Machine Learning-Based UAV Localization for Under-Structure Sensor Deployment in GPS-Denied Environments

Md Asifuzzaman Khan[a], Joud N. Satme[a], Mark Zheng[a], Samuel Tadamatla[a,b], Evan Phillips[a,c], Korebami Adebajo[a], Nolan Shute[a], Ryan Yount[a], and Austin R.J. Downey[a,d]

[a]Department of Mechanical Engineering, University of South Carolina, Columbia, USA
[b]Department of Computer Science, University of South Carolina, Columbia, USA
[c]Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, USA
[d]Department of Civil and Environmental Engineering, University of South Carolina, Columbia, USA

ABSTRACT

Structural health monitoring (SHM) is vital for ensuring the safety and longevity of civil infrastructure such as bridges and overpasses. Portable sensors affixed directly to structural surfaces provide high-quality data on strain, vibration, and fatigue, supporting timely maintenance decisions. However, deploying sensors in under-structure environments remains challenging due to limited accessibility and safety concerns. Unmanned aerial vehicles (UAVs) offer an efficient alternative for sensor deployment, but their navigation stability is severely affected in GPS-denied areas such as beneath bridges. Reliable localization methods are therefore essential to enable autonomous under-structure sensor deployment. This study presents a machine learning (ML)-based vision localization framework tailored for UAV operation in GPS-denied under-structure environments. A dual-camera system captures real-time UAV imagery, and an open-source object tracking algorithm extracts image-plane coordinates. These coordinates are paired with ground-truth positions from an external reference system to create training datasets for supervised ML models. The trained models map pixel-level observations to absolute spatial coordinates, providing a vision-based alternative to GPS for UAV localization. Several ML algorithms are implemented and comparatively analyzed for localization accuracy. Results demonstrate improved UAV stability and navigation reliability under bridge-like conditions. The contributions of this work include: (1) an experimental procedure for developing supervised ML models for UAV localization, and (2) a grid search method to evaluate and identify the most optimal ML model for accurate 3D localization. Together, these deliverables advance UAV-enabled SHM by enabling safe, autonomous, and efficient sensor deployment in GPS-denied under-structure environments.

Keywords: UAV, ML, GPS-denied localization, structural health monitoring, UAV deployable sensor packages, object detection, YOLO, vision-based navigation

1. INTRODUCTION

Unmanned aerial vehicles (UAVs) are increasingly used in civil infrastructure applications that require close interaction with structural surfaces, such as deploying portable sensors on the underside of bridges and overpasses. These sensors, affixed directly to structural surfaces, provide high-resolution measurements of strain, vibration, and other response quantities that support structural health monitoring (SHM) and condition assessment.[1] As shown in Figure 1, successful sensor placement requires the UAV to approach the structure, position itself accurately relative to a target location, and maintain a stable hover, all of which depend on reliable localization. However, standard localization solutions like the global positioning system (GPS) are not well-suited to under-structure environments. GPS signals are frequently attenuated, reflected, or completely blocked beneath bridge

Further author information: (Send correspondence to Austin Downey)
Austin Downey: Email: austindowney@sc.edu

