MOTION ESTIMATION OF AN UNDERWATER PLATFORM USING IMAGES FROM TWO SONAR SENSORS

José Enrique Almanza-Medina [1], Benjamin Henson [2], and Yuriy Zakharov [2]

[1]University of York
[2]Affiliation not available

October 30, 2023

Abstract

Many underwater applications that involve the use of autonomous underwater vehicles require accurate navigation systems. Image registration from acoustic images is a technique that can be used to achieve this task by comparing two consecutive sonar images and estimate the motion of the vechicle. The use of deep learning (DL) techniques for motion estimation can significantly reduce the processing complexity and achieve high-accuracy position estimates. In this paper we investigate the performance improvement when using two sonar sensors compared to using a single sensor. The DL network is trained using images generated by a sonar simulator. The results show an improvement in the estimation accuracy when using two sensors.

MOTION ESTIMATION OF AN UNDERWATER PLATFORM USING IMAGES FROM TWO SONAR SENSORS

José E. Almanza-Medina, Benjamin Henson, and Yuriy V. Zakharov

Department of Electronic Engineering, University of York, U.K.
{jeam502, bth502, yury.zakharov}@york.ac.uk

ABSTRACT

Many underwater applications require precise localization. This can be achieved by techniques such as image registration applied to two consecutive acoustic images obtained by a sonar. However, this can be a complex task to implement in real time. The use of deep learning (DL) techniques for motion estimation can significantly reduce the processing complexity and achieve high-accuracy position estimates. In this paper we investigate the performance improvement when using multiple sonar sensors compared to a single sensor. The DL network is trained using images generated by a sonar simulator. The results show an improvement in the estimation accuracy when using two sensors.

***Index Terms*—Deep learning, motion estimation, underwater micronavigation.**

I. INTRODUCTION

For exploration and surveying in underwater environments, autonomous underwater vehicles (AUVs) and remotely operated underwater vehicles (ROVs) are widely used [1]. The operation of such vehicles requires an accurate estimation of their position relative to the seafloor. Micronavigation techniques have been developed for this purpose [2]–[6]. Motion estimation based on optical images is a well known approach in terrestrial [7], [8] and aerial [9], [10] applications. Recently, this approach has been used to estimate the trajectory of an underwater platform by applying a deep learning (DL) network to a sequence of images from a camera [11]. However, the use of optical images is not reliable in underwater environments where the visibility can be poor [12].

The work [13] presents a method for attitude and trajectory estimation using sonar (acoustic) images. This method is capable of obtaining accurate position estimates by analyzing the pixel displacement between consecutive images. However, due to its complexity, this method is difficult to implement in real-time. In [14], we presented a method based on DL networks to estimate the motion of an underwater platform and its trajectory using sonar images. The method significantly reduces the complexity and processing time compared to the method in [13]. The low processing time makes the methods in [14] suitable for real-time applications. The DL networks in [14] allow a millimeter accuracy in positioning between two sonar images. However, higher estimation accuracy is required for some applications, e.g., synthetic aperture sonars [15]. In [14], a single sonar sensor was considered for motion estimation. The purpose of this paper is to consider the use of two sensors separated from the sonar transmitter to find out how it can improve the accuracy even further.

J. E. Almanza-Medina acknowledges financial support from CONACyT. The work of Y. Zakharov and B. Henson was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through Grants EP/R003297/1 and EP/V009591/1.

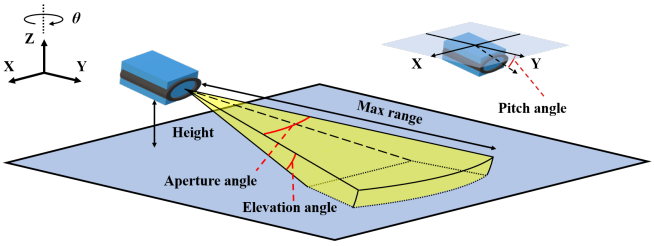

Fig. 1: Sonar FoV parameters and the coordinate system relative to the sonar. The motion in forward and backward directions corresponds to the y-axis, the motion in sideways direction corresponds to the x-axis and the rotation around the z-axis is represented with the parameter θ. The pitch angle is measured from the xy-plane, which is parallel to the seafloor.

