Seamless Underwater Navigation with Limited Doppler Velocity Log Measurements

Nadav Cohen ⬥ and Itzik Klein ⬥

Abstract—Autonomous Underwater Vehicles (AUVs) commonly utilize an inertial navigation system (INS) and a Doppler velocity log (DVL) for underwater navigation. To that end, their measurements are integrated through a nonlinear filter such as the extended Kalman filter (EKF). The DVL velocity vector estimate depends on retrieving reflections from the seabed, ensuring that at least three out of its four transmitted acoustic beams return successfully. When fewer than three beams are obtained, the DVL cannot provide a velocity update to bind the navigation solution drift. To cope with this challenge, in this paper, we propose a hybrid neural coupled (HNC) approach for seamless AUV navigation in situations of limited DVL measurements. First, we drive an approach to regress two or three missing DVL beams. Then, those beams, together with the measured beams, are incorporated into the EKF. We examined INS/DVL fusion both in loosely and tightly coupled approaches. Our method was trained and evaluated on recorded data from AUV experiments conducted in the Mediterranean Sea on two different occasions.The results illustrate that our proposed method outperforms the baseline loosely and tightly coupled model-based approaches by an average of 96.15%. It also demonstrates superior performance compared to a model-based beam estimator by an average of 12.41% in terms of velocity accuracy for scenarios involving two or three missing beams. Therefore, we demonstrate that our approach offers seamless AUV navigation in situations of limited beam measurements.

Index Terms—Autonomous Underwater Vehicle, Doppler Velocity Log, Extended Kalman Filter, Deep Learning

I. INTRODUCTION

NAVIGATING underwater environments has garnered significant attention in recent years due to its importance across scientific missions and various industries, such as oil and gas exploration, marine, subsea, and military operations. Underwater navigation poses challenges because traditional methods reliant on global navigation satellite system (GNSS) signals are ineffective, as these signals cannot penetrate water. The most promising and used sensors for autonomous underwater vehicle (AUV) navigation are the inertial navigation system (INS) and a Doppler velocity log (DVL) [1].

As a self-contained system, the strapdown INS can offer continuous high-rate updates of the platform's position, velocity, and orientation. The INS derives its navigation solution by integrating the inertial data from the inertial sensors, which are susceptible to inherent errors and noises. These factors contribute to the accumulation of errors in the navigation solution over time [2]. To mitigate these errors, an aiding sensor in the form of the DVL is utilized. It operates by transmitting four acoustic beams to the seabed. Upon receiving

N. Cohen and I. Klein are with the Hatter Department of Marine Technologies, Charney School of Marine Sciences, University of Haifa, Israel. Corresponding author: ncohe140@campus.haifa.ac.il (N.Cohen)

the reflected beams, the Doppler effect is utilized to calculate the velocity vector of the platform by evaluating the frequency shift between the transmitted and received beams [3]. To effectively integrate these sensors, a nonlinear filter is employed due to the nonlinear characteristics of the INS equations of motion. Common choices include the extended Kalman filter (EKF) or the unscented Kalman filter (UKF) [4].

The DVL operates under a condition known as "bottom lock," which occurs when at least three out of the four transmitted beams are reflected back to the sensor [5]. When this condition is met, the velocity vector updates provided by the DVL are integrated with the INS using a loosely coupled (LC) approach. This involves initially calculating the velocity vector of the platform through parameter estimation using the beam velocity measurements and the least squares estimator. Subsequently, once the velocity vector is obtained, it is incorporated into the nonlinear filter to mitigate the inertial errors. In real-world scenarios, the bottom lock condition is not always achieved, rendering the LC approach impractical as the LS estimator struggles to estimate Cartesian velocity. To circumvent this challenge, the raw DVL measurements, specifically the beam velocity readings, can be directly integrated into the nonlinear filter using a tightly coupled (TC) approach. This approach is adaptable to any number of beams, ranging from one to four. However, as the number of beams decreases, the system's capability to mitigate navigation errors diminishes [6].

Scenarios involving partial DVL measurements can occur due to various factors, including extreme roll and pitch maneuvers like while diving, passing over underwater structures such as trenches and rocks that may deflect the beams, and obstructions such as marine wildlife blocking the sensor's view. A visualization of these scenarios can be seen in Fig.1. To tackle these scenarios, research has started to emerge, beginning with Tal *et al.*, who proposed an extended loosely coupled (ELC) approach [7]. This method utilizes partial DVL beam measurements along with external information to compute the three-dimensional velocity vector of the AUV, which is then fed into the navigation nonlinear filter. Additionally, in [8], the authors proposed a TC implementation of INS/DVL along with a pressure sensor to address scenarios involving partial beam measurements. In [9], the authors analyzed the geometric relationship between the DVL beam configuration and a zero velocity vector assumption. They derived a zero velocity update-aided virtual beam method to continuously operate the TC fusion and also proposed an LS support vector machine-aided virtual beam method to integrate into the TC INS/DVL fusion. Moreover, an approach has been proposed to estimate the platform's velocity vector based on past DVL measurements and a motion model specifically designed for short

time periods [10]. With recent advancements in computational efficiency and proven results of data-driven approaches, deep learning (DL) has started to emerge as a method to address the problems associated with partial beam measurements. In [11], the authors employed a convolutional neural network (CNN) to regress a single missing beam in cases where it is absent. Subsequently, they investigated various combinations of missing beam regression using a long short-term memory (LSTM) based network in [12]. Our previous work consists of a series of networks referred to as BeamsNet, which demonstrated how one-dimensional CNNs can enhance DVL measurements [13], compensate for missing beams [14, 15], and provide velocity updates in cases of complete DVL outage [16].

In this paper, we leverage our previous work by adapting and modifying the BeamsNet framework to directly regress missing beam measurements instead of the AUV's velocity vector. Subsequently, we integrate the regressed missing beams together with the measured ones into the EKF using both LC and TC integration approaches to form a seamless AUV navigation solution. Specifically, we focus on scenarios involving two and three missing beams. The contributions made in this study are outlined below:

1) A tailored BeamsNet architecture, uniquely crafted to forecast missing beams in situations where two or three beams are absent, leveraging both past DVL beam measurements and the current partial measurement.
2) Hybrid neural LC (HNLC) methodology designed to incorporate incomplete DVL measurements with regressed beam data forming a velocity vector update to the EKF.
3) Hybrid neural TC (HNTC) methodology fusing regressed and measured beams as an update to the EKF.

To emphasize our approach's robustness, we evaluate it using real-world AUV data collected on two separate occasions. The training and validation datasets consisted of approximately four hours of data, spanning nine distinct missions with varied maneuvers, speeds, depths, and other factors. In contrast, the test dataset comprised a mission of four hundred seconds, conducted on a separate date under different sea conditions. This diverse test scenario aimed to underscore the robustness of the HNC approach. In situations of limited beam measurements, we compare HNLC and HNTC to the baseline TC and LC (pure inertial) and a model-based average beam estimator. Our approach offers an improvement of 96.15% on average over the baseline model-based LC and TC approaches. Additionally, it demonstrates superior performance compared to a model-based beam estimator by an average of 12.41% in terms of velocity accuracy for scenarios involving two or three missing beams.

