Deep Inertial Underwater Odometry System

Takuma Uno[1], Naoya Isoyama[1], Hideaki Uchiyama[1,*], Nobuchika Sakata[2] and Kiyoshi Kiyokawa[1]

[1] *Nara Institute of Science and Technology, Ikoma, Japan*
[2] *Ryukoku University, Otsu, Japan*

Abstract

Underwater 6-DOF odometry is an essential technology for robotics, navigation, and motion-based control systems. Existing solutions have issues to be solved: complicated sensor configurations and limited usage conditions. This paper presents a robust, stand-alone, light-weighted IMU-based underwater odometry system, referred to as deep inertial underwater odometry (DIUO). Our system is based on an inertial odometry technique based on deep learning used in the air. The main issue to be solved is the ground-truth data collection of inertial underwater odometry for supervised learning. Therefore, we design a stick-shaped jig that rigidly connects a fiducial marker and an IMU for the data collection. Inertial underwater odometry is computed by tracking the marker with a motion capture system in the air and using the rigid transformation from the marker to the IMU. Furthermore, the performance analysis of DIUO was conducted with several motions in the evaluation. Finally, we discuss limitations and perspectives.

Keywords

IMU, odometry, underwater, deep learning, data collection

1. Introduction

Underwater 6-DOF odometry is an essential technology for motion-based control systems. For example, autonomous underwater robots compute the odometry by fusing sonars and several sensors when they perform the localization tasks [1, 2, 3, 4, 5]. Especially, the Doppler shift of reflected and scattered waves from the seafloor and the underwater is used to estimate absolute speed or the speed relative to water. However, the only 3-DOF position is computed from sonars. Also, the estimation accuracy is affected by the surrounding conditions, such as shapes and materials. Another application based on underwater odometry is a virtual reality (VR) system for enhancing the swimming experience [6]. Multiple cameras are installed in a pool to track the head poses. Then, VR content is visualized according to the poses. However, the system is available only within the observable range of the cameras under the transparent water. Existing underwater 6-DOF odometry solutions have several limitations: complicated sensor configuration and limited usage conditions.

As a device for overcoming the limitations, we focus on an IMU that measures the device's motion with acceleration and angular velocity. The surrounding conditions do not affect IMU

IPIN 2022 WiP Proceedings, September 5 - 7, 2022, Beijing, China

*Corresponding author.

uno.takuma.ur1@is.naist.jp (T. Uno); isoyama@is.naist.jp (N. Isoyama); hideaki.uchiyama@is.naist.jp (H. Uchiyama); sakata@rins.ryukoku.ac.jp (N. Sakata); kiyo@is.naist.jp (K. Kiyokawa)

data measurement compared with vision-based sensors. Also, IMU is a more light-weighted and energy-saving device. IMU-based odometry with deep learning, referred to as deep inertial odometry, has been proposed [7, 8, 9, 10, 11, 12, 13, 14]. It trains a neural network to infer IMU odometry from IMU readings in supervised learning. Since its accuracy has approached vision-based methods, deep inertial odometry can be one promising solution for localization and navigation in the future. Therefore, we extend such a deep inertial odometry technique to 6-DOF underwater one.

One important issue for inertial odometry based on supervised learning is IMU odometry measurement for creating the training dataset. Generally, a motion capture system or a visual SLAM technique on a smartphone is used to collect the ground-truth IMU odometry in general environments such as rooms or buildings [15, 10]. However, it is difficult to use such vision-based systems underwater due to refraction and other effects. Therefore, designing the data collection framework for underwater inertial odometry is a crucial issue. Furthermore, the performance of 6-DOF underwater inertial odometry was not well-analyzed in the literature. This analysis will open up a new seamless indoor navigation application under underwater conditions, including a pool and an aquarium.

This paper presents a system for achieving deep inertial underwater odometry (DIUO). Specifically, we propose a framework to measure underwater odometry by designing a stick-shaped jig that rigidly connects a rigid marker and an IMU. In other words, using the jig, we can compute underwater odometry by tracking the marker with a motion capture system in the air. We construct our dataset with several motions to analyze the performance of DIUO. The evaluation shows that DIUO can be one promising solution for underwater localization and navigation. In addition, the performance was further analyzed when the training one was constructed differently from the test one to investigate motion transfer. This analysis clarified the problem of domain gaps between motions for future work.

In summary, this paper has the following contributions: (i) we propose to solve the underwater odometry problem by using an IMU, (ii) we design several jigs to correct underwater odometry data for supervised learning, and (iii) we evaluate the performance of DIUO and discuss limitations and perspectives.

2. Related Work

2.1. Odometry in 3D

General odometry techniques in 3D space can be classified into two categories: those that use devices installed in the environment and those that use sensors attached to a tracking device itself [16]. The former includes motion capture systems [17, 18, 19], referred to as outside-in tracking. First, cameras are fixed in the environment and calibrated. Next, fiducial markers are installed on the target tracking device. Finally, the device odometry is computed by observing it with multiple fixed cameras. A similar system using magnetic sensors has been proposed [20]. Also, visible lights are installed under the water to compute the pose by using a camera [21]. The underwater VR system was developed with a special underwater motion capture system [6]. An example of the latter is visual-inertial odometry [22, 23]. It uses both a camera and an IMU as input, referred to as inside-out tracking. Other odometry methods in this category have been proposed such as using LiDAR [24], wireless LAN [25, 26], and IMU [7, 8, 9, 10, 11, 13, 14]. Several solutions are extended to apply to underwater conditions [27, 28, 29].

