Long-term visual localization in deep-sea underwater environment

Clémentin Boittiaux, Claire Dune, Aurélien Arnaubec, Ricard Marxer, Maxime Ferrera, Vincent Hugel

Long-term visual localization in deep-sea underwater environment

Clémentin Boittiaux[1,2,3] Claire Dune[2] Aurélien Arnaubec[1]
Ricard Marxer[3] Maxime Ferrera[1] Vincent Hugel[2]

[1] Ifremer, Zone Portuaire de Bregaillon, La Seyne-sur-Mer, France
[2] Université de Toulon, COSMER, Toulon, France
[3] Université de Toulon, Aix Marseille Univ, CNRS, LIS, Toulon, France

boittiauxclementin@gmail.com

Résumé

Les récentes avancées liées aux véhicules sous-marins autonomes impliquent la nécessité de pouvoir localiser précisément ces derniers dans leur environnement. Cependant, la précision des systèmes de positionnement acoustique n'est pas suffisante. Une localisation plus fine pourrait alors être obtenue en utilisant les observations visuelles du robot. Dans cette étude, nous évaluons des méthodes de l'état de l'art de localisation visuelle développées en milieu terrestre sur le jeu de données sous-marin Eiffel Tower. Celui-ci contient des images issues de quatre visites de la même cheminée hydrothermale étendues sur cinq ans. Nous montrons que ces méthodes ont du mal à localiser des images issues d'années différentes. Nous menons ensuite une analyse pour évaluer les facteurs qui peuvent être responsables de cette baisse de performance.

Mots Clef

Localisation visuelle, robotique sous-marine.

Abstract

With the advent of autonomous underwater vehicles comes the need to localize them precisely in their environment. The robot's location is usually retrieved using acoustic positioning systems. However, in the context of autonomous vehicles, these systems may not be available or sufficiently accurate. A finer localization could be obtained using the robot's visual observations. In this study, we benchmark state-of-the-art visual localization methods that were developed for terrestrial applications on the Eiffel Tower deep-sea dataset. The latter embeds four visits of the same hydrothermal vent over five years. We show that these methods struggle to localize images collected in different years. We conduct an analysis to assess which factors may be responsible for this performance hit.

Keywords

Visual localization, marine robotics.

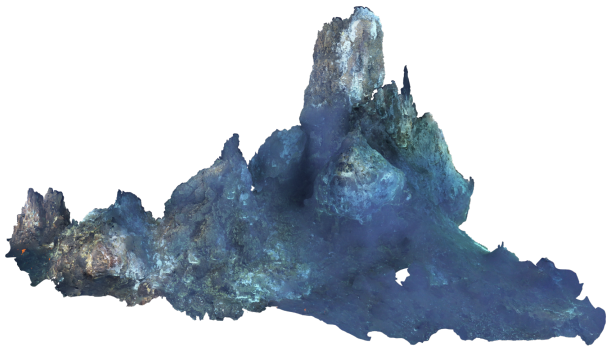


FIGURE 1 – Structure-from-Motion (SfM) reconstruction of the *Eiffel Tower* hydrothermal vent off the Azores Islands, visited four times between 2015 and 2020.

1 Introduction

Autonomous robots such as self-driving cars or delivery drones are increasingly developed. However, while most popular applications are either terrestrial or aerial, underwater vehicles have also much to gain from advances in this area. Underwater applications such as coastal surveillance, underwater cartography, or even robots that regularly monitor a site of scientific interest could benefit from the extended autonomy. A key step towards this goal is the ability for the robot to localize itself in its surroundings. In underwater environments, and more specifically in deep sea, absolute localization has proven to be a challenging problem. To tackle it, underwater robots make use of multiple sensors. Working class Remotely Operated underwater Vehicles (ROV) or Autonomous Underwater Vehicles (AUV) are equipped with high grade sensors including an Utra-Short Baseline acoustic positioning system (USBL), an Inertial Navigation System (INS), a Doppler Velocity Log (DVL) and a depth sensor. The sensor data are fused to compute the navigation of the system with regard to the ship. Absolute position is then retrieved by localizing the ship with GPS. However, this estimate is coarse and its margin of error is proportional to the distance between the vehicle and the ship.

