Journal of Computational Design and Engineering, 2022, 9(3), 1097–1106

https://doi.org/10.1093/jcde/qwac046
Journal homepage: www.jcde.org
Advance Access Publication Date: 12 May 2022

RESEARCH ARTICLE

Learning-based essential matrix estimation for visual localization

Moongu Son and Kwanghee Ko *

School of Mechanical Engineering, Gwangju Institute of Science and Technology, Cheomdangwagi-ro 123, Buk-gu, Gwangju 61005, Republic of Korea

*Corresponding author. E-mail: khko@gist.ac.kr https://orcid.org/0000-0001-7668-5796

Abstract

Visual localization is defined as finding the camera pose from two-dimensional images, which is a core technique in many computer vision tasks, including robot navigation, autonomous driving, augmented/mixed/virtual reality, mapping, etc. In this study, we address the pose estimation problem from a single-color image using a neural network. We propose a coarse-to-fine approach based on a deep learning framework, which consists of two steps: direct regression-based coarse pose estimation that obtains a pose by finding a pose-based similar image retrieval and Siamese network-based essential matrix estimation to obtain a refined pose. Experimental results using the 7-scenes, Cambridge, and RobotCar datasets demonstrate that the proposed method performs better than the existing methods in terms of accuracy and stability.

Keywords: visual localization; 6-DoF pose estimation; AI and machine learning

1. Introduction

Visual localization is a method of estimating the 6-DoF (Degree of Freedom) camera pose from a query image. It is a key technique utilized in computer vision-based tasks, such as augmented reality, simultaneous localization and mapping, and indoor navigation. Traditional feature-based approaches acquire image features through feature detection and description algorithms. They compute the local features of two-dimensional (2D) images and match them to create 3D scene structure features. When the 2D–3D matching pairs are assigned, a perspective-n-point solver is used to compute the camera pose. Mostly, these approaches use well-known hand-crafted feature detection algorithms, such as SIFT (Scale Invariant Feature Transform) by Lowe (2004), SURF (Speeded Up Robust Features) by Bay *et al.* (2006), and ORB (Oriented FAST and Rotated BRIEF) by Rublee *et al.* (2011). The problem with these methods is that such features are not robust when texture-less scenes are given, illumination changes drastically, and occlusions and repetitive structures exist in the scenes. In such cases, a serious feature mismatch occurs, causing significant errors in the pose estimation.

Recently, the success of the deep learning framework in computer vision has motivated researchers to use a deep learning network to estimate the camera pose from a single image. PoseNet by Kendall *et al.* (2015) is an end-to-end network that directly estimates the 6-DoF camera pose from a single RGB (Red Green Blue) image. It modifies InceptionV3 and replaces the classification part with fully connected layers to regress the camera poses. Many studies have been conducted to improve the performance of PoseNet. Namely, MapNet by Brahmbhatt *et al.* (2018) enforces geometric constraints by minimizing the total loss consisting of the relative camera pose loss between image pairs and direct pose loss. LSTM PoseNets by Walch *et al.* (2017) replaces the fully connected layers with LSTM (Long Short-Term Memory) units for structured feature correlation before the final direct camera pose regression. Hourglass PoseNets by Melekhov *et al.* (2017) integrates hourglass architecture layers with shortcut connections to the direct pose regression network. These

Received: 29 December 2021; **Revised:** 5 April 2022; **Accepted:** 28 April 2022

Table 1: The pros and cons of the proposed and existing methods.

Category	RGB	3D	Intrinsic parameter	Generalized	Processing time	Accuracy
Structure based	O	(O)	(O)	No	Slow	High
Direct pose regression	O	–	–	No	Fast	Low
Relative pose regression	O	–	–	Yes	Fast	Low
Ours	O	–	–	Yes	Fast	Medium

methods are effective for featureless environments, where most traditional methods do not work well. Learning-based pose regression methods are more robust compared to the traditional methods for images with viewport changes, repeated structures, weak textures, and blurred scenes. However, they are generally less accurate and difficult to generalize when the input scenes are not related to the training dataset.

Unlike traditional feature-based methods, some methods estimate the 3D coordinates of each pixel from a query image and the scene's world coordinate, which are called the structure-based methods. Shotton *et al.* (2013) demonstrated that a scene coordinate regression forest on RGBD (Red Green Blue Depth) features mapped from image locations to the corresponding scene coordinates can be trained directly. DSAC (Differentiable RANSAC), Brachmann *et al.* (2017), and DSAC++, Brachmann and Rother (2018) replaced the conventional RANSAC (RANdom SAmple Consensus)-based n-point solver with a novel differentiable RANSAC method to enable end-to-end training of the structure-based method pipeline. The structure-based methods work with high robustness and accuracy on small indoor scenes. However, they are not robust for large-scale outdoor scenes because of local and global ambiguities (Schönberger *et al.*, 2018).

In addition, most structure-based methods do not provide real-time performance. They rely on depth information or 3D models, which are memory-intensive and time-intensive to construct, and require inputs for training. Therefore, they require a significant amount of time for training and inference. Furthermore, they cannot handle unseen scenes as effectively as direct pose regression methods.

Li *et al.* (2020) introduced HSC-Net, a hierarchical joint learning framework to estimate scene coordinates and to eliminate environmental ambiguities. Brachmann and Rother (2021) proposed DSAC$*$, a two-stage approach consisting of scene coordinate regression and differentiable pose estimation. It uses RGB images and poses without 3D models for training. However, it cannot be generalized to unseen scenes.