For underwater odometry, the potential of adopting each technique is discussed here. We envision the applications of underwater odometry in various environments, such as a swimming pool and the ocean. For this reason, we targeted the aforementioned latter category that does not need to install equipment in the environment because the installation may not be possible at some locations. Since wireless technologies cannot be used or are less stable underwater, the methods using cameras, LiDAR, and IMU were further considered. For cameras and LiDAR, the estimation accuracy generally depends on the surrounding conditions. It is degraded due to the difference in the refractive index when the motion is between under and above the water. Finally, we focused on IMU because the data measurement is independent of the surrounding environments. Next, the detail of inertial odometry is further discussed.

2.2. Deep inertial odometry

The computation of classical inertial odometry based on kinematics consists of orientation computation by single-integration of angular velocity and position computation by double-integration of acceleration after removing gravity. The acceleration and angular velocity measured in MEMS-based IMU generally contain noise and bias. Also, the error on gravity removal is an issue. The design of a methodology to suppress the error has been investigated. For example, global navigation satellite system (GNSS) is used to compensate for positional error [30] Also, pedestrian dead reckoning (PDR) based on IMU uses constraints on walking behavior [31]. However, it is not easy to use GNSS or PDR underwater. Therefore, we focus on a method that can compensate for the error using deep learning owing to its recent significant advancement.

Several approaches on deep inertial odometry have been proposed [7, 8, 9, 10, 11, 12, 13, 14]. Generally, the basic idea is to construct a motion model with a neural network. In other words, the network is trained as a mapping function from IMU readings to IMU odometry. This means that IMU odometry can be computed only from the IMU readings in the inference process. One important aspect is that end-to-end deep inertial odometry may ignore the kinematics. The motion model can be built from the training data only. In other words, the neural network implicitly models the phenomenon of motion, bias, noise, and other factors from the dataset. Since TLIO is an open-source project [11], we develop our deep inertial underwater odometry system based on it.

The method in TLIO is summarized here. It uses both deep inertial odometry and Kalman filtering. In the training process, a neural network is trained with many pairs of IMU displacement and IMU readings. In the inference process, TLIO first computes IMU poses by double-integration of acceleration and single-integration of angular velocity, based on kinematics. In parallel, the 3D displacment is inferred by a neural network with IMU reading. Finally, the inferred displacement is incorporated into an extended Kalman filter framework as an observation to compensate for the error in kinematics-based 6-DOF inertial odometry.

3. Underwater kinematics

Classical inertial odometry is computed by using kinematics with acceleration and angular velocity. Since the acceleration contains gravity as an internal force, it is necessary to remove it from the acceleration before computing the odometry. As a theory of underwater inertial odometry, the identity of kinematics in the air and underwater is briefly investigated.

When using an IMU, the physical quantities related to motion are acceleration caused by external and internal forces and angular velocity. Gravity exists as an internal force in the air and underwater on earth. Compared with the air, underwater conditions additionally have buoyancy and viscous forces. They can be categorized into external forces. In other words, there is no additional internal force underwater to be removed from the acceleration. Therefore, the kinematic equations in the air can be used for underwater conditions.

4. Proposed system

4.1. Overview

We propose a system for achieving DIUO using TLIO. For inertial underwater odometry based on supervised learning, it is crucial to collect the ground-truth odometry while acquiring IMU readings. We design jigs for the data collection and their simple calibration technique. After collecting the data, the neural network is trained and is used for inferring IMU odometry from IMU readings. Note that the jigs are necessary for the training process only. Therefore, inertial odometry is computed from IMU readings only for the inferance process.

4.2. Devices for data collection

The data collection process consists of IMU data measurement underwater, rigid marker tracking in the air, and underwater inertial odometry computation. As a system configuration, the devices for data collection are explained here.

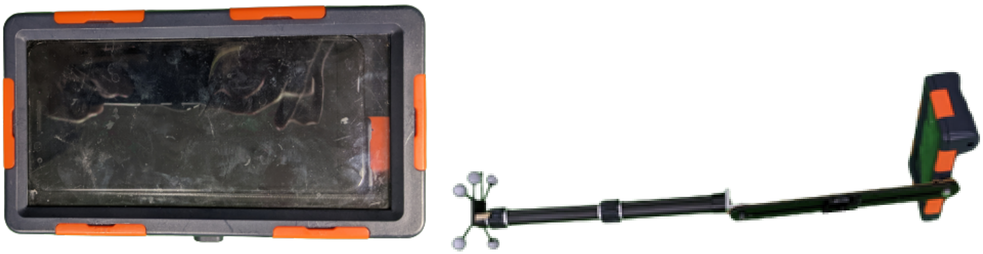
First, it is necessary to measure the readings of an IMU attached to a moving device underwater. As a prototype system assuming underwater VR goggles, we used a Google's Pixel 3a smartphone as an IMU data logger and put it in a waterproof box, as illustrated in Figure. 1(a). The implementation of the data logger is based on SensorManager provided in Andriod's sensor API[1].

Second, our proposed system uses a motion capture system to track a rigid marker. In our prototype, we used NaturalPoint's OptiTrack and Motive software[2]. A rigid marker used in OptiTrack is composed of multiple spherical markers, as illustrated in the left part of Figure. 1(b).

