Regressing Poses from Monocular Images in an Underwater Environment

Luyuan Peng and Mandar Chitre
ARL, Tropical Marine Science Institute and Department of Electrical and Computer Engineering
National University of Singapore

OCEANS 2022 - Chennai | 978-1-6654-1821-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/OCEANSChennai45887.2022.9775281

***Abstract*—Re-localization is crucial to underwater vehicles especially for short-range missions, such as inspection and maintenance. While re-localization has been tackled successfully in the land and aerial environments, it remains to be a challenge in the underwater environment where radio communications and global positional systems are unavailable. Traditionally, underwater localization methods include dead reckoning (DR), inertial navigation systems (INS) and acoustic sensors. However, DR and INS tend to accumulate errors along time and are vulnerable to changes in water speed while acoustic sensors often require complex architectures which are expensive. Inspired by recent progress in land re-localization methods, we implemented a regression learning method, which is simple and cost-effective, for re-localization in short-range missions. The method is able to regress a 6-DOF pose from a single RGB image. We trained and evaluated the method on datasets collected from an underwater simulator. We also investigated its robustness towards changes in lighting and made improvements.**

I. Introduction

Autonomous underwater vehicles (AUVs) and remotely controlled vehicles (ROVs) play an important role in marine engineering [1], [2]. For both types of vehicles, localization is an essential task as it is crucial to mapping, navigation, inspection and intervention [3]. Localization remains a challenge in the underwater environment where radio communications and global positional systems (GPS) are unavailable [3]. Traditionally, underwater localization methods use dead reckoning (DR), inertial navigation systems (INS) and acoustic sensors [2]. However, DR and INS accumulate errors with time while acoustic localization requires deployment of additional infrastructure and is therefore often inconvenient or expensive [2]. In recent years, there are some advancements in underwater localization using optical sensors such as cameras [4]. However, such methods often involve deployment of active markers [5], [6], or careful setup using divers [4], [7].

This paper focuses on the problem of localization in a known underwater environment for an inspection mission. An environment is considered "known" if information, such as a 3D map, of the environment is available. Such information is normally collected from previous exploration and navigation tasks. Underwater inspection refers to inspection on various types of marine structures, ranging from ship to tunnels. The main challenge for localization in underwater inspection is that pose estimation must be accurate, as the vehicle is close to the structure during an inspection mission. Moreover, the presence of the structure makes acoustic navigation with beacons difficult in practice. We wish to find a cost-effective solution to this problem, using a simple RGB camera. As we will be close to the structure in inspection missions, poor underwater visibility of camera should have insignificant impact on the effectiveness of our solution.

Camera-based localization methods can be separated into feature-based methods such as Active Search [8], and deep-learning methods such as PoseNet [9]. Active Search is able to achieve state-of-the-art results in outdoor scenes, but its performance deteriorates in indoor scenes, especially in textureless scenes [9]. As underwater scenes are often textureless and featureless, we choose to implement a learning-based regression method which was inspired by PoseNet, which is able to regress a 6-degree-of-freedom (DOF) pose from a single 224×224 RGB image. The method is able to achieve real-time localization, and can obtain approximately 6 cm position accuracy and 1.7°orientation accuracy, for small underwater scenes. Previous work has also shown that PoseNet is able to perform well on navigation tasks in large underwater scenes [14], giving us more confidence in applying this approach on visual inspection tasks.

We implemented this learning-based regression localization method on datasets collected from an underwater simulator [15]. We showed that the implemented model is able to work well on underwater scenes with limited visibility and limited landmarks or features. Fig. 1 shows some examples of underwater scenes. We also investigated the robustness of the implemented method against changes in lighting conditions and implemented different approaches to improve the model performance and robustness.

II. Related Works

A. Visual-based re-localization

Visual-based re-localization is closely related to visual simultaneous localization and mapping (SLAM). While visual SLAM focuses on mapping a new environment and tracking poses of the sensors simultaneously [11], visual-based re-localization focuses on estimating current poses of the sensors using prior information, such as a 3D map, of an environment and current images captured by the camera. Recent advancements in visual-based re-localization methods can be categorized into feature-based methods and deep-learning-based methods [10].

Fig. 1. As compared with typical outdoor terrestrial scenes used in PoseNet [9] (bottom panels), underwater scenes (top panels) have lower visibility and less texture.

B. Feature-based re-localization

Most state-of-the-art feature-based localization methods rely on local scale-invariant feature transform (SIFT) features. Using a 3D model reconstructed from Structure-from-Motion (SfM), one can estimate poses by firstly creating 2D-to-3D matches between image features and 3D points in SfM and then using a n-point solver for pose estimation inside a RANSAC loop [10]. Using a visual-vocabulary-based quantization of descriptor space as the prioritization scheme, Active Search speed up the 2D-to-3D matching process. It then uses 3D-to-2D matching to improve the localization accuracy, achieve the SOTA results [8].

