https://doi.org/10.2352/EI.2023.35.17.3DIA-102

Learned Visual Localization with Camera Pose Refinement and Verification Based on Differentiable Renderer

Chanchang Tsai, Hajime Taira, and Masatoshi Okutomi
Tokyo Institute of Technology; Tokyo, Japan

Abstract

This manuscript presents a new CNN-based visual localization method that seeks a camera location of an input RGB image with respect to a pre-collected RGB-D images database. To determine an accurate camera pose, we employ a coarse-to-fine localization manner that firstly finds coarse location candidates via image retrieval, then refines them using local 3D structure represented by each retrieved RGB-D image. We use a CNN feature extractor and a relative pose estimator for coarse prediction that do not sufficiently require a scene-specific training. Furthermore, we propose a new pose refinement-verification module that simultaneously evaluates and refines camera poses using differentiable renderer. Experimental results on public datasets show that our proposed pipeline achieves accurate localization on both trained and unknown scenes.

Introduction

Determining a camera of a given image is an essential ability for computer vision problems and often plays a vital role in several applications, such as Structure from Motion (SfM) and Simultaneous Localization and Mapping (SLAM). Visual localization [1–5] solves the problem as a relocalization task that estimates a 6DoF camera pose of a query image relatively to the given scene represented by pre-collected knowledge (database) such as 3D point cloud, *e.g.*, solving Perspective-n-Point (PnP) problem associated with 2D-to-3D correspondences. Thanks to recent progress in the machine learning area, several works encode such database into a learned camera pose regressor often built as a convolutional neural network (CNN) model [6–15], which enables a compact representation of the scene and a fast prediction in the testing phase. However, the accuracy of these approaches often is inferior to that of "hand-crafted" localization pipelines [16, 17], which constructs a crucial challenge for them. Furthermore, a pose estimation module designed as a deeply-learnable component is often specialized to the training scene, thus cannot be diverted to unknown scenes. This also prevents them from addressing to large-scaled scenes [18].

In this paper, we propose a learned visual localization pipeline to tackle these problems. Our contributions are threefold; (1) Instead of designing an end-to-end pose regressor that directly predicts an absolute camera pose in the scene, we construct a coarse-to-fine camera localization scheme that progressively update camera location. For a given query, we firstly perform an image retrieval among pose-known database images to find the relevant location of the query and subsequent pose prediction module estimate a 6DoF camera pose as a relative pose from the relevant database image. Since the pose prediction module focuses only on the local geometric relationship between the query and retrieved images, it can be expanded to unknown scenes that are not included in the training images; (2) We introduce "Top-N verification" strategy [3–5] into a deeply learned visual localization. Instead of determining one single prediction for each query, our localization pipeline stores several candidates of relevant location in the scene and find the best of them using known scene structure. Furthermore, we propose a new pose verification module (Fig. 1) that simultaneously evaluates and refines cameras using a differentiable renderer [19]. Our pose verification is simple enough to be applied to any CNN–based camera pose prediction, yet be effective for constantly improving the accuracy; (3) We tested our pipeline on two public datasets for visual localization task. Results show that our method achieves an accurate localization on both known (trained) and unknown scenes.

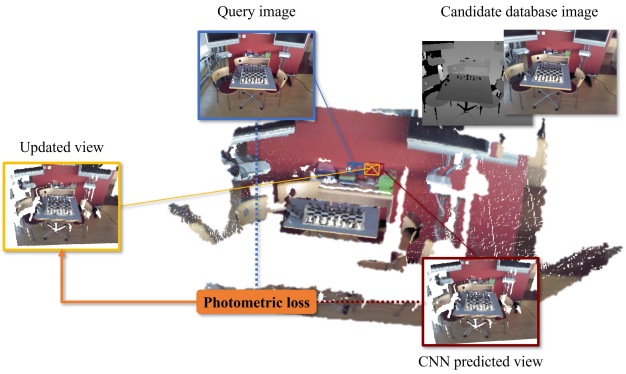

Figure 1. *Conceptual visualization of our differentiable-renderer-based pose refinement module. For a given query, we firstly obtain the initial prediction of query camera while referencing the relevant RGB-D database image chosen as a candidate and render the view using its 3D structure. Rendered synthetic image is then evaluated by the photometric distance from the original query image and iteratively updated by back-propagation towards the camera extrinsics.*

Related Works

Visual localization has often been studied as a camera reconstruction task that solves PnP to determine a 6DoF camera in a known 3D model, *e.g.*, SfM model [2–5]. Image retrieval [1, 4] is often used to determine a coarse location of the camera. One familiar strategy here is to hold several similar candidates instead of seeking the best one [3–5]. Taira *et al.* proposed camera pose verification strategy [5] that finds the best location in candidates by evaluating photometric consistency of the rendered 3D model.

