IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Real-Time Millimeter-Accurate Underwater Pose Estimation via Tightly-Coupled Fusion of Vision and Optical Tracking

Yuer Gao, *Student Member, IEEE*, Tongqing Xu, and Yi Cai, *Member, IEEE*

***Abstract*—Precise and high-frequency state estimation is required for advanced underwater robotic applications such as physical interaction and agile control, yet no single sensor can simultaneously provide both high accuracy and high update rates. Vision-based methods offer high-frequency updates but suffer from drift, while optical tracking systems are highly accurate but may not provide sufficiently high update rates for real-time control loops. This letter presents a tightly-coupled sensor fusion framework that combines a high-frequency (62 FPS) monocular vision-based pose estimator with a high-accuracy (millimeter-level) optical tracking system. Our approach uses a visual estimator for high-frequency state propagation—with a latent variable motion model to compensate for underwater disturbances—while the optical tracker provides periodic corrections. In a controlled underwater testbed, this achieves a position RMSE of 5.65 mm at 62 FPS, improving accuracy 1.6 × compared to the best baseline method (EfficientPose + EKF: 9.20 mm) and 6.4 × compared to vision-only estimation (36 mm). Our dataset and code are available upon request.**

***Index Terms*—Underwater robotics, sensor fusion, pose estimation, optical tracking, visual odometry.**

I. Introduction

UNDERWATER robotics has enabled significant progress in marine research, infrastructure inspection, and underwater archaeology. These applications require a precise and robust robot localization. However, unlike terrestrial applications where GNSS provides a standard solution, electromagnetic signal attenuation underwater requires alternative localization approaches [1].

In controlled environments such as research testbeds and validation facilities, establishing high-fidelity ground truth for localization remains challenging. This is essential for calibrating onboard sensors and validating advanced control algorithms. Existing high-precision systems face a trade-off. While acoustically aided systems can provide drift-free global localization, they are limited by the low-frequency nature of the acoustic signal (typically 1 Hz), which constrains real-time performance for agile maneuvers [2]. Onboard vision-based methods like Visual-Inertial Odometry (VIO) [3] can operate at high frequencies but are prone to long-term drift that often accumulates to several meters, as demonstrated in underwater evaluations [4]. This drift makes them unsuitable for tasks requiring high-fidelity metric accuracy, such as precise subsea manipulation or the validation of control algorithms. While optical tracking systems achieve millimeter-level accuracy in controlled laboratory conditions, their performance degrades significantly in practical underwater deployments. This speed-accuracy trade-off creates a practical bottleneck in developing high-performance underwater robotic systems.

To address this trade-off, this letter proposes the real-time fusion of complementary sensor modalities. The proposed approach integrates two systems: (i) a high-frequency (62 FPS) monocular vision pose estimator that tracks known markers on the robot, and (ii) a high-accuracy (±0.5˜mm) commercial optical tracking system (NOKOV) that provides precise position measurements at lower frequencies. In this framework, the vision system handles prediction while the optical tracker provides corrective updates, creating a fusion loop that maintains both speed and accuracy.

The main contributions of this work are fourfold:

1) A fusion framework that combines the accuracy of optical tracking (when available) with the robustness of learned visual-dynamics models, enabling reliable pose estimation in challenging underwater scenarios where optical-only solutions degrade or fail
2) A visual front-end with a kinematic model that includes a latent dynamics variable, allowing the system to learn and compensate for unmodeled underwater disturbances
3) Experimental validation showing that the fused output outperforms either individual sensing modality in both accuracy and robustness
4) An underwater localization dataset including synchronized video, control inputs, and high-precision ground truth from the optical tracking system

II. Related Work

Underwater robot localization has been addressed through various sensing approaches. We first discuss localization systems for open-water and controlled environments, followed by a review of high-frequency vision-based methods that offer real-time performance. Finally, we examine sensor fusion strategies to situate our contribution within the field.

Received 29 July 2025; accepted 10 November 2025. Date of publication 8 December 2025; date of current version 12 December 2025. This article was recommended for publication by Associate Editor E. Kelasidi and Editor G. Loianno upon evaluation of the reviewers' comments. This work was supported in part by Guangzhou-HKUST(GZ) Joint Funding Program under Grant 2024A03J0680 and in part by Guangzhou Municipal Science and Technology Project under Grant 2024A04J6464. *(Corresponding author: Yi Cai.)*

The authors are with the Smart Manufacturing Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511400, China (e-mail: ygao438@connect.hkust-gz.edu.cn; txu313@connect.hkust-gz.edu.cn; yicai@hkust-gz.edu.cn).

Digital Object Identifier 10.1109/LRA.2025.3641116

A. Localization in Open-Water and Controlled Environments

Industry Context: Commercial underwater navigation relies on acoustic positioning systems like Long Baseline (LBL) or Ultra-short Baseline (USBL) combined with Doppler Velocity Logs (DVLs) and Inertial Measurement Units (IMUs). These systems provide robust, drift-free localization in open water but have significant limitations: high cost, complex deployment requirements (LBL systems need pre-calibrated seafloor transponders), and low update rates, typically around 1 Hz. These constraints render them impractical for agile operations in controlled laboratory environments.

Controlled Environment Localization: High-precision localization in controlled testbeds typically uses external infrastructure like ceiling-mounted cameras tracking above-water features on floating platforms [5] or customized ultrasonic systems. These approaches fail when robots operate fully submerged or when water surface disturbances interfere with tracking. Our approach addresses this limitation through entirely underwater sensing.

However, the slow speed of sound in water limits these systems to low update rates and precisions around 15 mm [6], making them unsuitable for real-time control of agile robots. Commercial optical motion capture systems (e.g., Vicon, OptiTrack), or the NOKOV [7] system used in our work, can provide sub-millimeter accuracy. These systems require multi-camera installations, and while their individual cameras may operate at high frame rates (e.g., > 100 Hz), the final pose output rate available for real-time control is often limited by system-level processing delays.