The use of the DL approach has the problem of acquiring big volumes of labeled data for training the networks. In [14], synthetic images are generated by a sonar simulator from [16] to solve this problem. In this work, we modify the sonar simulator from [16] to allow acoustic images to be generated for more complicated sonar configurations and use these images for training and validation of DL networks.

II. SONAR SIMULATOR

The sonar simulator proposed in [16] and used in [14] for training DL networks, is built upon the development software Unity [17]. It is based on a ray-tracing technique to generate the images. The sonar has a field of view (FoV) that is

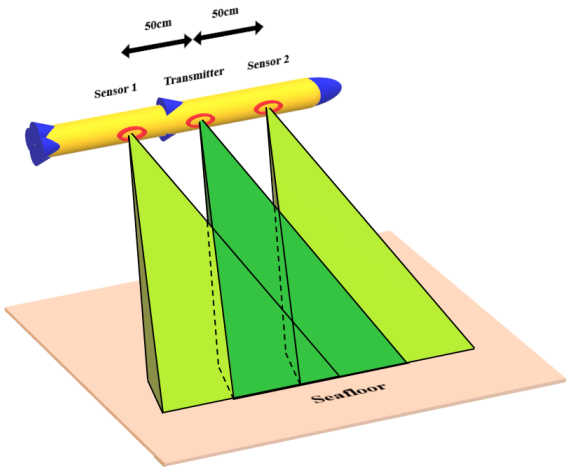

Fig. 2: The transmitter and two sensors facing perpendicularly to the direction of the underwater platform.

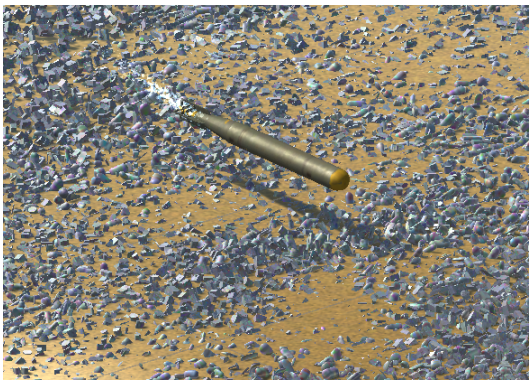

Fig. 3: Example of the simulated underwater environment used to create the data sets.

determined by an aperture angle, elevation angle, maximum range and pitch angle as shown in Fig. 1. The sonar images are generated following a hop-and-generate process, where the simulated platform generates an image at a particular position in a simulated environment, then it moves to a different position and generates another image and so on. When an image is generated, the position and orientation of the sonar sensor in the underwater environment is also stored.

The simulator in [16] only uses a single sonar sensor with the same transmit-receive antenna. We expanded the sonar simulator to separate the sonar transmitter from the receiver. Also, the capability to simulate a sonar with multiple sensors in different positions at the same time is added. A transmitter illuminates the environment while the sensors generate the sonar images. This is shown as the dark green beam in Fig. 2. Then, the platform moves to another position by following a randomly generated trajectory. An example of a simulated environment is shown in Fig. 3. The procedure for simulating the underwater environment is the same that was used in [14], where multiple scenarios with rocks randomly positioned on the seafloor were created. The generated images from each sensor and the position in the scenario where they were acquired are stored in data sets for training and validation of DL networks.

