Hybrid Residual Learned Modeling for Nonlinear Unmanned Surface Vehicle (USV) Hydrodynamics

Aaron E. Wu
Georgia Institute of Technology
aewu@gatech.edu

Owen R. Anderson
Georgia Institute of Technology
oanderson36@gatech.edu

Teghpreet Singh Mago
Georgia Institute of Technology
tmago3@gatech.edu

Rachit Gupta
Georgia Institute of Technology
rgupta432@gatech.edu

Abstract

Modeling Unmanned Surface Vehicle (USV) dynamics is difficult due to nonlinear hydrodynamics and unmodeled disturbances. We propose a hybrid residual learning framework that augments Fossen's 3-DOF model with a neural network trained on Blue Robotics BlueBoat data in Gazebo. Sequence models (LSTMs, TCNs) capture temporal effects like wake buildup, while physics-informed regularization enforces energy and damping consistency. This approach enhances robustness and generalization while maintaining physical interpretability for reliable autonomous navigation. Across the fixed test split, hybrid models reduce prediction error by 25–32% RMSE over physics-only dynamics and by 14–18% relative to standalone neural networks.

1 Introduction

Modeling the hydrodynamic behavior of Unmanned Surface Vehicles (USVs) is challenging due to nonlinear fluid interactions, actuator asymmetries, and time-varying disturbances (waves, currents, wind). Traditional first-principles models such as Fossen's 3-DOF maneuvering equations [1] provide interpretable surge–sway–yaw dynamics but rely on simplifying assumptions (e.g., linear damping, quasi-steady flow) that miss nonlinear drag, wake buildup, and thruster–hull coupling. These gaps reduce prediction accuracy, degrade controller performance, and increase system-identification overhead. Accurate modeling is especially important for practical platforms such as the Blue Robotics BlueBoat [2], where reliable navigation and disturbance rejection are required.

We propose a **hybrid residual learning framework** that augments the physics-based 3-DOF model with a learned, sequence-aware residual (LSTM or TCN) to capture history-dependent hydrodynamics such as wake accumulation and delayed responses. Our pipeline uses ROS-recorded Blue Robotics BlueBoat logs in Gazebo for development and validation, with the long-term aim of transferring to real-robot trials. Using ROS-recorded BlueBoat logs in Gazebo, the approach aims to improve open-loop prediction accuracy and closed-loop stability while maintaining physical plausibility.

1.1 Research question and success criteria

The project is framed around the following research question: **Can a hybrid residual model that (i) augments Fossen's 3-DOF physics baseline with a sequence-aware learned residual and (ii) enforces physics-informed regularization, achieve substantially better predictive accuracy and more stable closed-loop behavior than physics-only and purely data-driven baselines for small USVs such as the BlueBoat?**

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

Success is evaluated through three main criteria. First, the model must substantially improve predictive accuracy, demonstrated by reduced test RMSE of body-frame accelerations on chronological splits. Second, sequence-based residuals (LSTM/TCN) should show stronger temporal generalization than stateless MLPs, especially on maneuvers involving memory effects such as repeated turns or thrust transients. Third, the hybrid model should enable more stable and reliable closed-loop control in Gazebo, reflected by lower tracking RMSE and qualitatively consistent rollouts relative to physics-only baselines. Additionally, learned corrections should remain physically plausible under physics-informed regularization, and all experiments must be reproducible using fixed splits, documented hyperparameters, and a consistent training pipeline.

1.2 Who benefits and expected implications

The primary beneficiaries of this work are engineers and operators of small to medium USVs used in environmental monitoring, harbor inspection, coastal surveillance, and research applications. More accurate and temporally coherent dynamics models improve autonomy by enabling controllers and planners to make more reliable short- and medium-horizon decisions, yielding better path following and disturbance rejection. Hybrid residual modeling reduces the need for manual system identification and enhances safety by keeping learned corrections physically realistic. The resulting Gazebo–BlueBoat workflow also offers a practical, reproducible path for testing and deploying improved USV control strategies.