B. High-Frequency Vision-Based Pose Estimation

Researchers have turned to vision-based methods to address the low update rates of acoustic and optical systems. Monocular VIO, such as VINS-Mono [3], fuses camera images with IMU data to provide high-frequency state estimates. These odometry-based approaches suffer from error accumulation over time and lack an absolute global reference.

Deep learning-based 6D pose estimation offers another approach. Methods like YOLO6D [8], EfficientPose [9], and ROPE [10] can estimate the full 6D pose of an object from a single image in real-time. These methods work well for object detection and initial pose estimation. However, when applied frame-by-frame, these single-shot estimators are known to exhibit jitter and lack temporal consistency, often requiring a subsequent refinement or filtering step to ensure smooth trajectory estimates [11]. While they [12] achieve high speeds (40 to 60 FPS), their absolute accuracy degrades under challenging underwater visual conditions such as variable lighting and turbidity [13].

C. ROV Motion and Disturbance Modeling

Underwater vehicles experience complex hydrodynamic disturbances. Existing approaches either model these through system identification [14] or learn dynamics from direct disturbance measurements [15]. Our approach uses a latent variable $\mathbf{w}_f$ in the state propagation model that compensates for disturbances by minimizing pose error from visual feedback, requiring no vehicle-specific calibration or force measurements. Recent work has also explored learning-based approaches to underwater vehicle dynamics. Singh and Alexis [16] proposed DeepVL, which learns velocity from dynamics and inertial measurements for underwater odometry. Saksvik et al. [17] developed a deep learning approach to dead-reckoning navigation for AUVs with limited sensors. Cai et al. [18] presented a data-driven velocity estimator handling unmeasurable flow and wave disturbances. Our latent variable approach differs by learning implicit disturbance compensation within a pose estimation framework rather than explicit velocity prediction.

D. Sensor Fusion Strategies in Underwater Robotics

Underwater robotics commonly uses sensor fusion to address individual sensor limitations [19]. Visual-inertial fusion provides high-frequency dead-reckoning capabilities [3]. To reduce the inherent drift of such systems in large-scale environments, many works fuse vision with acoustic sensors. For instance, Li et al. [2] developed a robust underwater visual SLAM system that fuses vision with data from a DVL to provide long-term drift correction. Similarly, advanced systems like SVIn2 [20] fuse stereo vision, IMU, and sonar data to create dense, accurate maps of unknown underwater areas. While these multi-sensor SLAM systems are powerful for exploration and large-scale navigation, they focus on robustness over large areas, and their computational cost can limit real-time performance, as seen in Shkurti et al. [21], which was limited to 10 Hz.

While the aforementioned SLAM systems are ideal for mapping unknown environments, their final metric accuracy may be insufficient for tasks like millimeter-level control algorithm validation or high-precision robotic manipulation studies. While optical tracking systems offer millimeter-level precision, a framework that tightly couples a high-frequency visual estimator with such a system for real-time, high-fidelity state estimation in controlled environments has, to our knowledge, remained unexplored. Most approaches either operate offline or do not combine these specific modalities. Our framework fuses vision-based high-frequency estimation with optical tracking accuracy in real-time for laboratory and testbed applications.

III. System Design

The real-time sensor fusion framework presented in this letter for underwater robot localization is shown in Fig. 1. The framework uses a tightly-coupled Extended Kalman Filter (EKF) that integrates two complementary data streams: high-frequency pose estimates from a monocular vision front-end, and high-accuracy measurements from an external optical tracking system.

A. System Overview and State Representation

Our system processes two asynchronous input streams (Fig. 1). The Monocular RGB Camera provides a high-frequency sequence of images, which are fed into our High-Frequency Visual Pose Estimator. This module produces a rapid stream of state predictions ($\hat{\mathbf{x}}_{k|k-1}$), enabling real-time responsiveness but subject to drift over time. The Optical Tracking System provides a stream of highly accurate but lower-frequency absolute position data (z_k), processed by a Measurement Model. The Extended Kalman Filter (Fusion Core) combines the high-frequency predictions with the high-accuracy measurement updates to generate a final Fused 6D Pose ($\hat{\mathbf{x}}_{k|k}$) that maintains both high-frequency and high-accuracy.

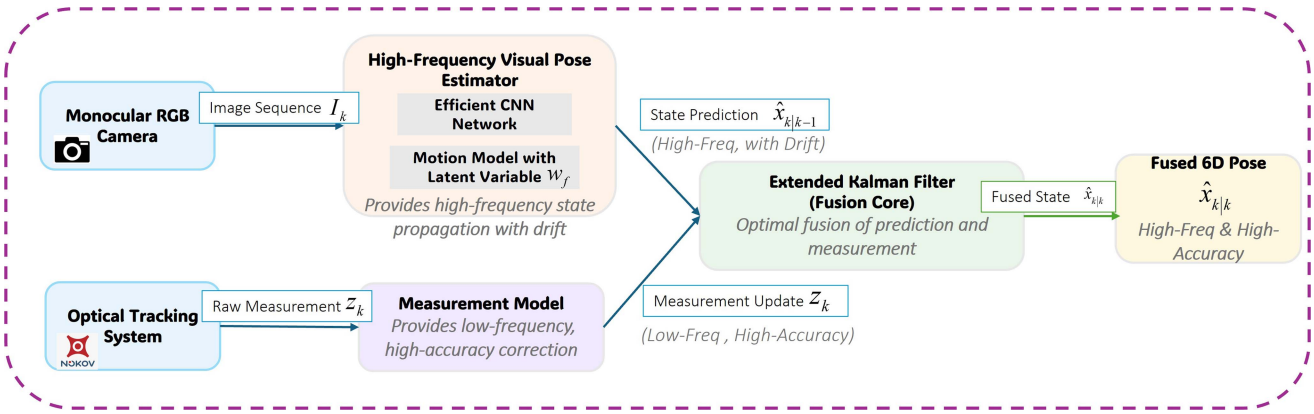

Fig. 1. An overview of our proposed real-time sensor fusion framework. The system integrates two complementary data streams: a high-frequency visual pose estimator provides state predictions, while a high-accuracy optical tracking system provides measurement updates. An EKF serves as the fusion core, combining these streams to produce a final state estimate that is both high-frequency and high-accuracy.