This paper investigates a specific scenario in which a ROV is used to visit a site of scientific interest on a yearly basis in order to monitor its evolution. The costs of ROV operations being high, we sought to perform future visits using an AUV. In this situation, the AUV needs to localize itself with respect to the observations made in previous visits by the ROV. Because of the margin of error of aforementioned sensors, we take an interest in the information provided by the camera of the robot. More specifically, this paper focuses on localizing the robot in a previously visited environment based on its visual observation, a problem termed visual localization [1, 2]. Most existing methods have been designed for terrestrial applications, and their transfer to underwater environments may not be trivial. Indeed, the explored site may be subject to changes, *e.g.* sedimentation, and underwater images exhibit very singular characteristics, *e.g.*, backscattering and wavelength absorption.

Most state-of-the-art algorithms rely on deep-learning based features [3, 4], or estimate 3D scene coordinates directly from a single image using neural networks [5, 6, 7]. However, these approaches require a large amount of data for training. While such data is available in great quantity for terrestrial applications, they are much more scarce in underwater scenarios, hence the interest in evaluating the performance of existing methods in this environment. We have therefore used the publicly available Eiffel Tower dataset [8] with data from four different ROV visits of the same hydrothermal vent between 2015 and 2020 (Fig. 1). A 3D model of the scene was built from acquired images by performing Structure-from-Motion (SfM) using COLMAP [9, 10] and loose poses priors computed from the system navigation. This dataset presents a vast panel of underwater image characteristics. Moreover, it includes some particularly challenging scenarios because of the evolution of the site and the changes in the environment during the different visits. We benchmarked four renowned methods in terrestrial localization on the Eiffel Tower dataset. Some of the techniques rely on deep-learned features trained on terrestrial datasets [3, 4] while in others the features are learned specifically for each scene [7, 11]. Results show that on the underwater dataset, during the same visit and thus without change of environment, these methods achieve results comparable with their performance on terrestrial datasets. Conversely, none of them perform close to the terrestrial results when evaluated across different visits, even in comparable scenarios where visits are performed in mismatched day/night conditions or in different seasons. This suggests the need to develop methods specific to the underwater conditions.

Section 2 reviews visual localization datasets, benchmarks and previous work on underwater pose estimation. Section 3 is dedicated to the presentation of the dataset and the benchmarked localization methods. Section 4 details the experimental setup and parameters used for each method. Section 5 presents and discusses the benchmark results.

2 Related work

Because of the interest in visual localization, numerous localization methods have been proposed [12, 11, 6, 3, 7, 4]. With recent advances in machine-learning, many of these methods rely on data-driven approaches [11, 6, 3, 7, 4], *e.g.*, deep-learned features. Such models require a large amount of data for training, which led to the creation of several visual localization datasets. Common datasets for benchmarking visual localization algorithms include Aachen Day-Night, RobotCar Seasons and CMU Seasons introduced in [2] and Cambridge [11], 7-Scenes [13] and 12-Scenes [14]. All of them are terrestrial. 7-Scenes and 12-Scenes are collected in an indoor setting, while all others are composed of outdoor environments. Sattler *et al.* datasets [2] exhibit some difficult localization scenarios like day/night observations. In some cases, hand-labelling 2D-3D matches were even necessary to accurately estimate the camera poses. While there exist many terrestrial datasets, such data is scarce and difficult to access in underwater scenarios. Moreover, while terrestrial images might be coupled with GPS or odometry data, AUVs operate in a GPS-less environment, making it much more difficult to have access to georeferenced data. Existing underwater datasets [15, 16] focus on providing data for the development of underwater SLAM algorithms. AQUALOC dataset [16] provides underwater images synchronized with inertial and depth data for 3 different sites off Corsica's shore. One of these sites is a harbor lying at a depth of 3 to 4 m and the other two are archaeological sites that lie at a depth of 270 m and 380 m. Images were acquired with a monochromatic camera. While sequences follow different trajectories, all different visits occurred during the same day, not covering all the possible changes that can happen in this environment, *e.g.*, salinity variation that can alter the pinhole model, increased turbidity, sedimentation or marine population changes.

Visual localization benchmarks on terrestrial datasets were already conducted in [1, 2, 17]. Nielsen *et al.* evaluated PoseNet [11], an end-to-end visual localization neural network, on an underwater dataset [18]. Other works also tackled the particularities of underwater images in other scenarios. Some studies [19, 20] focused on the estimation of the pose of known objects in an underwater environment. Other researchers trained a neural network to estimate the pose between two teamed-up underwater robots [21].