While the direct pose regression-based methods regress the camera pose in the global scene coordinate, relative pose regression-based methods take two images as input and estimate the relative pose between them. They need a reference regarding the global scene coordinates. Thus, they use an image retrieval method to obtain reference information. The image retrieval method typically builds a large image dataset with known camera poses, retrieves the most similar image in the database, and obtains the camera pose of the retrieved image. It provides a rough estimate, which is often used for general place recognition. NetVLAD (Vector of Locally Aggregated Descriptors) (Arandjelovic *et al.*, 2018) transformed the existing hand-crafted feature algorithm, VLAD, into a learnable feature-based deep learning approach with a triplet ranking loss adapted to weakly labeled data. NN-Net by Laskar *et al.* (2017) combined pairwise relative pose estimation and nearest neighbor retrieval between the query and top N ranked references with pose hypothesis filtering and evaluated unseen scene performance through experiments with 7Scenes dataset. Six of the scenes are taken for training, and the remaining scene is used for test. RP-Net by En *et al.* (2019) utilized a sequence-based image retrieval approach to estimate relative poses using an end-to-end trained neural network with various inference methods. Bai *et al.* (2018) combined CNN (Convolutional Neural Network) features and a sequence matching method to improve the accuracy when viewpoints and conditions change simultaneously. RelocNet (Balntas *et al.*, 2018) introduced a network that can retrieve the nearest pose neighbors using a frustum overlap. CamNet by Ding *et al.* (2019) applied a three-stage coarse-to-fine retrieval approach that shares one encoder network throughout the overall pipeline. The image retrieval method can handle unseen scenes. However, most image retrieval methods suffer from low accuracy in direct pose regression. Nevertheless, these methods are scalable and robust against varying conditions.

In this study, we propose a visual localization method that integrates direct pose regression, pose-based image retrieval, and essential matrix estimation for pose refinement in one framework. We regress the initial pose using a pose regression. Then, we find the nearest image in the database and compute a relative pose estimated by an essential matrix estimation network for pose refinement. The overall comparison of the proposed method with the existing methods is made. The pros and cons of the methods are summarized in Table 1. The table shows that the proposed method takes only RGB images and does not use 3D models. It does not require intrinsic parameters of the camera to work, which is more convenient than the others. Moreover, it can be generalized to unseen scenes and estimates the poses fast. The estimation accuracy is higher than direct and relative pose regression methods, while it is inferior to the structure-based method.

2. Overview of the Approach

The overall process of the proposed framework is presented in Fig. 1, which is a retrieval-based coarse-to-fine framework with deep neural networks. The framework consists of two modules: a direct pose regression module for coarse retrieval and an essential matrix estimation module for relative pose refinement. The direct pose regression model is based on a single-stream architecture with an RGB query image as the input. The essential matrix estimation network is based on a Siamese architecture using the image pair of the query image and associated image in the training dataset.

We first estimate the rough pose of a query image using the direct pose regression method. Next, we locate the image with the closest pose to the estimated rough pose of the query image in the training dataset. Once we retrieve the image from the training dataset, we estimate an essential matrix with a scale between the query and retrieved images. We use the estimated

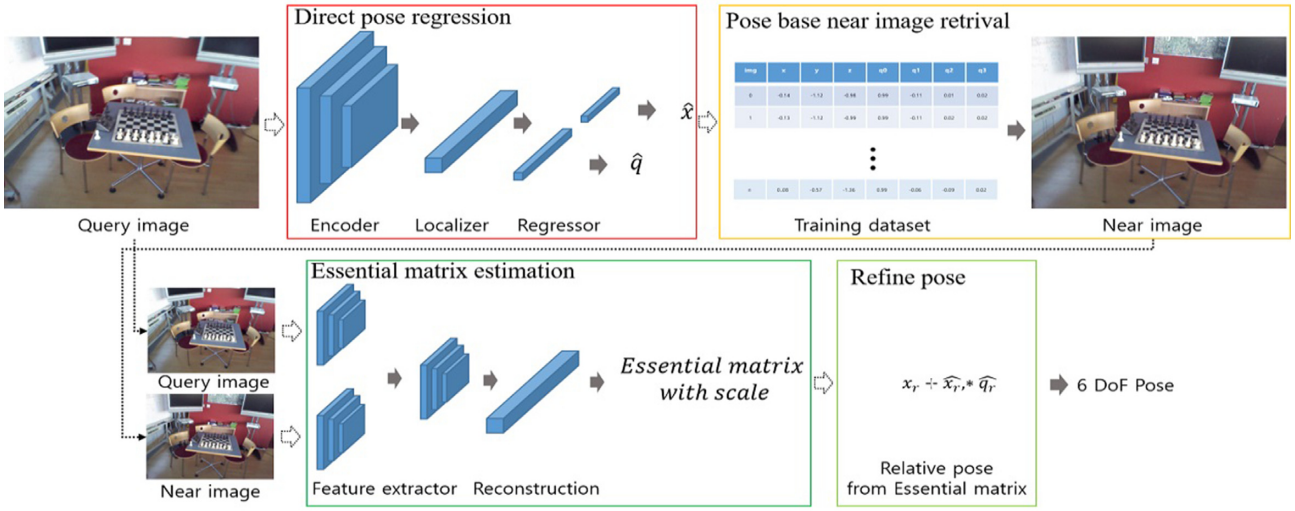

Figure 1: Overall process of the proposed framework.

essential matrix to compute the relative pose. Finally, we estimate an accurate camera pose through pose refinement.

Sections 3–4 describe the proposed approach in detail. Section 5 presents the experimental results. Finally, section 6 reports the conclusions of our study.

3. Direct Pose Estimation for Coarse Retrieval

3.1. Direct pose regression

The direct pose regression step estimates the pose of a query image directly using a neural network. For the backbone network, we used ResNet-34 pre-trained on the ImageNet dataset. The classification part of the network was replaced with fully connected layers for pose regression. As shown at the top of Fig. 1, we trained the network using a training database. The network processes and RGB query image I_t were used to predict the camera pose relative to the scene global coordinate system. We took two measurements of position and rotation:

$$L_x(I_t) := \|x_t - \hat{x}_t\|_\gamma, \tag{1}$$

$$L_q(I_t) := \|q_t - \hat{q}_t\|_\gamma, \tag{2}$$

where x_t and q_t are the ground-truth camera position and rotation components, respectively, $\hat{x}_t$ and $\hat{q}_t$ are the estimated camera position and rotation, respectively, and γ refers to the L^γ-norm. In this study, we use the L^2 Euclidean norm. The loss function is given as a combination of equations (1) and (2) defined by Kendall and Cipolla (2017) as the following:

$$L_s(I_t) := L_x(I_t)\exp(-\hat{s}_x) + \hat{s}_x + L_q(I_t)\exp(-\hat{s}_q) + \hat{s}_q, \tag{3}$$

where $\hat{s}_x$ and $\hat{s}_q$ are the two learnable parameters that balance the scale difference between the position and rotation in the loss function.