The rest of the paper is organized as follows: Section II introduces the mathematical background of INS/DVL fusion, examining both the LC and TC approaches. Section III presents the suggested hybrid neural coupled approach. Finally, Sections IV and V discuss the results and the conclusions, respectively.

II. INS/DVL FUSION

In this section, we describe the implementation of the error-state EKF for INS/DVL fusion. Throughout the implementation, three main reference frames will be indicated with an upper script. The first frame, denoted as b, represents the body frame centered at the vehicle's center of mass. In this frame, the x-axis aligns with the vehicle's longitudinal axis, pointing in the forward direction. The z-axis extends downward, and the y-axis extends outward, completing the right-hand orthogonal coordinate system. The second frame, denoted as n, represents the navigation frame, which is locally defined relative to the Earth's geoid. We utilize the north-east-down (NED) coordinate system within this frame, where the x-axis aligns with true north, the y-axis points eastward, and the z-axis points downward following the direction of gravity. The final coordinate, denoted as DVL, represents the DVL frame, with its sensitive axes determined by the manufacturer. The transformation from the DVL frame to the body frame is described by a fixed transformation matrix, known in advance [17].

The error-state EKF implementation is employed for the INS/DVL fusion. We denote the error-state vector as $\delta \boldsymbol{x} \in \mathbb{R}^{n \times 1}$, which represents an n-dimensional vector, formulated as:

$$\boldsymbol{x}^t = \boldsymbol{x}^e - \delta \boldsymbol{x} \qquad (1)$$

in this context, $\boldsymbol{x}^t \in \mathbb{R}^{n \times 1}$ and $\boldsymbol{x}^e \in \mathbb{R}^{n \times 1}$ denote the true state and the estimated state, respectively. Specifically, in the scenario of INS/DVL fusion, involving $n = 12$ error-states, we utilize $\delta \boldsymbol{x}^n$:

$$\delta \boldsymbol{x}^T = [(\delta \boldsymbol{v^n})^T \quad (\boldsymbol{\epsilon^n})^T \quad \delta \boldsymbol{b_a}^T \quad \delta \boldsymbol{b_g}^T]^T \in \mathbb{R}^{12 \times 1} \quad (2)$$

in this formulation, $\delta \boldsymbol{v^n} \in \mathbb{R}^{3 \times 1}$, $\boldsymbol{\epsilon} \in \mathbb{R}^{3 \times 1}$, $\boldsymbol{b_a} \in \mathbb{R}^{3 \times 1}$, and $\boldsymbol{b_g} \in \mathbb{R}^{3 \times 1}$ denote the velocity error-states expressed in the navigation frame, misalignment error, accelerometer bias residual error, and gyroscope bias residual error, respectively [18]. The linearized differential equation governing the error state is represented as follows:

$$\delta \dot{\boldsymbol{x}} = \mathbf{F} \delta \boldsymbol{x} + \mathbf{G} \boldsymbol{n} \qquad (3)$$

where $\boldsymbol{n} \in \mathbb{R}^{12 \times 1}$ denotes the system noise vector, $\mathbf{F} \in \mathbb{R}^{12 \times 12}$ represents the system matrix, and $\mathbf{G} \in \mathbb{R}^{12 \times 12}$ is the system noise distribution matrix. The system incorporates multiple independent sources of noise, each assumed to follow a zero-mean Gaussian distribution. These sources of noise can be represented as:

$$\boldsymbol{n} = [\boldsymbol{n_a}^T \quad \boldsymbol{n_g}^T \quad \boldsymbol{n_{a_b}}^T \quad \boldsymbol{n_{g_b}}^T]^T \in \mathbb{R}^{12 \times 1}. \qquad (4)$$

where $\boldsymbol{n_a} \in \mathbb{R}^{3 \times 1}$ and $\boldsymbol{n_g} \in \mathbb{R}^{3 \times 1}$ represent the additive noise for the accelerometer and gyroscope, respectively. Additionally, $\boldsymbol{n_{a_b}} \in \mathbb{R}^{3 \times 1}$ and $\boldsymbol{n_{g_b}} \in \mathbb{R}^{3 \times 1}$ denote the Gaussian random walk distributions for the accelerometer and gyroscope, respectively. The system matrix, $\mathbf{F}$, is defined as:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{vv} & \mathbf{F}_{v\epsilon} & \mathbf{0}_{3\times3} & \mathbf{C}_b^n \\ \mathbf{F}_{\epsilon v} & \mathbf{F}_{\epsilon\epsilon} & \mathbf{C}_b^n & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \qquad (5)$$
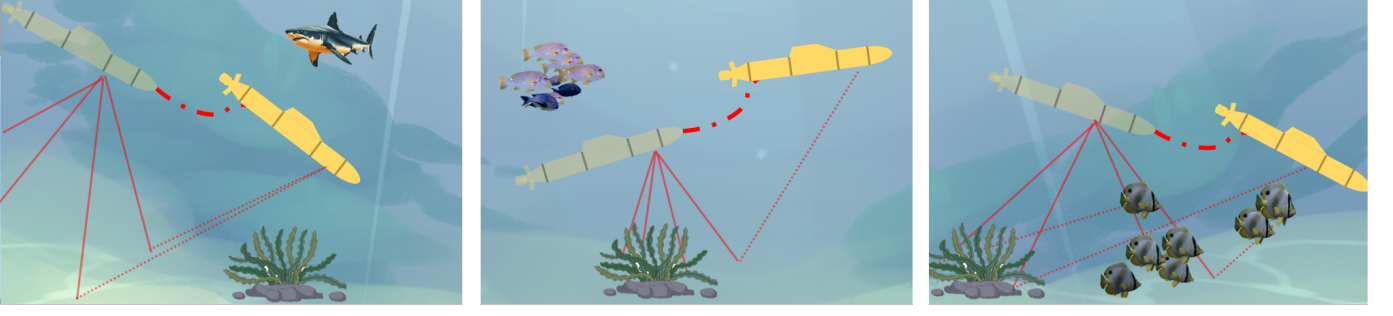
Fig. 1: A visualization of various scenarios where an AUV may encounter conditions limiting DVL beam measurements. On the left-hand side, the initial operation of the AUV, diving, is illustrated, characterized by a significant pitch angle. In the middle, the acoustic beams encounter uneven terrain, while on the right-hand side, the DVL view is obstructed by sea animals.

where $\mathbf{C}_b^n$ denotes the transformation matrix from the body frame to the navigation frame. The specific sub-matrices of $\mathbf{F}$ can be referenced in textbooks like [19, 20]. Concerning the distribution of system noise, the matrix $\mathbf{G}$ is defined as follows:

$$\mathbf{G} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{C}_b^n & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{C}_b^n & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{6}$$

where $\mathbf{0}_{3\times3}$ represents a three-by-three zero matrix, and $\mathbf{I}_{3\times3}$ denotes a three-dimensional identity matrix.