C. Deep-learning-based re-localization

Deep-learning-based re-localization methods requires large datasets about the environment [10]. Such methods usually make use of deep convolutional neural networks (CNN) as a feature extractor and then use an affine regressor to regress poses of the camera. Using transfer learning, PoseNet leverages models pretrained on ImageNet and reduce the dependency on large datasets.

III. Method

We developed the model based on PoseNet [9]. Given a monocular RGB image, *I*, the model outputs a pose vector $\mathbf{y}$, which contains a position vector, $\mathbf{p}$, and a orientation vector, $\mathbf{q}$:

$$\mathbf{y} = [\mathbf{p}, \mathbf{q}] \tag{1}$$

We implemented the model using PyTorch to avail several pretrained models in the package.

We constructed a composite loss function as a weighted sum of the position loss and orientation loss [9], when regressing the poses from images:

$$\mathcal{L} = \mathcal{L}_{\mathbf{p}} + \beta \mathcal{L}_{\mathbf{q}}, \tag{2}$$

where β is used to balance between the position and orientation losses. We fine tuned β to find the optimal value for each dataset. It was found that for underwater scenes, the optimal β was typically between 1 and 10.

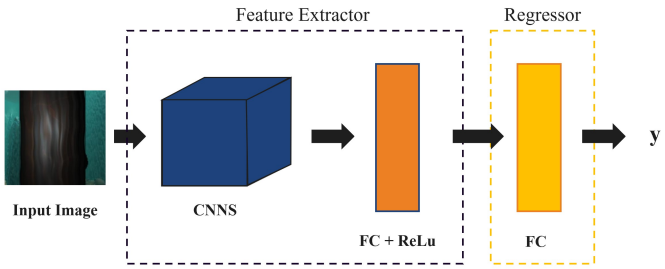

Fig. 2. Neural network architecture overview.

The position loss and orientation loss were calculated using Euclidean distance:

$$\mathcal{L}_{\mathbf{p}} = \|\hat{\mathbf{p}} - \mathbf{p}\|_2, \tag{3}$$

and,

$$\mathcal{L}_{\mathbf{q}} = \left\| \hat{\mathbf{q}} - \frac{\mathbf{q}}{\|\mathbf{q}\|} \right\|_2 \tag{4}$$

We use quaternions to represent rotation, to avoid wrap around problems associated with Euler angles [12].

As shown in Fig. 2, we used deep CNN architectures, such as GoogLeNet [13], pretrained on ImageNet as a feature extractor and then used an affine regressor to predict 6-DOF poses. Pretrained models were used to leverage the benefits of transfer learning as large underwater datasets are not widely available and it is computationally expensive to train the model on underwater datasets from scratch. Even though the ImageNet dataset is very different from underwater structure images, using pretrained models has shown better performance than training the model from scratch. This is probably because pretrained model learned how to extract features which is its sole purpose in our method. Before the last layer of the deep CNN, another fully connected (FC) layer of size 2048 was inserted. This FC layer represents the features extracted from the images. The last layer of the deep CNN was then modified to a fully connected layer to output 7-dimensional pose vectors.

The input images were rescaled to 256×256 pixels before cropping to the 224×224 input using centre cropping. To speed up the training, the images were also normalized using the mean and standard deviation of ImageNet. The poses were also normalized for the same reason. using the The outputs are 7-dimension pose vectors which contain 3-dimension position vectors and 4-dimension quaternion vectors.

IV. Datasets

To train and test our model, we collected datasets from an underwater robotics simulator [15] as shown in Fig. 3. In the underwater simulator, we placed a ROV 0.5 m from a vertical pillar with 0.7 m diameter. We then operated the ROV to inspect the pillar in a downwards spiral motion. The total spatial extent covered by the ROV was about 2 m×4 m×2 m. Using Robot Operating System (ROS), we recorded the images

Fig. 3. Underwater simulator (left) and the image captured by the ROV (right).

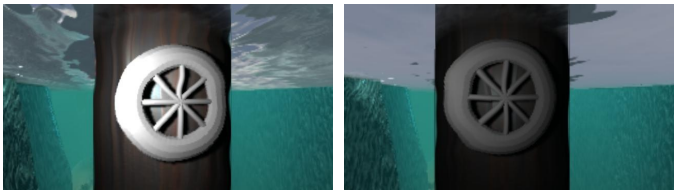

Fig. 4. Image from original dataset (left) and the image from dimmer dataset (right).

captured by the front camera of the ROV as well as the poses of the ROV in terms of position vectors and unit quaternions.

We first collected a dataset containing 14,400 images and corresponding poses. We randomly selected 70% of the dataset for training and used the remaining 30% for testing. We used random selection to ensure that both train and test sets had information of all scenes covered.