The direct pose regression method can always provide an initial pose, although not satisfactorily accurate for practical applications. However, it is more stable than traditional approaches that fail to function when texture-less images or images containing repetitive structures are provided. The estimated rough pose is used to obtain an image that is close to the given query image.

3.2. Pose-based near image retrieval

The image retrieval step locates the most similar image in the dataset to the query image. A method using bag-of-words developed by Csurka *et al.* (2004) may lead to significant memory consumption and fail to work in some cases. A network-based image retrieval method requires offline training, which is time-intensive and memory-intensive. Therefore, we use a pose-based image search method to obtain a "near image" from the training dataset.

The near image search step is performed when the following conditions are satisfied:

$$X = \| x_r - \hat{x}_r \|, \tag{4}$$

$$Q = \|2 * \arccos(conj(q_r, \hat{q}_r))\|, \tag{5}$$

$$L = \min(X + \beta Q), \tag{6}$$

where x_r and q_r are the reference translation and rotation components, respectively, $\hat{x}_r$ and $\hat{q}_r$ denote the estimated position and rotation components, respectively, and β is a scale balancing factor (e.g. $\beta = 1$ for indoor scenes.). Here, the image that yields the minimum value for equation (6) is the near image. The translation represents the distance error in 3D space. The rotation represents the angle difference between the two quaternions. Note that the query and near images sufficiently overlap in the camera frustum in most cases, namely, more than 50%, as presented in Fig. 2, which is an essential condition for relative pose calculation.

4. Essential Matrix Estimation for Fine Pose Refinement

4.1. Essential matrix estimation with respect to a scale

The relative pose between two images can be trained using a Siamese network using the following weighted loss function:

$$L_r(I, I^*) := \|x_r - x_r^*\|_2 + \beta\|q_r - q_r^*\|_2, \tag{7}$$

where $I = (x_r, q_r)$ denotes the true relative pose between an image pair and $I^* = (x_r^*, q_r^*)$ denotes the estimated relative pose. The hyper-parameter β plays a similar role in the pose-based

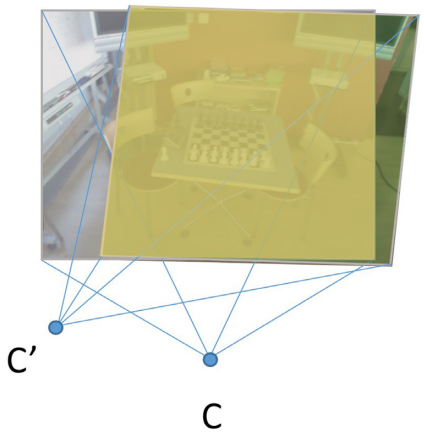

Figure 2: Camera frustrum overlap between query and near image. Here, C and C' denote query and near image camera poses, respectively.

near image retrieval process. Namely, β is needed because rotation (degrees) and translation (meters) are expressed in different units. The effect of this difference is not significant for indoor scenes where the position and rotation do not change significantly. However, it can be critical for outdoor scenes where the difference between the two camera positions is large because it is difficult to obtain a suitable weighting factor. Such a problem can be eliminated by an essential matrix regression, which implicitly defines the weights between the orthogonal rotation matrix and unit norm translation vector. However, the unit norm transform vector can only estimate the direction without a scale factor. Therefore, by adding a scale factor to the final regression step of the network, the 2-DoF translation vector is corrected with a scale. The detailed network structure is presented in Fig. 3.

We use a Siamese neural network based on deep fundamental matrix estimation without correspondences (Poursaeed *et al.*, 2019). Using image pairs as input, we train the entire network to regress the essential matrix directly. It comprises two feature extraction networks and a reconstruction network. The feature extractor networks are based on the universal correspondence network (Choy *et al.*, 2016), with the spatial transformers part removed (Jaderberg *et al.*, 2015). They are then concatenated into two layers. Subsequently, six parameters (t_x, t_y, t_z, r_x, r_y, and r_z) are mapped using the loss function presented in equation (7). The essential matrix is then reconstructed using equations (8–10) with these six parameters. The scale factor is calculated using equation (11). Moreover, the reconstruction layer regresses an essential matrix E with a scale factor, using equations (12) and (13).

The multiple loss function L_T contains the hyper-parameter beta in L_F. However, the essential matrix is reconstructed using equation (12) that does not contain the parameter, leading to the reduced effect of the parameter in the computation. This makes the proposed method different from the relative pose regression approach. Here, L_F is used for the process to improve the convergence of the essential matrix estimation learning stage.

$$L_F = \|t_X - \widehat{t_X}\|_2 + \beta\|R - \hat{R}\|_2 \tag{7}$$

$$[t]_X = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \tag{8}$$

$$R = R_x(r_x)\,R_y(r_y)\,R_z(r_z) \tag{9}$$

$$E = [t]_X R \tag{10}$$

$$s = \sqrt{t_x^2 + t_y^2 + t_z^2} \tag{11}$$

$$L_E = \|e - \hat{e}\|_2 \tag{12}$$

$$L_T = L_F + L_E \tag{13}$$

Here, $e \in R^{10}$ is the vectorized $E \in R^{3\times3}$ with a scale factor and $\hat{e}$ is the vectorized predicted essential matrix with a scale factor. The elements in e are the elements of the essential matrix. The first element in the essential matrix is e_0. The last element in the matrix is e_8. From equation (10), the multiplication of the translation vector and the rotation matrix produces the 3 by 3 essential matrix that corresponds to the nine elements in e. The scale s is computed using the translation vector components by equation (11).