The Kalman filtering process typically consists of two distinct phases: the prediction and update steps. During the prediction step, the *a priori* error state, denoted as $\delta\boldsymbol{x}^-$, is initialized to zero to facilitate linearization, which is a crucial aspect of the EKF method.

$$\delta\boldsymbol{x}^- = 0 \tag{7}$$

Next, the state covariance is propagated using the known model, which in this case corresponds to the INS nonlinear equations of motion:

$$\mathbf{P}_k^- = \boldsymbol{\Phi}_{k-1}\mathbf{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \mathbf{Q}_{k-1} \tag{8}$$

Here, $\mathbf{P}_k^-$ denotes the *a priori* state covariance estimate at time k, and $\mathbf{P}_{k-1}^+$ represents the *a posteriori* state covariance estimate at time $k-1$. The transition matrix, denoted as $\boldsymbol{\Phi}_{k-1}$, is typically derived through a power-series expansion of the system matrix, $\mathbf{F}$, and the propagation interval, τ_s:

$$\boldsymbol{\Phi}_{k-1} = \sum_{r=0}^{\infty} \frac{\mathbf{F}_{k-1}^r}{r!}\tau_s^r \tag{9}$$

The discrete process noise covariance matrix is derived from its continuous form $\mathbf{Q} = \mathbb{E}[\boldsymbol{n}\boldsymbol{n}^T]$. It accounts for the inherent uncertainty in the model and is typically approximated as follows: [21]:

$$\mathbf{Q}_{k-1} = \frac{1}{2}\left(\boldsymbol{\Phi}_{k-1}\mathbf{G}_{k-1}\mathbf{Q}\mathbf{G}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}\mathbf{G}_{k-1}^T\boldsymbol{\Phi}_{k-1}^T\right)\Delta t. \tag{10}$$

The next stage of the Kalman filter is the update step, which is executed using the following equations:

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1} \tag{11}$$

$$\mathbf{P}_k^+ = [\mathbf{I} - \mathbf{K}_k\mathbf{H}_k]\mathbf{P}_k^- \tag{12}$$

$$\delta\boldsymbol{x}_k^+ = \mathbf{K}_k\delta\boldsymbol{z}_k \tag{13}$$

where $\mathbf{K}_k$ represents the Kalman gain, which balances between incorporating new measurements and predictions from the system's dynamic model. The matrices $\mathbf{H}_k$ and $\mathbf{R}_k$ correspond to the measurement matrix and noise covariance matrix, respectively. Finally, $\delta\boldsymbol{x}_k^+$ and $\mathbf{P}_k^+$ denote the *a posteriori* error state and the estimated covariance state, respectively.

To determine the measurement matrix $\mathbf{H}_k$, it is essential to first inspect the geometry of the DVL operation. The DVL is installed inside the AUV such that its transducers are oriented toward the seabed. The DVL operates in an "×" type configuration, also known as the "Janus Doppler configuration" [22]. In the "×" type configuration, the transactors are located with a relative yaw and pitch angles to the DVL's sensor body frame, which usually differs from the platform's body frame, in the following manner:

$$\boldsymbol{b}_i = \begin{bmatrix} \cos\psi_i\sin\theta & \sin\psi_i\sin\theta & \cos\theta \end{bmatrix}_{1\times3} \tag{14}$$

where $\boldsymbol{b}_i$ such that $i = 1, 2, 3, 4$ represents the beam number, and ψ and θ denote the yaw and pitch angles relative to the body frame, respectively. The pitch angle is fixed and predetermined by the manufacturer, maintaining the same value for each beam. The yaw angle can be expressed, for example, by [7]:

$$\psi_i = (i-1)\cdot 90° + 45° , \; i = 1, 2, 3, 4 \tag{15}$$

A. Loosely Coupled INS/DVL Fusion

In the LC approach, the AUV velocity vector estimation through raw DVL beam measurements, is conducted separately to the EKF framework, utilizing a parameter estimation method known as least squares (LS) parameter estimation. Let $\boldsymbol{v}^{DVL}$ represent the velocity vector expressed in the DVL frame, and $\mathbf{T}$ be the transformation matrix mapping it to the velocity in the beam directions, denoted by $\boldsymbol{v}^{Beam}$. Therefore, the relation between them can be defined as follows:

$$\boldsymbol{v}^{Beam} = \mathbf{T}\boldsymbol{v}^{DVL}, \quad \mathbf{T} = \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \boldsymbol{b}_3 \\ \boldsymbol{b}_4 \end{bmatrix}_{4\times3} \tag{16}$$

The beam measurements are subject to inherent errors, which are modeled by:

$$\boldsymbol{y} = \mathbf{T}[\boldsymbol{v}^{DVL}(\mathbf{1} + \boldsymbol{s}_{DVL})] + \boldsymbol{b}_{DVL} + \boldsymbol{n}_{DVL} \qquad (17)$$

where, $\boldsymbol{b}_{DVL} \in \mathbb{R}^{4 \times 1}$ represents the bias vector, $\boldsymbol{s}_{DVL} \in \mathbb{R}^{3 \times 1}$ denotes the scale factor vector, and $\boldsymbol{n}_{DVL} \in \mathbb{R}^{4 \times 1}$ stands for a zero-mean white Gaussian noise. After obtaining the raw measurements, the subsequent step involves extracting $\boldsymbol{v}^d$ by filtering the data based on the following cost function:

$$\hat{\boldsymbol{v}}^{DVL} = \underset{\boldsymbol{v}^{DVL}}{\operatorname{argmin}} \| \boldsymbol{y} - \mathbf{T}\boldsymbol{v}^{DVL} \|^2. \qquad (18)$$

The solution to this LS problem is obtained by multiplying the observations with the pseudo-inverse of the matrix $\mathbf{T}$ [23]:

$$\hat{\boldsymbol{v}}^{DVL} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\boldsymbol{y}. \qquad (19)$$

Finally, the estimated velocity vector from the DVL is transformed to the body frame using:

$$\hat{\boldsymbol{v}}_{DVL}^b = \mathbf{C}_d^b \hat{\boldsymbol{v}}^{DVL} \qquad (20)$$

where $\mathbf{C}_d^b$ represents a known constant transformation matrix from the DVL frame to the body frame and $\hat{\boldsymbol{v}}_{DVL}^b$ denotes the DVL velocity in the body frame.

To derive $\mathbf{H}_k$ for the LC approach, we examine the innovation residual and employ a first-order perturbation approximation while neglecting higher-order error components:

$$\begin{aligned}
\boldsymbol{\delta z} &= \hat{\mathbf{C}}_n^b \hat{\boldsymbol{v}}^n - \boldsymbol{v}_{DVL}^b \\
&= \mathbf{C}_n^b(\mathbf{I}_3 + \boldsymbol{\epsilon}[\times])(\boldsymbol{v}^n + \delta\boldsymbol{v}^n) - \boldsymbol{v}_{DVL}^b \\
&= \overset{\boldsymbol{v}_{DVL}^b}{\overbrace{\mathbf{C}_n^b\boldsymbol{v}^n}} + \mathbf{C}_n^b\delta\boldsymbol{v}^n + \mathbf{C}_n^b\boldsymbol{\epsilon}[\times]\boldsymbol{v}^n - \overset{0}{\overbrace{\mathbf{C}_n^b\boldsymbol{\epsilon}[\times]\delta\boldsymbol{v}^n}} - \boldsymbol{v}_{DVL}^b \\
&\approx \mathbf{C}_n^b\delta\boldsymbol{v}^n - \mathbf{C}_n^b\boldsymbol{v}^n[\times]\boldsymbol{\epsilon}
\end{aligned} \qquad (21)$$

therefore $\mathbf{H}_k$ can be formulated into the following:

$$\mathbf{H}_k^{LC} = [\mathbf{C}_n^b \quad -\mathbf{C}_n^b\boldsymbol{v}^n[\times] \quad \mathbf{0}_{3\times 3} \quad \mathbf{0}_{3\times 3}] \qquad (22)$$

The loosely coupled integration offers two primary advantages: simplicity and redundancy. However, its main drawbacks arise from the use of cascaded filters (LS and EKF), which results in a sub-optimal solution, and the requirement of at least three available beams to estimate the velocity vector.