III. DL NETWORKS FOR MOTION ESTIMATION

In [14], several DL networks were evaluated for motion estimation using sonar images. We found that the PoseNet network [18] is well suited for the task after optimizing the network parameters to get best possible performance. The architecture of the optimized PoseNet is shown in Fig. 4. The input of the network is an image made of two consecutive sonar images. This input is connected to a series of 9 convolution layers with stride 2. The first 8 convolutional layers have a ReLU activation layer and batch normalization layer at their output before being connected to the next convolutional layer. The output of the last convolutional layer is connected to an average pooling layer with an averaging window of 4, then an output regression layer is connected to generate the motion estimates in 3 degrees of freedom (DoF). The regression layer uses the Mean Squared Error (MSE) loss function. In this paper we continue to use this network to validate the motion estimation using the new proposed sonar configuration.

IV. EXPERIMENTS USING ONE AND TWO SENSORS

IV-A. Sonar configurations

With the modified simulator, two sonar configurations were built:

1) *Two sensors:* One transmitter (dark green beam in Fig. 2) and two sonar sensors (two light green beams in Fig. 2) are placed on an underwater platform to side look perpendicularly to the forward motion of the platform as shown in Fig.2. The distance between the transmitter and each sensor is 50 cm. The FoV of the transmitter and the sensors is $29^\circ \times 14^\circ$ (azimuth and elevation angles, respectively), with 96 beams in the azimuth and a pitch angle of 35°. This is based on the parameters of the Didson 300 sonar [19].
2) *One Sensor:* This configuration is the same as used in [14]. One transmitter with a single sonar sensor is used with no separation between them. They both have the same FoV as described in the case of two sensors.

Three DoF are considered for the motion. The displacement of the sensors between consecutive images is described by a vector $\Delta = [\Delta_x, \Delta_y, \Delta_\theta]$, representing translations along the x and y-axes and rotation around the z-axis (denoted by θ), respectively (see Fig. 1). For this work, the maximum displacement between two images is 2.0 cm and 0.45° for the translations and rotation, respectively. The

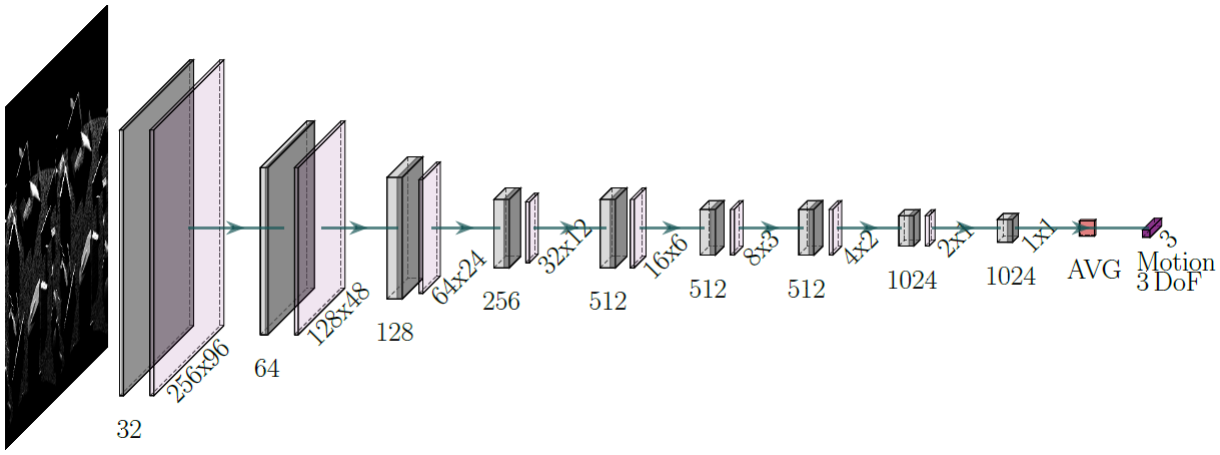

Fig. 4: Architecture of the DL network. The input is an image of size 192×512. The grey and purple squares represent the convolutional and the ReLU with batch normalization layers, respectively. The number of channels (32, 64, ..., 1024) is specified below the convolutional layers. The output size (256×96, 128×48, ..., 1×1) of a layer is specified below the ReLU function. The red square represents the average pooling layer and the rightmost purple square is the output regression layer.