Third, we design a jig to rigidly connect an IMU with a rigid marker by using a bar as illustrated in Figure. 1(b). While moving the IMU underwater, the rigid marker is moved in the air and tracked. By using the calibration explained in Section. 4.3, the marker pose in the air can be converted into the underwater IMU pose . This conversion enables the data collection of IMU odometry as ground truth while recording IMU readings underwater.

[1] https://developer.android.com/reference/android/hardware/SensorManager

[2] https://optitrack.com/



(a) Waterproof box for smartphone

(b) Connection jig

Figure 1: Jigs for underwater data collection

4.3. Calibration

After the data collection, we convert maker poses into IMU poses for computing IMU odometry. The conversion is based on the following calibration.

4.3.1. Coordinate systems

First, we define the coordinate system and pose variables used in our system in Figure. 2. The coordinate system fixed to an IMU is referred to as IMU coordinate system $\mathbf{X}_\mathrm{i}$. The marker has its marker coordinate system $\mathbf{X}_\mathrm{m}$. OptiTrack has the world coordinate system $\mathbf{X}_\mathrm{o}$. The marker pose represented by $\mathbf{M}_\mathrm{m}$ containing rotation matrix and translation vector in a 4×4 matrix is defined as

$$\tilde{\mathbf{X}}_\mathrm{m} = \mathbf{M}_\mathrm{m}\tilde{\mathbf{X}}_\mathrm{o}$$

where $\tilde{}$represents a homogeneous coordinate. In OptiTrack, the origin of $\mathbf{X}_\mathrm{m}$ is defined to the center of gravity of a rigid marker. The axis direction of $\mathbf{X}_\mathrm{m}$ is also defined to be the same as that of $\mathbf{X}_\mathrm{o}$. In other words, both coordinate systems share the same axis direction at the beginning.

Inertial odometry is the process to compute $\mathbf{M}_\mathrm{i}$ while OptiTrack provides $\mathbf{M}_\mathrm{m}$. Since IMU and a rigid marker are rigidly fixed, $\mathbf{M}_\mathrm{mi}$ is also fixed. Therefore, $\mathbf{M}_\mathrm{m}$ can be computed from $\mathbf{M}_\mathrm{i}$ once $\mathbf{M}_\mathrm{mi}$ is calibrated.

4.3.2. Jig based calibration

We design a jig to align the IMU coordinate system into the world one, as illustrated in Figure. 3. The jig comprises two bars, which are perpendicularly aligned and attached to a waterproof box containing a smartphone. At the edges and intersection of the bars, spherical markers are placed. These markers are used to define the origin and axis direction of the world coordinate system. Then, the edges of the waterproof box are aligned with the bars. Finally, the axis direction of the two coordinate systems can be aligned.

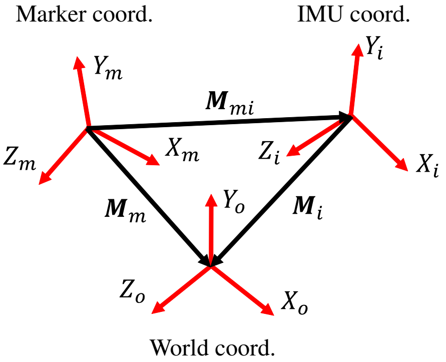


Figure 2: Definition of coordinate systems and pose variables

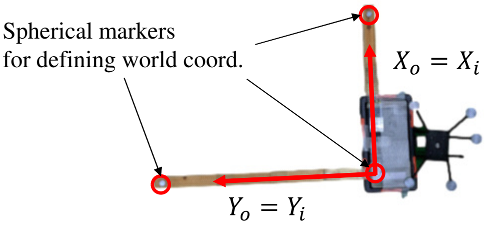


Figure 3: Calibration jig

By using the jig, the variables in Figure. 2 satisfies

$$\begin{aligned} \mathrm{M_i} &= \mathbf{I} \\ \mathrm{M_{mi}} &= \mathrm{M_m} \end{aligned}$$

because the world coordinate system is the same as the IMU one. Therefore, we simply acquire $\mathrm{M_m}$ from OptiTrack once in the calibration step. After removing the two bars from the calibration jig, we can freely move the connection jig to collect IMU odometry. While moving the smartphone underwater, the rigid marker is tracked with a motion capture system in the air to acquire $\mathrm{M_m}$. Finally, we can compute $\mathrm{M_i}$ from $\mathrm{M_m}$ and $\mathrm{M_{mi}}$. Note that the calibration accuracy depends on the engineering quality of the jig. The method for mathematically optimizing the calibration result is our future work.

4.4. Underwater inertial odometry with TLIO

In TLIO [11], the neural network to infer IMU displacement from IMU readings is first trained with the training dataset. Specifically, the network is built as a mapping function from IMU readings within one second to IMU displacement within 100 milliseconds.

In the inference process, IMU readings during underwater motion is first processed to compute inertial odometry with kinematics. In parallel, it is input to the network to infer the displacement. Finally, the inferred displacement is used to correct the error in kinematics-based inertial

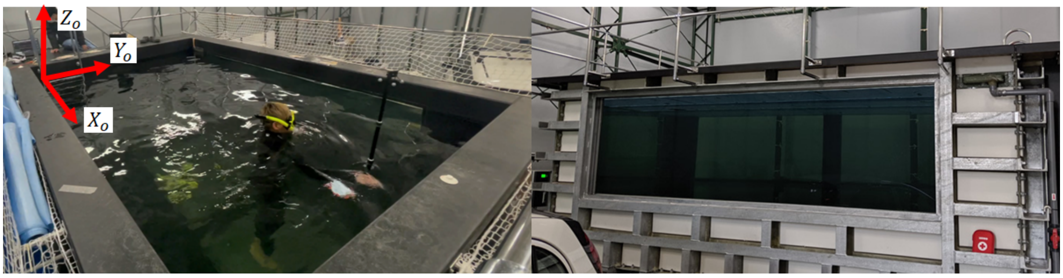


Figure 4: Experiment environment

odometry with Kalman filtering. By using supervised learning, underwater odometry can be achieved only with IMU readings after training the network.