B. Tightly Coupled INS/DVL Fusion

In the TC approach, the DVL raw beam measurements are directly processed through the EKF update stage. Each raw beam measurement is handled separately in the following manner:

$$\begin{aligned}
\boldsymbol{\delta z}_i &= \boldsymbol{b}_i^T \hat{\mathbf{C}}_n^b \hat{\boldsymbol{v}}^n - \boldsymbol{b}_i^T \boldsymbol{v}_{DVL}^b \\
&= \boldsymbol{b}_i^T \mathbf{C}_n^b(\mathbf{I}_3 + \boldsymbol{\epsilon}[\times])(\boldsymbol{v}^n + \delta\boldsymbol{v}^n) - \boldsymbol{b}_i^T \boldsymbol{v}_{DVL}^b \\
&= \overset{\boldsymbol{b}_i^T \boldsymbol{v}_{DVL}^b}{\overbrace{\boldsymbol{b}_i^T\mathbf{C}_n^b\boldsymbol{v}^n}} + \boldsymbol{b}_i^T\mathbf{C}_n^b\delta\boldsymbol{v}^n + \boldsymbol{b}_i^T\mathbf{C}_n^b\boldsymbol{\epsilon}[\times]\boldsymbol{v}^n \\
&\quad - \overset{0}{\overbrace{\boldsymbol{b}_i^T\mathbf{C}_n^b\boldsymbol{\epsilon}[\times]\delta\boldsymbol{v}^n}} - \boldsymbol{b}_i^T \boldsymbol{v}_{DVL}^b \\
&\approx \boldsymbol{b}_i^T\mathbf{C}_n^b\delta\boldsymbol{v}^n - \boldsymbol{b}_i^T\mathbf{C}_n^b\boldsymbol{v}^n[\times]\boldsymbol{\epsilon}
\end{aligned} \qquad (23)$$

where $\boldsymbol{\delta z}_i$ is the residual of the ith beam measurement and therefore $\mathbf{H}_{k,i}$, that corresponds to this beam, can be formulated into the following:

$$\mathbf{H}_{k,i}^{TC} = [\boldsymbol{b}_i^T\mathbf{C}_n^b \quad -\boldsymbol{b}_i^T\mathbf{C}_n^b\boldsymbol{v}^n[\times] \quad \mathbf{0}_{1\times 3} \quad \mathbf{0}_{1\times 3}] \qquad (24)$$

One of the advantages of the TC approach is its ability to operate in cases where there is no bottom lock, meaning that at least three beams are reflected back to the sensor from the seabed, including scenarios with fewer than three beams. The actual $\mathbf{H}_k$ is determined by the number of available beams by concatenating the $\mathbf{H}_{k,i}$. For example, in an ideal case with four beams, the measurement matrix would be:

$$\mathbf{H}_k^{TC} = \begin{bmatrix} \mathbf{H}_{k,1}^{TC} \\ \mathbf{H}_{k,2}^{TC} \\ \mathbf{H}_{k,3}^{TC} \\ \mathbf{H}_{k,4}^{TC} \end{bmatrix}_{4\times 12} \qquad (25)$$

It has been demonstrated in the literature that the TC approach consistently outperforms its LC counterpart in terms of both accuracy and robustness [24].

III. Hybrid Neural Coupled INS/DVL Fusion

In this section, we introduce a deep learning methodology inspired by the BeamsNet framework [13, 14] to allow regression of missing beams and thus enable seamless AUV navigation in scenarios with limited DVL measurements. Additionally, we explore the feasibility of integrating our hybrid neural approach into the error-state EKF framework using both LC and TC approaches.

A. Missing Beams Regression

Our proposed framework utilizes N past DVL measurements and the current partial beam measurements (one or two). Considering that AUVs generally do not engage in extreme maneuvers and commonly operate at speeds lower than 4 [m/s], we opted to concentrate on a brief time window N preceding the partial beam measurements. In doing so, we leverage a data-driven methodology to extract pertinent features facilitating the prediction of missing beams. The proposed approach's architecture relies on a one-dimensional convolutional neural network (1DCNN), as depicted in Fig. 2. The first step of the network is feature extraction from past DVL measurements using a 1D convolutional layer with a kernel size of 2×1 and stride of 2×1, followed by a hyperbolic tangent function. Then, a residual connection between the flattened output of this layer and the flattened input is established and inserted into a set of fully connected layers followed by rectified linear units. The output of this section is concatenated with the current partial beam measurements, along with an additional input, which is the average over the N past DVL measurements. Lastly, it passes through a fully connected layer that outputs the missing beams. The network can also be formulated mathematically in the following manner:

$$\text{Input:} \quad X = [x_1, x_2, \ldots, x_N] \qquad (26a)$$

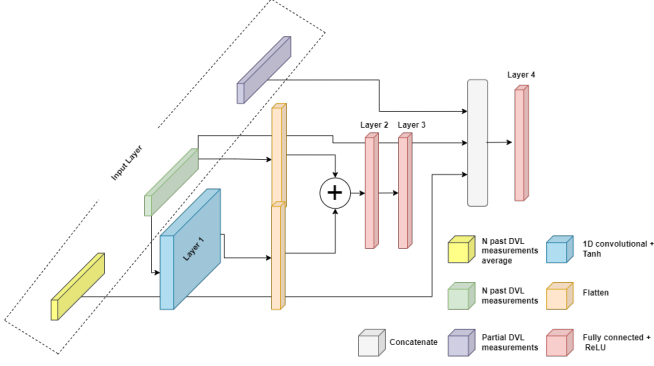$$\text{Input:} \quad P \qquad (26b)$$

Fig. 2: A schematic representation of the revised BeamsNet architecture, comprising of one-dimensional convolutional layers and fully connected layers.

where X is a vector of N past DVL beam measurements and P represents the current partial DVL beam measurements. M is the third input representing the mean of the past DVL beam measurements:

$$M = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{27}$$

In the ith convolutional layer, the output Z_i is obtained as follows:

$$Z_i = \sigma_{\text{Tanh}} \left(\sum_{j=1}^{m} (x_{i+(j-1)s} \cdot w_j) + b \right) \tag{28}$$

where m represents the window/kernel size, b signifies the bias, s denotes the stride, and w_j represents the weights. Additionally, σ_{Tanh} denotes the Tanh activation function, defined as follows:

$$\sigma_{\text{Tanh}}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{29}$$