Visual localization datasets require reference camera poses for each of the images, which can be constructed in different ways. For example, PoseNet's underwater evaluation [18] was conducted on an unconventional dataset where camera poses were obtained with an underwater motion capture system. Most common methods to access such information as well as the scene's geometry rely on SfM or depth-based SLAM. However, in a deep-sea environment, motion capture is out of the question due to the difficulties in deploying such a system, and depth-based SLAM is difficult to set up because of the absorption of infrared light in

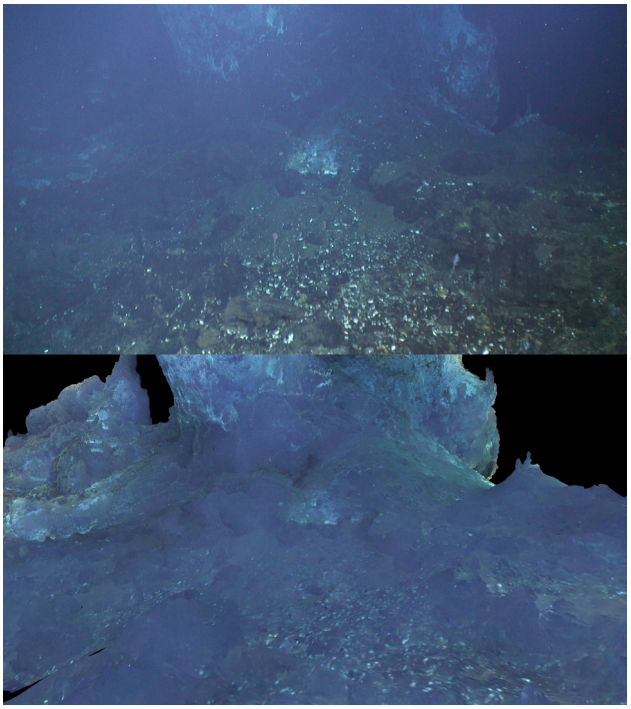


FIGURE 2 – Example of the challenging nature of the targeted underwater data. Backscatter and color absorption attenuate the image signal in a distance-dependent manner, making it difficult to observe far off elements of the scene.

water. Thereby, SfM appears to be a sensible choice for estimating the camera poses and the scene's geometry in the underwater environment. Nevertheless, Brachmann *et al.* showed that the performance of a localization method on a given dataset is greatly impacted by the method used to build the "ground-truth" of this dataset [17]. Indeed, methods that minimize the same error as the algorithm used for estimating the ground-truth poses have an advantage because they lead to the same local minima. This paper will discuss results obtained by taking into consideration that the ground-truth of the underwater dataset was constructed using SfM.

3 Method

The work presented here consists in generating a 3D model and corresponding image poses of a hydrothermal vent using SfM and evaluating the performance of different localization methods on this model.

3.1 Dataset

Over the last decade, the EMSO-Azores observatory monitored a hydrothermal vent field off the Azores Islands in the Mid-Atlantic Ridge. One of the hydrothermal vents, named *Eiffel Tower* was visited by a ROV operated by the French Research Institute for the Exploitation of the Sea (Ifremer) in 2015, 2016, 2018 and 2020. It lies at a depth of approximately 1700 meters and spans over 800 m^2.
During each visit, 4000 to 6000 images have been acquired by the ROV *Victor 6000*, embedding a camera with a special optical lens designed to correct glass-water diffraction. The ROV navigation was obtained from its USBL, INS, DVL and depth sensors. It provides an estimate of the vehicle's localization that is consistent for each individual visit. Because no light from the surface reaches such depths, the robot was also equipped with an artificial lighting system to illuminate the hydrothermal vent.
The dataset exhibits underwater imagery specificities, *i.e.*, strong distortion and poor range of vision because of light absorption (Fig. 2). Moreover, because the scene is always illuminated by a light source placed near the camera, backscattering is accentuated and illumination is constantly changing. In addition, the investigated site also shows some peculiar characteristics, like moirage (smoke) coming from the chimneys.

3.2 3D Model

Using aforementioned images and available navigation data, a sparse 3D model of all visits was built using COLMAP SfM [9]. Navigation data was used to perform spatial matching within each year. A vocabulary tree was then used for matching images between different years and images that lack navigation data. The resulting model is used as ground-truth for the camera poses and scene geometry. The scale of the model was retrieved by aligning 2016 poses with poses priors obtained with navigation.