5. Evaluation

5.1. Overview

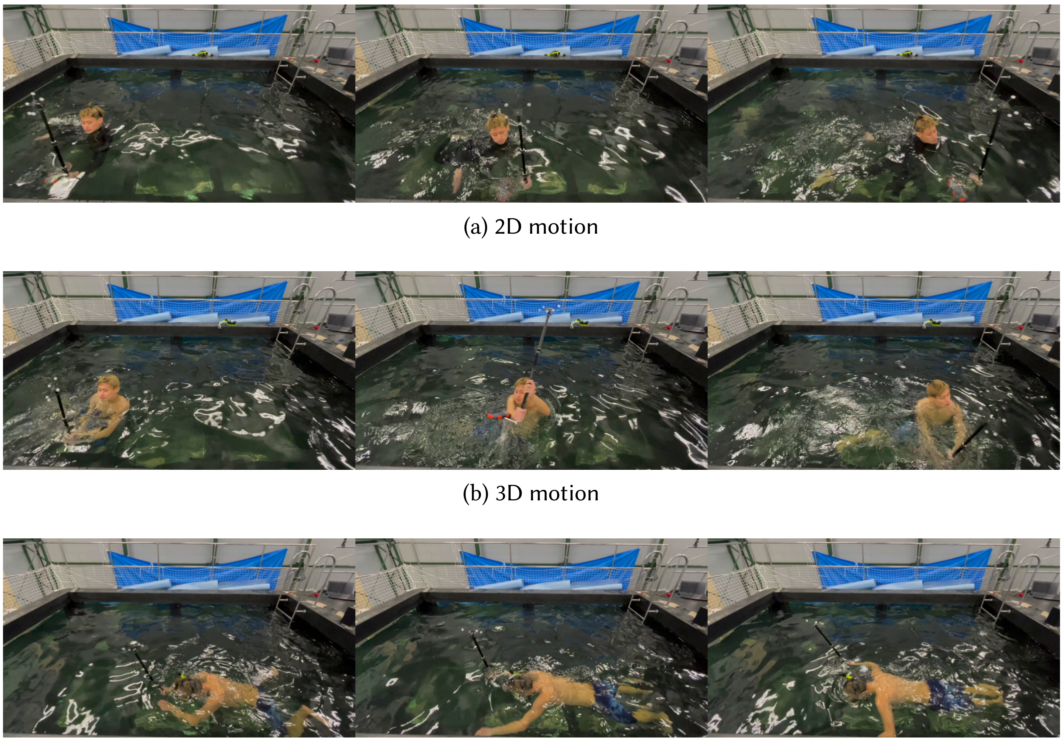
We construct the dataset with several motions and investigate the performance of DIUO. We used a laboratory pool, as illustrated in Figure. 4. The size is the length of 4 meters, the width of 2 meters, and the height of 1.5 meters. In the pool, water flow can be controlled by external forces. However, its functionality was not used in the experiment.

The cameras for OptiTrack were installed in the four corners of the fence around the pool. The camera view angles were carefully designed because marker tracking sometimes failed due to the reflection of spherical markers on the water surface. The cameras were calibrated to set the origin of the world coordinate system at one of the corners in the pool. Note that we followed the safety rules approved by the university during the experiments when a subject got into the pool.

5.2. Data collection

We captured the following three motions: 2D motion, 3D motion, and breaststroke swimming, as illustrated in Figure. 5. The 2D motion was selected as a simple motion by referring to the trolley dataset in OxIOD [15]. Since it was manually moved, the motion was performed approximately on a plane. The 3D motion was designed to verify if it is possible to seamlessly infer the movement between underwater and above water. It is not easy to track this motion with existing solutions, as discussed in Section. 2.1. Breaststroke swimming was selected as our potential future application.

The same subject collected all of the data by holding the jig in Figure. 1(b) and moving it according to the purpose of each motion. The length of each motion data was approximately one hour. IMU readings was saved at 400 Hz while marker poses from OptiTrack were saved at 120 Hz. Both framerates were upsampled at 1000 Hz by using linear interpolation to satisfy the input format of TLIO.



(a) 2D motion

(b) 3D motion

(c) Breaststroke swimming

Figure 5: Data collection for performance evaluation

5.3. Results

We conducted the performance analysis using our dataset for each motion. For the data split between training and testing, one-minute test data was randomly selected from the collected 60-minutes data for each motion. The remaining data was used for the training. This simple leave-one-out way was selected to avoid including the test data in the training one. The termination of the training was heuristically determined according to the convergence of the loss curve.

The result of each test data was illustrated in terms of position and orientation errors in Figure. 6. Since TLIO provides displacement estimation based on neural network, the result of TILO, displacement concatenation in TLIO as TLIO (NN only), and OptiTrack as ground truth are visualized. TLIO (NN only) is not include in the orientation estimations because it cannot compute the orientation. The horizontal axis is time in second in all of the figures.