Recently, several works attempted to replace the entire [6–9] or parts [10–15] of localization pipeline with deeply learned mod-

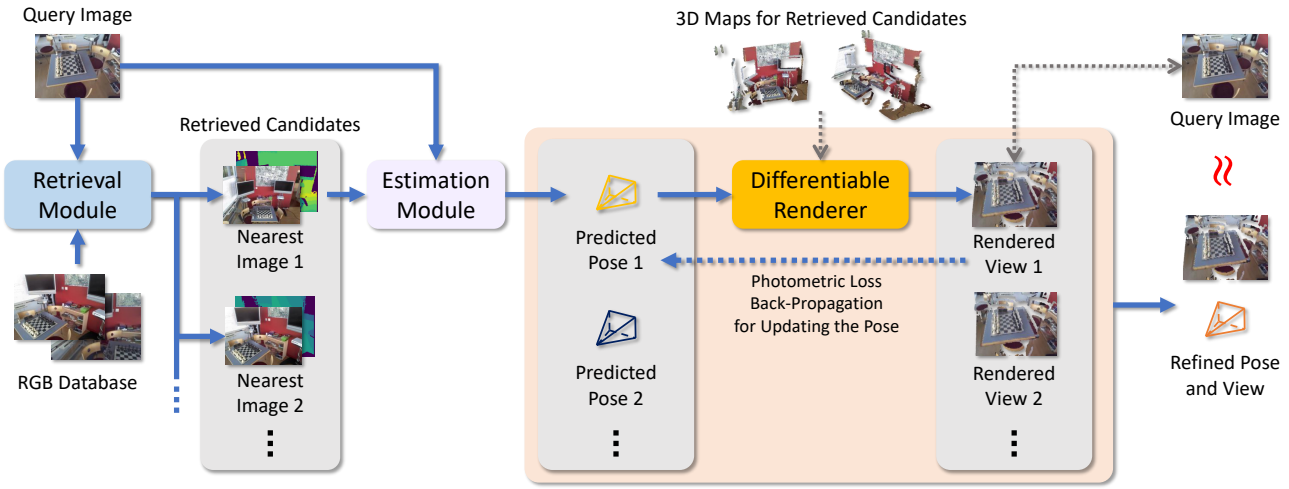

Figure 2. *The overview of our camera localization pipeline. (a) For an input of RGB query image, we firstly seek N relevant database images via image retrieval and predict a coarse query camera candidate with respect to each of database cameras via a pair-wise relative pose regressor. (b) The predicted candidates are then refined by back-propagating the photometric loss of rendered views. We finally select the most similar (refined) candidate as the final localization result.*

ules. Kendall *et al.* proposed to convert the whole process of localization to a learned CNN pose regressor that directly predicts absolute poses [6, 7]. Several works [10–13] achieved visual localization by predicting a relative camera pose with respect to the neighbor training image. Following these approaches, we carefully employ camera pose verification strategy to CNN–based visual localization, and extend it to also refine candidates to get a more accurate camera pose.

Differentiable renderer. Kato *et al.* proposed a differentiable scheme for mesh rendering [19] that enables to propagate residuals of the rendered view towards meshes, texture, and camera. This can also be used to train backbone CNN models for specific tasks represented by 3D appearances, such as object pose estimation. Alternatively, NeRF [20] learns a deep network encoding the density field and radiances of the scene. Since the network consisting of fully connections is differentiable in nature, it can also be used as a differentiable layer to render the nobel view. iNeRF [21] utilizes a pre-trained NeRF model to find a camera of a newly seen image, by back-propagating the photometric error of rendered view towards camera parameters. Our proposed pose refinement scheme seeks a camera pose in a similar manner as in iNeRF, but ours directly uses the known scene structure and does not require any pre-training on the scene.