We made the light on the ROV dimmer and collected another dataset. The only difference between these two datasets is the lighting condition. We randomly selected 30% of this dataset to test the robustness of the model against change in lighting conditions.

V. Experiments

We used a 22-layer GoogLeNet as the feature extractor for our baseline model. We trained the model using stochastic gradient descent with a base learning rate of 0.003 and with a momentum of 0.9. Training took about an hour with a batch size of 4. We set β as 4. The base learning rate and batch size were chosen through hyper-parameter tuning using random search while β was chosen using grid search. We found that the implemented method was able to perform localization effectively as it can achieve a 9 cm translational accuracy and a 3°rotational accuracy. Also, Fig. 5 shows that the trajectory predicted by the model is very close to the actual trajectory performed. Fig. 6 shows cumulative histograms of position and orientation error in scenes with both standard and dimmer lighting conditions. We noted that the localization errors increased significantly when the lighting became dimmer. Thus, the model was not robust to changes in lighting.

As changing backbones, using data augmentation and applying dropouts are common techniques to improve performance on test set, we applied these techniques separately to investigate their effectiveness on our model and datasets.

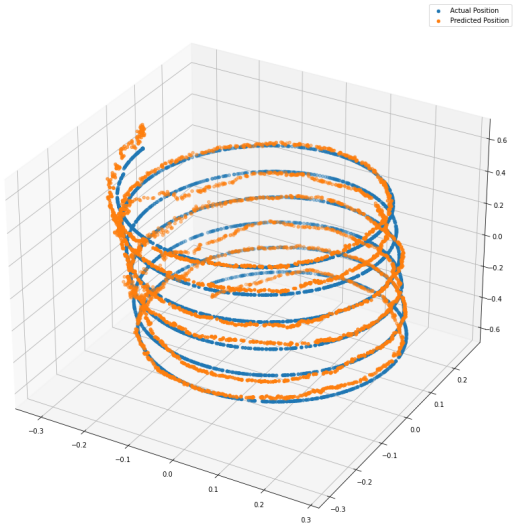

Fig. 5. Predicted trajectory (orange) vs real trajectory (blue)

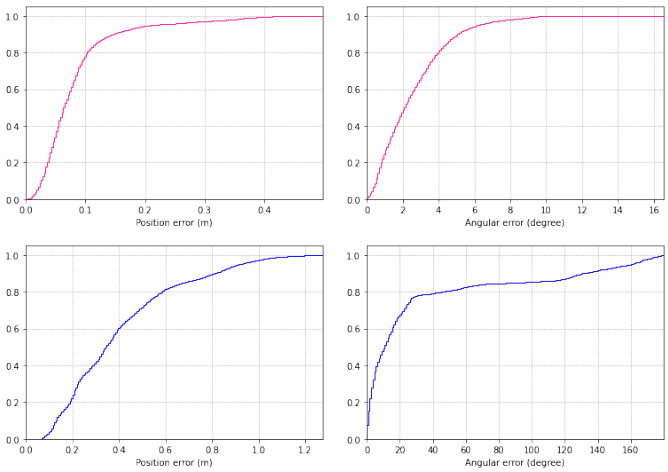

Fig. 6. **Localization Performance.** Localization on dimmer dataset (bottom) has bigger and more widely distributed errors that on standard dataset (top).

A. Data Augmentation

We applied color jittering to the standard dataset during training by randomly changing image brightness, contrast, saturation and hue. As shown in Table I, after applying color jittering, the robustness to changes in lighting condition improved slightly. We also applied contrast limited adaptive histogram equalization (CLAHE). The robustness was further improved.

TABLE I
Model Performance in Different Lighting Conditions

Data Augmentation	Dataset	Accuracy
disabled	standard	0.0891 m, 2.91°
disabled	dimmer	0.468 m, 37.8°
enabled	standard	0.125 m, 2.63°
enabled	dimmer	0.389 m, 15.3°

B. Using Deeper Network

To examine the effectiveness of deeper network on robustness, we used ResNet models pretrained on ImageNet [16]. Using residual blocks, ResNets can be much deeper than GoogLeNet without experiencing exploding or vanishing gradient. They also achieved state-of-the-art results on various visual recognition tasks, such as ImageNet localization [16]. From the results in Table II, we saw that using deeper networks was effective on improving robustness of the model although the improvement was small. We also noticed that deeper networks can achieve much higher accuracy on standard dataset during testing. Considering both training time and model accuracy, for our subsequent experiments, we adopted ResNet-50 as our backbone.