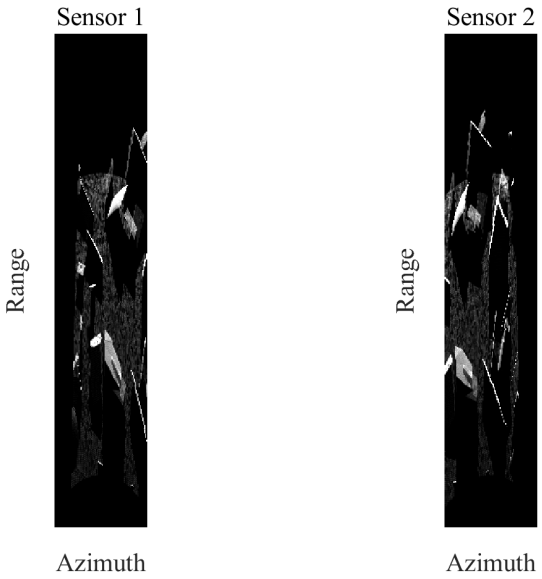

Fig. 5: Examples of sonar images generated by the two sensors at the same time.

height of the sensor from the seafloor is 2.5 m. The sonar image size is 512×96 pixels and the pixel values are integer numbers in the range from 0 to 255. Examples of sonar images generated by the sensors are shown in Fig. 5. Since they are situated on each side of the transmitter (as shown in Fig. 2), they have a slightly different point of view of the scenario. In the images, it can be seen that some area of the image is totally dark. For sensor 1, this area appears on the left and for sensor 2, the area appears on the right. This is because the FoV of the transmitter does not totally overlap with the FoV of the sensors, so this portion of the sensor's FoV does not receive signals from the transmitter.

IV-B. Training the DL network

To create the training data sets, pairs of consecutive images are concatenated into a single image. Each concatenated image is associated with a displacement label to make a training sample. The label corresponds to the vector Δ. The three elements in the labels are normalized to the range from -10 to 10 with respect to their maximum values. In this case, they have the same weight within the loss function. A data set of 20,000 pairs of concatenated images is generated for each sonar sensor.

For the two-sensor configuration, the already concatenated images from each sensor are concatenated with the corresponding concatenated images of the other sensor to make a larger image of 4 concatenated images. This larger image is put into the network. For the one-sensor case, the pairs of concatenated images are directly put into the DL network. The data sets for both the cases are split into 95% and 5% for training set and validation set, respectively.

The sonar images generated by the simulator are noiseless. We follow two approaches for training the networks. One consists in training with the noiseless images and the other consists in training with the same images, but with a low-level noise added to their pixels. The noise is generated according to two considerations [20]: (i) the pixels of acoustic shadows in the images are modified with additive Gaussian noise with the mean and standard deviation of 4% and 2% of the maximum pixel value, respectively. (ii) The rest of the pixels are affected by adding noise with the Rayleigh distribution with a scale parameter of 4% of the maximum pixel value, thus representing the scattering noise.

Table I: Validation RMSE when training the DL network with noiseless images and with low-level noise images.

Training approach	RMSE of motion estimation					
	Validation on noiseless images			*Validation on high-level noise images*		
	Δ_x *(mm)*	Δ_y *(mm)*	Δ_θ $(^\circ)$	Δ_x *(mm)*	Δ_y *(mm)*	Δ_θ $(^\circ)$
With noiseless images						
One-sensor	2.69	2.74	0.054	5.73	5.82	0.118
Two-sensor	2.17	1.42	0.044	7.47	6.13	0.156
With low-level noise images						
One-sensor	3.52	3.38	0.076	5.75	5.80	0.125
Two-sensor	3.05	2.33	0.070	4.37	3.34	0.089

After the networks have been trained, we validate the estimation accuracy using either the noiseless images or images with a high-level noise based on measures of noise in real Didson sonar images described in [21]. The high-level noise has a Gaussian distribution with the mean and standard deviation of 13.72% and 3.14% of the maximum pixel value, respectively, and a Rayleigh distribution with a scale parameter of 13.72% of the maximum pixel value.