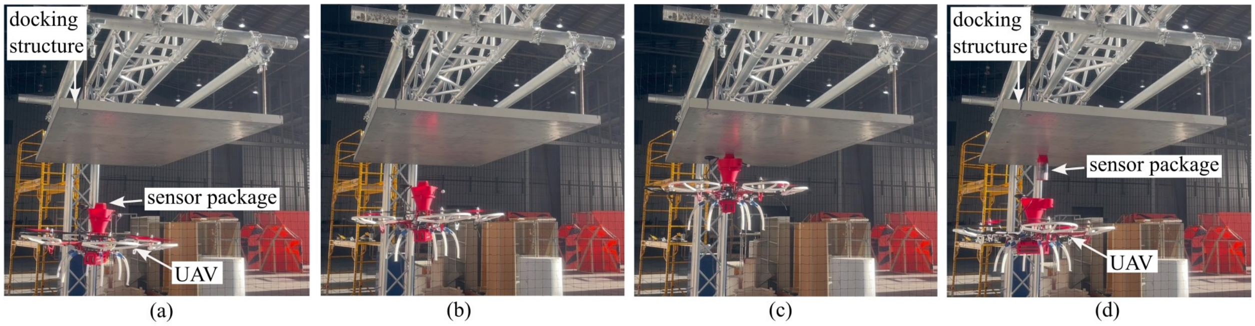

Figure 1. A demonstration of sensor package deployment showing a)UAV approaching the target structure, b) positioning relative to the target location, c) maintaining stable hover while pressing the sensor against the structure, and d) completion of deployment.

decks and in proximity to large structural members,[2] which leads to unreliable or unavailable position estimates. Navigation only using inertial measurement units (IMUs), also known as dead reckoning, suffers from error accumulation over time[3] during slow and precise maneuvers. Alternative technologies such as ultra-wideband ranging suffer from multi-path reflection near metal structures, affecting accuracy.[4] Manual line of sight control is feasible in very close proximity, but it is difficult to properly aim the UAV, seeing it only from one side when it's further back. Moreover, phenomena like ceiling effect[5] make manual control even more difficult.

Vision-based localization has emerged as an attractive option in this context. Cameras are capable of providing rich information about the environment. Fiducial markers require the UAV or the docking zone to have special tags of different sizes, which at times can lead to complexities.[6] Onboard vision methods, including visual odometry and visual simultaneous localization and mapping (SLAM), estimate the motion of the UAV from images recorded by cameras mounted on the vehicle. These methods have demonstrated good performance but are computationally heavy to be onboard a UAV. A complementary approach is to relocate the vision system off-board. In this case, one or more fixed cameras observe the UAV from the environment rather than from the vehicle. External cameras can be placed to cover the relevant workspace and are not subject to the same field-of-view and occlusion constraints that affect onboard sensors. Previous studies have considered monocular or stereo external camera systems that track a UAV using color segmentation[7] and the geometric shape of the UAV viewed by the camera.[8] These methods can achieve good accuracy in controlled settings. However, they typically rely on a UAV of a specific color and geometry, respectively. Moreover, lens distortions, slight camera misalignments, and partial occlusions introduce systematic errors that can reduce localization accuracy.

Machine learning (ML) offers a way to mitigate some of these limitations. Zheng et. al. proposed a technique that uses the open-source You Only Look Once (YOLO) algorithm to extract 3D pixel coordinates of a UAV using a stereo camera setup.[9] However, those pixel-level coordinates don't map linearly to the 3D real-world Cartesian coordinate system due to lens distortions and camera misalignment. Instead of relying solely on explicit geometric relationships, an ML-based regressor can be trained to map visual measurements directly to spatial coordinates. When provided with sufficiently rich training data, such a model can learn to compensate for systematic biases associated with non-ideal optics, imperfect calibration, and other unmodeled effects. In this study, a machine learning–based vision localization framework tailored for UAV operation in GPS-denied under-structure environments is developed. The framework employs a dual-camera system that continuously observes the UAV. Image-plane coordinates of the UAV are extracted from each camera view. These pixel-level observations are synchronized with ground-truth positions obtained from an external reference system, thereby forming training datasets for supervised learning. The objectives of this study are twofold: 1) to establish an experimental procedure for constructing supervised learning datasets for UAV localization using dual external cameras, and 2) to evaluate and compare multiple ML algorithms with respect to localization accuracy within the operational workspace.