TABLE II
MODEL PERFORMANCE USING DIFFERENT BACKBONES

	Standard		**Dimmer**	
	Position	***Orientation***	***Position***	***Orientation***
GoogLeNet	0.125 m	2.63°	0.389 m	15.3°
ResNet-18	0.0894 m	3.22°	0.421 m	17.0°
ResNet-34	0.0782 m	2.36°	0.330 m	21.3°
ResNet-50	0.0657 m	2.15°	0.348 m	11.1°
ResNet-101	0.0592 m	1.70°	0.370 m	6.59°

C. Using Larger Dataset

We combined the standard dataset and the dimmer dataset to form a larger dataset and randomly selected 70% of the larger dataset as the train set. We then tested the trained model on test sets from both lighting conditions separately. The results are shown in Table III.

TABLE III
MODEL PERFORMANCE USING LARGER DATASET

Dataset	**Position Accuracy**	**Orientation Accuracy**
Standard	0.211 m	1.83°
Dimmer	0.272 m	2.00°

D. Applying Dropouts

We found applying dropouts showed no improvement on model robustness towards different lighting conditions.

VI. CONCLUSION

In this paper, we implemented a deep-learning-based relocalization method in a simulated underwater environment. We showed that our approach can achieve high accuracy for both position and orientation. We found that using deeper networks can significantly improve the model performance. Moreover, data augmentation and using a larger dataset can improve the robustness of the model towards changes in lighting.

In the future, the research can be continued in several directions. Firstly, we can improve robustness of the model towards change in other factors such as distance. Secondly, we can test our method in real underwater environments. Thirdly, as the model does not restrict the input to be camera images, we can also experiment with using sonar images or point clouds. Lastly, We may explore how to improve the model performance through changing the architecture such as incorporating long short-term memory (LSTM) [17].

REFERENCES

[1] A. Kenge and A. Mali, "Design and Analysis of Underwater Remotely Operated Vehicle," 2019 International Conference on Nascent Technologies in Engineering (ICNTE), 2019, pp. 1-5, doi: 10.1109/ICNTE44896.2019.8945851.

[2] J. González-García, A. Gómez-Espinosa, E. Cuan-Urquizo, L. G. García-Valdovinos, T. Salgado-Jiménez, and J. A. E. Cabello, "Autonomous Underwater Vehicles: Localization, Navigation, and Communication for Collaborative Missions," Applied Sciences, vol. 10, no. 4, p. 1256, Feb. 2020.

[3] L. Paull, S. Saeedi, M. Seto and H. Li, "AUV Navigation and Localization: A Review," in IEEE Journal of Oceanic Engineering, vol. 39, no. 1, pp. 131-149, Jan. 2014, doi: 10.1109/JOE.2013.2278891.

[4] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough and A. Mouzakitis, "A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications," in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 829-846, April 2018, doi: 10.1109/JIOT.2018.2812300.

[5] A. D. Buchan, E. Solowjow, D. Duecker and E. Kreuzer, "Low-cost monocular localization with active markers for micro autonomous underwater vehicles," 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp. 4181-4188, doi: 10.1109/IROS.2017.8206279.

[6] D. Chiarella, M. Bibuli, G. Bruzzone, M. Caccia, A. Ranie Marconi, P. Cutugno, "Gesture-based language for diver-robot underwater interaction," in OCEANS, Genoa, Italy, 2015.

[7] A. G. Chavez, C. A. Mueller, T. Doernbach, D. Chiarella, and A. Birk, "Robust gesture-based communication for underwater human-robot interaction in the context of search and rescue diver missions," in IROS Workshop on Human-Aiding Robotics, Madrid, Spain, 2018, held in conjunction with IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

[8] T. Sattler, B. Leibe and L. Kobbelt, "Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 9, pp. 1744-1756, 1 Sept. 2017, doi: 10.1109/TPAMI.2016.2611662.

[9] A. Kendall, M. Grimes and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2938-2946, doi: 10.1109/ICCV.2015.336.

[10] F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck, and D. Cremers, "Image-based localization using lstms for structured feature correlation," 2017 IEEE International Conference on Computer Vision (ICCV), 2017.

[11] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual slam algorithms: A survey from 2010 to 2016," IPSJ Transactions on Computer Vision and Applications, vol. 9, no. 1, 2017.

[12] A. Kendall and R. Cipolla, "Geometric Loss Functions for Camera Pose Regression with Deep Learning," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6555-6564, doi: 10.1109/CVPR.2017.694.

[13] C. Szegedy et al., "Going deeper with convolutions," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594.

[14] B. Teixeira, H. Silva, A. Matos and E. Silva, "Deep Learning for Underwater Visual Odometry Estimation," in IEEE Access, vol. 8, pp. 44687-44701, 2020, doi: 10.1109/ACCESS.2020.2978406.

[15] A. Chaudhary, R. Mishra, B. Kalyan, and M. Chitre, "Development of an Underwater Simulator using Unity3D and Robot Operating Systems", 2021 OCEANS - San Diego, in press.

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

[17] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.