Combining the output of the flattened convolutional layer Z (28) with the input X (26a), resulting in layer Y:

$$Y = Z + X \tag{30}$$

The output L of the first fully connected layer is computed as:

$$L = \sigma_{\text{ReLU}}(W_2(\sigma(W_1Y + b_1)) + b_2) \tag{31}$$

where W_1 and W_2 represent the weight matrices, and b_1 and b_2 represent the bias vectors. Additionally, σ_{ReLU} denotes the ReLU activation function, defined as follows:

$$\sigma_{\text{ReLU}}(x) = \max(0, x) \tag{32}$$

After obtaining L in (31), it is stacked with P (26b) and M (27) to form U:

$$U = \text{stack}\{L, P, M\} \tag{33}$$

Finally, the output $\hat{O}$ of the neural network is computed as:

$$\hat{O} = \sigma_{\text{ReLU}}(W_3U + b_3) \tag{34}$$

where W_3 is the weight matrix and b_3 is the bias vector. During the training process, we employ the mean squared error (MSE) loss function:

$$\text{MSE}(O, \hat{O}) = \frac{1}{n} \sum_{i=1}^{n} (O_i - \hat{O}_i)^2 \tag{35}$$

where O represents the reference data and n in the number of data points.

The optimization process is carried out using the root mean square propagation (RMSprop) optimizer. The model is trained with a batch size of 4, a learning rate of 0.001, and a learning rate decay factor of 0.1 every 35 epochs. Training is conducted over 100 epochs. The differences in hyper-parameters between the architectures for handling two or three missing beams are presented in Table I. Notice that the same architecture is applied with differences only in the input and output shapes.

TABLE I: Variations in hyperparameters for two and three missing beam networks within the modified BeamsNet architecture.

Network	Type	Input Shape	Output Shape
2 Missing Beams	Layer1 - Conv1D	3	6
	Layer2 - FC	12	16
	Layer3 - FC	16	2
	Layer4 - FC	8	2
3 Missing Beams	Layer1 - Conv1D	5	10
	Layer2 - FC	20	16
	Layer3 - FC	16	3
	Layer4 - FC	8	3

B. Hybrid Neural Coupled EKF

Once the network is trained, it is incorporated into the navigation filter to address scenarios of partial beam measurements. When the system identifies the number of missing beams, (17) undergoes a slight modification. Now, the vector $\boldsymbol{y}$ depicted in (17) is no longer in $\mathbb{R}^{4\times1}$ but rather in either $\mathbb{R}^{2\times1}$ or $\mathbb{R}^{1\times1}$, depending on the number of beam measurements. The output of the suggested approach completes the missing beams to obtain a total of four beam measurements. Let the beams be denoted as beam1 to beam4 according to (14). For the two missing beams scenario, we arbitrarily choose beam1 and beam3 as the missing beams, while when three beams are missing, we arbitrarily choose beam2 to be available. Therefore, for the scenario of two missing beams, the measurement vector is defined as:

$$\boldsymbol{y}_{2beams}^{HNC} = [\hat{O}[1], \quad \boldsymbol{y}[1], \quad \hat{O}[2], \quad \boldsymbol{y}[2]] \in \mathbb{R}^{4\times1} \tag{36}$$

and for three missing beams as

$$\boldsymbol{y}_{3beams}^{HNC} = [\hat{O}[1], \quad \boldsymbol{y}[1], \quad \hat{O}[2], \quad \hat{O}[3]] \in \mathbb{R}^{4\times1} \tag{37}$$

where $\boldsymbol{y}[n]$ and $\hat{O}[n]$ represent the beam measurements in the nth position of the DVL and the network's output as defined in (34), respectively. These measurements can then be integrated into the filter using the HNLC approach or the HNTC approach. Notice that in both cases, the measurement matrix remains consistent with that of a regular situation involving complete four-beam measurements, even though the missing beams are obtained from a different source (network
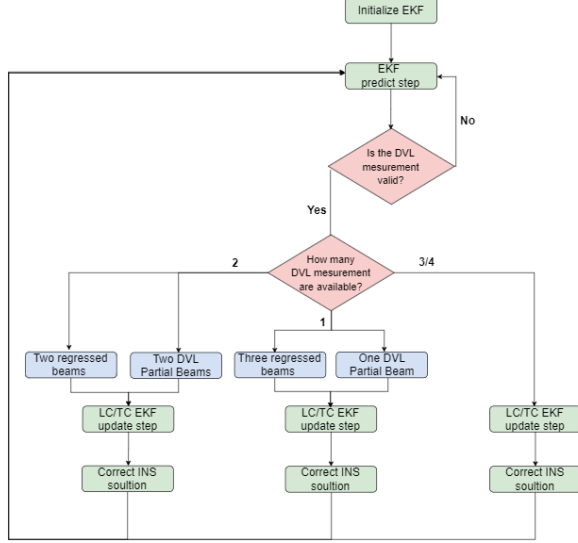
Fig. 3: Hybrid-Neural EKF information flow. The EKF is initialized and constantly propagates the INS model while awaiting the DVL velocity update. Upon receiving the DVL velocity update, if the bottom lock condition is met, the update procedure is carried out regularly, either in an LC or TC approach. However, if there are fewer than three beams, the data is directed through the appropriate path and the missing beams are forecasted using the proposed approach and subsequently utilized within the filter.

regression). This ensures the preservation of the filter update structure. Specifically, $\mathbf{H}_k^{HNLC}$ is equivalent to $\mathbf{H}_k^{LC}$ as defined in (22), and $\mathbf{H}_k^{HNTC}$ is equivalent to $\mathbf{H}_k^{TC}$ as defined in (24).

The flow of information, as outlined in Fig.3, begins by propagating the navigation solution using the INS model and its corresponding estimated covariance as per (8). Then, when a DVL sample is deemed valid, the first step is to determine the number of available beams. If there are three or four beams available, the update phase proceeds as usual, following either an LC or TC integration approach. If fewer beams are available, the previously stored past DVL beam measurements, along with the current partial measurements, are inputted into our proposed deep learning model to forecast the missing beams. Once all four beams are available, they are used to update the filter either by the LC or TC methods.

IV. ANALYSIS AND RESULTS

A. Field Experiments and Dataset

To evaluate our approach, we generated data by conducting missions with an AUV in the Mediterranean Sea. Specifically, we employed a modified ECA Group A18D mid-size AUV called Snapir. It is capable of autonomously performing missions up to 3000 meters in depth with 21 hours of endurance [25]. The Snapir AUV is outfitted with the iXblue

Phins Subsea INS, which utilizes fiber optic gyroscope (FOG) technology for precise inertial navigation [26]. Additionally, Snapir uses a Teledyne RDI Work Horse Navigator DVL [27], renowned for its capability to provide accurate velocity measurements with a standard deviation of $0.02\ [m/s]$. The INS operates at a frequency of 100 [Hz], while the DVL operates at 1 [Hz].