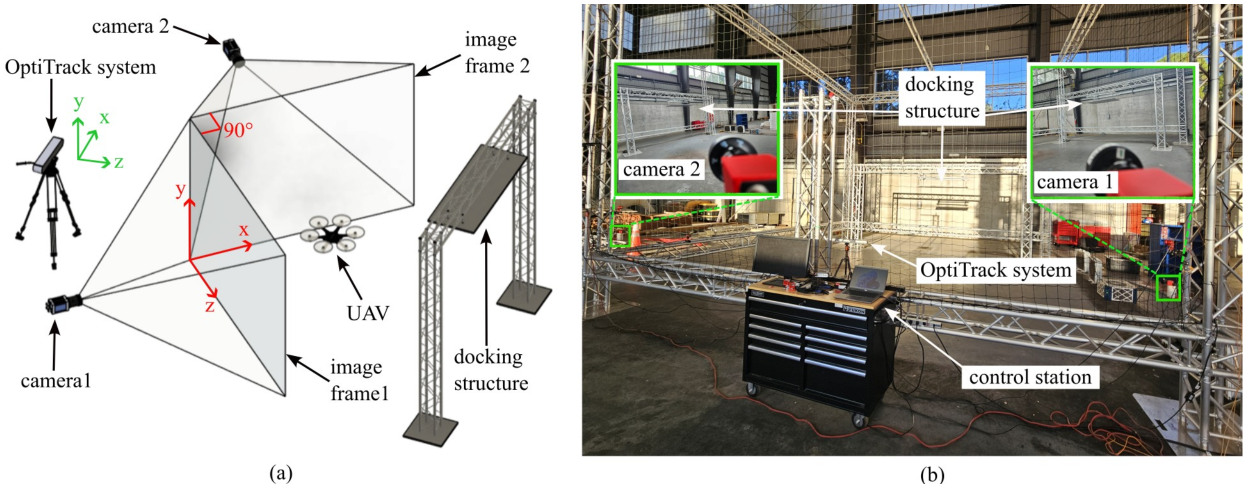

Figure 2. The camera and OptiTrack setup showing a) illustration of the axis for both coordinate systems (OptiTrack in green, camera in red), b) the physical implementation of the system within an UAV cage.

2. METHODOLOGY

This section describes the hardware setup for obtaining the dataset and the selection of machine learning algorithms for evaluation.

2.1 Hardware setup and data acquisition

To collect the pixel-level coordinates from the camera and reference ground truth coordinates, a test environment is set up within an UAV-safe cage as shown in Figure 2. Two monocular cameras were installed to cover the UAV operation area, with their image planes arranged in a way that they intersect at approximately 90° angle. This configuration provides an overlapping field of view in which the UAV remains visible to both cameras during the experiment. An open-source object detection and tracking algorithm was applied to each image stream to detect the UAV and to extract its horizontal and vertical pixel locations in every frame, as described in the study by Zheng et. al.[9] The image plane of camera 1 is used to define the x–y directions of the pixel-level coordinate system, while the image plane of camera 2 defines the z–y directions. Consequently, the horizontal pixel coordinate in image frame 1 corresponds to the x-axis, the horizontal coordinate in image frame 2 corresponds to the z-axis, and the vertical coordinate in both frames represents the common y-axis. Small camera misalignments can introduce slight differences between the vertical pixel locations obtained from the two images. Therefore, the y-coordinate is computed as the average of the two vertical measurements. The resulting triplet (x, y, z) forms the pixel-level representation of the UAV position.

Ground-truth spatial coordinates of the UAV are obtained from an external OptiTrack motion-capture system,[10] which tracks an optical marker mounted on the UAV and reports the corresponding x, y, and z positions with millimeter-level accuracy. These measurements are time-synchronized with the image data and are used to label the pixel-based coordinates for supervised learning. The flight path of the UAV in both coordinate systems is plotted and shown in Figure 3.

2.2 Selection and assessment of ML algorithms

As shown in Figure 3, the trajectories obtained from the stereo pixel coordinates and from the OptiTrack system do not coincide exactly, although they follow a similar overall trend. This visual mismatch suggests that a simple linear mapping between the two coordinate systems may not be sufficient. To examine this more formally, a Pearson correlation[11] analysis is performed between each axis of the stereo pixel system and each axis of the OptiTrack system. A correlation value of 1 indicates a perfect positive linear relationship, whereas a value of -1 indicates a perfect negative linear relationship. As shown in Figure 4, the diagonal terms exhibit relatively high correlations between corresponding axes, with values of 0.83 for x, 0.99 for y, and 0.81 for z. However, several off-diagonal terms also have appreciable magnitudes, such as the correlation of 0.58