Learned Visual Localization using Differentiable Renderer

Fig. 2 shows the overview of our retrieval–based visual localization scheme. We assume a database of RGB–D images with known camera poses and each image is represented as an image feature using a trained CNN feature extractor. For a queried RGB image, we firstly perform image retrieval to find similar database images representing the potential candidates of the location. For each pair of query and selected database image, a learned CNN pose regressor also predicts a relative camera pose, which is turned into a coarse camera candidate of the query. Our retrieval and pose estimation modules are based on an existing CNN localization pipeline [10], but ours remains multiple candidates to improve the final prediction, whereas most of relevant methods determine one specific camera pose for the input image. For each of coarse predictions, we additionally apply a pose refinement and verification strategy [5] that evaluates the photometric error between the original and synthetic images which is generated with respect to the relative camera pose and the given depth maps of database images. Our pose refinement is designed as a differentiable module, thus is suitable as a post-processing for CNN–based localization methods. We finally select the most similar camera pose in the refined candidates as a final location output.

Image retrieval and relative pose estimation

Because of its compactness and simplicity, we employ an existing localization scheme named as RelocNet [10] as the basis of our retrieval and pose estimation stage. This stage consists of CNN feature extractor and direct pose regressor that extracts features representing images for retrieval and predicts a relative pose between the input image pair. We use up to conv3 layer of ResNet18 [22] as the feature extractor, while adding average pooling layer to obtain a 256-dimensional feature vector used for image retrieval. The extracted feature is also used as an input of subsequent relative pose regressor. From the concatenated two features from paired images, we obtain a 6-dimensional pose vector through several linear layers, which represents camera pose in a $se(3)$ space [10]. As in [10], we trained both parts in a end-to-end manner using training image pairs while evaluating the frustum overlapping between cameras and pose error with respect to the ground-truth camera.

In the testing phase, we store image features for all database images and match them to the features extracted from the query image. For the most similar 10 database images, we also perform subsequent pose estimation module. The obtained relative pose

between query and each database image is treated as the coarse camera candidate and refined in the latter refinement-verification stage.

Pose refinement and verification using differentiable rendering

To refine the coarse prediction from the pose regressor, we exploit an idea of pose verification strategy [5] that utilizes the appearance of the scene seen from the candidate cameras. Assuming a fine 3D (colored) point cloud representing the whole scene, [5] renders candidate cameras by projecting 3D points onto the image and evaluates them by the photometric distance between the synthetic view and original input image. However, their approach requires a large known 3D model, thus costs a large computational resources and memory footprint. Also, their final output highly depends on the quality of the coarse prediction for candidates. Therefore, we propose a new *pose refinement–verification* strategy that simultaneously evaluates the camera candidates and also refine them to get more accurate camera poses. We employ a recent differentiable rendering module [19] that enables us to update cameras by the back-propagation operation. Furthermore, we renders each camera using only the local scene structure represented by the depth map of the retrieved database image, which leads a relatively small-scaled 3D model.

For each of camera candidates obtained via image retrieval and coarse prediction, we iteratively refine the camera while evaluating the appearance of the local scene. In each step we render the camera using a local 3D point cloud back-projected from the depth map of the candidate database image. Assuming i-th pixels in the original query image $I_q(i)$ and the rendered image $I_r(i)$, we compute the photometric loss $\mathscr{L}_{photometric}$ as:

$$\mathscr{L}_{photometric} = \frac{\sum_{i \in P(I_r)} \left\| I_r(i) - I_q(i) \right\|}{|P(I_r)|} \tag{1}$$

where $P(I_r)$ is the set of valid pixels in the rendered image. We then back-propagate the loss towards the camera pose and update it with a learning rate 0.003. We empirically set the iteration number as 200 times to sufficiently reaches to convergence. Finally, from the 10 refined candidates, we select the best one that leaves the lowest photometric loss.

Experiments

In this section, we test our localization pipeline on public datasets for visual localization. We firstly test our whole localization pipeline on 7scenes dataset [23], which is a popular benchmarking of visual localization task. We next conduct ablation studies to show that each of components in our pipeline produces complementary performance gains. We also perform visual localization on 12scenes dataset [24], while using the model trained on 7scenes. Even on the different scenes captured by different devices, our pre-trained model achieves comparable results to state-of-the-arts.