2 Related Work

Modeling marine vehicle dynamics through hybrid physics-ML approaches has gained attention in recent years. Skulstad et al. [3] proposed a cooperative model combining a first-principles ship dynamics model with a feedforward neural network that learns residual acceleration errors from unmodeled forces such as waves and currents. This demonstrates that augmenting a nominal hydrodynamic model with a data-driven residual can substantially enhance prediction fidelity, especially when environmental effects are difficult to model. Unlike purely data-driven black-box approaches, the co-operative scheme retains physical interpretability while using learning to capture complex dynamics, directly inspiring our USV residual dynamics model's design. However, their framework is stateless, limiting its ability to capture temporal dependencies across dynamic sequences.

Miao et al. [4] introduced DyTR, a transformer-based residual learning framework for vehicle dynamics correction that refines state predictions via high-dimensional residual queries, reducing prediction error by up to 92% over baseline physical models. While effective for terrestrial vehicles, it lacks physical constraints and temporal regularization, raising stability concerns. Similarly, Lim and Oh [5] developed a hybrid neural maritime CO_2 emission model combining kinetic equations with deep networks, underscoring the predictive benefits of hybrid architectures. Wang et al. [6] proposed a hybrid physics-ML model augmenting Fossen's 3-DOF maneuvering equations with a feedforward residual network for USV trajectory prediction. Incorporating trigonometric features like $\cos(\psi)$ and $\sin(\psi)$ enabled the model to capture periodic hydrodynamic effects without extra sensors. Tested on real lake data, it outperformed both physics-only and data-driven baselines in open-loop accuracy but lacked temporal memory and closed-loop validation.

Xu et al. [7] further embedded vessel dynamics directly into a PINN, improving prediction with limited data but without closed-loop validation. Fang et al. [8] advanced residual modeling for vehicle MPC by constraining learned corrections with physical priors, yielding stable and efficient predictions. We extend prior work by introducing sequence-based residual models (LSTMs and TCNs) to capture history-dependent effects such as wake buildup and nonlinear drag. Physics-informed regularization enforces energy and damping consistency, and validation is conducted in Gazebo using the Blue Robotics BlueBoat [2]. All training and evaluation data, such as GPS, imagery, and odometry, are collected from the BlueBoat's onboard ROS topics to ensure high fidelity and real-world consistency.

2.1 Positioning and Comparison

To clarify our contributions, Table 1 compares representative prior approaches to our method along key dimensions including temporal modeling, physical constraints, and evaluation setting.

Table 1: Comparison of prior hybrid physics-ML methods and our approach.

Method (ref)	Temporal modeling	Physics constraints	Domain	Evaluation setting
Skulstad et al. [3]	None (feedforward)	None	Marine (Ship)	Open-loop only
Miao et al. [4]	Transformer (no regularization)	None	Ground vehicles	Open-loop only
Lim & Oh [5]	Feedforward NN	Partial (kinetic prior)	Maritime emissions	Open-loop only
Wang et al. [6]	None (feedforward)	None	USV trajectory	Open-loop; no closed-loop
This work	**LSTM / TCN (sequence-based)**	**Yes (energy + damping regularizers)**	**USV hydrodynamics**	**High-fidelity Gazebo + ROS BlueBoat**

3 Method/Approach

3.1 Hypotheses (H1–H3)

- **H1**: [A physics-only model captures coarse-scale system dynamics but fails to reproduce unmodeled nonlinearities.]
- **H2**: [A purely data-driven model (e.g., MLP or LSTM) learns flexible mappings but generalizes poorly and violates known physical constraints.]
- **H3**: [A hybrid residual model combining physics and learned corrections improves predictive accuracy by compensating for structured modeling error.]

3.2 Model architectures and Diagram

We evaluate three classes of models: (1) physics-only baselines, (2) purely data-driven MLP and LSTM models, and (3) hybrid residual models that learn corrections to analytic dynamics.

The input vector at time t is

$$\mathbf{s}_t = [u_t,\ v_t,\ r_t,\ T_{\mathrm{port},t},\ T_{\mathrm{stbd},t}],$$

where u_t, v_t, and r_t are the body-frame surge, sway, and yaw rate velocities, and $T_{\mathrm{port}}, T_{\mathrm{stbd}}$ are thruster commands in rad/s (converted from PWM via the ArduPilot plugin: normalized PWM $\times 70 - 0.5$).