We define the system state vector $\mathbf{x}_k$ at time k as:

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{p}_k^T & \mathbf{q}_k^T & \mathbf{v}_k^T & \boldsymbol{\omega}_k^T \end{bmatrix}^T \in \mathbb{R}^{13}, \tag{1}$$

where $\mathbf{p}_k \in \mathbb{R}^3$ is the 3D position of the robot's center of mass, $\mathbf{q}_k \in \mathbb{R}^4$ is the orientation represented as a unit quaternion, $\mathbf{v}_k \in \mathbb{R}^3$ is the linear velocity, and $\boldsymbol{\omega}_k \in \mathbb{R}^3$ is the angular velocity. This state representation allows us to model the full 6-DOF dynamics of the robot.[1]

The robot is equipped with $N = 10$ visual markers whose positions $\{\mathbf{p}_i^B | i = 1, \ldots, N\}$ are known precisely in the robot's body frame B. Our vision module detects these markers to estimate the robot's pose. The Extended Kalman Filter was chosen as it provides a robust and computationally efficient framework for fusing non-linear models, which is well-suited for our real-time application.

B. High-Frequency State Propagation

The prediction step of our EKF uses a custom vision-based motion model that operates at high-frequency, providing state estimates between sparse updates from the optical tracker. This model takes the previous state estimate $\hat{\mathbf{x}}_{k-1|k-1}$ and control inputs $\mathbf{u}_{k-1}$ to propagate the state forward in time.

Our model includes a latent variable to learn and compensate for unmodeled hydrodynamic disturbances.

1) Kinematic Model With Latent Dynamics: We model the robot's motion using a discrete-time, non-linear state transition function f:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_f) + \mathbf{w}_k, \tag{2}$$

where $\mathbf{u}_{k-1}$ represents the control inputs (i.e., thruster commands) and $\mathbf{w}_k$ is the process noise, assumed to be a zero-mean Gaussian with covariance Q_k. The control inputs $\mathbf{u}_{k-1}$ are logged from the joystick commands used to teleoperate the ROV.

The latent dynamics variable $\mathbf{w}_f = [\mathbf{w}_{f,v}^\top, \mathbf{w}_{f,\omega}^\top]^\top \in \mathbb{R}^6$ models unmodeled disturbances in underwater environments, where $\mathbf{w}_{f,v} \in \mathbb{R}^3$ and $\mathbf{w}_{f,\omega} \in \mathbb{R}^3$ capture translational and rotational effects. Rather than measuring physical flow directly, the neural network estimates $\mathbf{w}_f$ from the image sequence at time $k-1$ to predict the state at time k. During training, the network learns $\mathbf{w}_f$ values that minimize pose error, implicitly compensating for external forces.

We define the state transition function f using a constant velocity motion model, augmented by our learned latent variable. The control inputs $\mathbf{u}_{k-1}$ correspond to commanded linear and angular velocities, while the latent variable $\mathbf{w}_f$ provides learned corrections to these velocities. The state propagates over the time interval $\Delta t = t_k - t_{k-1}$ as follows:

Position Update: The new position is the previous position plus the integrated velocity over the time step.

$$\mathbf{p}_k = \mathbf{p}_{k-1} + \mathbf{v}_{k-1}\Delta t \tag{3}$$

Although the kinematic model conceptually uses commanded velocity $u = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$, we employ a constant velocity assumption in (3)-(6) as actual velocities deviate from commands due to water currents and thruster dynamics. The learned latent dynamics w_f implicitly capture these deviations between commanded and actual motion.

Orientation Update: The new orientation is obtained by integrating the angular velocity using quaternion multiplication.

$$\mathbf{q}_k = \mathbf{q}_{k-1} \otimes \exp\left(\frac{1}{2}\begin{bmatrix} 0 \\ \boldsymbol{\omega}_{k-1} \end{bmatrix} \Delta t\right), \tag{4}$$

where $\otimes$ denotes quaternion multiplication.

Velocity Updates: The velocities follow a random walk model, influenced by the learned latent disturbance variable $\mathbf{w}_f$.

$$\mathbf{v}_k = \mathbf{v}_{k-1} + \mathbf{w}_{f,v}, \tag{5}$$

$$\boldsymbol{\omega}_k = \boldsymbol{\omega}_{k-1} + \mathbf{w}_{f,\omega}, \tag{6}$$

where the latent variable $\mathbf{w}_f$ is split into a linear component $\mathbf{w}_{f,v}$ and an angular component $\mathbf{w}_{f,\omega}$, both regressed by our neural network.

2) Theoretical Motivation for the Latent Variable Approach: Accurately modeling the complete hydrodynamic forces on an underwater vehicle is notoriously difficult. Traditional first-principles-based dynamic models require extensive system identification experiments, often in specialized facilities like towing tanks, to determine dozens of vehicle-specific hydrodynamic coefficients [14]. Furthermore, such explicit models struggle

[1]For state covariance propagation within the EKF, a minimal 3-DOF error-state representation is used for the 4-D quaternion, resulting in a 12x12 covariance matrix. This is a standard practice to avoid singularity issues.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

to account for complex, time-varying effects such as forces from a tether, changes in vehicle dynamics due to payload manipulation, or unpredictable water currents.

Our use of a learned latent variable $\mathbf{w}_f$ can be interpreted as a form of **data-driven residual modeling** [22]. In this paradigm, the simple kinematic model (3)–(6) represents our baseline physical understanding of the system's motion. The latent variable $\mathbf{w}_f$ then represents the network's estimate of the **residual dynamics**—that is, the net effect of all forces and disturbances not captured by our simple model.