In the training stage, training image pairs are selected, which have translation and angle differences within 0.5 m and 5°, respectively, for indoor scenes and 5 m and 5°, respectively, for outdoor scenes. Under these conditions, the selected image pairs can have overlapping frustums that are sufficient for pose refinement.

Unlike the direct pose regression, which is a scene-specific approach, the relative pose regression can be generalized to unseen scenes. However, the relative pose regression approach has the disadvantage of requiring scene-dependent hyperparameters. Direct regression of the essential matrix can solve the scene-dependent parameter problem. However, the essential matrix for a 2-DoF translation **t** is normalized to scale 1, which denotes the unit direction. A scale factor should be estimated to de-normalize the translation vector. In this study, we overcome these problems by estimating an essential matrix with a scale factor, using a deep learning network.

4.2. Pose refinement

The computed essential matrix containing the relative pose between the query and retrieved images is decomposed into a normalized translation vector $\tilde{t}$ and a rotation matrix R. There are four possible candidate poses: *(R, $\tilde{t}$), (R, $-\tilde{t}$), (R', $\tilde{t}$), and (R', $-\tilde{t}$)*, where, R' denotes the opposite of R. The correct translation and rotation pairs among the four candidates can be found using the chirality test. However, directly regressing the relative pose does not include matching information used for the chirality test. Therefore, the known reference pose of the dataset image is used to select candidates that match the query image. The directions of the rough and reference poses determine the direction of $\tilde{t}$ or $\widetilde{-t}$. Similarly, the minimum angle difference between the reference and candidate rotations determines R *or* R'. Additionally, $\tilde{t}$ is given as a normalized vector. Therefore, we use a scale factor s to estimate the actual translation with respect to the reference position. To correct the actual pose, a reference pose is necessary, which is obtained from the near image present in the training database. Then, we compute the refined pose by adding the relative pose to the reference. Figure 4 depicts the relationship between the four candidates, reference pose of the near image, and rough pose of the query image.

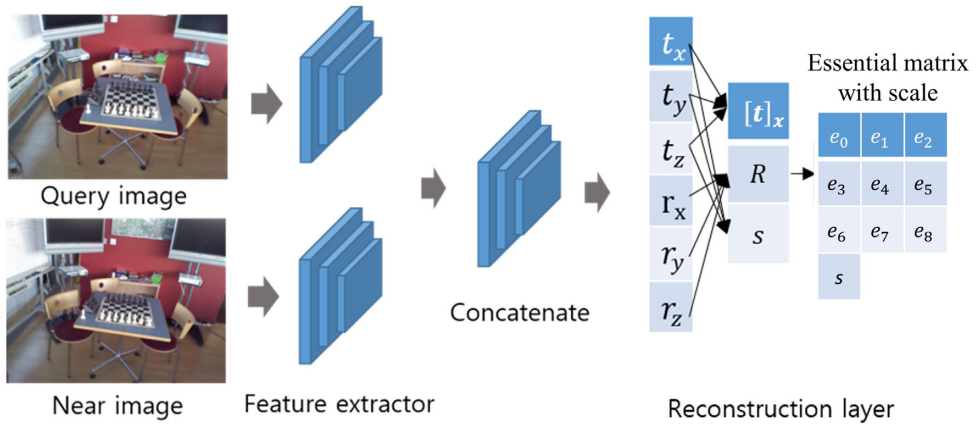

Figure 3: Essential matrix estimation network structure.

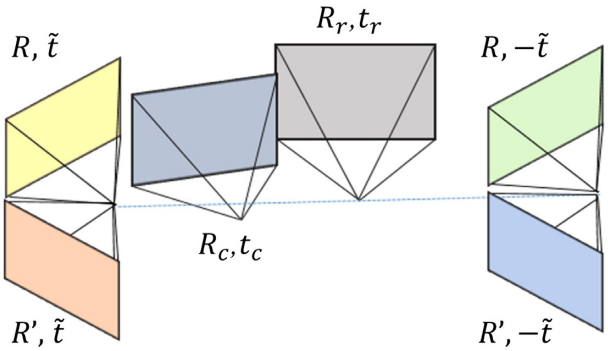

Figure 4: Essential matrix decomposed into the four candidates. Here, the pair (R_r, t_r) denotes a reference pose and (R_c, t_c) denotes a rough pose.

5. Experiment and Results

The proposed method is tested using various datasets and compared to other approaches.

5.1. Datasets

Three benchmark datasets, 7-scenes (Shotton *et al.*, 2013), Cambridge (Kendall *et al.*, 2015), and Oxford RobotCar dataset (Maddern *et al.*, 2017), were used in the experiments. In the 7-scenes dataset, seven indoor environments were measured with a 640 × 480 resolution handheld Kinect RGB-D device. The ground-truth camera poses were generated using the KinectFusion method (Newcombe *et al.*, 2011). Multiple sequences were recorded inside one indoor environment, which were split into training and test sequence sets of 500 or 1000 images. The Cambridge dataset contains images of large-scale outdoor environments taken around the Cambridge University. The dataset is divided into training and test sequence sets, where each sequence contains several hundred images. The ground-truth camera poses were generated using structure-from-motion techniques. The Oxford RobotCar dataset, again, contains images of large-scale outdoor environments. The images were repeatedly captured following a consistent route over a long period of time and documented various environmental changes. The ground-truth camera poses were generated by combining heterogeneous sensors, such as LiDAR, GPS, and camera. We used the LOOP subset of the dataset.