Implementation. We implement our retrieval-estimation module on PyTorch. For our camera pose refinement and verification module, we utilize a differentiable renderer implemented by PyTorch3D toolkit. For the rasterization setting, we set the blur radius as 0.0, faces per pixel as 1. And we set the environment light as ambient lights. For the shader, we choose the Hard Phong shader.

Visual localization on 7scenes dataset. 7scenes [23] consists of 7 indoor scenes captured as several RGB–D sequences with known camera poses, providing 26,000 training images and 17,000 images for testing. We evaluate our method using RGB test images, while using RGB–D training images as the database for each scene.

We firstly train our model using image pairs extracted from training images. We gather 80,000 training image pairs as the distances of pairs are lower than 2.0m, 70°. Training is performed in the same manner as in [10], whereas we set the learning rate at 10^{-4} and the weight decay at 10^{-5}.

Tab. 1 shows the localization accuracy of our proposed pipeline and other representative methods on 7scenes. Compared to methods predicting absolute camera poses [7–9], our coarse-to-fine localization pipeline consisting of coarse image retrieval and fine relative pose estimation provides constant performance gains. Our method also employs Top-N verification strategy that keeps multiple candidates for each query, resulting in better performance than other relative pose regressor–based methods [10–12] including RelocNet [10], which is the baseline of our pipeline. The effect is especially shown on the scenes Chess and Heads, which includes much repetitive and texture-less regions and ours can localize images more than double times accurate compared to the baseline. Whereas CamNet [13] shows the best performance in all scenes by introducing their own retrieval module that re-selects image pairs, note that our pose refinement–verification module can be applied to any localization methods which can obtain multiple pose candidates[1]. Also notice that whereas we do not train our model for each scene specifically, our whole pipeline still shows well localization results on most scenes, except for Stairs, which captures highly repetitive structure and specular materials.

Tab. 2 provides evaluation for several variants of our method that subtract components in the full pipeline. A variant that omits coarse pose prediction module and outputs the camera of the Top-1 retrieved database image (left-most) suggests that image retrieval step itself is often unstable so that the most similar image can actually be an erroneous candidate that has few view-overlapping with query. Results show ours achieves a progressive improvement, by adopting each of components including coarse relative pose regressor (Coarse pred.), differentiable renderer-based pose refinement (Refine), and Top-10 verification (Top-10).

Qualitative examples. Fig. 3 shows typical examples in 7scenes on which our localization pipeline can successfully find the query 6DoF camera poses. Our RelocNet-based image retrieval (b) firstly finds the relevant database image that partly shares view with query image, and subsequent coarse pose prediction (c) estimates the initial camera pose as the relative motion from the relevant database image. As in examples for Fire, Heads, RedKitchen, and Stairs, since the relevant database image captures less shared views with query, the predicted view (c) often obtains gaps when compared to the original query view. Our differentiable-renderer-based pose refinement module evaluates differences between the original and rendered query view, consequently gets the compensated view (d) that shows less gaps

[1] We do not provide results of our method combined with CamNet just because their full implementation is unavailable.

Table 1. Camera localization performance on 7scenes [23]. Each column shows the median translational and rotational errors reported in each original paper except for Ours. We highlight the best and second best results by red and blue letters, respectively.