This data-driven residual modeling approach works well for our underwater application. By training the network end-to-end to minimize pose error, we force it to learn a function that maps visual cues from the image sequence to a latent vector $\mathbf{w}_f$ that explains the discrepancy between kinematic prediction and observed motion. The latent variable $\mathbf{w}_f$ needs no external sensors or ground truth measurements for supervision—it emerges as a byproduct of the pose estimation objective. This eliminates laborious, vehicle-specific system identification while adapting to complex disturbances that are difficult to model explicitly.

3) Neural Network Architecture: Our motion model uses a deep neural network to estimate keypoint locations and latent dynamics from the input image I_k. The network takes a single RGB image, resized to 640×640 pixels, as input. The architecture prioritizes computational efficiency for real-time performance.

The network is trained end-to-end with a multi-task loss balancing three objectives:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{keypoint}} + \lambda_2 \mathcal{L}_{\text{pose}} + \lambda_3 \mathcal{L}_{\text{latent}} \tag{7}$$

The keypoint term $\mathcal{L}_{\text{keypoint}} = \frac{1}{N}\sum_{i=1}^{N} \|\hat{\mathbf{u}}_i - \mathbf{u}_i\|^2$ measures reprojection error for the $N = 10$ optical markers, with ground truth $\mathbf{u}_i$ computed from NOKOV poses and camera calibration.

The pose term $\mathcal{L}_{\text{pose}} = \|\hat{\mathbf{p}} - \mathbf{p}_{\text{gt}}\|^2 + \alpha\|\hat{\mathbf{q}} - \mathbf{q}_{\text{gt}}\|^2$ supervises the 6D estimate directly, with $\alpha = 10.0$ balancing position (mm) and quaternion scales. Poses $\hat{\mathbf{p}}, \hat{\mathbf{q}}$ are obtained via PnP from predicted keypoints $\hat{\mathbf{u}}_i$.

The regularization term $\mathcal{L}_{\text{latent}} = \|\mathbf{w}_f\|^2$ bounds the latent variable. We set $\lambda_1 = 1.0$, $\lambda_2 = 5.0$, $\lambda_3 = 0.01$. The ground truth keypoint projections $\mathbf{u}_i$ are computed by projecting the known 3D marker positions $\{\mathbf{p}_i^B\}$ using the NOKOV-provided poses and calibrated camera parameters.

The network uses an **EfficientRep** [23] backbone, specifically a medium-sized variant that balances feature extraction capability with inference speed. Features from multiple backbone stages are aggregated and fused by a **Rep-PAN** [8] neck, capturing both fine-grained local features for precise keypoint localization and high-level semantic features.

From the fused feature maps, two separate prediction heads perform the main tasks, as shown in Fig. 1:

1) **Keypoint Detection Head:** Convolutional layers followed by a fully-connected layer regress the 2D pixel coordinates $(\mathbf{u}_i, \mathbf{v}_i)$ for each of the $N = 10$ visible markers. The output is a vector in $\mathbb{R}^{20}$.
2) **Latent Variable Estimation Head:** A parallel head with a similar structure regresses the latent dynamics variable $\mathbf{w}_f \in \mathbb{R}^6$. This 6-dimensional vector consists of 3D linear and 3D angular velocity correction terms, corresponding to $\mathbf{w}_{f,v}$ and $\mathbf{w}_{f,\omega}$ respectively.

The shared backbone with separate heads enables specialized learning for each task while maintaining computational efficiency. The detected keypoints and estimated latent variable are then used to compute the state prediction $\hat{\mathbf{x}}_{k|k-1}$ using the kinematic model in Equation 2.

4) Jacobian Matrices for the EKF: To complete the EKF formulation, we define the Jacobian matrices F_k and H_k, which are the partial derivatives of the state transition and measurement functions with respect to the state.

The Jacobian of the measurement function $h(\mathbf{x}_k)$ with respect to the state is straightforward, as our measurement model is linear. From Equation 12, the measurement matrix H_k serves as its own Jacobian:

$$H_k = \frac{\partial h(\mathbf{x}_k)}{\partial \mathbf{x}_k} = \begin{bmatrix} I_{3\times3} & 0_{3\times4} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{8}$$

The Jacobian of the non-linear state transition function $f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ with respect to the state $\mathbf{x}_{k-1}$ is more complex. It is a 13×13 matrix, evaluated at the previous state $\hat{\mathbf{x}}_{k-1|k-1}$. In practice, this becomes a 12×12 matrix when using the 3-DoF error state representation for the quaternion. By linearizing our motion model from Equations 3-6, the Jacobian $\mathbf{F}_{k-1}$ is derived from standard EKF linearization of the state transition model.

5) EKF Prediction Step: With the state transition function f defined, the EKF prediction step propagates the state forward:

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}) \tag{9}$$

$$\hat{\mathbf{w}}^f_{k+1|k} = f(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, \hat{\mathbf{w}}^f_{k|k}) \tag{10}$$

The process covariance is predicted as:

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1} \tag{11}$$

where F_{k-1} is the Jacobian matrix of the state transition function f with respect to the state, evaluated at $\hat{\mathbf{x}}_{k-1|k-1}$. The process noise covariance Q_{k-1} reflects the uncertainty of our motion model.

C. High-Accuracy Measurement Update

The measurement update comes from the high-accuracy optical tracking system (NOKOV). This system provides sparse but precise measurements of the robot's absolute position, correcting drift accumulated during vision-based prediction.

We model the measurement process with a linear measurement function h:

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k = H_k \mathbf{x}_k + \mathbf{v}_k, \tag{12}$$

where $\mathbf{z}_k \in \mathbb{R}^3$ is the 3D position measurement provided by the NOKOV system at time k. The measurement noise $\mathbf{v}_k$ is assumed to be a zero-mean Gaussian with covariance matrix R_k, reflecting the high accuracy of the optical tracker. The measurement matrix H_k extracts the position components from the state vector $\mathbf{x}_k$. For our state representation in Equation 1,where $H_k = [I_{3\times3}\ 0_{3\times4}\ 0_{3\times3}\ 0_{3\times3}]$, $[\mathbf{0}_{3\times3}]$ and $I_{3\times3}$ is the identity matrix and $0_{3\times m}$ are zero matrices.