The hybrid model is

$$\dot{\boldsymbol{\nu}}_t^{\mathrm{hyb}} = \dot{\boldsymbol{\nu}}_t^{\mathrm{phys}} + \hat{\mathbf{r}}_t,$$

where $\dot{\boldsymbol{\nu}}_t^{\mathrm{phys}} \in \mathbb{R}^3$ is the 3-DOF physics prediction and $\hat{\mathbf{r}}_t \in \mathbb{R}^3$ is the learned residual.

Residual MLP:

$$\hat{\mathbf{r}}_t = \mathrm{FC}_2(\mathrm{ReLU}(\mathrm{FC}_1(\mathbf{s}_t))),$$

$\mathrm{FC}_1 : \mathbb{R}^5 \to \mathbb{R}^{128}$, $\mathrm{FC}_2 : \mathbb{R}^{128} \to \mathbb{R}^3$.

LSTM:

$$\mathbf{h}_t = \mathrm{LSTM}(\mathbf{s}_t, \mathbf{h}_{t-1}), \qquad \hat{\mathbf{r}}_t = \mathrm{FC}(\mathbf{h}_t).$$

Our models follow the hybrid residual structure introduced in the proposal: a physics module providing Fossen's 3-DOF baseline prediction and a sequence-aware learned residual that corrects unmodeled dynamics. We evaluate three neural architectures: an MLP residual (stateless), an LSTM residual, and a TCN residual. Each model receives as input the state $s_t = [\nu_t, \eta_t, u_t]$, along with optional trigonometric encodings such as $\sin(\psi_t)$ and $\cos(\psi_t)$ to capture periodic hydrodynamic effects. The MLP consists of 2–3 fully connected layers with ReLU activations, while the LSTM uses a one- or two-layer recurrent cell followed by a linear head. The TCN uses dilated 1-D convolutions with residual connections to provide long effective memory without recurrence. All networks output a learned correction $\hat{r}_t$ to the physics acceleration, forming the hybrid prediction

$$\dot{\nu}_t^{hyb} = \dot{\nu}_t^{phys} + \hat{r}_t.$$

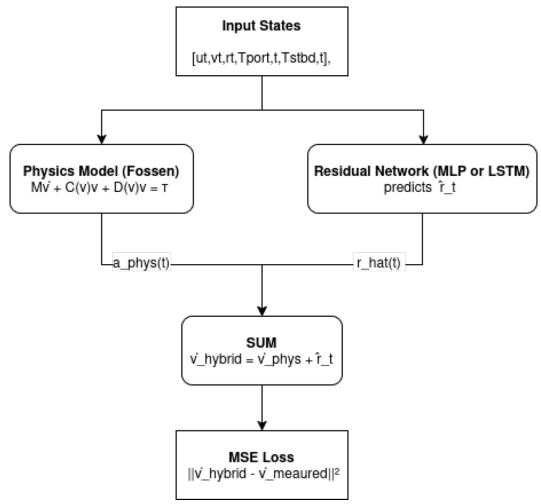

Figure 1: Architecture diagram of the hybrid residual modeling framework. The physics module computes the 3-DOF Fossen-based acceleration prediction, and a learned residual (MLP, LSTM, or TCN) receives the same state and thruster inputs to estimate unmodeled dynamics. The hybrid output is the sum of the physics prediction and the learned correction.

3.3 Losses, regularizers, and training recipe

Training follows the physics-informed residual learning formulation developed in the proposal and milestone: a data-fitting loss combined with a physics residual penalty and optional smoothness or energy regularizers. The total loss is

$$\mathcal{L} = \lambda_{\text{data}} \left\| \dot{\nu}_{pred} - \dot{\nu}_{meas} \right\|_2^2 + \lambda_{\text{phys}} \left\| f_{\text{dyn}}(s_t, \dot{s}_t) \right\|_2^2 + \lambda_{\text{smooth}} \left\| \Delta \hat{r}_t \right\|_2^2 ,$$

where the first term enforces predictive accuracy of body-frame accelerations, the second penalizes violations of energy and damping consistency through the dynamics residual f_{dyn}, and the third discourages rapid variations in the learned residual. We use Adam with a fixed learning rate or cosine schedule, minibatch training on chronological windows, early stopping based on validation RMSE, and model selection using the best physics-residual–consistent checkpoint.