3.3 Visual localization methods

We will evaluate 4 different visual localization methods, each using data-based models for different tasks.

PoseNet In [11] Kendall *et al.* replaced the classification layer of a deep neural network to regress camera poses. From a simple inference, this new network directly predicts a camera pose given its image. The model's parameters were optimized by minimizing a loss function that defines the error between ground-truth and estimated poses. However, the design of the loss function is challenging because it needs to embed an error in $SE(3)$ into a scalar to be optimized through gradient backpropagation. This paper evaluates a PoseNet-like network with two different loss functions : i) PoseNet loss [11] weights translation and rotation errors by a fixed factor β; ii) Homoscedastic loss [22] weights translation and rotation errors through two learnable factors $\hat{s}_x$ and $\hat{s}_q$ trained during the optimization.

hLoc This approach [3] divides the localization problem into two main steps. First, images that are similar to the current observation are retrieved. Then local features between the retrieved images and the current observation are matched. The final pose is then computed with PnP and RANSAC. Global retrieval and local matching are both performed with deep neural networks.

PixLoc In [4] Sarlin *et al.* designed an end-to-end pipeline for performing photometric alignment. They localized a query image by aligning it with multiple previously seen

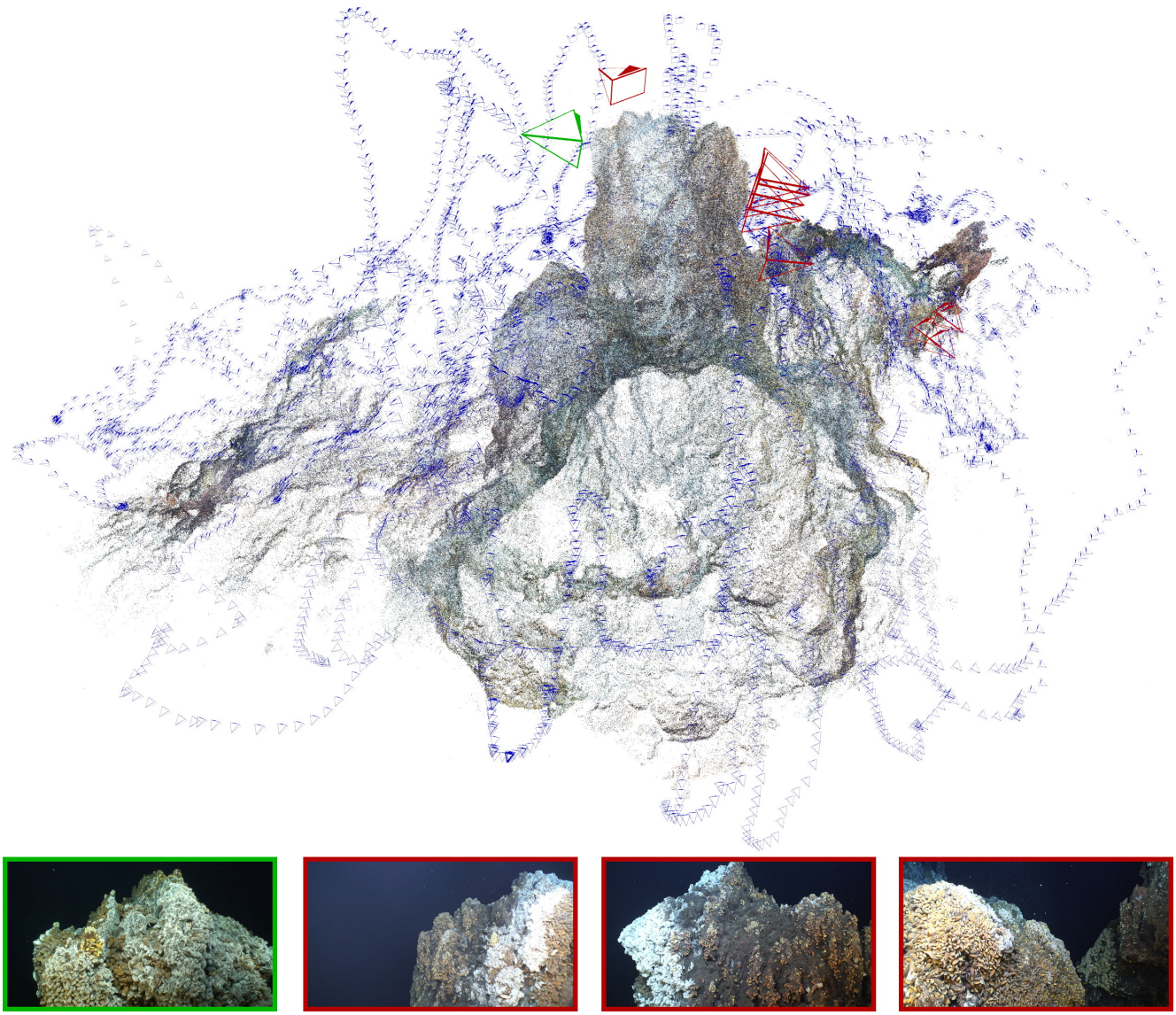


FIGURE 3 – Global retrieval of a 2016 test image on 2015 train images using NetVLAD. Test image pose is in green. Retrieved images are in red. All images in the train set are in blue.

images. To do so, the authors minimized a photometric error on multiple levels, from coarse to fine. This photometric error was computed on multi-level deep-learned features. Previously seen images were retrieved by using deep global features matching just as in the first step of hLoc. The authors also suggested that the pose may be initialized with hLoc and then further refined with this method. We refer to this approach as hLoc + PixLoc.

DSAC* With this method, Brachmann *et al.* once again improve their end-to-end localization pipeline introduced in DSAC [5] and revisited in DSAC++ [6]. Using a neural network, they first estimated 3D scene coordinates for some pixel grids of the image. Then, for RGB images, they followed a differentiable PnP/RANSAC scheme to estimate the pose of the camera.

4 Experiments