The DL networks were trained in MATLAB. The training uses the Adam optimization algorithm [22]. The learning rate starts at 0.0001 and halves every 12 epochs until the validation loss converges. During the training, a dropout regularization with a rate of 50% is applied.

IV-C. Numerical results

Table I presents results of training the networks with noiseless and low-level noise images in the terms of the root-mean-square error (RMSE) obtained when validating with noiseless and high-level noise images.

It can be seen that a better performance is obtained by the network trained with two sensors over the network trained with one sensor. For y-axis in the training and validation with noiseless images, the RMSE for the one-sensor configuration is 2.74 mm and for the two-sensor configuration is 1.42 mm, which is a reduction of almost twice. The other parameter estimates are also improved when training with the two-sensor configuration.

The only case when the two-sensor configuration presents a higher RMSE compared to the one-sensor configuration is when training with noiseless images and validation with high-level noise images. This can be caused by the black areas on the side of each sonar image which do not provide information about the motion. It is possible that since this area is always black (0 value), the network learns to ignore that part of the images for the motion estimation, but when randomly generated noise is added, it affects the estimates. This issue is eliminated when training with noisy images, even if it is not the same level of noise that is used for validation.

For both configurations, the best estimates are for the y-axis. In [14], it is found that there is a high correlation between estimates of translation along x-axis and the rotation, which affects the estimation accuracy. Therefore estimates along x-axis and the rotation are less accurate than the estimates for the y-axis.

Training with low-level noise images reduces the RMSE of the two-sensor configuration while the RMSE of the one-sensor configuration is not reduced. This suggests that using the two-sensor configuration is more suitable for motion estimation with real data, since the noise level in this case is the same as measured from real sonar images.

When training, the DL network parameters were tuned to provide the best performance for the one-sensor case; we used the same network with the same parameters as in [14]. However, a better performance is obtained by the network trained with two sensors even without optimizing the parameters. For future work, it is possible that the performance can be improved further by tuning the DL network to optimize the two-sensor case, removing the black area on the images before putting them into the DL network, adjusting the image noise before training, and/or optimizing the configuration of the sensors such as the distance and orientation relative to the transmitter and the FoV.

V. CONCLUSIONS

In this paper we present a DL-based motion estimation that combines sonar images from two sensors rather than using images from only one sensor. This is an attempt to improve the motion estimation accuracy obtained with a single sensor. The two-sensor configuration shows an improvement in the estimation accuracy compared to the one-sensor configuration, even without tuning the DL training parameters to try to optimize the estimation. For instance, there is an RMSE reduction of almost twice for the y-axis movement, while the RMSE for the other types of movement are also reduced.

The obtained results suggest that further work with the two-sensor configuration could improve even more the motion estimation accuracy. The future work can focus on optimizing the training parameters, removing image areas with no information about the motion and/or optimizing the sensors configuration relative to the sonar transmitter.

VI. REFERENCES

[1] H. Huang, J. Tang, B. Zhang, J. Chen, J. Zhang, and X. Song, "A novel nonlinear algorithm for non-gaussian noises and measurement information loss in underwater navigation," *IEEE Access*, vol. 8, pp. 118 472–118 484, 2020.

[2] K. Xu, Z. Tian, J. Tang, J. Wang, and S. Zhang, "An INS data-based micronavigation method for the imaging of multiple receiver synthetic aperture sonar," in *10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, 2017, pp. 1–5.

[3] S. Caporale and Y. Petillot, "A new framework for synthetic aperture sonar micronavigation," *arXiv preprint arXiv:1707.08488*, 2017, accessed: 2021-02-18. [Online]. Available: https://arxiv.org/pdf/1707.08488.pdf

[4] D. C. Brown, I. D. Gerg, and T. E. Blanford, "Interpolation kernels for synthetic aperture sonar along-track motion estimation," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 4, pp. 1497–1505, 2019.