5.2. Experiments on the 7-scenes dataset

We evaluated the proposed method against the state-of-the-art techniques on the 7-scenes dataset. The results are summarized in Table 2. ActiveSearch (Sattler *et al.*, 2017), which is a SIFT-based localization method, obtains the best accuracy but often fails during localization with texture-less scenes because the number of correspondence features was insufficient. The number of images in which the method failed during localization, is indicated in parentheses. ActiveSearch needs to build a scene-by-scene 3D model as well. In contrast, the proposed method works with single RGB query images without needing a 3D model. DSAC*, proposed by Brachmann and Rother (2021) is the state-of-the-art structure-based method that shows the best performance. It uses RGB images without any 3D models. However, it requires camera intrinsic properties to initialize a scene coordinate. Its average processing time is about 30 ms, whereas the proposed method takes about 5 ms. Here, the camera translation and rotation errors are given in meters (m) and degree angles (°), respectively. The median localization errors of the proposed method on the 7-scenes dataset are 0.10 m and 5.47°. The

Table 2: The experimental results of the methods on the 7-scenes dataset (m/°).

Scene	PoseNet	MapNet	NN-Net	RelocNet	ActiveSearch	DSAC* (RGB)	Ours
Chess	0.13/4.48	0.08/3.25	0.13/6.46	0.12/4.14	0.04/1.96	0.02/0.6	0.05/2.52
Fire	0.27/11.3	0.27/11.69	0.26/12.72	0.26/10.40	0.03/1.53(1)	0.02/0.9	0.10/8.81
Heads	0.17/13.0	0.18/13.25	0.14/12.34	0.14/10.50	0.02/1.45(1)	0.01/0.7	0.09/7.69
Office	0.19/5.55	0.17/5.15	0.21/7.35	0.18/5.32	0.09/3.61(34)	0.03/0.8	0.10/4.84
Pumpkin	0.26/4.75	0.22/4.02	0.24/6.35	0.26/4.17	0.08/3.10(68)	0.04/1.1	0.12/3.65
Red kitchen	0.23/5.35	0.23/4.93	0.24/8.03	0.23/5.08	0.07/3.37	0.04/1.3	0.13/4.10
Stairs	0.35/12.4	0.30/12.08	0.27/11.82	0.28/7.53	0.03/2.22	0.04/1.2	0.13/6.68

Figure 5: Cases in which ActiveSearch fails to estimate the poses.

proposed method succeeds in all cases with comparable performance, whereas ActiveSearch often fails to obtain the camera pose. ActiveSearch uses RootSIFT features to establish 2D–3D matches. It is based on prioritized matching, which terminates the correspondence search once 200 matches have been found. The pose is estimated via a PnP (Perspective-n-Point) solver inside a RANSAC loop, followed by non-linear refinement of the pose. The 3D model required by ActiveSearch is built by matching each training image against the nearby training images and triangulating the resulting matches using the provided training poses. Therefore, ActiveSearch fails or produces inaccurate pose estimates when there is little or no visual overlap between the test and training images. It also fails to estimate pose when no or little matching results are obtained if the images are blurred by a sudden motion or contain bright regions by a bright light source, as shown in Fig. 5. However, the proposed method can handle such images successfully with a faster computation performance. Fig. 6 shows pose estimation results on the 7-scenes chess dataset compare to the PoseNet and MapNet.

5.3. Experiments on the Cambridge dataset

The Cambridge dataset covers large areas of outdoor environments. It is larger compared to the 7-scenes dataset but contains less images per volume compared to the 7-scenes dataset. Each sequence has several hundred frames. Therefore, the experiment on the Cambridge dataset presents the influence of relative posture estimation through essential metrics. We compared our approach to PoseNet, MapNet, and RPNet methods using the Cambridge dataset. As reported in Table 3, our model performs better than the other methods. Observe that, compared to RPNet, our method can achieve a more accurate pose estimation because it reduces the influence of the balancing parameters on outdoor scenes.

5.4. Experiments on the Oxford RobotCar dataset

The structure-based methods, DSAC and DSAC++, cannot handle large outdoor scenes (Lee *et al.*, 2021). The Cambridge landmark dataset covers several tens or hundreds of meters. On the

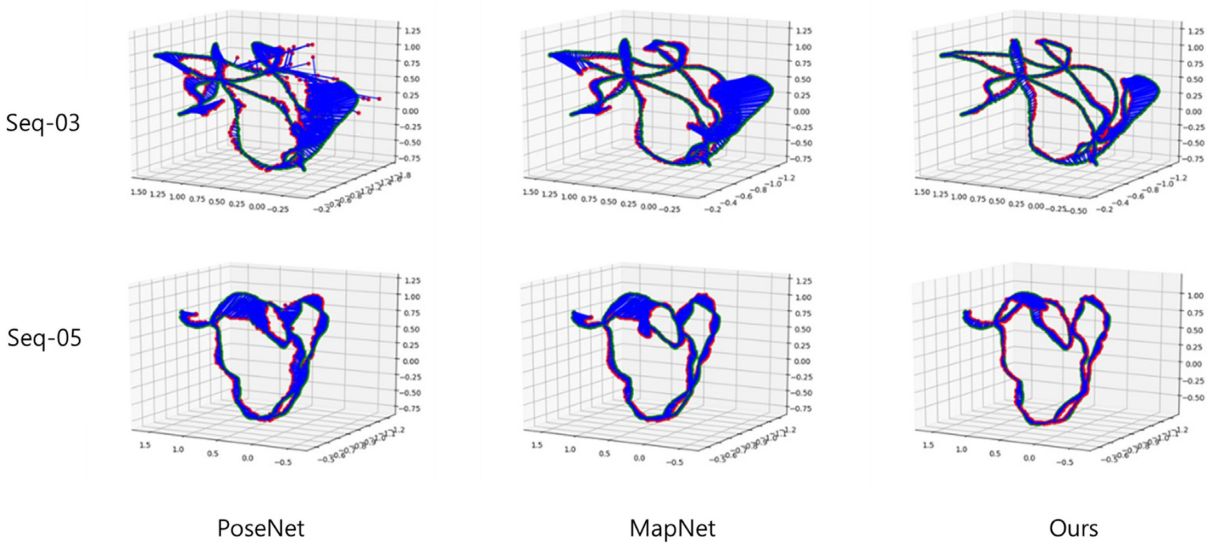

Figure 6: Pose estimation results on the chess dataset (green: GT, red: estimated pose, and blue: distance).