To thoroughly assess the robustness of our approach, we utilized data from two different dates. One dataset was employed for training and validating our data-driven approach, while the testing data comes from a separate mission. The dataset utilized for training and validation was collected in May 2021. The dataset comprises nine distinct missions characterized by various parameters, including maneuvers, depth, speed, sea conditions, and more. Eight of the missions were utilized for training, containing 12,799 DVL samples, equivalent to 3.5 hours of data. An example of two trajectories can be seen in Fig. 4. Additionally, 1758 DVL samples were reserved for validation, which we refer to as trajectory 1. For testing, an additional 400 seconds of data from another sea experiment conducted in June 2022 were employed. This data is denoted as trajectory 2. Additionally, corresponding inertial measurement unit (IMU) measurements were captured and utilized in the EKF to propagate the INS and filter model. The ground truth (GT) solution is the one provided by the Delph INS filter using a post-processing software designed for iXblue's INS-based subsea navigation [28].
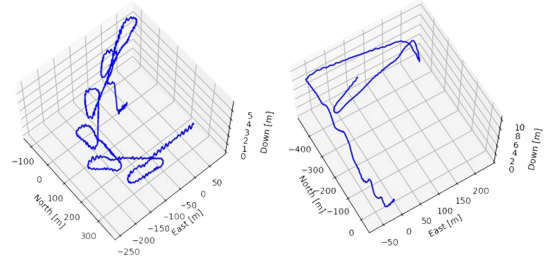

Fig. 4: Example of two out of the eight trajectories used to train our proposed network.

The data provided by the Teledyne RDI Work Horse Navigator DVL comprises velocity samples. To obtain the beam velocity measurements, we utilized the error model specified in (17), considering a DVL bias of 0.01 [m/s] and additive white noise with a standard deviation of 0.042 [m/s] in all three axes and for both missing beam cases. The angle θ in (14) was set to $20°$, and the angle ψ, defined in (15), is utilized to construct the matrix $\mathbf{T}$ as in (16).

B. Evaluation Metrics

To evaluate the effectiveness of our proposed method, we utilized the root mean square error (RMSE) metric, specifically looking at the velocity RMSE (VRMSE):

$$VRMSE(\boldsymbol{v}_i, \hat{\boldsymbol{v}}_i) = \sqrt{\frac{\sum_{i=1}^{M}(\boldsymbol{v}_i - \hat{\boldsymbol{v}}_i)^2}{M}} \qquad (38)$$

where, M represents the total number of samples, $\boldsymbol{v}_i$ represents the ground truth velocity vector, while $\hat{\boldsymbol{v}}_i$ signifies the predicted velocity vector.

Additionally, the velocity relative total error (VRTE) is utilized as a percentage to gauge the relative discrepancy between the model's predictions and the reference data using the VRMSE metric. The formula computes the absolute difference between the model's and GT VRMSE, and then normalized by the GT VRMSE. Finally, it's multiplied by 100 to express the error in percentage.

$$VRTE = \left(\frac{|\text{VRMSE}_{\text{model}} - \text{VRMSE}_{\text{reference}}|}{\text{VRMSE}_{\text{reference}}} \right) \times 100 \quad (39)$$

where $\text{VRMSE}_{\text{model}}$ and $\text{VRMSE}_{\text{reference}}$ denote the VRMSE of the model's predictions and the GT data, respectively.

C. Experimental Results

Initially, we explored the different durations of partial DVL measurements, ranging from 5 seconds up to 25 seconds, on the filter performance. We observed that the performance of the model-based approaches was consistently satisfactory. This is attributed to the high quality of the inertial sensors. Hence, we extended our investigation to a 30-second window of partial beam measurements and evaluated the performance of both LC and TC approaches in these scenarios. To ensure a fair comparison with the network's beam prediction, we applied also an average estimator to regress the missing beams. It is important to emphasize that the regressed beams are not actual DVL beam measurements. Thus, this should be reflected in different variance values in the measurement noise covariance matrix $\mathbf{R}_k$. Moreover, as the network utilizes all N past DVL beam measurements along with the current measured ones, it introduces correlations that should also be reflected in the measurement noise covariance matrix. However, for simplicity, we assume these effects are small and neglect them. In Fig. 5, the behavior of the velocity error states in the EKF is presented. The error remains bounded within the estimated covariance sleeve (1 sigma), except for a few measurements that exceed it. Additionally, the estimated covariance by the EKF appears to be converging over time. This behavior is expected when all four beams are available, as indicated by the tight constraint on the velocity error state. However, in scenarios where the bottom lock condition is not met, leading to fewer available beams, the uncertainty in the velocity measurements increases, resulting in a wider covariance sleeve.

Figure 6 displays a magnified perspective spanning a 30-second duration, depicting the covariance of velocity estimates derived from partial DVL measurements for instances where two beams are missing, covering both trajectory 1 and trajectory 2. This visualization elucidates the divergence of uncertainty during encounters with missing beam situations, followed by subsequent convergence once all the beams are available again. This behavior is also observed in the case of three missing beams, but as expected, with a higher divergence rate and final value. When two beams are missing, the baseline LC EKF operates solely on pure inertial navigation, as the velocity cannot be estimated using the LS parameter estimator. This explains the divergence of the estimated velocity

(a) Results of trajectory 1.
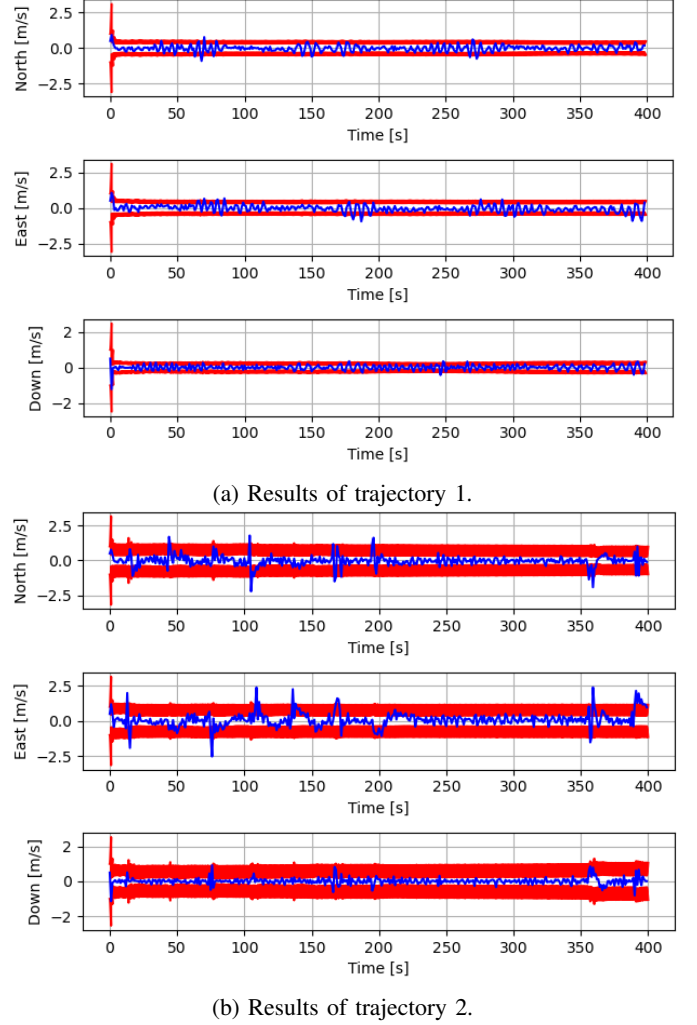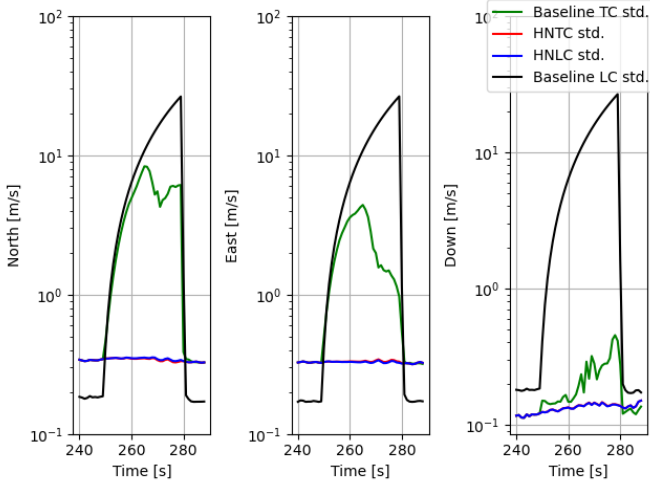

