Water Level Identification Using LiDAR and Machine Learning

*Applied Sensor Fusion for Predictive Flood Monitoring*

Project Summary

This project explored the use of LiDAR and machine learning (ML) to predict water levels in a controlled lab environment simulating river conditions. The goal was to evaluate whether non-contact sensors like LiDAR could reliably measure water level across varying turbidity (clarity) levels and whether ML models could be trained to predict those levels using sensor data.

The results demonstrated that LiDAR-based systems are highly effective for this task—even in murky or turbulent water—making them a strong candidate for real-world flood monitoring or environmental sensing.

Sensors & Data Overview

Sensor inputs were collected under three turbidity conditions (low, medium, high) from the following devices:

* LiDAR sensor – measured distance and signal strength
* Ultrasonic sensor – measured distance
* Accelerometer and Gyroscope – captured motion/orientation data

Each sensor stream was recorded alongside known water levels, producing a labeled dataset suitable for supervised learning.

Modeling Approach

This was structured as a regression task, where multiple ML models were trained to predict water level from sensor readings. The pipeline included:

* Exploratory Data Analysis (EDA) to assess variable correlation
* Data preprocessing and scaling (for distance-based models)
* Hyperparameter tuning via Grid Search
* Performance evaluation using:
  + Mean Absolute Error (MAE)
  + Mean Squared Error (MSE)
  + Root Mean Squared Error (RMSE)
  + Average residual

Key Findings

* LiDAR distance had the strongest and most consistent correlation to water level across all turbidity conditions.
* Tree-based models significantly outperformed others, especially:
  + Gradient Boosting – top performer in every error metric
  + Single Decision Tree – very accurate with minimal tuning
  + Random Forest – consistently strong performer
* Ultrasonic data was weakly predictive, particularly in high turbidity.
* Accelerometer and gyroscope data contributed little to prediction quality.
* XGBoost underperformed due to unexpected output behavior (flat predictions), and was excluded from ensemble results.

Performance Snapshot (Top Models)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | MAE | MSE | RMSE | Avg. Residual |
| Gradient Boosting | *Best* | *Best* | *Best* | < 100 units |
| Decision Tree | 2nd | 2nd | 2nd | *Lowest* |
| Random Forest | 3rd | 3rd | 3rd | 4th |
| Polynomial Regression | 7th | 7th | 7th | Several outliers |
| SVM | 8th | 8th | 8th | Skewed errors |
| XGBoost | *Last* | *Last* | *Last* | Constant value |

Project Impact & Next Steps

This project illustrates the practical use of LiDAR with ML for environmental monitoring, even in visually degraded conditions. The strong performance of tree-based models shows that effective solutions can be built without needing deep learning, keeping compute and deployment costs low.

Next directions could include:

* Training on real-world river datasets
* Testing neural network architectures
* Integrating models with IoT devices for edge deployment
* Exploring forecasting models based on sequential data

Conclusion

LiDAR combined with machine learning provides a powerful, scalable solution for water level prediction—even in challenging conditions. This project lays the groundwork for intelligent flood monitoring and real-time environmental sensing systems using affordable, off-the-shelf hardware.

References:

1. <https://harcresearch.org/news/when-technology-and-nature-collide-lidar-and-the-importance-of-biodiversity/?gad_source=1&gad_campaignid=9558509408&gbraid=0AAAAACg5UaQcPomhU5IOhcKP991S_Zt33&gclid=Cj0KCQjw-ZHEBhCxARIsAGGN96JqAcTmzLz95JuFdmWgYF7HZ0INDpa1naJ_Ixy2ex9dtBtA26Crut0aAgWUEALw_wcB>
2. <https://www.kaggle.com/datasets/caetanoranieri/water-level-identification-with-lidar>
3. See Attached File
4. <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>
5. <https://aws.amazon.com/what-is/hyperparameter-tuning/>
6. <https://medium.com/@abhishekjainindore24/svm-kernels-and-its-type-dfc3d5f2dcd8>
7. <https://duchesnay.github.io/pystatsml/machine_learning/ensemble_learning.html>
8. <https://medium.com/analytics-vidhya/what-is-gradient-boosting-how-is-it-different-from-ada-boost-2d5ff5767cb2>
9. <https://xgboost.readthedocs.io/en/stable/parameter.html>