Table 3: The experimental results of the methods on the Cambridge dataset (m/°).

Scene	PoseNet	MapNet	RPNet	Ours
Great Court	4.78/6.10	7.85/3.76	–	**2.15/3.42**
Kings College	5.97/9.24	1.07/1.89	1.93/3.12	**0.96/1.58**
Old Hospital	1.59/5.13	1.94/3.91	2.41/4.81	**1.51/3.86**
Shop Facade	1.05/6.19	1.49/4.22	1.68/7.07	**1.03/3.99**
St Marys Church	**1.32**/6.78	2.00/4.53	2.29/5.90	1.45/**3.35**

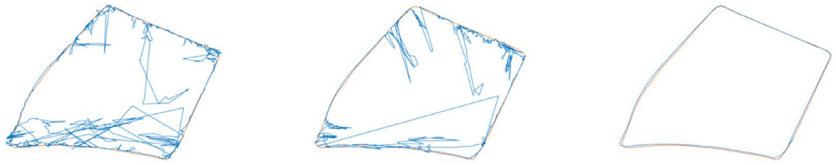

Figure 7: Pose estimation results on the Loop dataset.

Table 4: The experimental results of the methods on the Oxford RobotCar dataset (m/°).

Scene	DSAC++	PoseNet	MapNet	Ours
Loop	N/A	25.29/17.45	9.84/3.96	**2.15/3.71**

other hand, the Oxford RobotCar is an outdoor environment where the length of the loop is about 1120 m, the range of which is much larger than that of the Cambridge landmark dataset. In the case of the Loop dataset, DSAC++ learning is not possible due to a memory problem because it requires the entire 3D scene coordinates that must be reconstructed from the dataset, which may take an enormous amount of memory depending on the scene. Lee *et al.* (2021) also reported that the method failed in the experiments. DSAC++ has the Cambridge landmark dataset results. However, it required more information such as intrinsic parameters than RGB images for training, which means that it does not belong to the category of the methods that use RGB images only. Therefore, we do not consider DSAC++ in the comparison.

We compared our model to MapNet and PoseNet on the Loop dataset. PoseNet and MapNet resulted in position and direction errors of 25.29 m and 17.45°, and 9.84 m and 3.96°, respectively. The proposed method resulted in position and direction errors of 2.15 m and 3.71°, respectively, which underlines that it is more accurate compared to both PoseNet and MapNet, as depicted in Fig. 7. and Table 4.

The proposed method shows similar or inferior performance to the existing methods with a few datasets. For example, as shown in Table 3, the proposed method shows about 10% less accurate estimation result than PoseNet with the St Marys Church dataset, one scene of the Cambridge dataset. However, for others, the proposed method works better. The same result is obtained from the experiments with the Oxford RobotCar dataset, which shows much better accuracy than the others by about 78%, as shown in Table 4. For the 7-scenes dataset, the proposed method shows better accuracy than the others except for the structure-based methods (ActiveSearch and DSAC∗) by about 40–60%. The structure-based methods show higher accuracy than the proposed method. However, they require 3D models and additional parameter values in addition to RGB images, and they need more computation time to estimate the pose than the proposed method.

5.5. Pose-based retrieval results

In this section, we evaluate the pose-based near-image method of the retrieval stage in the pipeline, which returns images in the dataset related to a query image. Traditional image retrieval methods locate the image most similar to the query image from a dataset. The similarity between the two images is computed as follows: Images are converted into vectors (descriptors) using VLAD or bag-of-visual-words. Next, the Euclidean distances between them are calculated. We experimentally compared the performance of our method to that of NetVLAD, an advanced image retrieval method, with respect to seven scenarios. We identify the candidate images in the dataset using each method, given a target image. Then, we locate the reference image, the nearest one to the target image from the pose data present in the dataset, and calculate the error between the candidates and reference images. Our method relies on the direct pose regression method; therefore, it cannot always locate the most similar image. However, when there are enough datasets in a small volume, such as the 7-scenes dataset, the results are similar to or better than those of NetVLAD. This shows that the proposed method is sufficiently effective in relative pose refinement. The images obtained using NetVLAD and proposed method are presented in Fig. 8 when the target images are given as query images. Table 5 showes our pose-based retrieval results on 7Scenes dataset. The proposed method can retrieve images closer to the target images than those obtained by the NetVLAD method.

5.6. Generalization performance

The direct pose regression and structure-based methods have limitations on unseen scenes. In contrast, relative pose regression can be generalized to handle unseen scenes. In this section, we experiment on the generalization performance of the proposed method and compare it to NN-Net and DSAC∗. Six scenes from the 7-scenes dataset were used for training, and the remaining scene data were used as queries. The results are summarized in Table 6. The final pose estimation result by query-

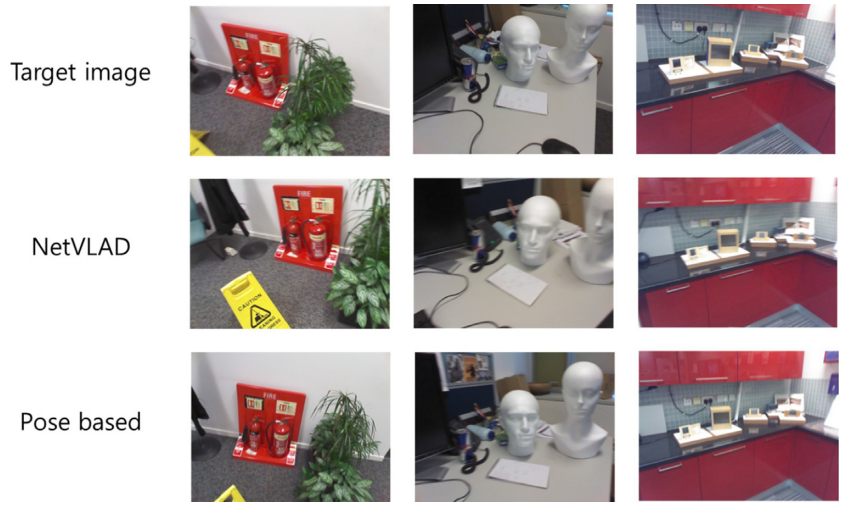

Figure 8: Examples of target images and images retrieved by the NetVLAD and pose-based methods.