We evaluate the performance of the aforementioned methods in two different scenarios. To begin with, the methods are validated by localizing images acquired within the same visit, *i.e.*, during the 2015 dive. One every five frame is sequentially selected to be part of the test set, and the rest of the frames is used for the train set. This particular dataset split does not suffer from any environment change, furthermore we ensure no regions of the scene remain unseen during training. Subsequently, all methods are benchmarked on every year starting from 2016, in a setting quite similar to the target application yet much more challenging due to mismatches between train and test. Following a chronological rationale, image poses of each year are estimated using data from all the previous available years. For instance, 2018 image poses are retrieved using

Set	Method	Median errors	1 cm, 1°	2 cm, 2°	3 cm, 3°	5 cm, 5°	25 cm, 2°	50 cm, 5°	500 cm, 10°
2015	PixLoc	**0.001m**, 0.019°	96.72%	97.95%	98.26%	98.77%	99.08%	99.18%	99.28%
	hLoc	**0.001m**, 0.021°	98.46%	**99.59%**	**99.79%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**
	hLoc+PixLoc	**0.001m**, **0.014°**	**99.08%**	**99.59%**	99.59%	**100.00%**	**100.00%**	**100.00%**	**100.00%**
	DSAC*	0.533m, 4.900°	0.00%	0.21%	0.31%	1.44%	18.67%	42.36%	69.64%
	PoseNet	0.250m, 0.837°	0.00%	0.10%	0.31%	0.92%	45.44%	87.38%	99.79%
	Homoscedastic	0.138m, 0.820°	0.21%	0.62%	2.36%	6.36%	76.10%	96.62%	**100.00%**
2016	PixLoc	7.741m, 45.254°	0.32%	0.86%	1.68%	3.59%	7.40%	8.81%	13.94%
	hLoc	0.437m, 4.827°	**2.51%**	**8.08%**	12.94%	24.86%	44.91%	49.15%	51.88%
	hLoc+PixLoc	**0.426m**, **4.696°**	2.19%	7.74%	**13.07%**	**25.28%**	**45.03%**	**49.26%**	**51.96%**
	DSAC*	12.080m, 67.454°	0.00%	0.00%	0.00%	0.00%	0.16%	1.30%	6.46%
	PoseNet	4.805m, 26.199°	0.00%	0.00%	0.00%	0.00%	0.03%	0.41%	21.72%
	Homoscedastic	4.045m, 22.277°	0.00%	0.00%	0.00%	0.00%	0.08%	1.78%	27.91%
2018	PixLoc	12.635m, 61.073°	0.25%	0.47%	0.55%	0.73%	1.25%	4.22%	8.68%
	hLoc	1.632m, 15.320°	1.23%	**2.32%**	**2.91%**	**3.64%**	9.07%	**27.69%**	47.80%
	hLoc+PixLoc	**1.628m**, **15.289°**	**1.37%**	2.24%	2.88%	3.60%	**9.21%**	27.51%	**47.83%**
	DSAC*	13.239m, 70.808°	0.00%	0.00%	0.00%	0.00%	0.00%	0.30%	3.49%
	PoseNet	3.289m, 17.974°	0.00%	0.00%	0.00%	0.00%	0.02%	0.61%	31.57%
	Homoscedastic	2.817m, 14.407°	0.00%	0.00%	0.00%	0.02%	0.00%	1.57%	38.30%
2020	PixLoc	10.580m, 58.616°	0.00%	0.03%	0.08%	0.18%	2.21%	4.00%	7.33%
	hLoc	4.199m, 32.348°	**0.03%**	**0.20%**	**0.46%**	**2.67%**	28.79%	36.53%	41.22%
	hLoc+PixLoc	4.190m, 32.338°	**0.03%**	0.18%	0.43%	2.42%	**29.03%**	**36.55%**	41.27%
	DSAC*	14.066m, 93.291°	0.00%	0.00%	0.00%	0.00%	0.00%	0.03%	0.66%
	PoseNet	2.586m, 10.047°	0.00%	0.00%	0.00%	0.00%	0.08%	1.76%	45.62%
	Homoscedastic	**1.795m**, **9.054°**	0.00%	0.00%	0.00%	0.00%	0.31%	4.25%	**52.06%**