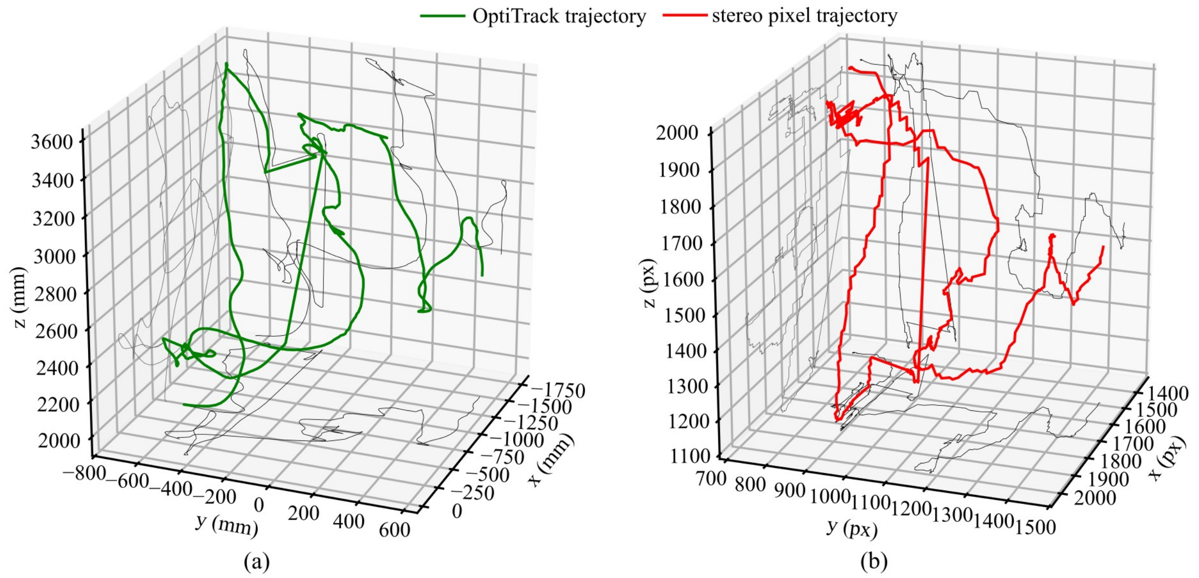

Figure 3. Flight path of the UAV represented in 3D coordinates in a) the OptiTrack system in mm and b) image pixel coordinates combining the two camera feeds.

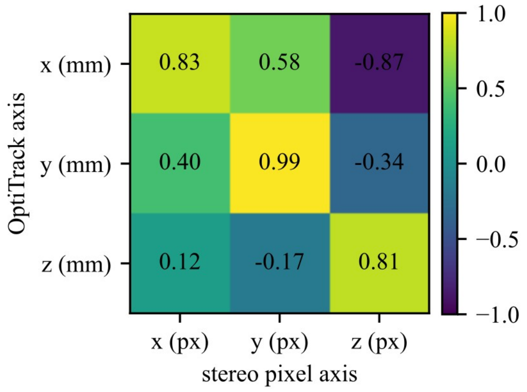

Figure 4. Pearson correlation matrix showing the non-linear relation between the two coordinate systems.