Overall, the results of TLIO gradually deviate from the ground truth over time. This phenomenon is a natural property of odometry due to error accumulation. The orientation error is relatively small compared with the positional one. This result represents that the position estimation is more difficult in inertial odometry. In Figure. 6(c), the level of the water surface was 0 on Z axis. As discussed in Section. 3, the result shows that DIUO can track IMU poses under and above the water. Figure. 6(e) shows that even small movements of the hand during the swimming can be tracked owing to the framerate of the IMU. For some results, TLIO was

Table 1
Absolute and relative errors on trajectory and yaw angle

	2D		3D		Swim	
Criterion	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.
Trajectory (meters)	0.78	0.14	1.00	0.17	1.62	0.16
Yaw (degrees)	3.90	1.11	3.12	2.97	3.16	2.85

worse than TLIO (NN only). This is sometimes caused by the insufficiency of calibration and optimization of parameters used in Kalman filtering.

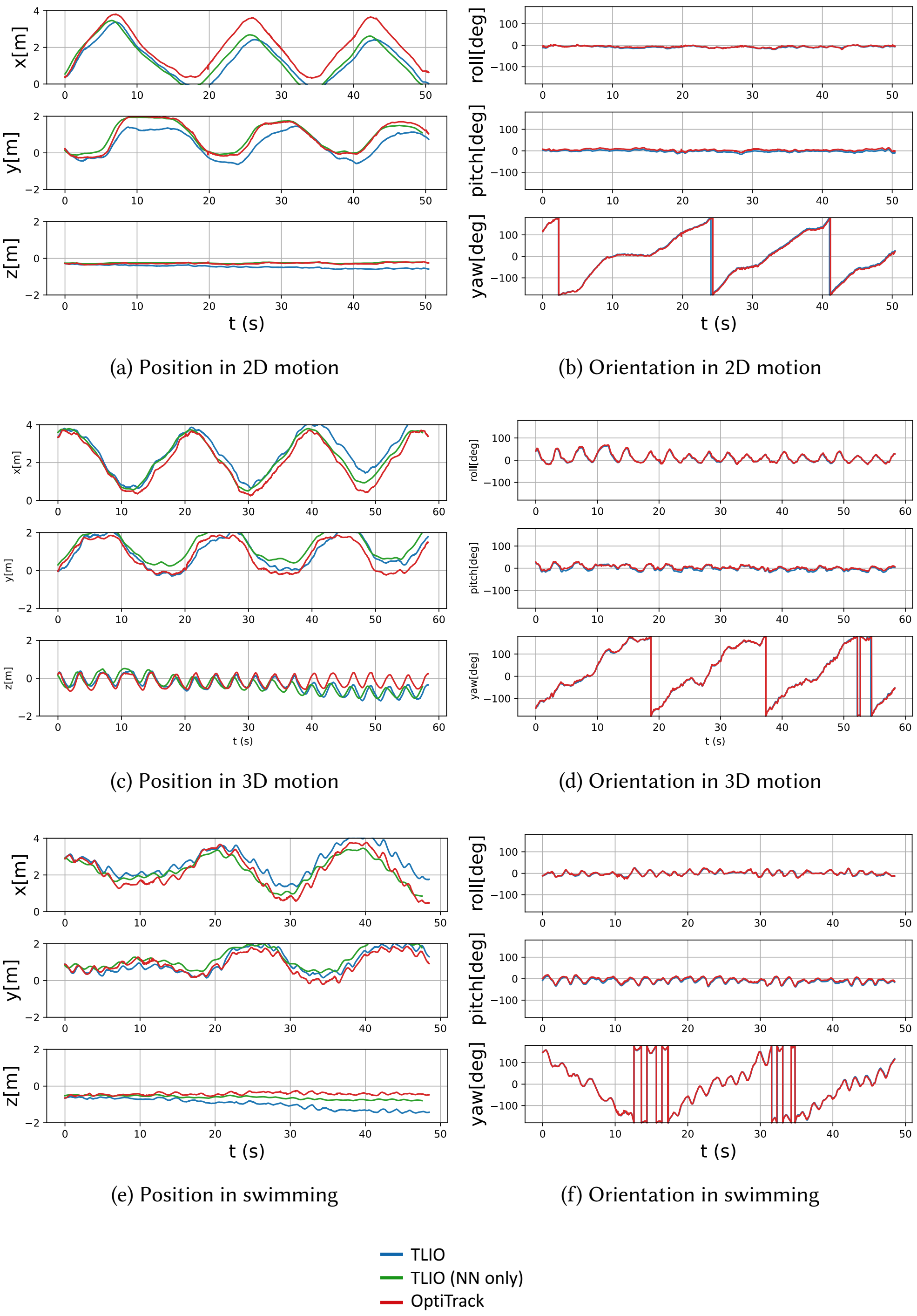
For quantitative evaluation, we show absolute and relative errors used in TLIO in Table. 1. The absolute error is computed by using the root mean square deviation (RMSE) between the ground truth and the predicted result. Since this error varies with the data length, it is not fair to compare the results when the data length and motion speed are different. In addition, we also compute the relative error, which is the RMSE of the locally-defined error such as the RMSE of the displacement error in one second. This criterion does not depend on the data length. For the relative error, 2D motion was the smallest while 3D motion was the largest. Since the swimming motion was approximately performed on a plane, the result was close to 2D motion. Even though the result of the 3D motion was the worst, the difference between the others was a few centimeters.