	Absolute Pose Regression			Relative Pose Regression				
Scene	PoseNet2 [7]	MapNet [8]	Atloc [9]	NN-Net [12]	RelocNet [10]	AnchorNet [11]	CamNet [13]	**Ours**
Chess	0.13m, 4.48°	0.08m, 3.25°	0.10m, 4.07°	0.13m, 6.46°	0.12m, 4.14°	0.06m, 3.89°	0.04m, 1.73°	0.04m, 1.61°
Fire	0.27m, 11.3°	0.27m, 11.69°	0.25m, 11.4°	0.26m, 12.72°	0.26m, 10.4°	0.15m, 10.3°	0.03m, 1.74°	0.13m, 3.32°
Heads	0.17m, 13.0°	0.18m, 13.25°	0.16m, 11.8°	0.14m, 12.34°	0.14m, 10.5°	0.08m, 10.9°	0.05m, 1.98°	0.06m, 3.65°
Office	0.19m, 5.55°	0.17m, 5.15°	0.17m, 5.34°	0.21m, 7.35°	0.18m, 5.32°	0.09m, 5.15°	0.04m, 1.62°	0.07m, 1.82°
Pumpkin	0.26m, 4.75°	0.22m, 4.02°	0.21m, 4.37°	0.24m, 6.35°	0.26m, 4.17°	0.10m, 2.97°	0.04m, 1.64°	0.08m, 2.10°
RedKitchen	0.23m, 5.35°	0.23m, 4.93°	0.23m, 5.42°	0.24m, 8.03°	0.23m, 5.08°	0.08m, 4.68°	0.04m, 1.63°	0.08m, 1.99°
Stairs	0.35m, 12.4°	0.30m, 12.08°	0.26m, 10.5°	0.27m, 11.82°	0.28m, 7.53°	0.10m, 9.26°	0.04m, 1.51°	0.34m, 9.81°
Average	0.23m, 8.12°	0.21m, 7.77°	0.20m, 7.56°	0.21m, 9.30°	0.21m, 6.73°	0.09m, 6.74°	0.04m, 1.69°	0.12m, 3.47°

Table 2. Ablation studies. Each column represents a variant of our method and reports the average of median localization error on 7scenes [23].

	Variants			Ours
Coarse Pred.		✓	✓	✓
Refine			✓	✓
Top-10				✓
Average	0.363m, 12.050°	0.219m, 7.153°	0.141m, 4.206°	**0.117**m, **3.471**°

Table 3. Camera localization median error on 12scenes [24]. For PoseNet and ESAC, we report results provided by [25].

	Trained on 12sc.			Trained on 7sc.
Scene	PoseNet [6]	PnLP [26]	ESAC [18]	**Ours**
kitchen1	0.29m, 15.48°	0.09m, 4.1°	0.01m, 0.44°	0.06m, 1.04°
living1	0.29m, 15.31°	0.08m, 2.9°	0.01m, 0.43°	0.03m, 0.80°
kitchen2	0.21m, 18.18°	0.10m, 3.7°	0.01m, 0.46°	0.01m, 0.72°
living2	0.31m, 23.58°	0.10m, 4.7°	0.01m, 0.40°	0.02m, 0.71°
bed	0.57m, 17.85°	0.12m, 5.7°	0.01m, 0.46°	0.04m, 0.83°
luke	0.35m, 20.07°	0.14m, 5.5°	0.01m, 0.59°	0.07m, 1.11°
5a	0.57m, 14.55°	0.09m, 3.6°	0.01m, 0.59°	0.10m, 0.99°
5b	0.47m, 15.49°	0.10m, 4.7°	0.02m, 0.59°	0.02m, 0.68°
lounge	0.29m, 18.42°	0.10m, 3.5°	0.02m, 0.61°	0.03m, 1.18°
manolis	0.22m, 17.45°	0.09m, 3.7°	0.01m, 0.53°	0.02m, 0.96°
gates362	0.27m, 16.71°	0.10m, 4.7°	0.01m, 0.46°	0.06m, 0.68°
gates381	0.37m, 20.52°	0.11m, 4.4°	0.01m, 0.67°	0.10m, 1.23°
Average	0.35m, 17.80°	0.10m, 4.27°	0.01m, 0.52°	0.05m, 0.91°

from the original view.

In Fig. 4, we also show failure cases in 7scenes that present limitations of our pose refinement module. The first example on Fire represents the dependency of our pose refinement module on the bottleneck of coarse pose prediction. Because of few landmarks in the query and sparsity of database, image retrieval (b) selects a far distant database image, and thus coarse prediction module (c) wrongly initializes the query pose. Consequently, our pose refinement (d) cannot converge even after 200 epochs refinement. The second example on Stairs shows the effect of typical scene nature on differentiable renderer. The query (a) includes specular regions that do not appear in database image (b), which highly affects the photometric evaluation in the refinement module. As a potential future work, adapting feature-based loss [15] instead of photometric loss could mitigate such issue derived from the scene and material nature.