Table 5: The experimental image retrieval results (m/°).

Scene	NetVLAD	Pose based
Chess	0.08/5.22	**0.07/3.26**
Fire	0.18/7.29	**0.15/5.12**
Heads	**0.16/6.32**	0.17/7.10
Office	0.36/**9.28**	**0.16**/10.52
Pumpkin	0.24/10.52	**0.21/8.28**
Red kitchen	0.23/7.31	**0.18/5.35**
Stairs	**0.27/5.50**	**0.27**/11.27

ing the chess scene image has a translation and rotation error of 0.25 m and 12.07°, respectively. Similarly, the errors related to the heads and red kitchen scenes are considered. The results indicate that the proposed method has a similar accuracy to that of NN-Net. The direct pose regression step of the proposed method sometimes yields a significant error in the rough pose estimation process. However, a stable result can be obtained after a few frames using the proposed method, which locates the nearest image using the previous pose estimation.

5.7. Ablation study

In this section, we assess how the components used in the proposed method affect the visual localization system performance through an ablation study. The study considered the following four networks to highlight the contributions of each part of the framework: (i) Base, which regresses the direct pose; (ii) Base + DE (Directly estimate Essential matrix), which regresses an essential matrix directly using nine parameters; (iii) Base + RE, which regresses an essential matrix with a reconstruction layer; and (iv) Base + RE (with Reconstruction layer estimate Essential matrix) + s, which regresses an essential matrix with a scale factor.

Table 6: Generalization test results on the 7-scenes dataset (m/°).

Removed scene	NN-Net	DSAC*	Ours
Chess	0.27/13.05	2.61/99.9	**0.25/12.07**
Heads	**0.23/15.03**	2.80/89.7	0.26/17.04
Red kitchen	**0.36/12.60**	3.29/105.3	**0.36**/13.41

The ablation study results are presented in Table 7. The first case (Base) denotes the initial pose estimation results, which can be used as a reference to show how each part contributes to the performance. In the second case (Base + DE), the results do not improve much compared to the first case. Because the scale of each element of the essential matrix is very different, the direct regression method needs a long time to converge, whose results are not promising. Therefore, the proposed method does not consider the direct regression of the essential matrix and the scale. The third case (Base + RE) indicates that the pose error has been reduced by approximately 40% when an essential matrix is obtained with a reconstruction layer. The reconstruction network is significantly effective in producing an accurate essential matrix. In the fourth case (Base + RE + s), the results improved substantially compared to the third case. The essential matrix has a unit translation vector, whereas the third case only corrects the direction. Therefore, considering the scaling effect in the estimation can improve the performance.

6. Conclusions

In this study, we propose a coarse-to-fine visual localization framework that combines direct pose regression, pose-based retrieval, and essential matrix regression to estimate the 6-DoF camera pose when only a 2D query image is given, without using 3D information about the scene.

The proposed method integrates direct pose regression, pose-based image retrieval, and essential matrix estimation for pose refinement in one framework, which combines the characteristics and strengths of each approach to strike a balance between robustness and accuracy. Although it employs some elements of the existing methods, it reduces the processing time by using an estimation method that takes pose information in the image retrieval step. After that, a method is developed to estimate the relative pose through the essential matrix estimation, which reduces the sensitivity of the hyperparameter, resulting in higher estimation accuracy in the outdoor environments compared to the existing methods.

Table 7: Ablation study results on the 7-scenes dataset (m/°).

Scene	Base	Base + DE	Base + RE	Base + RE + s
Chess	0.14/6.17	0.12/4.14	0.07/3.25	**0.05/2.52**
Fire	0.24/10.08	0.23/10.4	0.17/9.04	**0.10/8.81**
Head	0.18/11.6	0.14/10.5	0.12/8.45	**0.09/7.69**
Office	0.21/5.77	0.18/5.32	0.15/5.38	**0.10/4.84**
Pumpkin	0.25/5.92	0.16/4.17	0.13/4.93	**0.12/3.65**
Red kitchen	0.23/7.01	0.23/5.08	0.20/5.01	**0.13/4.10**
Stairs	0.27/10.6	0.25/7.53	0.15/6.94	**0.13/6.68**

The pose-based image retrieval approach is more efficient in terms of speed and memory, while presenting similar results to those obtained by traditional image retrieval methods. Essential matrix-based relative pose estimation can be applied to unseen scenes, which underlines that the proposed method is more robust and stable with respect to various scenes. The proposed method is motivated by the image retrieval approach. Therefore, it experiences the limitations that the image retrieval method has. Namely, it should maintain a large RGB image and the corresponding pose dataset. It also requires training before it can be used for pose estimation.

Our proposed framework results in a more accurate pose estimation compared to the state-of-the-art visual localization methods, which is confirmed based on the experimental results obtained on indoor and outdoor datasets.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (2017R1A2B4012124) and by Research Program of Civil-Military Technology Cooperation through Defense Acquisition Program Administration of Korea and Ministry of Trade, Industry and Energy, Korea (20-SN-GU-01).

Conflict of interest statement

None declared.

References

Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., & Sivic, J. (2018). NetVLAD: CNN architecture for weakly supervised place recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(6), 1437–1451. https://doi.org/10.1109/TPAMI.2017.2711011.