[5] A. J. Hunter, S. Dugelay, and W. L. Fox, "Repeat-pass synthetic aperture sonar micronavigation using redundant phase center arrays," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 4, pp. 820–830, 2016.

[6] V. Myers, I. Quidu, B. Zerr, T. O. Sæbø, and R. E. Hansen, "Synthetic aperture sonar track registration with motion compensation for coherent change detection," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 3, pp. 1045–1062, 2019.

[7] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1851–1858.

[8] M. Aladem and S. A. Rawashdeh, "Lightweight visual odometry for autonomous mobile robots," *Sensors*, vol. 18, no. 9, p. 2837, 2018.

[9] D. Scaramuzza, M. C. Achtelik, L. Doitsidis, F. Friedrich, E. Kosmatopoulos, A. Martinelli, M. W. Achtelik, M. Chli, S. Chatzichristofis, L. Kneip *et al.*, "Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in gps-denied environments," *IEEE Robotics & Automation Magazine*, vol. 21, no. 3, pp. 26–40, 2014.

[10] Y. Lin, F. Gao, T. Qin, W. Gao, T. Liu, W. Wu, Z. Yang, and S. Shen, "Autonomous aerial navigation using monocular visual-inertial fusion," *Journal of Field Robotics*, vol. 35, no. 1, pp. 23–51, 2018.

[11] B. Teixeira, H. Silva, A. Matos, and E. Silva, "Deep learning for underwater visual odometry estimation," *IEEE Access*, vol. 8, pp. 44 687–44 701, 2020.

[12] H. Saç, M. K. Leblebicioğlu, and G. Bozdağı Akar, "2D high-frequency forward-looking sonar simulator based on continuous surfaces approach," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 23, no. 1, pp. 2289–2303, 2015.

[13] B. T. Henson and Y. V. Zakharov, "Attitude-trajectory estimation for forward-looking multibeam sonar based on acoustic image registration," *IEEE Journal of Oceanic Engineering*, vol. 44, no. 3, pp. 753–766, 2018.

[14] J. E. Almanza-Medina, B. Henson, and Y. V. Zakharov, "Deep learning architectures for navigation using forward looking sonar images," *IEEE Access*, vol. 9, pp. 33 880–33 896, 2021.

[15] L. Rixon Fuchs, C. Larsson, and A. Gällström, "Deep learning based technique for enhanced sonar imaging," in *5th International Conference and Exhibition on Underwater Acoustics (UACE), Hersonissos, Crete, Greece*, 2019, pp. 1021–1028.

[16] J. E. Almanza-Medina, B. T. Henson, and Y. V. Zakharov, "Imaging sonar simulator for assessment of image registration techniques," in *MTS/IEEE OCEANS 2019, Seattle*, 2019.

[17] "Unity," https://unity.com/, accessed: 2021-03-01.

[18] Z. Yin and J. Shi, "GeoNet: Unsupervised learning of dense depth, optical flow and camera pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1983–1992.

[19] *DIDSON 300 Standard Version Specifications*, Sound Metrics Corp., accessed: 2021-03-03. [Online]. Available: http://www.soundmetrics.com/products/DIDSON-Sonars/DIDSON-300m/DIDSON-300-Standard-Version-Specifications.

[20] F. Schmitt, M. Mignotte, C. Collet, and P. Thourel, "Estimation of noise parameters on sonar images," in *Statistical and Stochastic Methods for Image Processing*, vol. 2823. International Society for Optics and Photonics, 1996, pp. 2–13.

[21] B. T. Henson, "Image registration for sonar applications," Ph.D. dissertation, University of York, 2017, accessed: 2020-09-14. [Online]. Available: http://etheses.whiterose.ac.uk/19536/1/thesis.pdf

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.