5.4. Discussion on data collection

It is preferable to create a dataset in the air in practice because creating underwater datasets is not easy. We investigate the performance of DIUO with the training dataset created in the air. In other words, the training dataset is created in the air while the test one is created underwater. For the data collection in the air, the subject moved the jig by mimicking the underwater motion.

The results are illustrated in Figure. 8. TLIO gave huge position error because the displacement estimation inferred by TLIO was not accurate. Therefore, the estimated trajectory is completely dissimilar from the ground truth. Even though the result on the Z axis appeared to be accurate, this is because the motion on Z axis was almost constant in the training dataset due to 2D motion. Surprisingly, the rotation error was small. Wrong observation in Kalman filtering did not affect the performance of orientation estimation.

We further investigated the reason for the inaccuracy. Figure. 9 shows the acceleration distribution on X and Y axes for the motion in the air and the one underwater. Even though the subject tried to move the jig in the same manner, the distribution range was different. It is important to carefully prepare the data distribution for training and inference in supervised learning. If the size of the dataset is huge such as dozens of hours of data, it may be possible to cover all of the possible motions. With such a dataset, the stability of the inference accuracy increases [11, 10]. However, collecting underwater odometry is not an easy task, even with our jigs. One possible direction is to correct the data with autonomous underwater robots. Another one would be to use a domain adaptation technique to fill the gap between the training data and the test one in our future work [32].



(a) Position in 2D motion

(b) Orientation in 2D motion

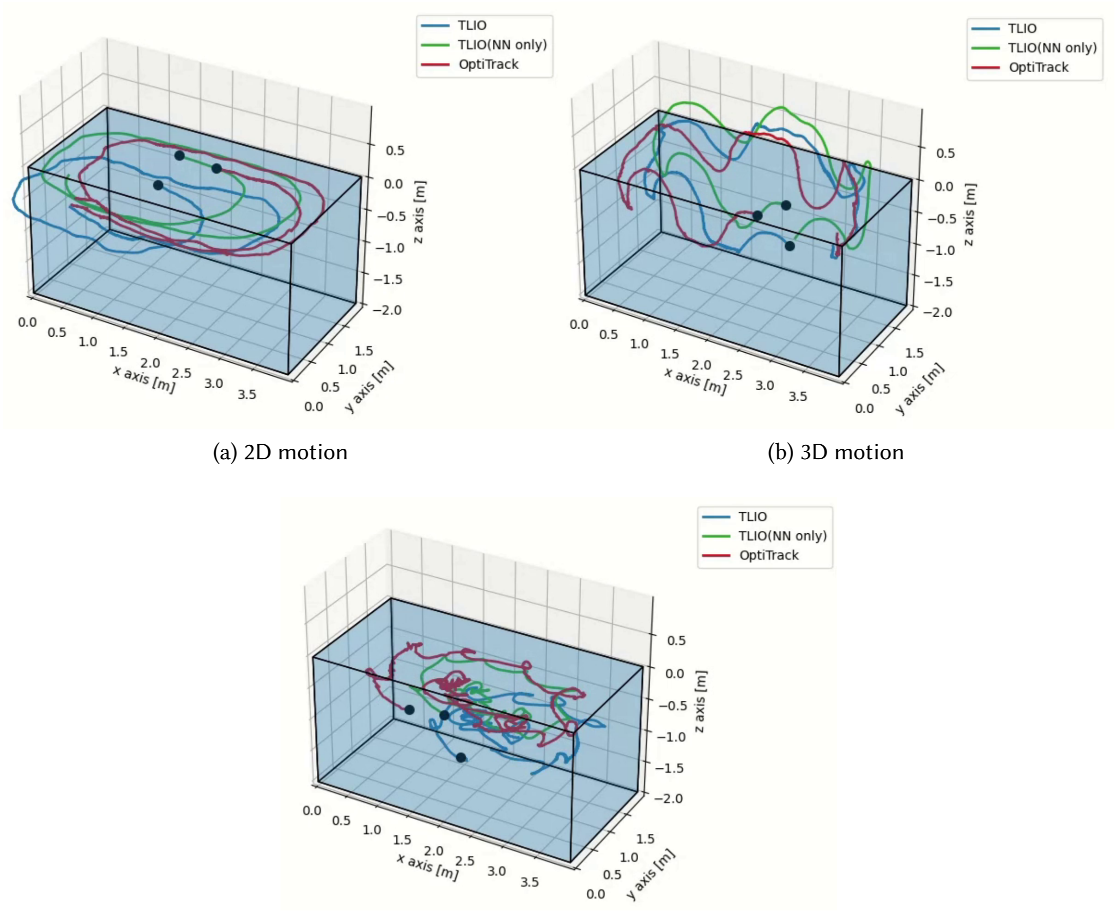
(c) Position in 3D motion

(d) Orientation in 3D motion

(e) Position in swimming

(f) Orientation in swimming

Figure 6: Comparison among TLIO, displacement concatenation in TLIO (NN only), and OptiTrack