(b) Results of trajectory 2.

Fig. 5: Velocity error states. The blue line represents the velocity error state vector bounded within the estimated covariance sleeve (shown in red) of the EKF for trajectory 1 in (a) and trajectory 2 in (b).
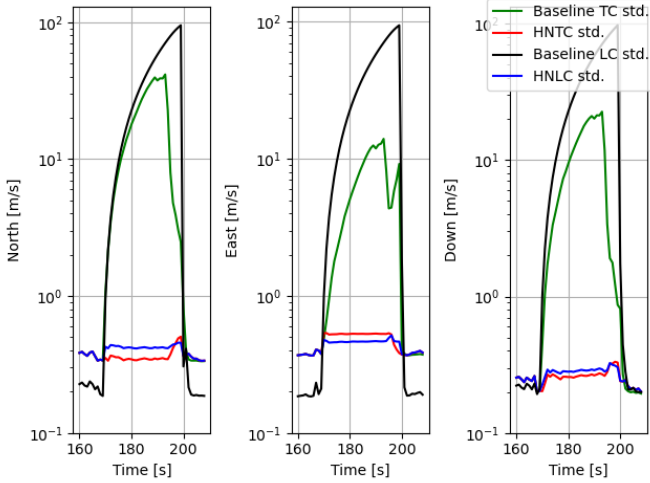
covariance, represented by a black curve. On the other hand, in the baseline TC approach, partial DVL measurements are utilized to mitigate error accumulation. Although the partial measurements do not provide full velocity information, they offer more certainty compared to the LC approach, as indicated by the magnitude of the green curve. When examining the HNLC and HNTC approaches, no divergence is observed, and the standard deviation values in both the neural loosely and neural tightly coupled methods are similar, with only a small difference as illustrated in Fig.6.

All of the factors mentioned above contribute to a significant improvement in VRMSE. Our HNLC and HNTC approaches outperform the model-based methods in both cases of two and three missing beams. Specifically, for two missing beams, the proposed approach demonstrates an improvement of over 93.6%, and for three missing beams, more than 94.9% in terms of VRTE. We also compared our approach to an average estimator, which takes the past DVL beam measurements and calculates the average to forecast the missing beam. Then,

utilizes it in the LC or TC methods to update the EKF. This comparison is considered fair as both methods receive the same inputs. The average estimator yields good results primarily because the AUV's maneuvers are not extreme and do not change frequently. However, the neural coupled approach is able to outperform the estimator in terms of VRMSE. With two missing beams, the proposed approach demonstrates an improvement of over 5.8% for two missing beams and over 11.4% for three missing beams. All the results are summarized in Table II, and the performance of all the methods in terms of position accuracy in the North-East plane is illustrated in Fig.7.
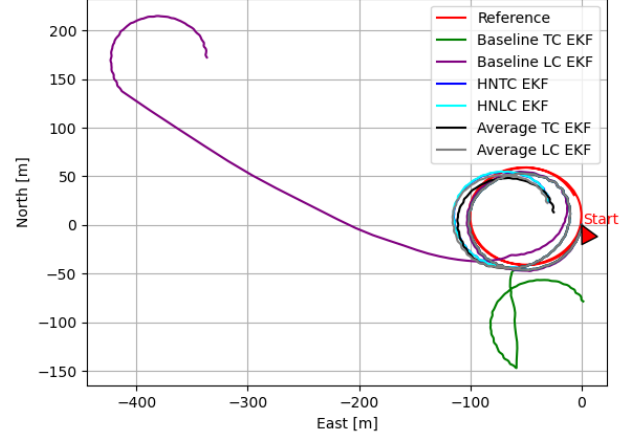

(a) Trajectory 1 in the North-East plane.

(a) The estimated velocity standard deviation for trajectory 1 over time.
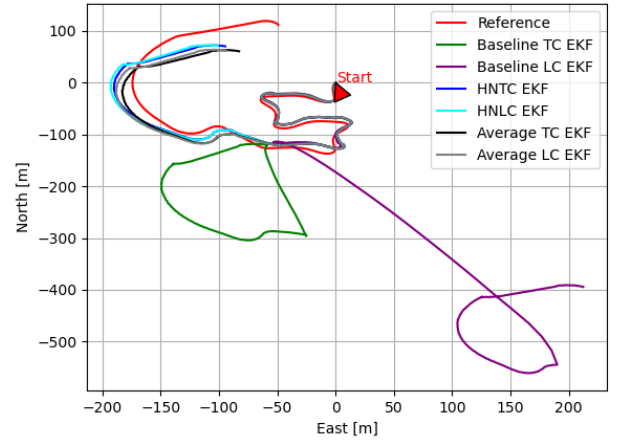
(b) Trajectory 2 in the North-East plane.

Fig. 7: Visualization of the trajectory of all inspected approaches with respect to the reference trajectory in red, plotted in the North-East plane. The model-based LC and TC approaches exhibit divergence during the 30-second period of missing beams, whereas the other methods maintain good results. The suggested approach outperforms both model-based methods.


(b) The estimated velocity standard deviation for trajectory 2 over time.

Fig. 6: The estimated velocity standard deviation trajectory 1 (a) and trajectory 2 (b) over time, comparing the baseline LC and TC approaches to the suggested HNLC and HNTC approaches. The suggested approaches were able to maintain an almost constant standard deviation, whereas in the model-based approaches, it diverges.

V. CONCLUSIONS

The underwater environment poses significant challenges for navigation due to its harsh conditions and the unavailability of GNSS signals for precise position updates. Typically, navigation systems integrate an INS with a DVL using an EKF. The DVL relies on a condition known as bottom lock, which requires at least three received beams out of the four transmitted. When fewer beams are available, the navigation solution suffers, whether in a TC or LC approach. In this paper, we proposed the HNLC and HNTC approaches to cope with limited beam scenarios. We designed a deep learning architecture to regress the missing beams using past beam measurements. In addition to comparing the results to the baseline model-based filter performance, we also compared

TABLE II: A comparison of the HNLC and HNTC approaches' performance with respect to the model-based LC and TC baseline, and the model-based beam regressor (average estimator) in terms of both the VRMSE and the VRTE.