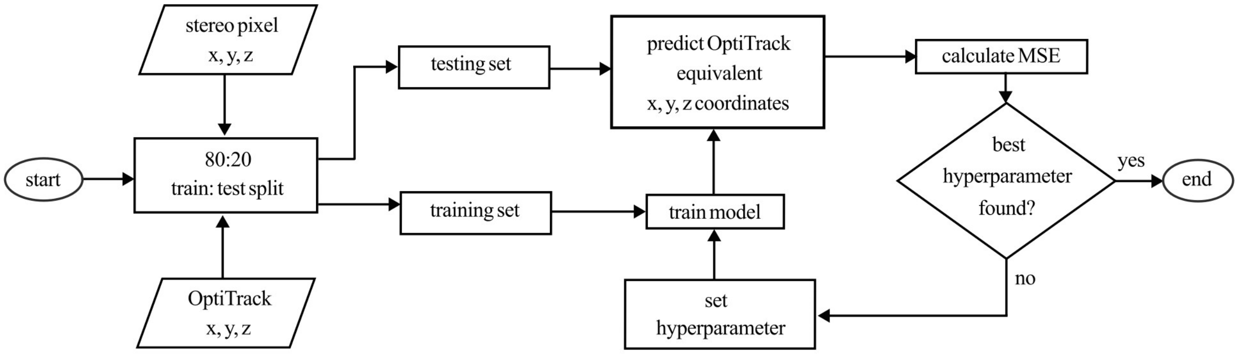

Figure 5. Flow chart describing the analysis of each machine learning algorithm used to predict the 3D spatial coordinate of the UAV from the stereo pixel coordinate.

between OptiTrack x and stereo y, and -0.87 between OptiTrack x and stereo z. These results indicate that the axes of the two coordinate systems are not independent and that there is significant cross-coupling between them. Moreover, the correlations deviate from ±1, which implies that even along the dominant directions, the relationship is not perfectly linear. These observations motivate the use of more flexible machine-learning models rather than a simple coordinate transformation to map between the two systems. Since the study deals with non-linear continuous time data, the prediction task is formulated as a multivariate non-linear regression problem. Let $\mathbf{p}_{\mathrm{px}} = [x_{\mathrm{px}}, y_{\mathrm{px}}, z_{\mathrm{px}}]^{\mathsf{T}}$ denote the stereo pixel coordinates and $\mathbf{p}_{\mathrm{mm}} = [X_{\mathrm{mm}}, Y_{\mathrm{mm}}, Z_{\mathrm{mm}}]^{\mathsf{T}}$ denote the corresponding OptiTrack coordinates. Regression ML models are trained to approximate the mapping $f : \mathbf{p}_{\mathrm{px}} \mapsto \mathbf{p}_{\mathrm{mm}}$. Several non-linear supervised regression algorithms are considered, including polynomial regression, support vector regression, random forest regression, and gradient boosting regression. The stereo pixel coordinates and corresponding OptiTrack coordinates are first partitioned into training and testing subsets using an 80:20 split, as illustrated in Figure 5. The training subset is used to fit a set of supervised regression models, where a grid search procedure explores combinations of hyperparameters for each algorithm. For every candidate hyperparameter configuration, the model is trained on the training data and then applied to the testing set to predict the OptiTrack-equivalent x, y, and z coordinates. The performance of each configuration is quantified by the 5-fold cross-validated[12] mean squared error (MSE) between the predicted coordinates and the corresponding ground-truth OptiTrack measurements. The hyperparameter set that yields the lowest MSE is selected as the optimal configuration for that algorithm.

3. RESULTS

3.1 Model comparison and trajectory reconstruction

Table 1 summarizes the performance of the four regression models and their selected hyperparameters. Among the candidate models, the support vector regression (SVR) model with a radial bias function (RBF) kernel achieved the lowest validation error. Its five-fold cross-validated mean squared error (MSE) was 271.82 mm^2, corresponding to a root mean squared error (RMSE) of 16.48 mm. The gradient boosting regression model yielded a slightly higher MSE of 283.73 mm^2 and an RMSE of 16.84 mm. Polynomial regression with degree six and ridge regularization produced an MSE of 402.67 mm^2 and an RMSE of 18.66 mm, while the random forest model obtained an MSE of 353.82 mm^2 and an RMSE of 18.81 mm. These results indicate that the SVR model provides the best overall accuracy for the considered data set and is therefore selected for further analysis.

The selected SVR model with its optimal hyperparameters is used to predict the OptiTrack-equivalent coordinates for the test data. Figure 6 shows a three-dimensional comparison between the reconstructed trajectory and the actual OptiTrack trajectory. The predicted path closely follows the ground-truth path over the entire flight. The two curves overlap over most segments of the motion, and only small local deviations are visible near regions of rapid manoeuvres. This visual agreement confirms that the regression model is able to reproduce the overall spatial evolution of the UAV.