Visual localization on unknown scenes. We additionally evaluate our trained model on unknown scenes that are not included in the training set. We choose 12scenes dataset [24] because many state-of-the-arts reported results on the dataset, while trained on its own scenes. Tab. 3 shows the performance of our model trained on 7scenes. Even the feature extractor is trained on different dataset, ours still can achieve comparable results to state-of-the-arts.

Conclusion

This paper presents a visual localization pipeline that is based on a learned image retrieval and relative pose regressor. Instead of determining one single camera by a learned regressor, we keep several camera candidates during the pipeline and verify them to find the most similar camera. Our pipeline employs a new camera pose refinement–verification module that not only evaluates but also refines the candidate with respect to the appearance of local 3D structure seen from the candidate camera. The module based on a differentiable renderer does not require any pre-training since it directly uses the depth map of database images, and suit as a post-processing of learned relative pose regressors. Experiments on a public dataset for visual localization task show that ours can bring a clear improvement compared to baselines, while each component produces a complementary performance gains. We also show that the proposed method can work well on unknown scenes which is not included in the training set. We believe this work raises one potential approach for generalization of recent learned pose regressors.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic, NetVLAD: CNN Architecture for Weakly Supervised Place Recognition, Proc. CVPR, pg. 5297-5307. (2016).

[2] Torsten Sattler, Bastian Leibe, and Leif Kobbelt, Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization, IEEE PAMI, vol. 39, no. 9, pg. 1744–1756. (2017).

[3] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman, Object Retrieval with Large Vocabularies and Fast Spatial Matching, Proc. CVPR, pg. 1-8. (2007).

[4] Akihiko Torii, Hajime Taira, Josef Sivic, Marc Pollefeys, Masatoshi Okutomi, Tomas Pajdla, and Torsten Sattler, Are Large-Scale 3D Models Really Necessary for Accurate Visual Localization?, IEEE PAMI, vol. 43, no.3, pg. 814-829. (2019).

[5] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis, IEEE PAMI, vol. 43, no.4, pg. 1293-1307. (2019).

[6] Alex Kendall, Matthew Grimes, and Roberto Cipolla, Posenet: A Convolutional Network for Real-Time 6-dof Camera Relocalization,