Scenario	Method	VRMSE [m/s]		VRTE w.r.t Baseline		VRTE w.r.t Average Estimator	
		Trajectory 1	Trajectory 2	Trajectory 1	Trajectory 2	Trajectory 1	Trajectory 2
2 Missing Beams	Baseline LC	8.62	12.01	N/A	N/A	N/A	N/A
	Baseline TC	2.80	6.08	N/A	N/A	N/A	N/A
	Average LC	0.19	0.42	97.7 %	96.5 %	N/A	N/A
	Average TC	0.18	0.41	93.2 %	93.1 %	N/A	N/A
	HNLC (Ours)	0.17	0.35	97.9 %	97.0 %	10.6 %	14.5 %
	HNTC (Ours)	0.17	0.35	93.6 %	94.1 %	5.8 %	14.2 %
3 Missing Beams	Baseline LC	9.53	11.23	N/A	N/A	N/A	N/A
	Baseline TC	5.17	10.53	N/A	N/A	N/A	N/A
	Average LC	0.28	0.35	96.9 %	96.8 %	N/A	N/A
	Average TC	0.29	0.34	94.3 %	96.6 %	N/A	N/A
	HNLC (Ours)	0.25	0.29	97.3 %	97.3 %	11.4 %	15.8 %
	HNTC (Ours)	0.25	0.29	94.9 %	97.1 %	12.2 %	14.8 %

it to an average estimator, which receives the same inputs as our suggested network. Our approach was evaluated on real-world AUV data from two different occasions to emphasize its robustness. The results demonstrate that the suggested approach outperforms the model-based approaches by an average of 95.65% and the performance of the average estimator by an average of 11.27% in scenarios with two missing beams. In the case of three missing beams, it outperforms the model-based approaches by an average of 96.65% and the performance of the average estimator by an average of 13.55%. Our proposed approach requires only software modifications to enable seamless AUV navigation, even in situations of limited beam measurement.

In future work, we intend to delve deeper into investigating the cross-correlations that arise from using DL approaches with past and current DVL beam measurements, particularly in how they impact the measurements' noise covariance matrix. Additionally, prior research has indicated that incorporating inertial data alongside past and current DVL measurements may lead to improved beam and velocity estimation. However, this approach introduces additional correlations between the process data and the measurement data, which must be carefully addressed when integrating them into the EKF. We aim to analyze the effects of these correlations on performance in future studies.

REFERENCES

[1] P. A. Miller, J. A. Farrell, Y. Zhao, and V. Djapic, "Autonomous underwater vehicle navigation," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 3, pp. 663–678, 2010.

[2] D. Titterton and J. L. Weston, *Strapdown inertial navigation technology*. IET, 2004, vol. 17.

[3] N. A. Brokloff, "Matrix algorithm for Doppler sonar navigation," in *Proceedings of OCEANS'94*, vol. 3. IEEE, 1994, pp. III–378.

[4] M. Karimi, M. Bozorg, and A. Khayatian, "A comparison of DVL/INS fusion by UKF and EKF to localize an autonomous underwater vehicle," in *2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*. IEEE, 2013, pp. 62–67.

[5] J. Y. Taudien, *Doppler velocity log algorithms: detection, estimation, and accuracy*. The Pennsylvania State University, 2018.

[6] D. Rudolph and T. A. Wilson, "Doppler Velocity Log theory and preliminary considerations for design and construction," in *2012 Proceedings of IEEE Southeastcon*. IEEE, 2012, pp. 1–7.

[7] A. Tal, I. Klein, and R. Katz, "Inertial navigation system/Doppler velocity log (INS/DVL) fusion with partial DVL measurements," *Sensors*, vol. 17, no. 2, p. 415, 2017.

[8] P. Liu, B. Wang, Z. Deng, and M. Fu, "INS/DVL/PS tightly coupled underwater navigation method with limited DVL measurements," *IEEE Sensors Journal*, vol. 18, no. 7, pp. 2994–3002, 2018.

[9] Y. Yao, X. Xu, X. Xu, and I. Klein, "Virtual beam aided SINS/DVL tightly coupled integration method with partial DVL measurements," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 1, pp. 418–427, 2022.

[10] I. Klein, Y. Gutnik, and Y. Lipman, "Estimating DVL velocity in complete beam measurement outage scenarios," *IEEE Sensors Journal*, vol. 22, no. 21, pp. 20 730–20 737, 2022.

[11] M. Yona and I. Klein, "Compensating for partial Doppler velocity log outages by using deep-learning approaches," in *2021 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*. IEEE, 2021, pp. 1–5.

[12] ——, "MissBeamNet: learning missing Doppler velocity log beam measurements," *Neural Computing and Applications*, vol. 36, no. 9, pp. 4947–4958, 2024.

[13] N. Cohen and I. Klein, "BeamsNet: A data-driven approach enhancing Doppler velocity log measurements for

autonomous underwater vehicle navigation," *Engineering Applications of Artificial Intelligence*, vol. 114, p. 105216, 2022.

[14] ——, "LiBeamsNet: AUV velocity vector estimation in situations of limited DVL beam measurements," in *OCEANS 2022, Hampton Roads*. IEEE, 2022, pp. 1–5.

[15] ——, "Data-Driven Strategies for Coping with Incomplete DVL Measurements," *arXiv preprint arXiv:2401.15620*, 2024.

[16] N. Cohen, Z. Yampolsky, and I. Klein, "Set-transformer BeamsNet for AUV velocity forecasting in complete DVL outage scenarios," in *2023 IEEE Underwater Technology (UT)*. IEEE, 2023, pp. 1–6.

[17] J. Farrell, *Aided navigation: GPS with high rate sensors*. McGraw-Hill, Inc., 2008.

[18] R. M. Rogers, *Applied mathematics in integrated navigation systems*. AiAA, 2003, vol. 1.

[19] P. G. Savage *et al.*, *Strapdown analytics*. Strapdown Associates Maple Plain, MN, 2000, vol. 2.

[20] P. D. Groves, "Principles of GNSS, inertial, and multisensor integrated navigation systems," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 2, pp. 26–27, 2015.

[21] A. Gelb *et al.*, *Applied optimal estimation*. MIT Press, 1974.

[22] N. Cohen and I. Klein, "A-KIT: Adaptive Kalman-informed transformer," *arXiv preprint arXiv:2401.09987*, 2024.

[23] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[24] B. Zhang, D. Ji, S. Liu, X. Zhu, and W. Xu, "Autonomous underwater vehicle navigation: a review," *Ocean Engineering*, vol. 273, p. 113861, 2023.

[25] ECA Group, "A18-D AUV: Autonomous Underwater Vehicle," https://www.ecagroup.com/en/solutions/a18-d-auv-autonomous-underwater-vehicle, Accessed: March 2024.

[26] iXblue, "PHINS Subsea," https://www.ixblue.com/store/phins-subsea/, Accessed: March 2024.

[27] Teledyne Marine, "Doppler Velocity Logs," https://www.teledynemarine.com/products/product-line/navigation-positioning/doppler-velocity-logs, Accessed: March 2024.

[28] iXblue, "DELPH INS," https://www.ixblue.com/store/delph-ins/, Accessed: March 2024.