Table 1. performance comparison between the non-linear regression models with their best selected hyperparameters

	polynomial regression	support vector regression (RBF kernel)	random forest regression	gradient boosting regression
hyperparameter 1	number of degrees	regularization strength (c)	maximum depth	maximum depth
hyperparameter 1 value	6	1e5	20	4
hyperparameter 2	ridge regularization (α)	kernel width (γ)	number of trees	number of trees
hyperparameter 2 value	1e-5	1	350	350
5-fold cross validated MSE (mm^2)	402.67	271.82	353.82	283.73
5-fold cross-validated RMSE (mm)	18.66	16.48	18.81	16.84

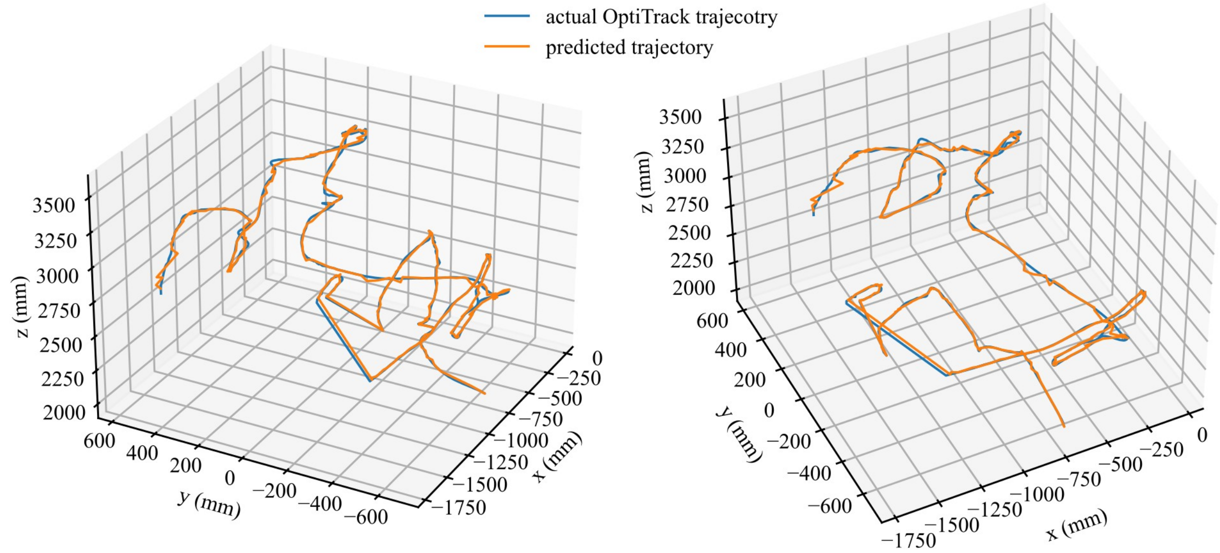

Figure 6. Comparison of the drone's 3D trajectory, showing predicted flight paths using a support vector regression model and true OptiTrack flight paths in mm units.

A more detailed assessment is obtained from an axis-wise absolute error analysis of the SVR model. This analysis is carried out using only the 20% testing subset, which is held out during training and hyperparameter tuning and is therefore not seen by the model beforehand. Figure 7 presents the absolute differences between the predicted and OptiTrack coordinates for the x, y, and z axes. The plots show that the majority of errors remain relatively small for all three directions, with occasional peaks that correspond to the maximum absolute error on each axis. These peak values quantify the worst-case deviation of the model on previously unseen data. For x axis, the max deviation is around 125 mm, for y axis it's around 72 mm, and for z axis it's around 57 mm.