(a) 2D motion

(b) 3D motion

(c) Breaststroke swimming

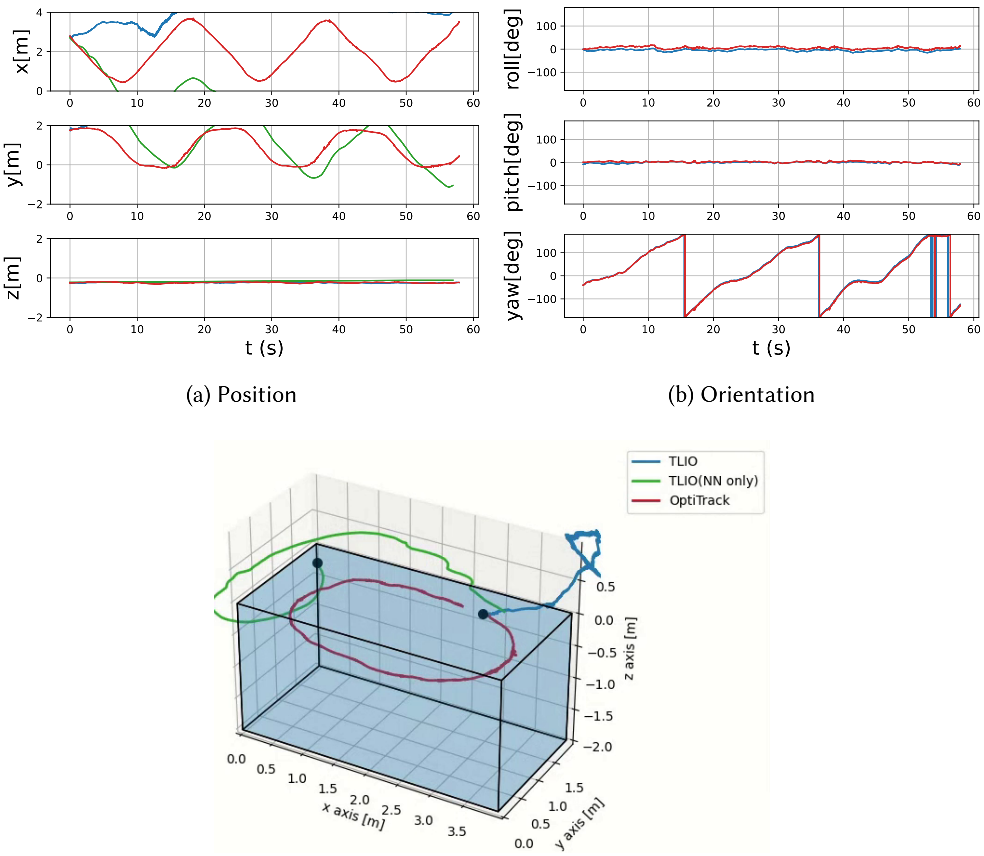
Figure 7: Trajectory visualization

6. Conclusion

We proposed a system for deep inertial underwater odometry, referred to as DIUO. We designed and calibrated a stick-shaped jig that rigidly connects a fiducial marker and an IMU for the data collection used in supervised learning. In the evaluation, the performance of DIUO was investigated with several motions: 2D motion, 3D motion, and swimming. The inference was accurate if the training dataset covered the test one. However, the accuracy of DIUO with the training dataset created in the air degraded when the motion dataset were manually created. Therefore, filling the domain gap between the two datasets is a remaining issue.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP20K11891.

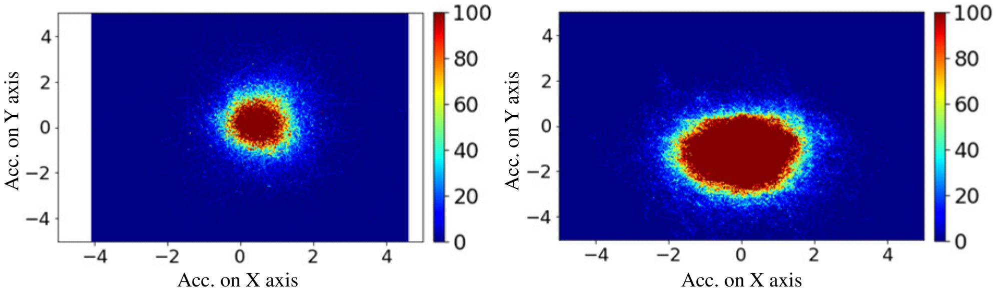


(a) Position

(b) Orientation

(c) Trajectory

Figure 8: Result with the dataset created in the air



(a) In the air

(b) Underwater

Figure 9: Acceleration distribution in 2D motion

References

[1] A. Mallios, P. Ridao, D. Ribas, F. Maurelli, Y. Petillot, Ekf-slam for auv navigation under probabilistic sonar scan-matching, in: International Conference on Intelligent Robots and Systems, IEEE, 2010, pp. 4404–4411.

[2] J. Aparicio, A. Jiménez, F. J. Alvarez, C. De Marziani, J. Urena, C. Diego, Underwater acoustic relative positioning system based on complementary set of sequences, in: International Conference on Indoor Positioning and Indoor Navigation, 2011, pp. 21–23.

[3] Y. Yang, G. Huang, Acoustic-inertial underwater navigation., in: ICRA, 2017, pp. 4927–4933.

[4] J. H. Kepper, B. C. Claus, J. C. Kinsey, A navigation solution using a mems imu, model-based dead-reckoning, and one-way-travel-time acoustic range measurements for autonomous underwater vehicles, Journal of Oceanic Engineering 44 (2018) 664–682.

[5] J. González-García, A. Gómez-Espinosa, E. Cuan-Urquizo, L. G. García-Valdovinos, T. Salgado-Jiménez, J. A. E. Cabello, Autonomous underwater vehicles: Localization, navigation, and communication for collaborative missions, Applied sciences 10 (2020) 1256.