Note that detected keypoints are used only to regress the latent dynamics variable $\mathbf{w}_f$ via the MLP, and not directly for pose correction. The EKF state updates rely solely on optical tracking measurements when available, with the learned dynamics providing predictions between measurements and during optical dropouts.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

D. EKF Update and Fusion

The EKF update stage corrects the state prediction $\hat{\mathbf{x}}_{k|k-1}$ and its covariance $P_{k|k-1}$ using the high-accuracy measurement $\mathbf{z}_k$ from the optical tracker.

First, the Kalman Gain K_k is computed:

$$K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1} \tag{13}$$

The state estimate is updated using the measurement residual:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k(\mathbf{z}_k - h(\hat{\mathbf{x}}_{k|k-1})) \tag{14}$$

The state covariance matrix is updated:

$$P_{k|k} = (I - K_kH_k)P_{k|k-1} \tag{15}$$

The resulting state estimate, $\hat{\mathbf{x}}_{k|k}$, is the final output of our fusion framework at time k.

E. Sensor Time Synchronization

Accurate temporal alignment between the asynchronous data streams from the high-frequency camera and the high-accuracy optical tracker is critical for our fusion algorithm performance. In our experimental setup, time synchronization between the monocular camera and the NOKOV system was achieved through a dedicated hardware trigger signal. This signal simultaneously initiates frame capture on the camera and pose recording on the NOKOV system, ensuring a temporal alignment error of less than 1 ms between corresponding measurements. This precise synchronization is necessary for accurate EKF measurement updates.

IV. Experimental Validation

The fusion framework was evaluated through experiments in a controlled underwater environment.

A. Experimental Setup

1) Test Environment: All experiments were conducted in a $6.0 \times 5.0 \times 1.5$ meters indoor test tank. Water temperature was maintained at 20 ± 1 °C. We varied water turbidity levels within 0.5–5 NTU and generated water flow speeds from 0 to 0.5 m/s using a configurable pump system. The varying levels of turbidity were created by incrementally adding small amounts of milk into the water tank. Milk is a standard agent used in underwater vision research to simulate the light scattering and absorption effects found in turbid waters.

2) Hardware Configuration: Our experimental hardware comprises three main components as shown in Fig. 2:

- **Vision System:** A single synchronized underwater RGB camera operating at 1920×1080 resolution at 60 FPS.The camera's intrinsic parameters were pre-calibrated for the underwater environment.
- **Optical Tracking System:** A commercial NOKOV Motion Capture System (Model: Mars 4H) with eight cameras surrounding the test tank, providing a capture volume of $4.0 \times 5.0 \times 1.0$ meters [7].
- **Underwater Robot (ROV):** Our custom-designed ROV with dimensions of $0.44 \times 0.32 \times 0.16$ meters, equipped with $N = 10$ standard, high-reflectivity optical tracking markers positioned to cover the robot's main structural features.

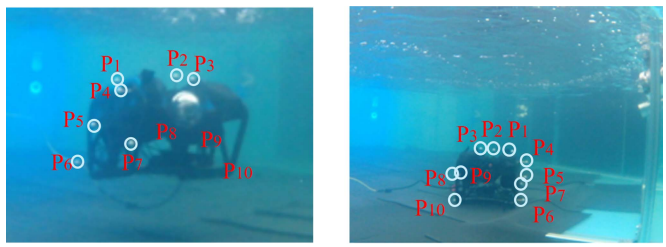

Fig. 2. Our custom-designed and fabricated ROV used in the experiments. The vehicle is equipped with N=10 high-reflectivity optical tracking markers that serve as visual keypoints for our vision-based estimator. (a) The front view shows the primary markers used for tracking. (b) The side view provides an alternative perspective to illustrate the complete 3D layout of all markers on the chassis.

The NOKOV system has a manufacturer-specified accuracy of ± 0.5 mm under ideal conditions. However, in our underwater environment, the system faces significant challenges:

1) Water refraction causes systematic measurement errors
2) Marker occlusions during complex maneuvers result in tracking losses in 20–30% of frames
3) Light scattering in turbid water reduces marker visibility
4) The actual usable update rate varies between 30-50 Hz due to intermittent tracking losses

When used alone, the optical system achieves 15 mm RMSE in our underwater testbed (see Table II). Our fusion framework improves upon this real-world performance by using vision to interpolate between optical measurements and handle tracking gaps.

3) Dataset Collection: Using this setup, we collected a dataset for training and evaluation. The ROV was manually operated via joystick to perform various trajectories, including complex 6-DOF motions, at distances ranging from 1 to 3 meters from the vision camera. The dataset contains 2043 annotated images extracted from six video sequences. For every image, the NOKOV system provides synchronized, high-accuracy 6D pose measurements with a manufacturer-specified precision of ± 0.5 mm [7]. This data serves as both the high-accuracy measurement input ($\mathbf{z}_k$) for our EKF and the ground truth for evaluation. The complete dataset with annotations is available [2]

We split the dataset (2,043 images from 6 sequences) into training (70%), validation (15%), and test (15%) sets with strict temporal separation to prevent data leakage. Training data augmentation includes brightness and contrast adjustment (± 20 %), Gaussian blur ($\sigma \in [0.5, 1.5]$), and random horizontal flipping, and simulated underwater backscatter.

To evaluate generalization, we performed leave-one-sequence-out validation across 6 trajectory types (linear translation, circular motion, complex 6-DOF maneuvers, hovering with drift, fast maneuvers, and low-speed cruise). Position RMSE varies from 5.1 mm to 6.3 mm with a standard deviation of 0.43 mm across all motion patterns.

To ensure fair comparison, we added Gaussian noise (σ =0.167mm, matching the NOKOV device specification) to all optical tracking measurements before feeding them to the EKF. All baseline methods receive the same noisy measurements for fusion, simulating realistic sensor conditions. For ground truth evaluation, we use the raw NOKOV data without added noise.