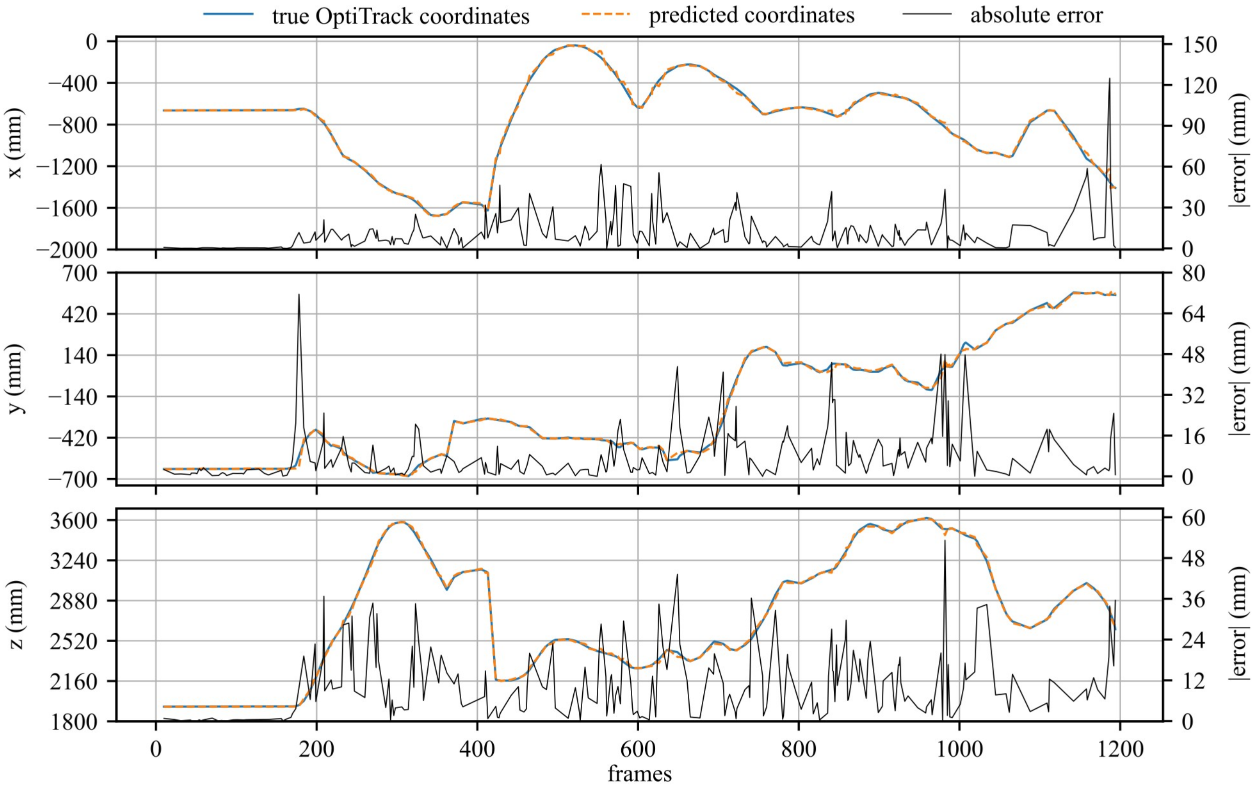

Figure 7. Comparison between the SVR model predicted coordinates and true OptiTrack coordinates in each axis for the 20 percent test data set.

4. CONCLUSION

This study presented a machine-learning-based external vision localization framework for UAV operation in GPS-denied under-structure environments. A dual-camera setup was used to obtain stereo pixel coordinates of a UAV performing sensor-deployment maneuvers, while an OptiTrack motion-capture system provided millimeter-accurate ground-truth positions. The mapping from stereo pixel space to OptiTrack coordinates was formulated as a multivariate nonlinear regression problem, and several regression algorithms were evaluated, including polynomial regression, support vector regression, random forest regression, and gradient boosting regression. Hyperparameters for each model were selected using a grid search with five-fold cross-validation, and performance was assessed using the cross-validated MSE and RMSE. Among the considered models, the support vector regression (SVR) model with an RBF kernel achieved the lowest error, with an RMSE of 16.48 mm. The reconstructed trajectory generated by the SVR model closely matched the OptiTrack trajectory, with strong overlap over most of the flight path and only small local deviations near rapid maneuvers. An axis-wise absolute error analysis on the held-out 20% test set showed that the majority of errors remained relatively small, with maximum deviations of approximately 125 mm in x, 72 mm in y, and 57 mm in z axis.