[6] S. Yamashita, X. Zhang, T. Miyaki, J. Rekimoto, Aquacave: An underwater immersive projection system for enhancing the swimming experience., in: ICAT-EGVE, 2016, pp. 25–28.

[7] C. Chen, X. Lu, A. Markham, N. Trigoni, Ionet: Learning to cure the curse of drift in inertial odometry, in: AAAI Conference on Artificial Intelligence, 2018.

[8] J. P. Silva do Monte Lima, H. Uchiyama, R.-i. Taniguchi, End-to-end learning framework for imu-based 6-dof odometry, Sensors 19 (2019) 3777.

[9] S. Chen, Y. Zhu, X. Niu, Z. Hu, Improved window segmentation for deep learning based inertial odometry, in: International Performance Computing and Communications Conference, 2020, pp. 1–7.

[10] S. Herath, H. Yan, Y. Furukawa, Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, & new methods, in: International Conference on Robotics and Automation, 2020, pp. 3146–3152.

[11] W. Liu, D. Caruso, E. Ilg, J. Dong, A. I. Mourikis, K. Daniilidis, V. Kumar, J. Engel, Tlio: Tight learned inertial odometry, IEEE Robotics and Automation Letters 5 (2020) 5653–5660.

[12] Q. A. Dugne-Hennequin, H. Uchiyama, J. P. S. D. M. Lima, Understanding the behavior of data-driven inertial odometry with kinematics-mimicking deep neural network, IEEE Access 9 (2021) 36589–36619.

[13] S. Sun, D. Melamed, K. Kitani, Idol: Inertial deep orientation-estimation and localization, arXiv preprint arXiv:2102.04024 (2021).

[14] C. Zhou, X. Cao, D. Zeng, Y. Wang, Rio: Rotation-equivariance supervised learning of robust inertial odometry, arXiv preprint arXiv:2111.11676 (2021).

[15] C. Chen, P. Zhao, C. X. Lu, W. Wang, A. Markham, N. Trigoni, Oxiod: The dataset for deep inertial odometry, arXiv preprint arXiv:1809.07491 (2018).

[16] L. Zheng, W. Zhou, W. Tang, X. Zheng, A. Peng, H. Zheng, A 3d indoor positioning system based on low-cost mems sensors, Simulation Modelling Practice and Theory 65 (2016) 45–56.

[17] A. Dasgupta, Y. Nakamura, Making feasible walking motion of humanoid robots from

human motion capture data, in: International Conference on Robotics and Automation, volume 2, 1999, pp. 1044–1049.

[18] M. Windolf, N. Götzen, M. Morlock, Systematic accuracy and precision analysis of video motion capturing systems—exemplified on the vicon-460 system, Journal of biomechanics 41 (2008) 2776–2780.

[19] M. Field, D. Stirling, F. Naghdy, Z. Pan, Motion capture in robotics review, in: International Conference on Control and Automation, 2009, pp. 1697–1702.

[20] S. Yabukami, H. Kikuchi, M. Yamaguchi, K. Arai, K. Takahashi, A. Itagaki, N. Wako, Motion capture system of magnetic markers using three-axial magnetic field sensor, IEEE transactions on magnetics 36 (2000) 3646–3648.

[21] M. Hammouda, A. M. Vegni, V. Loscrí, On the noise effect of fingerprinting-based positioning errorin underwater visible light networks, Sensors 21 (2021) 5398.

[22] Z. Zhang, D. Scaramuzza, A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry, in: International Conference on Intelligent Robots and Systems, 2018, pp. 7244–7251.

[23] T. Qin, P. Li, S. Shen, Vins-mono: A robust and versatile monocular visual-inertial state estimator, IEEE Transactions on Robotics 34 (2018) 1004–1020.

[24] C. Debeunne, D. Vivet, A review of visual-lidar fusion based simultaneous localization and mapping, Sensors 20 (2020) 2068.

[25] B. Ferris, D. Fox, N. D. Lawrence, Wifi-slam using gaussian process latent variable models., in: IJCAI, 2007, pp. 2480–2485.

[26] L. Bruno, P. Robertson, Wislam: Improving footslam with wifi, in: International Conference on Indoor Positioning and Indoor Navigation, 2011, pp. 1–10.

[27] Y. Cho, A. Kim, Visibility enhancement for underwater visual slam based on underwater light scattering model, in: International Conference on Robotics and Automation, IEEE, 2017, pp. 710–717.

[28] X. Su, I. Ullah, X. Liu, D. Choi, A review of underwater localization techniques, algorithms, and challenges, Journal of Sensors 2020 (2020).

[29] E. Vargas, R. Scona, J. S. Willners, T. Luczynski, Y. Cao, S. Wang, Y. R. Petillot, Robust underwater visual slam fusing acoustic sensing, in: International Conference on Robotics and Automation, 2021, pp. 2140–2146.

[30] J. Kim, S. Sukkarieh, et al., 6dof slam aided gnss/ins navigation in gnss denied and unknown environments, Positioning 1 (2005).

[31] Y. Wu, H.-B. Zhu, Q.-X. Du, S.-M. Tang, A survey of the research status of pedestrian dead reckoning systems based on inertial sensors, International Journal of Automation and Computing 16 (2019) 65–83.

[32] A. Farahani, S. Voghoei, K. Rasheed, H. R. Arabnia, A brief review of domain adaptation, Advances in Data Science and Information Engineering (2021) 877–894.