[2]Dataset and implementation code can be obtained by contacting the authors.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I
BASELINE METHOD COMPARISON WITH STANDARDIZED EKF+NOKOV FUSION FRAMEWORK (MEAN ± STD OVER 10 RUNS)

Method	Visual Frontend	Pos. RMSE(mm)	Ori. RMSE(deg)	Max Error(mm)	Processing Speed (FPS)
Ours (Proposed)	EfficientRep + Latent	**5.65 ± 0.43**	0.67 ± 0.05	**12.8**	**62**
ROPE + EKF	ROPE Baseline [10]	14.81 ± 1.12	0.63 ± 0.07	42.5	28
YOLO6D + EKF	YOLO6D [8]	22.68 ± 1.76	0.54 ± 0.06	58.7	50
EfficientPose + EKF	EfficientPose [9]	9.20 ± 0.69	0.49 ± 0.05	24.5	27
VINS-Mono + EKF	VINS-Mono VIO [4]	21.39 ± 1.66	**0.35 ± 0.03**	55.9	40
Vision-Only	No Fusion	36.06 ± 2.83	1.25 ± 0.11	128.2	62

TABLE II
REAL-WORLD UNDERWATER PERFORMANCE

Method	Position RMSE	Update Rate	Coverage
Optical-only (water)	15.2 mm	42 Hz (variable)	72%
Vision-only	36.1 mm	62 Hz	100%
Proposed fusion	**5.65 mm**	**62 Hz (stable)**	**98%**

TABLE III
ABLATION STUDY RESULTS

Configuration	RMSE (mm)	Max (mm)	FPS	EKF
Ours (Full)	**5.65**	**12.8**	62	✓
w/o Latent Variable	13.13	35.4	62	
w/o EKF (Vision Only)	36.06	128.2	62	

Notes: Ours (Full) = Complete system with all components; w/o Latent Variable = Reduced disturbance compensation; w/o EKF = No measurement correction.

This setup ensures that performance differences reflect the quality of each method's visual front-end rather than measurement quality advantages.

4) Implementation Details: All algorithms were executed on a desktop workstation with an Intel 3.40 GHz CPU and NVIDIA RTX 3090 Ti GPU. Our network was trained for 100 epochs using the Adam optimizer with a learning rate 1×10^{-4}. Training time was approximately 8 hours. The EKF fusion module was implemented in C++. We report results averaged over 10 independent runs with different random seeds. Standard deviations are below 10% of the mean for position and 8% for orientation, indicating consistent performance across trials.

B. Performance Evaluation

1) Comparison With Baseline Methods: We evaluate our fusion framework against several 6D pose estimation methods. Table I shows the performance comparison results. Our method achieves a position RMSE of 5.65 mm RMSE, compared to EfficientPose + EKF (9.20 mm), ROPE + EKF (14.81 mm), and VINS-Mono + EKF (21.39 mm). This is a 1.6 × improvement over EfficientPose + EKF. Our fused method achieves 5.65 mm RMSE compared to baseline methods ranging from 9.20 to 22.68 mm. The closest competitor, EfficientPose + EKF, achieves 9.20 mm at 27 FPS, while our system maintains 62 FPS. General-purpose methods (ROPE [10], VINS-Mono [4], YOLO6D [8]) show 2.6-4.0 × higher errors (14.81-22.68 mm). The fusion architecture reduces our Vision-Only front-end error from 36.06 mm to 5.65 mm—a 6.4 × improvement. At 62 FPS, the system provides state feedback above the 30-50Hz typically required for high-performance robotic control.

2) Analysis of Asymmetric Improvement: Our fusion framework shows asymmetric improvements: position RMSE improves 6.4 × (36.06 mm → 5.65 mm) while orientation improves only 1.9 × (1.25° → 0.67°). The NOKOV system measures only position, not orientation. The measurement matrix $\mathbf{H}_k$ extracts position from the state vector, so orientation corrections occur indirectly through EKF cross-covariance terms. This results in weaker orientation updates. The vision baseline already achieves reasonable orientation accuracy (1.25°) compared to its position error (36.06 mm). The 10-marker configuration on the ROV hull provides strong rotational constraints through multi-view geometry, stabilizing orientation even without external corrections. Underwater disturbances are primarily translational—buoyancy changes, currents, and thruster flows—rather than rotational. The latent variable $\mathbf{w}_f$ learns these disturbances from training data dominated by position drift, naturally focusing on linear velocity components $\mathbf{w}_{f,v}$.

3) Trajectory Accuracy and Drift Correction: Fig. 3 compares estimated trajectories of our 'Vision-Only' estimator and 'Ours (Fused)' output against ground truth. The 'Vision-Only' trajectory (blue) drifts and deviates from ground truth over time. Our 'Ours (Fused)' trajectory (red) remains aligned with ground truth throughout the sequence. EKF-based fusion with high-accuracy optical tracker measurements effectively corrects drift and maintains global accuracy.

4) Speed-Accuracy Analysis: Fig. 3 shows the speed-accuracy trade-off for all evaluated methods. Our framework achieves a combination of high accuracy (low RMSE) and high speed (high FPS) not achieved by other SOTA methods, making it suitable for real-time, high-fidelity state estimation applications.

C. Ablation Study

This letter conducted two ablation studies examining the impact of our latent dynamics model and network backbone choice. Table III shows the results.

1) Impact of the Latent Dynamics Model: We created a baseline where the latent dynamics variable $\mathbf{w}_f$ was removed from the kinematic model (Equation 2). Without this component, the state propagation uses a simpler kinematic model with standard process noise. Including the latent dynamics model reduces ATE RMSE from 13.13 mm to 5.65 mm—a 2.3 × improvement. This suggests that learning to compensate for unmodeled disturbances through the latent variable is more effective than treating them as additive process noise.