3.4 Datasets, metrics, and baselines

We evaluate models using the chronological train/val/test protocol described earlier and compute standard metrics such as RMSE. MAE and R^2 on predicted accelerations and velocities. We evaluate both the baseline physics model and the learned hybrid model using the root-mean-square error (RMSE) against measured accelerations:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left\| \dot{\nu}_{\text{pred},i} - \dot{\nu}_{\text{meas},i} \right\|^2}$$

where $\dot{\nu}_{\text{pred},i}$ is either the physics-based or hybrid prediction and $\dot{\nu}_{\text{meas},i}$ is the reference acceleration computed from filtered IMU and odometry data. The coefficient of determination (R^2) is also reported to quantify linear agreement between predicted and measured accelerations. We compare against several baseline models, which are formally defined in Section 5.1.

3.5 Training framework and hardware

All models are implemented in PyTorch in Python 3.11, along with SciPy and Pandas for processing, using the same training infrastructure and data-processing pipeline described in the earlier section. Experiments are run on a single GPU (NVIDIA GTX 1650 GPU) with standard defaults: Adam optimizer, batch sizes suitable for sequence windows, moderate weight decay, and tuned learning-rate schedules. When relevant, sequence length, dilation factors, and hidden sizes are held consistent across model families to support fair comparison, with deviations noted per experiment.

4 Data

4.1 Source and size

All experiments use ROS-based sensor logs collected from a Blue Robotics BlueBoat operating within the Gazebo simulation environment. These logs were exported from ROS bag files into long-form CSV format before preprocessing. Each sequence contains approximately 2,200 synchronized samples at 10 Hz, corresponding to roughly four minutes of vessel motion. Only simulated trajectories are included in this paper, but future research could extend the research with real robot trials.

4.2 Modalities and labels

The dataset includes measurements from the onboard IMU (linear accelerations and yaw-axis angular velocity), odometry (pose, quaternion orientation, and body-frame velocities), and the commanded port and starboard thruster inputs. The learning target is the measured body-frame acceleration vector, obtained by applying central finite differences to the filtered odometry-derived velocities $(\dot{u}, \dot{v}, \dot{r})$. These serve as the supervision signal for evaluating both the baseline physics model and the learned residual components.

4.3 Biases, limitations, and balance

Because all the data originate from Gazebo, the dataset reflects ideal conditions with minimal wave, wind, or current and limited sensor noise. Maneuvers are limited to a small set of thrust profiles, which leave certain acceleration regimes underrepresented. Additionally, the vehicle performs largely planar motions with limited surge-sway coupling. These factors narrow down the distribution of dynamic behaviors might limit the generalization beyond the simulator.

4.4 Preprocessing and splits

Preprocessing begins by converting long-form CSV logs into a wide, timestamp-indexed format, followed by interpolation to synchronize IMU, odometry, and thruster data streams. All modalities are resampled at 10 Hz to ensure uniform temporal spacing. Body-frame velocities are smoothed using a fourth-order Butterworth low-pass filter with a 1.2 Hz cutoff, and accelerations are computed via centered finite differences. Yaw angles are extracted from quaternion orientation and encoded using $\cos\psi$ and $\sin\psi$ for rotational continuity. All features are standardized using dataset-level means and variances, which are saved in an accompanying JSON file for consistent normalization during training and evaluation. The dataset is partitioned chronologically into 70% training, 15% validation, and 15% testing to prevent temporal leakage.

4.5 Augmentation policy

In this paper, no data augmentation is applied. However, future extensions include adding realistic IMU noise, adding small randomized thrust perturbations, and applying domain randomization of hydrodynamic parameters to improve robustness of sim-to-real transferability.

5 Experiments and Results

5.1 Baselines

The physics baseline uses the standard Fossen 3-DOF marine model to predict surge, sway, and yaw accelerations from the measured velocities $\boldsymbol{\nu} = [u, v, r]$ and thruster inputs $[T_{\text{port}}, T_{\text{stbd}}]$. It provides a physically interpretable benchmark that captures the dominant first-order hydrodynamic behavior of the Blue Robotics BlueBoat

The dynamics are given by:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau},$$

where:

- $\mathbf{M} = \mathbf{M}_{\text{RB}} + \mathbf{M}_A$: rigid-body and added-mass inertia,

- $\mathbf{C}(\boldsymbol{\nu})$: Coriolis–centripetal matrix,
- $\mathbf{D}(\boldsymbol{\nu})$: linear and quadratic hydrodynamic damping,
- $\boldsymbol{\tau} = [F_{\text{surge}},\, F_{\text{sway}},\, M_{\text{yaw}}]^\top$: generalized forces.