TABLE 1 – Median localization errors and percentage of poses localized within given thresholds in meters and degrees

2015 and 2016 images, poses and scene geometry.

Hereby, we present the parameters used for each of the methods.

PoseNet We re-implemented the network as described in [11], except for replacing the GoogLeNet backbone with a more modern MobileNetV2 [23]. We used $\beta = 500$ for PoseNet loss as suggested in [11] for the outdoor Cambridge dataset, and initialized the Homoscedastic loss as suggested in [22], *i.e.*, $\hat{s}_x = 0.0$ and $\hat{s}_q = -3.0$.

hLoc We used the pipeline presented in [3], *i.e.*, NetVLAD [24] for global retrieval and SuperPoint [25] alongside SuperGlue [26] pre-trained on outdoor scenes for local matching.

PixLoc We used weights of the network pre-trained on the MegaDepth dataset [27].

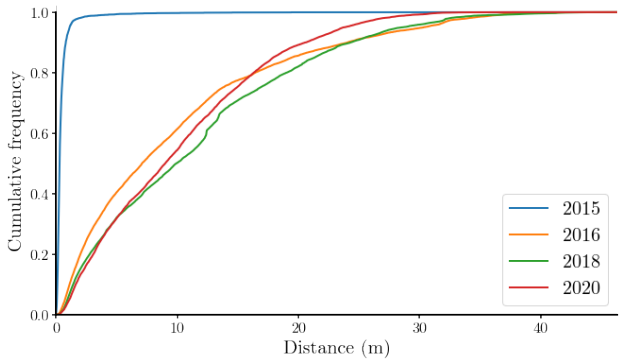
DSAC* We initialized the network as suggested by training it for 1000000 iterations to directly regress sparse scene coordinates resulting from SfM. We then trained the network end-to-end for 100000 iterations.

5 Results & analysis

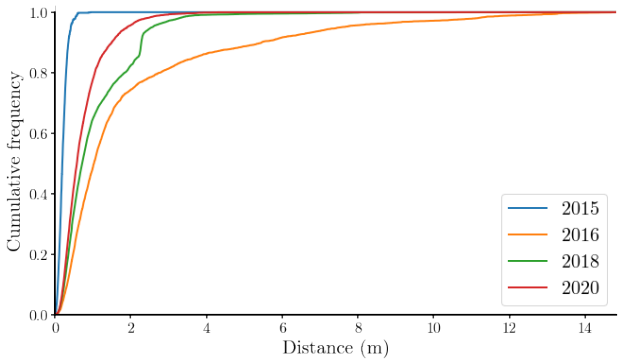
For each method, we report the median translational and rotational errors in meters and in degrees, as in [11, 4, 22]. We also report the classical percentage of poses localized within given thresholds in cm and degrees [3, 4, 2]. Table 1 shows the results on two scenarios : i) train and test sets composed of images from the same visit in 2015; ii) for every year starting from 2016 the train set consists in all visits prior to the given year, and the test set is the given year.