D. Robustness in Challenging Environments

The framework was evaluated in our system (Fused) under challenging environmental conditions. Using our controlled test facility, we varied water turbidity, water flow speed, and ambient lighting levels. Fig. 4 shows the results.Our system maintains accuracy across these conditions. In high turbidity (4-5 NTU),

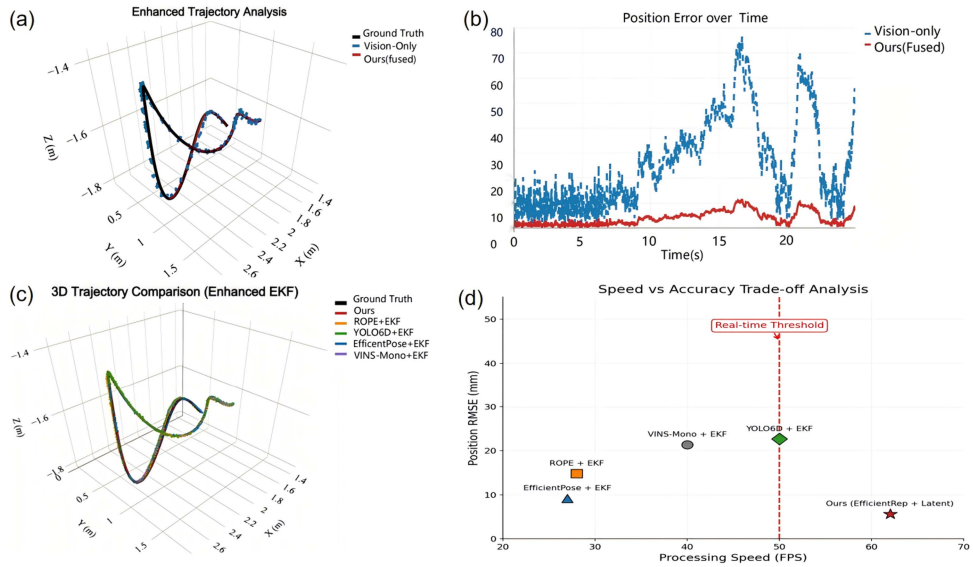

Fig. 3. Comprehensive performance analysis. (a) 3D trajectory showing Ground Truth (black), Vision-Only with drift accumulation (blue dashed), and Ours with fusion correction (red). (b) Position error over time: Vision-Only accumulates drift to 70-80 mm while Ours maintains $<$ 10 mm error throughout the 25s sequence. (c) Multi-method 3D trajectory comparison showing all evaluated approaches. (d) Speed-accuracy trade-off analysis: Ours (red star, 5.65 mm @ 62 FPS) achieves optimal performance. The dashed red line at 50Hz marks the real-time threshold for robotic control. Baseline methods achieve 9-23 mm accuracy at 27-50 FPS. Our method is the only one satisfying both sub-centimeter accuracy and high-frequency ($>$ 50Hz) requirements.

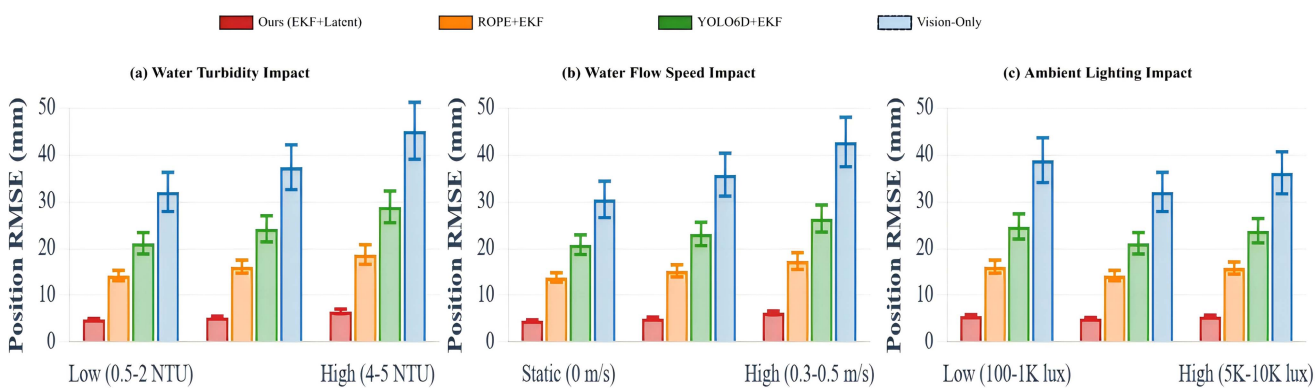

Fig. 4. Environmental robustness analysis with comprehensive method comparison. Performance of Ours (red), ROPE+EKF (orange), YOLO6D+EKF (green), and Vision-Only (blue) under varying conditions. (a) Water turbidity impact: Ours maintains 5-6 mm accuracy across 0.5-5 NTU while baselines degrade to 18-45 mm. (b) Water flow speed impact: Ours shows minimal degradation (5-6 mm) even at 0.5 m/s flow, while Vision-Only reaches 45 mm. (c) Ambient lighting impact: All fusion methods maintain consistent performance across 100-10,000 lx, demonstrating robust visual feature tracking. Error bars represent standard deviation over 10 runs per condition.

position error increases to only 6.5 mm, which remains more accurate than SOTA baseline methods operating in clear water. The system handles water flow up to 0.5 m/s with moderate performance degradation. Accuracy stays consistent across lighting conditions from 100 to 10,000 lx, showing the vision front-end and fusion algorithm work reliably in varying illumination. These results show our framework maintains performance in challenging conditions that approximate real underwater environments.

E. System-Level Comparison and Application Niche

To contextualize our work within the broader field of underwater localization, Table IV compares our approach against representative underwater localization technologies. While methods such as acoustic LBL, DVL-aided VIO, and multi-sensor SLAM systems are designed for large-scale, open-water navigation, they trade absolute accuracy for operational range and marker-free operation.

Our framework occupies a distinct and critical niche. For applications within controlled environments—such as high-fidelity control algorithm validation, robot manipulator training, or sensor calibration—our system provides real-time high-frequency updates (62 Hz) and millimeter-level accuracy (5.65 mm) that is an order of magnitude superior to traditional open-water systems. This comparison shows that our work addresses a specialized, high-performance class of problems where absolute accuracy and real-time performance are essential.