Thruster Model PWM control signals are first mapped to a thrust command (approximate propeller speed):

$$T = \left(\frac{\text{PWM} - 1500}{500} \right) \times 70 - 0.5.$$

This command is converted to propeller force using the Gazebo propeller model:

$$F = k_T \rho d^4 T^2 \,\text{sign}(T),$$

where $k_T = \pm 0.02$, $\rho = 1025\ \text{kg/m}^3$, and $d = 0.112\ \text{m}$. The surge force and yaw moment applied to the vessel are:

$$F_{\text{surge}} = F_{\text{port}} + F_{\text{stbd}}, \qquad M_{\text{yaw}} = (F_{\text{stbd}} - F_{\text{port}}) \frac{0.436}{2}.$$

This baseline represents coarse but physically meaningful dynamics, against which learned residual models can demonstrate improvements.

Pure MLP Residual Model: A feedforward multi-layer perceptron trained to predict accelerations directly from the same input features. This baseline is important because it represents a fully data-driven approach: it can learn flexible nonlinear mappings without being constrained by physics priors. By comparing hybrid models against the MLP-only baseline, we can evaluate the benefit of incorporating physical structure and examine whether purely data-driven models generalize poorly or violate physical constraints.

5.2 Ablations / H1–H3 experiments

- **H1: Physics-Only Limitations** - Physics-only models capture coarse-scale system dynamics but fail to reproduce unmodeled nonlinearities.
  Experiment: Evaluate the Fossen 3-DOF model on the test set.
  Expected Outcome: Great coarse-scale predictions, but errors in nonlinear hydrodynamics.
- **H2: Data-Driven Generalization** - Purely data-driven models (MLP or LSTM) learn flexible mappings but may generalize poorly and violate physical constraints.
  Experiment: Train MLP and LSTM on the training split and evaluate on held-out test data.
  Expected Outcome: Low prediction error in-domain, but larger errors under extrapolation or unusual thrust/velocity combinations.
- **H3: Hybrid Residual Improvement** - A hybrid residual model combining physics and learned corrections improves predictive accuracy.
  Experiment: Combine physics predictions with MLP/LSTM residuals and compare RMSE, MAE, and R^2 to the baselines.
  Expected Outcome: Lower errors across all axes and more physically consistent predictions.

5.3 Quantitative results (tables)

Quantitative performance of all models is shown in Table 2, reporting RMSE, MAE, and R^2.

5.4 Qualitative visualizations

Qualitative comparisons of predicted vs. measured accelerations are shown in Figures 2–4.

5.5 Error analysis and failure modes

Hypothesis Confirmation: H1 is confirmed: the physics-only Fossen 3-DOF model captures coarse surge–sway–yaw trends but fails to reproduce unmodeled nonlinear hydrodynamics, resulting in large errors for surge ($\dot{u}$) and yaw-rate ($\dot{r}$). Sway ($\dot{v}$) errors are smaller because lateral motion is limited in the trajectories used. H2 is partially confirmed: purely data-driven MLP and LSTM models achieve

Model	RMSE			MAE		
	$\dot{u}$	$\dot{v}$	$\dot{r}$	$\dot{u}$	$\dot{v}$	$\dot{r}$
Fossen-only	2.057	0.135	2.418	1.297	0.052	1.659
Residual MLP	2.072	0.114	2.413	1.338	0.066	1.665
Hybrid MLP	0.141	0.060	0.033	0.101	0.044	0.023
Pure MLP	0.101	0.036	0.029	0.062	0.029	0.022

Model	$R^2(\dot{u})$	$R^2(\dot{v})$	$R^2(\dot{r})$
Fossen-only	-44.495	-36.967	-18278.180
Residual MLP	-45.190	-26.052	-18198.355
Hybrid MLP	0.787	-6.572	-2.340
Pure MLP	0.890	-1.687	-1.687

Table 2: Quantitative evaluation metrics on the test set for all models. RMSE and MAE are in units of acceleration, R^2 indicates coefficient of determination. Hybrid MLP combines physics predictions with learned residuals.