As previously mentioned, the performance of the methods is first validated on the 2015 subset. Because this subset was acquired during the same visit, there is almost no change in the environment and all test camera poses lie on the trajectory of the train camera poses. These two conditions greatly ease the localization task. This is reflected by the results obtained on this subset. hLoc and PixLoc achieve a localization of almost every pose within 1 cm/1°.

However, very poor results are obtained with all methods on the cross-years splits. Unintuitively, DSAC* shows the worst results, even though it is directly trained on the underwater scene. The network appears to have failed to generalize learned scene coordinates to new observations. Interestingly, the PosetNet architecture achieves performance on par with the state-of-the-art localization method hLoc. Thanks to its much simpler pipeline, and because it was directly trained on underwater data, the network managed to generalize to new observations, resulting in more robust but still coarse localization. The results of hLoc and PixLoc can be partly explained by the struggle of NetVLAD to match cross-years images. Figure 3 shows the train images retrieved by NetVLAD for a given test image. It also shows



(a) Cumulative histogram of the distances between each test image and its three best NetVLAD matches.



(b) Cumulative histogram of the distance between a test image and its closest image in the train set.

FIGURE 4 – Comparison between NetVLAD and best candidate matches.

all poses for images in the train set, that are possible candidates for image retrieval. In this case, it is easy to notice that there were many better candidates for the test image than the retrieved NetVLAD images. It appears that the network failed to identify which side of the top of the chimney was observed, resulting in a large error. As seen in Fig. 3, this task is not trivial, even for an experimented pilot. To evaluate the performance of NetVLAD across the whole dataset, Figure 4a displays a cumulative histogram of the distance between test images and their top 3 matches in the train set as retrieved by NetVLAD. According to the histogram on the 2015 subset, built with train and test images of the same visit, the network matches similar images whose poses are close when there is no change in the environment. On the contrary, it shows poor performance on cross-years matching, with more than half of the retrieved images taken at least 7 meters apart from the test image. This histogram can be compared with Fig. 4b that provides a cumulative histogram of the distance between test images poses and their closest poses in the train set. All things considered, since more than 74% of test images have a candidate in the train set within 2 meters, errors of an amplitude of Fig. 4a can only be explained by the difficulties encountered by NetVLAD for matching cross-years images. It is interesting to note that except for the 2015 subset, test images have closer retrieval candidates each year. This is because over the different visits, the area covered by the train set expands. We can also notice that localization results deteriorates over the years for all methods except PoseNet and Homoscedastic, which is unintuitive because more data is available. By leveraging the observations made on NetVLAD, we argue that the more data is available, the more the network struggles to accurately match images between different years, as it promotes images with similar environment conditions.

As shown by Brachmann *et al.* in [17], different localization algorithms greatly benefit from different methods for estimating the ground-truth of the poses and scene geometry. Because SfM was used as ground-truth in the present study, methods that minimize the same quantities as SfM are expected to perform better on the benchmarked dataset. Such methods include hLoc and to a lesser extent, DSAC*. While DSAC* estimates poses with a PnP/RANSAC scheme, it still relies on "fake" scene coordinates generated by a neural network, unlike hLoc. In a way, PixLoc also regresses similar quantities as SfM. However, in its final localization pipeline, it minimizes a deep-learned photometric error. However, these benefits are negligible compared to the aforementioned problems.

All benchmarked visual localization algorithms rely on data-driven modules. Typically, these methods require GPU hardware for real-time processing, which may not always be readily available, especially on embedded robotic systems. Despite this limitation, these methods are often necessary to achieve satisfactory performance. For instance, SIFT features may struggle to provide accurate matches between images taken in different years, whereas SuperPoint [25] and SuperGlue [26] provide a clear improvement in such scenarios.

6 Conclusion

In this paper, we benchmarked four different visual localization methods on a very challenging underwater dataset. While they perform on par with terrestrial applications when localizing within the same visit, these methods struggle to generalize to new observations with major changes in the environment. Nevertheless, hLoc shows some promising results and future work may involve training global and local descriptors on underwater data to be more robust to underwater environment changes that can be much different from their terrestrial counterparts. Global retrieval could also be improved by using sequential images to disambiguate hard cases.