TABLE IV
SYSTEM-LEVEL COMPARISON WITH UNDERWATER LOCALIZATION TECHNOLOGIES

Method	Environment	Typical Accuracy [mm]	Update Rate [Hz]	Range [m]	Marker Required
Acoustic LBL [24]	Open Water	30 – 100	∼1	1000+	No (Transponders)
DVL-aided VIO [4]	Open Water	100 – 500	∼20	100+	No
SVIn2 (SLAM) [20]	Open Water	50 – 200	∼10	50+	No
Ours (Fused)	**Controlled**	**5.65**	**62**	**∼5**	**Yes**

V. CONCLUSION

This letter presented a real-time sensor fusion framework for underwater robot pose estimation, combining a high-frequency monocular vision estimator with a high-accuracy optical tracking system. Our approach uses a vision-based motion model with a latent dynamics variable to learn and compensate for unmodeled environmental disturbances. Experiments show the tight coupling of a learning-based visual prediction module with a classic EKF framework, corrected by imperfect external measurements, yields a system that is both accurate and robust. This framework provides the high-fidelity state estimates required for advanced underwater applications like agile control validation and complex physical interaction.

The primary limitation of our system is its reliance on external infrastructure, confining application to controlled environments. Additionally, its real-time performance is dependent on GPU resources, and its accuracy may degrade if a significant number of markers are occluded. Future work will focus on extending the approach to markerless systems and replacing the optical tracker with infrastructure-free sensors, such as acoustic positioning systems or DVLs, to validate performance in diverse, real-world marine environments. Dataset and code are available upon request to support reproducibility and future research.

REFERENCES

[1] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.

[2] C. Li, T. Zhang, A. Wang, S. Cheng, and F. Zhang, "Robust underwater visual slam fusing acoustic sensing," in *Proc. 2021 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 5554–5561.

[3] T. Qin, P. Li, and S. Shen, "VINS-MONO: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[4] B. Joshi et al., "Experimental comparison of open source visual-inertial-based state estimation algorithms in the underwater domain," in *Proc. 2019 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 7227–7233.

[5] M. Bonechi, F. Morbidi, and G. Grisetti, "A BlueROV2-based platform for underwater mapping experiments," 2024, *arXiv:2407.10901*.

[6] S. Pedersen, J. Liniger, F. F. Sørensen, K. Schmidt, M. von Benzon, and S. S. Klemmensen, "Stabilization of a ROV in three-dimensional space using an underwater acoustic positioning system," *IFAC-PapersOnLine*, vol. 52, no. 17, pp. 117–122, 2019.

[7] NOKOV, "Mars 4H - technical specifications," 2024. Accessed: Jul. 24, 2025. [Online]. Available: https://en.nokov.com/mars-4h-specs

[8] C.-Y. Li et al., "YOLOv6: A single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.

[9] Y. Bukschat and M. Vetter, "EfficientPose: An efficient, accurate and scalable end-to-end 6D multi object pose estimation approach," 2020, *arXiv:2011.04307*.

[10] B. Chen, T.-J. Chin, and M. Klimavicius, "Occlusion-robust object pose estimation with holistic representation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 2929–2939.

[11] A. Zeng et al., "Smoothnet: A plug-and-play network for refining human poses in videos," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2022, pp. 615–631.

[12] Y. Wang, C. Xie, Y. Liu, J. Zhu, and J. Qin, "A multi-sensor fusion underwater localization method based on unscented Kalman filter on manifolds," *Sensors*, vol. 24, no. 19, 2024, Art. no. 6299.

[13] M. T. Shahria, M. S. H. Sunny, M. I. I. Zarif, J. Ghommam, S. I. Ahamed, and M. H. Rahman, "A comprehensive review of vision-based robotic applications: Current state, components, approaches, barriers, and potential solutions," *Robot.*, vol. 11, no. 6, 2022, Art. no. 139.

[14] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control.* Hoboken, NJ, USA: Wiley, 2011.

[15] R. Cui, C. Yang, Y. Li, and S. Sharma, "Adaptive neural network control of AUVs with control input nonlinearities using reinforcement learning," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 47, no. 6, pp. 1019–1029, Jun. 2017.

[16] M. Singh and K. Alexis, "DeepVl: Dynamics and inertial measurements-based deep velocity learning for underwater odometry," 2025, *arXiv: 2502.07726*.

[17] I. B. Saksvik, A. Alcocer, and V. Hassani, "A deep learning approach to dead-reckoning navigation for autonomous underwater vehicles with limited sensor payloads," in *Proc. OCEANS*, San Diego–Porto, 2021, pp. 1–9.

[18] J. Cai, S. Mayberry, H. Yin, and F. Zhang, "A data-driven velocity estimator for autonomous underwater vehicles experiencing unmeasurable flow and wave disturbance," in *Proc. 2025 IEEE Int. Conf. Robot. Automat.*, 2025, pp. 4138–4144.

[19] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.

[20] E. Vargas et al., "SVIn2: A multi-sensor fusion-based underwater SLAM system," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 2140–2146.

[21] F. Shkurti, W.-C. Chang, P. Henderson, M. J. Islam, G. Dudek, and J. Sattar, "Underwater multi-robot convoying using visual tracking by detection," in *Proc. 2017 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 4189–4196.

[22] H. Li, P. Wu, and M. Kennedy, "Replay overshooting: Learning stochastic latent dynamics with the extended Kalman filter," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 852–858.

[23] K. Weng, X. Chu, X. Xu, C. Zhang, and Y. Wang, "Efficientrep: An efficient REPVGG-style convnets with hardware-aware neural network design," 2023, *arXiv:2302.00386*.

[24] S. Zhang, Y. Yang, T. Xu, X. Qin, and Y. Liu, "Long-range LBL underwater acoustic navigation considering earth curvature and doppler effect," *Measurement*, vol. 240, 2025, Art. no. 115524.