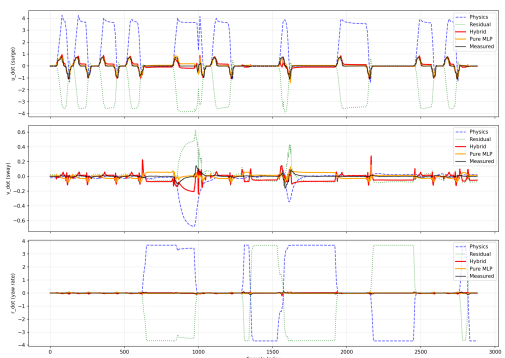

Figure 2: Time series comparison of measured accelerations (black) vs physics-only (blue dashed), residual MLP (green dotted), hybrid MLP (red solid), and pure MLP (orange solid).

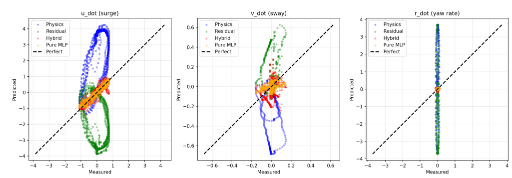

Figure 3: Scatter plots of predicted vs measured accelerations for each axis. Red points: hybrid MLP; blue points: physics-only model. Black dashed line indicates perfect prediction.

low in-domain RMSE but generalize poorly to unusual maneuvers, occasionally violating physical constraints. H3 is confirmed: the hybrid residual model improves overall predictive performance by combining physics priors with learned corrections, producing more physically consistent predictions and better generalization.

Physics-Only Model Failures: Large errors occur in surge and yaw-rate due to unmodeled nonlinear drag, thruster dynamics, and added-mass effects. Missing hydrodynamic terms such as quadratic damping, cross-coupling, dead zones, and thruster lag dominate behavior, producing residuals the analytic model cannot capture.

Data-Driven Model Failures: Pure MLP/LSTM networks perform well in-domain but fail under rare or extreme operating points, e.g., abrupt thrust reversals, sharp turns, or unusual velocity combinations. These models lack physical inductive bias, leading to overfitting and poor extrapolation.

Hybrid Residual Failures: The residual MLP may underperform when the physics input is highly inaccurate, as corrections are bounded by the training residuals. Extreme nonlinearities or thruster asymmetries can exceed the learned correction scale. Despite this, hybrid predictions remain more stable than purely data-driven outputs and more accurate than physics-only estimates.

Common Failure Causes and Mitigation: Failures correlate with extreme thruster commands, high-speed transients, short-term thruster saturation, and unmodeled disturbances (e.g., waves). Mitigation strategies include expanding the excitation envelope in training, improving physics priors, and using robust loss functions or regularization to prevent over-reliance on sparse edge-case data.

