compare the median position and orientation errors when using a 4-layer MLP without Fourier Features and when applying Fourier Features with L the number of levels set to 3 and 6 (selected architecture). The performance is reported for the KingsCollege scene (CambridgeLandmarks dataset). The Teacher is a PoseNet APR with a MobileNet architecture.

Auto Encoder Architecture	**Position [m]**	**Orientation [deg]**
4-Layers MLP (No Fourier Features)	1.26	3.54
4-Layers MLP + Fourier Features, $L = 3$	1.36	3.27
4-Layers MLP + Fourier Features, $L = 6$	1.15	3.58

Architecture Choices

- Dimension of $\hat{\mathbf{z}}_\mathbf{x}$ and $\hat{\mathbf{z}}_\mathbf{q}$: the dimension of the PAE's latent vectors should match the dimension of the latent output of the APR teacher. For the APRs used in our paper, the dimension is the same for both vectors.
- Separate branches for position and orientation estimation: In our work, we use both single- and multi- scene APRs with separate branches for position and orientation. Nevertheless, PAEs can be applied to any APR architecture.
- Image size (image decoding): The choice of 64×64 pixels as the size of the reconstructed image is set to maintain a short runtime. We note that similar results (in terms of position error and image quality) were achieved with a higher image resolution (256 pixels).

1.4 Test-time Position Refinement: Additional Results

Table 3. Median position error in meters when sampling a random guess around the ground-truth pose and when refining the initial guess with our test-time optimization. We report the results for the CambridgeLandmarks dataset.

Method	**K. College**	**Old Hospital**	**Shop Facade**	**St. Mary**
Initial Guess	1.47	1.45	1.53	1.8
Refined Guess	0.59	0.57	0.56	0.4

Table 4. Median position error of single-scene APRs, with and without our test-time position refinement. Performance is reported for the KingsCollege scene (CambridgeLandmarks dataset).

APR Architecture	**Without Position Refinement** [m]	**With Position Refinement** [m]
PoseNet+MobileNet	1.24	0.91
PoseNet+ResNet50	1.56	1.31
PoseNet+EfficientNet	0.88	0.81

Test-time Position Refinement with a Random Pose Guess We perform an additional verification of our test-time optimization, where instead of using an APR to estimate the pose of the query and its latent representation, we take a random guess around the ground truth pose and encode it (i.e., perform pose estimation *without* images). Table 3 reports the results for the CambridgeLandmarks dataset, showing the accuracy of the position of the initial guess and the refined estimate, obtained with our test-time optimization. Our method can significantly reduce the error of the initial guess.

Test-time Position Refinement with Single-scene APRs and PAEs We apply our test time position refinement to single scene APRs and their respective student PAEs, trained on the KingsCollege scene from the CambridgeLandmarks dataset. Table 4 shows the position error achieved by applying each single scene APR with and without our PAE-based position refinement. Our test-time optimization yields a consistent improvement, regardless of the APR architecture used.

1.5 Test-time Orientation Estimation with Affine Combination

Our test-time refinement focuses on position estimation through affine combination of train positions, fetched based on PAE encoding. We further evaluate this procedure to refine the orientation estimation. Table 5 shows the results for MS-Transformer with and without applying our affine combination to orientation estimation for the CambridgeLandmarks dataset. The affine combination leads to degradation, suggesting that additional research is needed to extend the PAE-based test time refinement to improve orientation estimation. Natural extensions are estimating the weights for position and orientation separately as well as combining the quaternions through quaternion averaging algorithms such as [9] (rather than directly applying a weighted average as done in our proposed procedure).

1.6 Comparison of Camera Localization Methods

Our work focuses on encoding camera poses and demonstrating their usages for absolute pose regression. However, absolute pose regression is one family of

Table 5. Median orientation error in degrees for the CambridgeLandmarks dataset, obtained with MS-Transformer[13] with and without the proposed test-time affine combination.

Method	K. College	Old Hospital	Shop Facade	St. Mary
MS-Transformer	1.47	2.39	3.07	3.99
MS-Transformer with Affine Combination	2.83	4.04	3.44	7.96

methods out of several clusters of techniques for camera localization, namely: structure-based methods, image retrieval, and relative pose regression (see our Related Work section). In order to support a more complete comparison, Table 6 shows the results for representative methods for the CambridgeLandmarks and 7Scenes datasets. Structure-based method achieve the best localization accuracy. However, they require the intrinsics of the query camera, which might not be accurate or available.

Table 6. Comparison of localization methods when applied to the Cambridge Landmarks and 7Scenes datasets. We show results for representative methods from each localization family: structure-based (STR), image retrieval (IR), relative pose regression (RPR) and image based absolute pose regression (APR). We report the average of median position/orientation errors across scenes in meters/degrees for each method.