Bai, D., Wang, C., Zhang, B., Yi, X., & Yang, X. (2018). Sequence searching with CNN features for robust and fast visual place recognition. *Computers and Graphics*, *70*, 270–280. https://doi.org/10.1016/j.cag.2017.07.019.

Balntas, V., Li, S., & Prisacariu, V. (2018). RelocNet: Continuous metric learning relocalisation using neural nets. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision – ECCV 2018. ECCV 2018. Lecture Notes in Computer Science*(Vol. *11218*, pp. 782–799). Springer. https://doi.org/10.1007/978-3-030-01264-9_46.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In A. Leonardis, H. Bischof, & A. Pinz (Eds.), *Computer Vision – ECCV 2006. ECCV 2006. Lecture Notes in Computer Science*(Vol. *3951*, pp. 404–417). Springer. https://doi.org/10.1007/11744023_32.

Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., & Rother, C. (2017). DSAC – Differentiable RANSAC for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 6684–6692). https://doi.org/10.1109/CVPR.2017.267.

Brachmann, E., & Rother, C. (2018). Learning less is more – 6D camera localization via 3D surface regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 4654–4662). https://doi.org/10.1109/CVPR.2018.00489.

Brachmann, E., & Rother, C. (2021). Visual camera re-localization from RGB and RGB-D images using DSAC. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2021.3070754.

Brahmbhatt, S., Gu, J., Kim, K., Hays, J., & Kautz, J. (2018). Geometry-aware learning of maps for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 2616–2625). https://doi.org/10.1109/CVPR.2018.00277.

Choy, C. B., Gwak, J., Savarese, S., & Chandraker, M. (2016). Universal correspondence network. *Advances in Neural Information Processing Systems*, *29*. https://doi.org/10.5555/3157096.3157366.

Csurka, G., Dance, C., Fan, L., Willamowski, J., & Bray, C. (2004). Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*(Vol. 1, No. 1–22, pp. 1–2).

Ding, M., Wang, Z., Sun, J., Shi, J., & Luo, P. (2019). CamNet: Coarse-to-fine retrieval for camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*(pp. 2871–2880). https://doi.org/10.1109/ICCV.2019.00296.

En, S., Lechervy, A., & Jurie, F. (2019). RPNet: An end-to-end network for relative camera pose estimation. In L. Leal-Taixé, & S. Roth (Eds.), *Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science*(Vol. *11129*, pp. 738–745). Springer. https://doi.org/10.1007/978-3-030-11009-3_46.

Jaderberg, M., Simonyan, K., & Zisserman, A. (2015). Spatial transformer networks. *Advances in Neural Information Processing Systems*, *28*, 2017–2025. https://doi.org/10.5555/2969442.2969465.

Kendall, A., & Cipolla, R. (2017). Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 5974–5983). https://doi.org/10.1109/CVPR.2017.694.

Kendall, A., Grimes, M., & Cipolla, R. (2015). PoseNet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision*(pp. 2938–2946). https://doi.org/10.1109/ICCV.2015.336.

Laskar, Z., Melekhov, I., Kalia, S., & Kannala, J. (2017). Camera relocalization by computing pairwise relative poses using convolutional neural network. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*(pp. 929–938). https://doi.org/10.1109/ICCVW.2017.113.

Lee, S. J., Kim, D., Hwang, S. S., & Lee, D. (2021). Local to global: Efficient visual localization for a monocular camera. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*(pp. 2231–2240). https://doi.org/10.1109/WACV48630.2021.00228.

Li, X., Wang, S., Zhao, Y., Verbeek, J., & Kannala, J. (2020). Hierarchical scene coordinate classification and regression for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*(pp. 11983–11992). https://doi.org/10.1109/CVPR42600.2020.01200.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94.

Maddern, W., Pascoe, G., Linegar, C., & Newman, P. (2017). 1 year, 1000 km: The Oxford RobotCar dataset. *International Journal of Robotics Research*, *36*(1), 3–15. https://doi.org/10.1177/0278364916679498.

Melekhov, I., Ylioinas, J., Kannala, J., & Rahtu, E. (2017). Image-based localization using Hourglass networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 879–886). https://doi.org/10.1109/ICCVW.2017.107.

Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., & Fitzgibbon, A. (2011). KinectFusion: Real-time dense surface mapping and tracking. In *10th IEEE International Symposium on Mixed and Augmented Reality, 2011*(pp. 127–136). IEEE Publications. https://doi.org/10.1109/ISMAR.2011.6092378.

Poursaeed, O., Yang, G., Prakash, A., Fang, Q., Jiang, H., Hariharan, B., & Belongie, S. (2019). Deep fundamental matrix estimation without correspondences. In L. Leal-Taixé, & S. Roth (Eds.), *Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science*(Vol. *11131*, pp. 485–497). Springer. https://doi.org/10.1007/978-3-030-11015-4_35.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In *International Conference on Computer Vision, 2011*(pp. 2564–2571). IEEE Publications. https://doi.org/10.1109/ICCV.2011.6126544.

Sattler, T., Leibe, B., & Kobbelt, L. (2017). Efficient and effective prioritized matching for large-scale image-based localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(9), 1744–1756. https://doi.org/10.1109/TPAMI.2016.2611662.

Schönberger, J. L., Pollefeys, M., Geiger, A., & Sattler, T. (2018). Semantic visual localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 6896–6906). https://doi.org/10.1109/CVPR.2018.00721.

Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., & Fitzgibbon, A. (2013). Scene coordinate regression forests for camera relocalization in RGB-D images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 2930–2937). https://doi.org/10.1109/CVPR.2013.377.

Walch, F., Hazirbas, C., Leal-Taixe, L., Sattler, T., Hilsenbeck, S., & Cremers, D. (2017). Image-based localization using LSTMs for structured feature correlation. In *Proceedings of the IEEE International Conference on Computer Vision*(pp. 627–637). https://doi.org/10.1109/ICCV.2017.75.