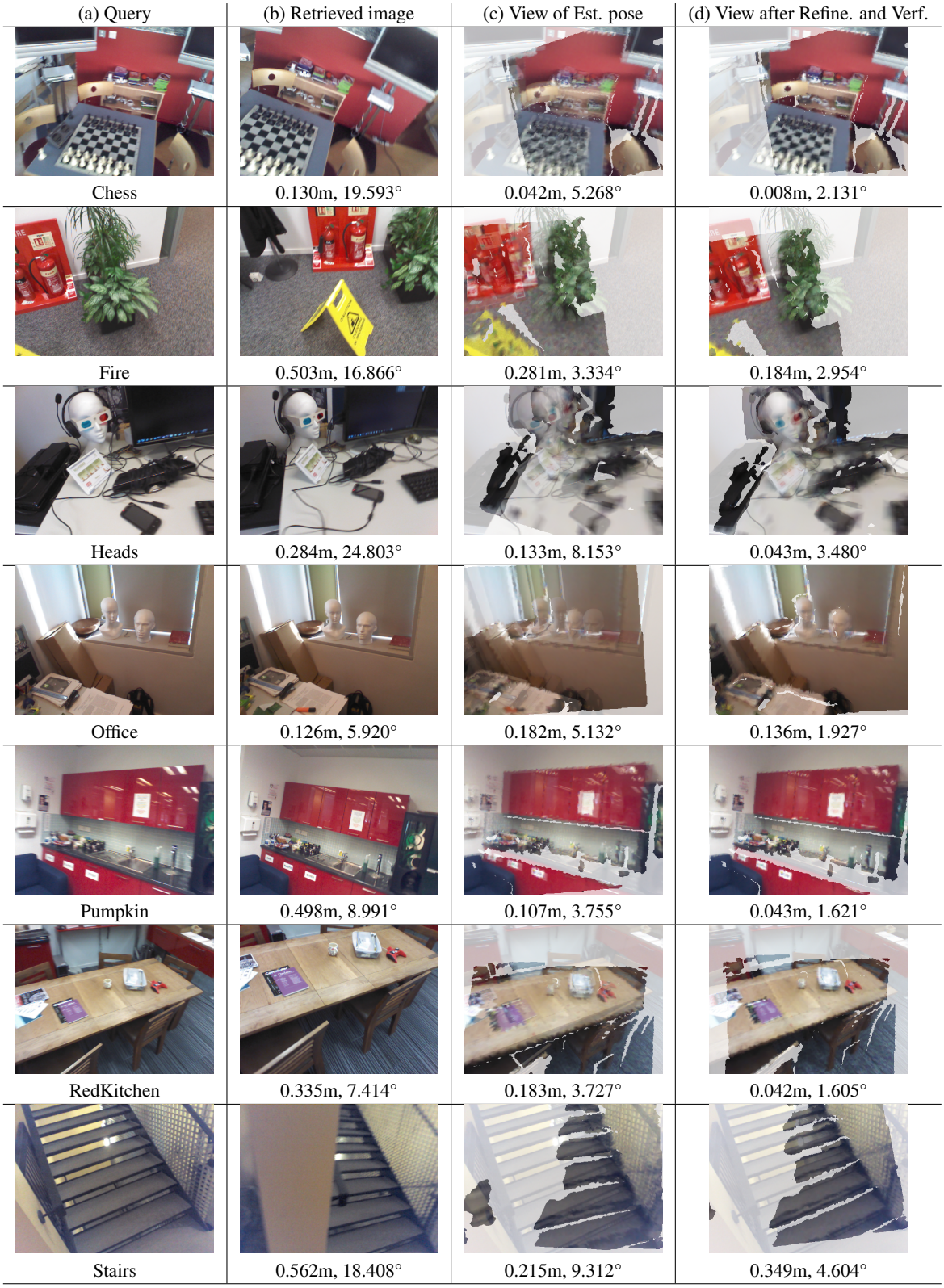

Figure 3. *Qualitative examples. We select a typical example of our visual localization results for each scene of 7scenes. In each row, we show (a) original query image, (b) retrieved database image relevant to query, (c) rendered view after our coarse pose estimation, and (d) the final rendered view after our pose refinement and verification. For each stage of (b)-(d), we report the translational and rotational localization error. In (c) and (d), we additionally overlay the original query image for reference.*

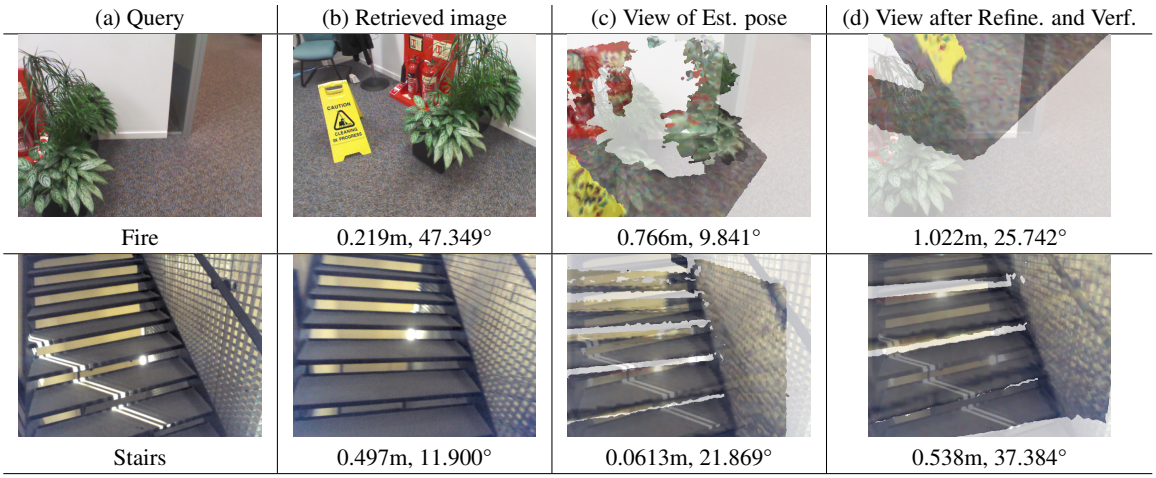

Figure 4. *Failure cases. We show the typical failure cases of our localization pipeline in Fire and Stairs of 7scenes, in the same format as in Fig. 3*

Proc. ICCV, pg. 2938-2946. (2015).