These results demonstrate that the proposed regression-based localization framework can provide accurate and reliable position estimates for UAVs operating beneath bridge-like structures, without requiring GPS, fiducial markers, or additional infrastructure on the structure itself. Future work will focus on expanding the data set to include a wider range of UAV trajectories and operating conditions, reducing latency in the real-time detection and localization of the UAV, and implementing a closed-loop control system that uses the vision-based localization to stabilize the UAV during docking under a structure in the absence of GPS.

5. ACKNOWLEDGMENTS

This material is based upon work supported by the Air Force Office of Scientific Research (AFOSR) through award no. FA9550-21-1-0083. This work is also partly supported by the National Science Foundation (NSF) grant numbers CCF - 1956071, CMMI - 2152896, CCF-2234921 and, CPS - 2237696. Additional funding for this work comes from the Office of Naval Research through the award number N000142412727. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force, the National Science Foundation, or the United States Navy.

REFERENCES

[1] Carroll, S., Satme, J., Alkharusi, S., Vitzilaios, N., Downey, A., and Rizos, D., "Drone-based vibration monitoring and assessment of structures," *Applied Sciences* **11**, 8560 (Sept. 2021).

[2] Kos, T., Markezic, I., and Pokrajcic, J., "Effects of multipath reception on gps positioning performance," in [*Proceedings ELMAR-2010*], 399–402 (2010).

[3] Petritoli, E., Leccese, F., and Spagnolo, G. S., "Inertial navigation systems (ins) for drones: Position errors model," in [*2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*], 500–504, IEEE (June 2020).

[4] Silva, B. and Hancke, G. P., "Ir-uwb-based non-line-of-sight identification in harsh environments: Principles and challenges," *IEEE Transactions on Industrial Informatics* **12**, 1188–1195 (June 2016).

[5] Conyers, S. A., Rutherford, M. J., and Valavanis, K. P., "An empirical evaluation of ceiling effect for small-scale rotorcraft," in [*2018 International Conference on Unmanned Aircraft Systems (ICUAS)*], IEEE (June 2018).

[6] Wang, R., Guo, H., Wang, X., and Han, L., "The effect of aruco marker size, number, and distribution on the localization performance of fixed-point target10.1109/rcae59706.2023.10398770s," in [*2023 6th International Conference on Robotics, Control and Automation Engineering (RCAE)*], 118–123, IEEE (Nov. 2023).

[7] Andrean, S. Y., Joelianto, E., Widyotriatmo, A., and Adiprawita, W., "Low cost vision-based 3d localization system for indoor unmanned aerial vehicles," in [*2013 International Conference on Robotics, Biomimetics, Intelligent Computational Systems*], 237–241, IEEE (Nov. 2013).

[8] Himawan, R. W., Baylon, P. B. A., Sembiring, J., and Jenie, Y. I., "Development of an indoor visual-based monocular positioning system for multirotor uav," in [*2023 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*], 1–7, IEEE (Oct. 2023).

[9] Zheng, Q. M., Khan, A., N., J., Adebajo, K., Yount, R., and Downey, A. R., "Stereo yolo uav localization and tracking enabling autonomous sensor deployment on critical infrastructure," in [*AIAA SCITECH 2026 Forum*], American Institute of Aeronautics and Astronautics (Jan. 2026).

[10] NaturalPoint Inc., "OptiTrack Motion Capture System." https://www.optitrack.com (2024). Accessed: 2026-01-15.

[11] Benesty, J., Chen, J., Huang, Y., and Cohen, I., [*Pearson Correlation Coefficient*], 1–4, Springer Berlin Heidelberg (2009).

[12] Arlot, S. and Celisse, A., "A survey of cross-validation procedures for model selection," *Statistics Surveys* **4** (Jan. 2010).