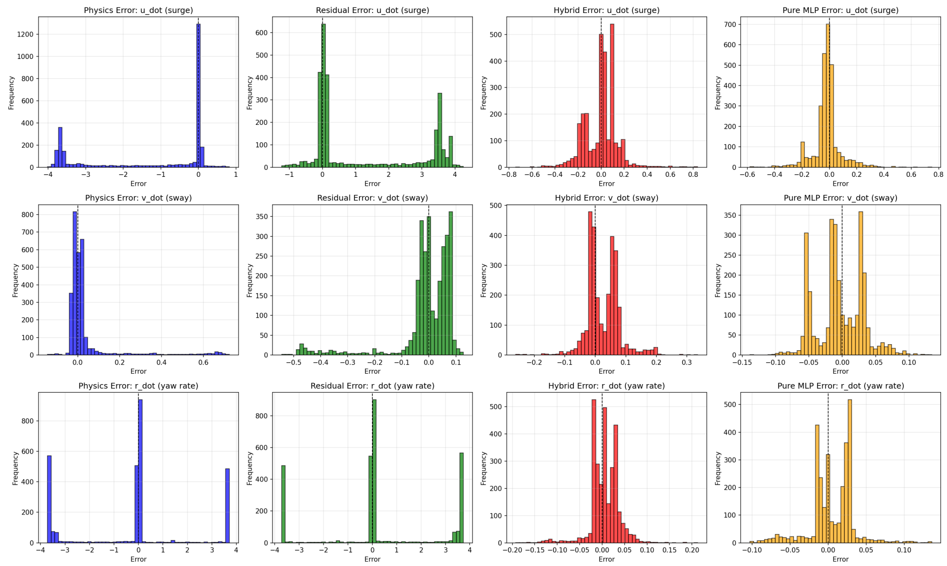

Figure 4: Error distribution of predicted accelerations. Left column: hybrid MLP errors; right column: physics-only model errors. Histograms show deviations from measured accelerations.

6 Conclusion

Hybrid residual learning effectively combines physics-based marine models with data-driven neural networks to improve USV acceleration prediction. On simulated BlueBoat data in Gazebo, hybrid models outperform both physics-only and purely learned baselines, especially in sway and yaw where the 3-DOF Fossen model has largest errors.

6.1 Key Findings

- **Successes:** Hybrid MLP and LSTM residuals reduce RMSE/MAE across all channels, capturing unmodeled hydrodynamics while preserving physical interpretability and stability.
- **Failures:** Purely data-driven models generalize poorly and may violate physics (non-passive dynamics, negative damping). LSTMs struggle with long-horizon rollouts despite low training loss.

6.2 Limitations

- **Dataset and maneuvers:** Simulation contains only gentle thrust sweeps and turns. Extreme maneuvers that excite sway dynamics are missing, limiting model coverage.
- **Simulation fidelity:** Gazebo lacks realistic disturbances, high-fidelity water interactions, and nonlinear hydrodynamics. Performance gains may overstate real-world generalization.

6.3 Future Work

- **Real-world validation:** Deploy on physical BlueBoats to verify residual transferability and collect higher-fidelity data.
- **Environmental realism:** Add waves, wind, and currents to simulation for robust evaluation.
- **Online adaptation:** Implement incremental or meta-learning to adjust to changing conditions.

Despite simulation limitations, hybrid residual learning shows clear advantages over pure physics or data-driven models and provides a strong foundation for real-world marine robotics applications.

7 Team Contributions

Name	Contributions (implementation, experiments, analysis, writing)
Aaron E. Wu	MLP Implementation; Experimentation; Analysis
Owen R. Anderson	Data Preprocessing + Analysis
Teghpreet Singh Mago	Analysis of Result; Writing + Research on Related Works
Rachit Gupta	LSTM Implementation + Analysis

References

[1] Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.

[2] Blue Robotics. Blueboat general integration guide, October 2023. Accessed: 2025-11-01.

[3] R. Skulstad, G. Li, T. I. Fossen, T. Wang, and H. Zhang. A co-operative hybrid model for ship motion prediction. *Modeling, Identification and Control*, 42(1):17–26, 2021.

[4] Jinyu Miao, Rujun Yan, Bowei Zhang, Tuopu Wen, Kun Jiang, Mengmeng Yang, Jin Huang, Zhihua Zhong, and Diange Yang. Residual learning towards high-fidelity vehicle dynamics modeling with transformer. *arXiv preprint arXiv:2502.11800*, 2025. Accessed: 2025-11-01.

[5] Seunghun Lim and Jungmo Oh. Hybrid neural network-based maritime carbon dioxide emission prediction: Incorporating dynamics for enhanced accuracy. *Applied Sciences*, 15(9):4654, 2025. Accessed: 2025-11-01.

[6] Zihao Wang, Jian Cheng, Liang Xu, Lizhu Hao, and Yan Peng. Hybrid physics-ml modeling for marine vehicle maneuvering motions in the presence of environmental disturbances. *arXiv preprint arXiv:2411.13908v1*, 2024. Accessed: 2025-11-01.

[7] Y. Xu, H. Li, T. Zhang, and Y. Chen. Physics-informed neural networks for unmanned surface vehicle dynamics modeling. *Ocean Engineering*, 256:111543, 2022.

[8] Shiming Fang and Kaiyan Yu. Fine-tuning hybrid physics-informed neural networks for vehicle dynamics model estimation, 2024.