[7] Alex Kendall and Roberto Cipolla, Geometric Loss Functions for Camera Pose Regression with Deep Learning, Proc. CVPR, pg. 5974-5983. (2017).

[8] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz, Geometry-Aware Learning of Maps for Camera Localization, Proc. CVPR, pg. 2616-2625. (2018).

[9] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham, Atloc: Attention Guided Camera Localization, Proc. AAAI, vol. 34, no.6, pg. 10393-10401. (2020).

[10] Vassileios Balntas, Shuda Li, and Victor Prisacariu, Relocnet: Continuous Metric Learning Relocalisation using Neural Nets, Proc. ECCV, pg. 751-767. (2018).

[11] Soham Saha, Girish Varma, and C. V. Jawahar, Improved Visual Relocalization by Discovering Anchor Points, Proc. BMVC, (2018).

[12] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala, Camera Relocalization by Computing Pairwise Relative Poses using Convolutional Neural Network, Proc. ICCV, pg. 929-938. (2017).

[13] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo, CamNet: Coarse-to-Fine Retrieval for Camera Re-Localization, Proc. ICCV, pg. 2871-2880. (2019).

[14] Eric Brachmann and Carsten Rother, Learning Less is More – 6D Camera Localization via 3D Surface Regression, Proc. CVPR, pg. 4654-4662. (2018).

[15] Paul-Edouard Sarlin, Ajaykumar Unagar, Måns Larsson, Hugo Germain, Carl Toft, Victor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, and Torsten Sattler, Back to the Feature: Learning Robust Camera Localization from Pixels to Pose, Proc. CVPR, pg. 3247-3257. (2021).

[16] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, Proc. CVPR, pg. 3302-3312. (2019).

[17] Changhao Chen, Bing Wang, Chris Xiaoxuan Lu, Niki Trigoni, and Andrew Markham, A Survey on Deep Learning for Localization and Mapping: Towards the Age of Spatial Machine Intelligence, arXiv preprint, arXiv:2006.12567. (2020).

[18] Eric Brachmann and Carsten Rother, Expert Sample Consensus Applied to Camera Re-Localization, Proc. ICCV, pg. 7525-7534. (2019).

[19] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada, Neural 3D Mesh Renderer, Proc. CVPR, pg. 3907-3916. (2018).

[20] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng, NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis, Proc. ECCV, pg. 405-421. (2020).

[21] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin, iNeRF: Inverting Neural Radiance Fields for Pose Estimation, Proc. IROS, pg. 1323-1330. (2021).

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, Deep Residual Learning for Image Recognition, Proc. CVPR, pg. 770-778. (2016).

[23] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi, Real-Time RGB-D Camera Relocalization, Proc. ISMAR, pg. 173-179. (2013).

[24] Julien Valentin, Angela Dai, Matthias Nießner, Pushmeet Kohli, Philip Torr, Shahram Izadi, and Cem Keskin, Learning to Navigate the Energy Landscape, Proc. 3DV, pg. 323-332. (2016).

[25] Hunter Blanton, Revisiting Absolute Pose Regression, PhD thesis, University of Kentucky, (2021)

[26] Nathan Piasco, Désiré Sidibé, Cédric Demonceaux, and Valérie Gouet-Brunet, Perspective-n-Learned-Point: Pose Estimation from Relative Depth, Proc. BMVC, (2019).

Author Biography

Chanchang Tsai *is a researcher graduated from Okutomi & Tanaka lab, Tokyo Institute of Technology. His research interests include camera localization and machine learning. He received his bachelor's degree from Zhejiang University in 2019 and his master's degree from Tokyo Institute of Technology in 2022.*

Hajime Taira

Masatoshi Okutomi *received the B.E. degree from The University of Tokyo in 1981 and the M.E. degree from Tokyo Institute of Technology in 1983. He joined Canon Research Center in 1983. From 1987 to 1990, he was a Visiting Research Scientist with the School of Computer Science, Carnegie Mellon University. He received the PhD degree by dissertation from Tokyo Institute of Technology in 1993. Since 1994, he has been with Tokyo Institute of Technology, where he is currently the Professor with the Department of Systems and Control Engineering.*