Acknowledgment

The authors would like to thank Ifremer for providing the data that allowed us to conduct this study.

Références

[1] N. Piasco, D. Sidibé, C. Demonceaux, and V. Gouet-Brunet, "A survey on visual-based localization : On the benefit of heterogeneous data," *Pattern Recognition*, vol. 74, pp. 90–109, 2018.

[2] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla, "Benchmarking 6dof outdoor visual localization in changing conditions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[3] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine : Robust hierarchical localization at large scale," in *CVPR*, 2019.

[4] P.-E. Sarlin, A. Unagar, M. Larsson, H. Germain, C. Toft, V. Larsson, M. Pollefeys, V. Lepetit, L. Hammarstrand, F. Kahl, and T. Sattler, "Back to the Feature : Learning robust camera localization from pixels to pose," in *CVPR*, 2021.

[5] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother, "DSAC-Differentiable RANSAC for camera localization," in *CVPR*, 2017.

[6] E. Brachmann and C. Rother, "Learning less is more - 6D camera localization via 3D surface regression," in *CVPR*, 2018.

[7] E. Brachmann and C. Rother, "Visual camera relocalization from rgb and rgb-d images using dsac," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5847–5865, 2022.

[8] C. Boittiaux, C. Dune, M. Ferrera, A. Arnaubec, R. Marxer, L. Van Audenhaege, M. Matabos, and V. Hugel, "Eiffel tower : A deep-sea underwater dataset for long-term visual localization," 2022.

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

[10] J. L. Schönberger, T. Price, T. Sattler, J.-M. Frahm, and M. Pollefeys, "A vote-and-verify strategy for fast spatial verification in image retrieval," in *Asian Conference on Computer Vision (ACCV)*, 2016.

[11] A. Kendall, M. Grimes, and R. Cipolla, "Posenet : A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[12] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient & effective prioritized matching for large-scale image-based localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1744–1756, 2017.

[13] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proc. Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2013.

[14] J. Valentin, A. Dai, M. Niessner, P. Kohli, P. Torr, S. Izadi, and C. Keskin, "Learning to navigate the energy landscape," in *2016 Fourth International Conference on 3D Vision (3DV)*, 2016, pp. 323–332.

[15] A. Mallios, E. Vidal, R. Campos, and M. Carreras, "Underwater caves sonar data set," *The International Journal of Robotics Research*, vol. 36, no. 12, pp. 1247–1251, 2017.

[16] M. Ferrera, V. Creuze, J. Moras, and P. Trouvé-Peloux, "Aqualoc : An underwater dataset for visual–inertial–pressure localization," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1549–1559, 2019.

[17] E. Brachmann, M. Humenberger, C. Rother, and T. Sattler, "On the limits of pseudo ground truth in visual camera re-localisation," *CoRR*, vol. abs/2109.00524, 2021.

[18] M. C. Nielsen, M. H. Leonhardsen, and I. Schjølberg, "Evaluation of posenet for 6-dof underwater pose estimation," in *OCEANS 2019 MTS/IEEE SEATTLE*, 2019, pp. 1–6.

[19] M. Jeon, Y. Lee, Y.-S. Shin, H. Jang, and A. Kim, "Underwater object detection and pose estimation using deep learning," *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 78–81, 2019, 12th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2019.

[20] A. Mohammed, J. Kvam, J. T. Thielemann, K. H. Haugholt, and P. Risholm, "6d pose estimation for subsea intervention in turbid waters," *Electronics*, vol. 10, no. 19, 2021.

[21] B. Joshi, M. Modasshir, T. Manderson, H. Damron, M. Xanthidis, A. Q. Li, I. Rekleitis, and G. Dudek, "Deepurl : Deep pose estimation framework for underwater relative localization," 2020.

[22] A. Kendall and R. Cipolla, "Geometric loss functions for camera pose regression with deep learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[23] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2 : Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[24] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad : Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[25] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint : Self-supervised interest point detection and description," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[26] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue : Learning feature matching with graph neural networks," in *CVPR*, 2020.

[27] Z. Li and N. Snavely, "Megadepth : Learning single-view depth prediction from internet photos," in *Computer Vision and Pattern Recognition (CVPR)*, 2018.