	Method	**CambridgeLand.**	**7Scenes**
STR	DSAC [2]	0.15/0.4	0.03/1.4
STR	DSAC* [3]	0.15/0.4	–/–
IR	VLAD [14]	2.56/7.1	0.26/12.5
IR	VLAD+Inter [11]	1.67/4.9	0.24/11.7
RPR	EssNet [19]	1.08/3.4	0.22/8.0
RPR	VLocNet [15]	0.78/2.8	0.05/3.8
RPR	GL-Net [18]	1.22/2.4	0.19/6.3
RPR	NC-EssNet [19]	0.85/2.8	0.21/7.5
RPR	RelocGNN[3]	0.91/2.3	0.91/2.3
APR	PoseNet [7]	2.09/6.84	0.44/10.4
APR	BayesianPN [8]	1.92/6.28	0.47/9.81
APR	LSTM-PN [16]	1.30/5.52	0.31/9.86
APR	SVS-Pose [10]	1.33/5.17	−−
APR	GPoseNet [5]	2.08/4.59	0.31/9.95
APR	PoseNetLearn [6]	1.43/2.85	0.24/7.87
APR	GeoPoseNet [6]	1.63/2.86	0.23/8.12
APR	MapNet [4]	1.63/3.64	0.21/7.78
APR	IRPNet [12]	1.42/3.45	0.23/8.49
APR	AttLoc[17]	−−	0.20/7.56
APR	MS-TransFormer[13]	1.28/2.73	0.18/7.28
APR	MS-Transformer + Position Refinement	0.96/2.73	0.15/7.28

References

1. Balntas, V., Li, S., Prisacariu, V.: Relocnet: Continuous metric learning relocalisation using neural nets. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
2. Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., Rother, C.: Dsac - differentiable ransac for camera localization. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2492–2500. IEEE Computer Society, Los Alamitos, CA, USA (jul 2017). https://doi.org/10.1109/CVPR.2017.267, `https://doi.ieeecomputersociety.org/10.1109/CVPR.2017.267`
3. Brachmann, E., Rother, C.: Visual camera re-localization from rgb and rgb-d images using dsac. IEEE Transactions on Pattern Analysis and Machine Intelligence (01), 1–1 (apr 2021)
4. Brahmbhatt, S., Gu, J., Kim, K., Hays, J., Kautz, J.: Geometry-aware learning of maps for camera localization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
5. Cai, M., Shen, C., Reid, I.: A hybrid probabilistic model for camera relocalization (2019)
6. Kendall, A., Cipolla, R.: Geometric loss functions for camera pose regression with deep learning. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6555–6564 (2017). https://doi.org/10.1109/CVPR.2017.694
7. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-DOF camera relocalization. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 2938–2946 (2015). https://doi.org/10.1109/ICCV.2015.336
8. Kendall, A., Cipolla, R.: Modelling uncertainty in deep learning for camera relocalization. In: Proceedings of the International Conference on Robotics and Automation (ICRA) (2016)
9. Markley, F.L., Cheng, Y., Crassidis, J.L., Oshman, Y.: Averaging quaternions. Journal of Guidance, Control, and Dynamics **30**(4), 1193–1197 (2007)
10. Naseer, T., Burgard, W.: Deep regression for monocular camera-based 6-DoF global localization in outdoor environments. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 1525–1530 (2017)
11. Sattler, T., Zhou, Q., Pollefeys, M., Leal-Taixé, L.: Understanding the limitations of cnn-based absolute camera pose regression. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3297–3307 (2019). https://doi.org/10.1109/CVPR.2019.00342
12. Shavit, Y., Ferens, R.: Do we really need scene-specific pose encoders. In: To Appear in 2021 IEEE International Conference on Pattern Recognition (ICPR) (2021)
13. Shavit, Y., Ferens, R., Keller, Y.: Learning multi-scene absolute pose regression with transformers. In: 2021 IEEE International Conference on Computer Vision (ICCV) (2021)
14. Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M., Pajdla, T.: 24/7 place recognition by view synthesis. In: CVPR (2015)
15. Valada, A., Radwan, N., Burgard, W.: Deep auxiliary learning for visual localization and odometry. ICRA pp. 6939–6946 (2018)
16. Walch, F., Hazirbas, C., Leal-Taixé, L., Sattler, T., Hilsenbeck, S., Cremers, D.: Image-based localization using lstms for structured feature correlation. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 627–637 (2017). https://doi.org/10.1109/ICCV.2017.75

17. Wang, B., Chen, C., Lu, C.X., Zhao, P., Trigoni, N., Markham, A.: Atloc: Attention guided camera localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 10393–10401 (2020)
18. Xue, F., Wu, X., Cai, S., Wang, J.: Learning multi-view camera relocalization with graph neural networks. In: CVPR (2020)
19. Zhou, Q., Sattler, T., Pollefeys, M., Leal-Taixe, L.: To learn or not to learn: Visual localization from essential matrices. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 3319–3326. IEEE (2020)