MegLoc: A Robust and Accurate Visual Localization Pipeline

Abstract

*This paper introduces a visual localization pipeline called **MegLoc**, used to obtain robust and accurate 6 D.O.F. pose estimation under challenging conditions and various scenarios, including indoor scenes and outdoor scenes, different times across the day, different seasons across the year, and even across many years.*

MegLoc can be divided into two stages: mapping and localization. We will elaborate on these two stages respectively below.

1. Mapping

Mapping includes five steps: image preprocessing, feature extraction, matching, sparse reconstruction, and map refinement.

1.1. Image Preprocessing

Each image was resized to make the largest dimension 1600 pixels while retaining its original aspect ratio. Some pixels of the bottom-right border might be cropped, to ensure each dimension is a multiple of 8.

Since the bottom part of the camera view was always occluded by the car shell, local features in that area impede subsequent feature matching and triangulation. We removed local features in that area by masking.

1.2. Feature Extraction

SuperPoint[1] was selected as our local feature extractor. The Non-Maximum Suppression radius and keypoints threshold were tuned empirically based on the experiments conducted in the CVPR2019 Image Matching Challenge[2]. As a result, approximately 1500 keypoints were extracted for each image on average.

1.3. Matching

The matching step provided sparse 2D keypoints correspondences between images that share some covisible viewing areas. Those correspondences are the key to recover 3D map point positions by triangulation.

Image Retrieval Before we found sparse correspondences by local feature matching, database images were retrieved to obtain the top-K nearest image pairs, between which a large number of local correspondences could be established potentially. There are two ways to find image-level correspondences, by global feature similarity or utilizing the temporal characteristics of the query sequences.

Sparse Correspondences SuperGlue[3] was implemented for local feature matching. After we obtained several local correspondences, a coarse pose was estimated and those correspondences conflicting with epipolar constraint were eliminated.

1.4. Sparse Reconstruction by Triangulation

We used Colmap Toolbox[4][5] to reconstruct the map sparsely. Global bundle adjustment was performed once only to save computations.

1.5. Map Refinement

In the 4Seasons dataset[6], the vehicle always traverses forward. The baseline in the x-y direction of the camera coordinate system is relatively narrow, which causes the triangulated 3D map points suffering from a large uncertainty in the z-direction.

We formulate the uncertainty of a 3D map point as follow:

A 3D map point p_w in the world frame can be observed in multiple cameras $C_i (i = 1...N)$ from different views. The camera pose C_i and its intrinsic parameters can be represented as $[R_{wc_i}, t_{wc_i}]$ and $[f_x, f_y, c_x, c_y]$, respectively. Supposing a 3D map point $P_c = [x, y, z]$ with respect to the camera reference frame corresponds to a observation in the image plane of camera C_i, and its 2D location can be represented as p_{uv}

The Jacobian of 2D observation dp_{uv} to the 3D map point dp_w is

$$J_i = \frac{dp_{uv}}{dp_w} = \frac{\frac{dp_{uv}}{dp_c} \cdot dp_c}{dp_w} = \begin{bmatrix} \frac{f_x}{z} & 0 & -\frac{x \cdot f_x}{z^2} \\ 0 & \frac{f_y}{z} & -\frac{y \cdot f_y}{z^2} \end{bmatrix} \cdot R_{wc_i} \quad (1)$$

Assuming the observation uncertainty of each pixel on the image plane is an identity matrix, i.e. $\Sigma_{uv} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

The information matrix of 3D point p_w is

$$\Sigma_{w_i}^{-1} = J_i^T * \Sigma_{uv}^{-1} * J_i \quad (2)$$

The total uncertainty is an addition of the certainty of all observations, as

$$\Sigma_w^{-1} = \Sigma_{w_1}^{-1} + \Sigma_{w_2}^{-1} + ... \quad (3)$$

The uncertainty of three orthogonal directions can be calculated by eigen-decomposition, and those map points

with large uncertainty can be removed from the map by setting a threshold. The value of the threshold should be determined based on many factors, such as the scale of the reconstructed scene, the number of mapping images, resolution of per image, etc.

Fig, 1 shows the effect of pose refinement.

2. Localization

Localization consists of four steps: image preprocessing, feature extraction, matching, pose estimation by rig Perspective-N-Points and pose refinement.

The image preprocessing and feature extraction steps are the same as those in the mapping section. We also used Superglue to find local correspondences. Global feature similarities between query images and database(mapping) images were evaluated by cosine function and the top-K nearest pairs were retrieved. Query images are also temporally sequential and thus we also utilized this characteristic to retrieve more image pairs.

2.1. Pose Estimation by Rig Perspective-N-Points

The 4Seasons dataset was recorded by a stereo camera with the given extrinsic parameters (i.e. transformation between two cameras). During the pose estimation step, we treated the images captured at the same timestamp as a whole, and only solved the pose of the left camera. We figured out this strategy advanced accuracy and robustness.

2.2. Pose Refinement

Once all the poses of the query images were estimated, we found some of them localized unsuccessfully. To tackle it, we further considered the temporal characteristics of the query images. We first assumed that there must exist some covisible parts between consecutive images. Hence some local correspondences could be established among sequential query images.

While fixing the camera poses of all mapping images and meanwhile involving those query-to-query correspondences into the map by triangulation, we ran a global optimization on camera poses of all query images. The effectiveness of outliers rejection by query-to-query correspondences can be demonstrated by Fig.2.

References

[1] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," *CoRR*, vol. abs/1712.07629, 2017. [Online]. Available: http://arxiv.org/abs/1712.07629

[2] X. Bi, Y. Chen, X. Liu, D. Zhang, R. Yan, Z. Chai, H. Zhang, and X. Liu, "Method towards CVPR 2021 image matching challenge," *CoRR*, vol. abs/2108.04453, 2021. [Online]. Available: https://arxiv.org/abs/2108.04453

[3] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *CVPR*, 2020. [Online]. Available: https://arxiv.org/abs/1911.11763

[4] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[6] P. Wenzel, R. Wang, N. Yang, Q. Cheng, Q. Khan, L. von Stumberg, N. Zeller, and D. Cremers, "4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving," in *Proceedings of the German Conference on Pattern Recognition (GCPR)*, 2020.