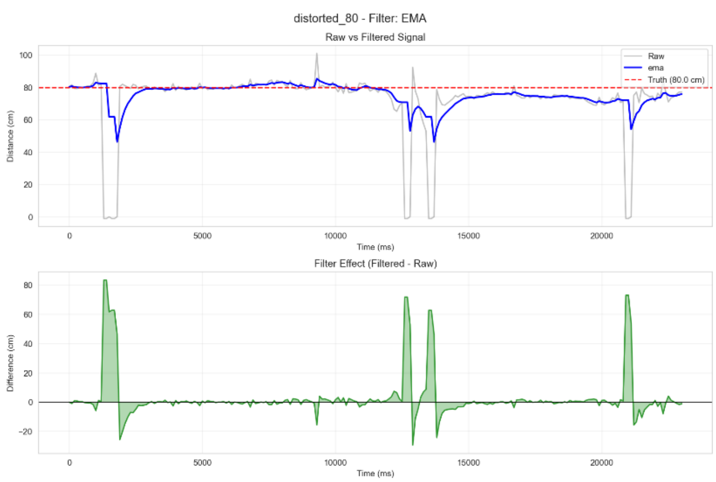
Filter Analysis

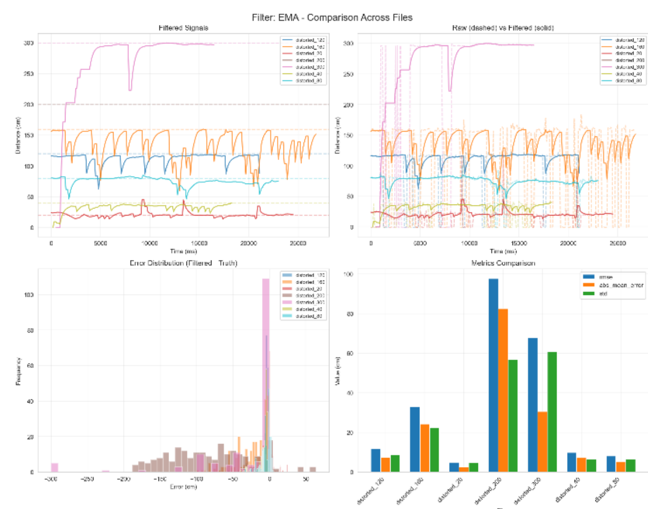
Distance Sensor HC-SR04P Analysis

We tested the HC-SR04P ultrasonic distance sensor using an ESP32 running our filtering firmware to compare raw and filtered readings at different fixed distances (20–300 cm). The sensor sampled every 100 ms, and data—including outputs from EMA, Low-Pass, Median, Outlier, and Kalman filters—was streamed over serial and logged with our Python script. For each distance, we collected 15–25 seconds of measurements. The analysis compared noise reduction, responsiveness, and stability across filters, showing that the Median and Outlier filters removed spikes, EMA and Low-Pass provided smooth curves with minor lag, and the Kalman filter achieved the best overall balance, resulting in cleaner and more reliable distance readings.

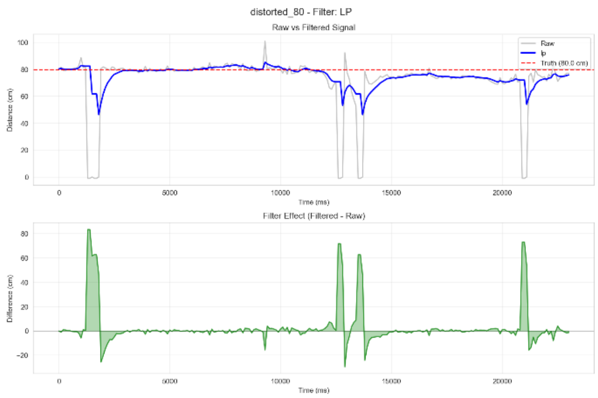
We'll show one example graph of a certain distance as a function of time. And all the distances graph and that is per filter:

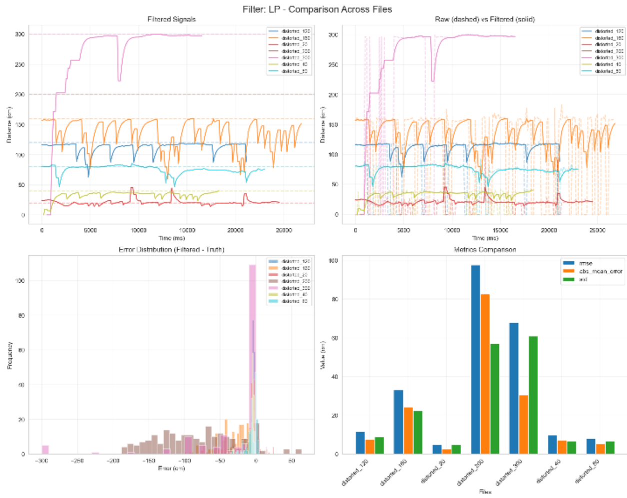
1) EMA



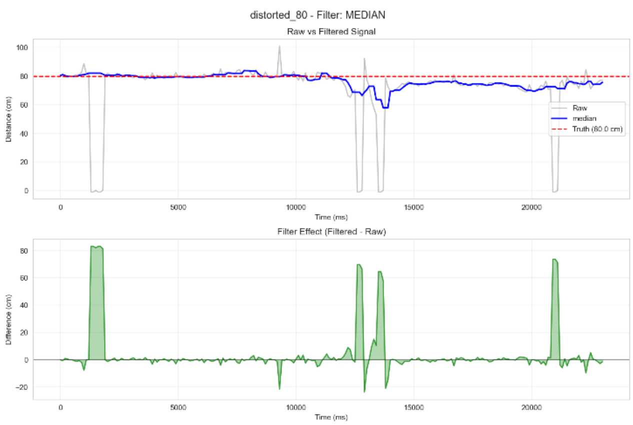


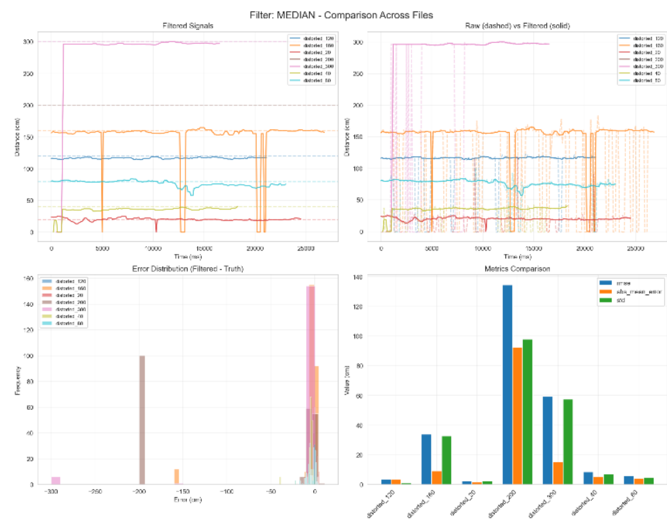
2) Low pass filter



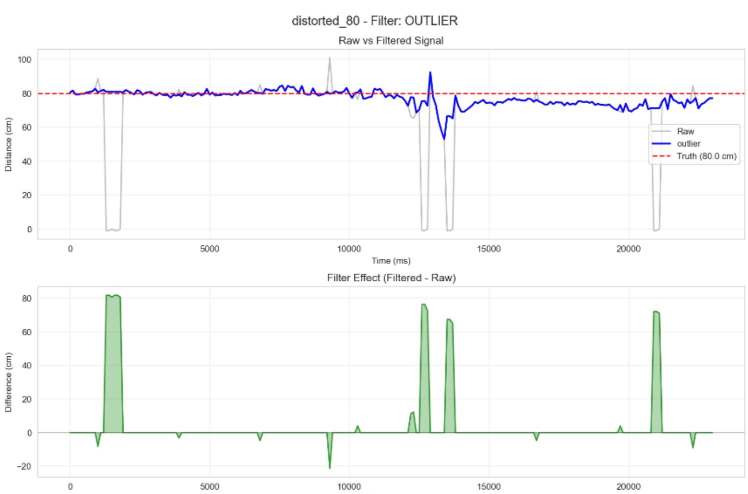


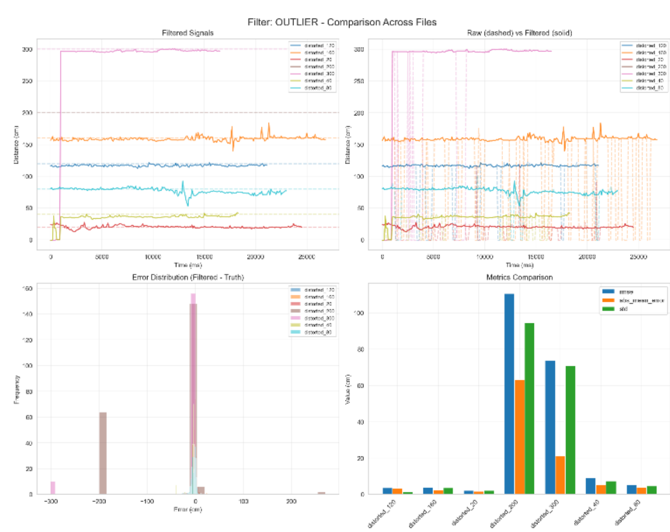
3) Median



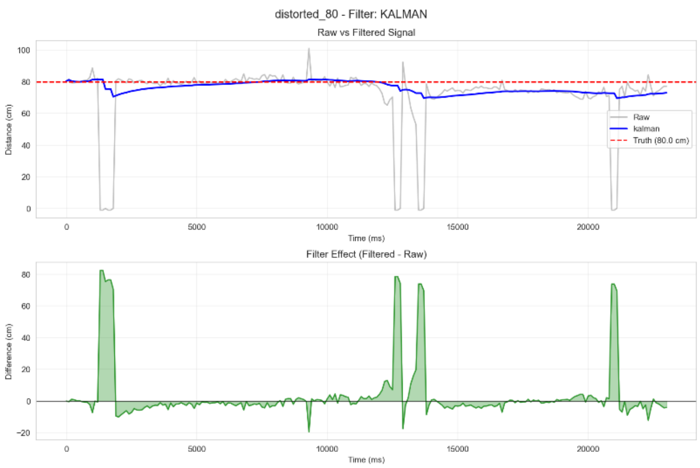


4) Outlier





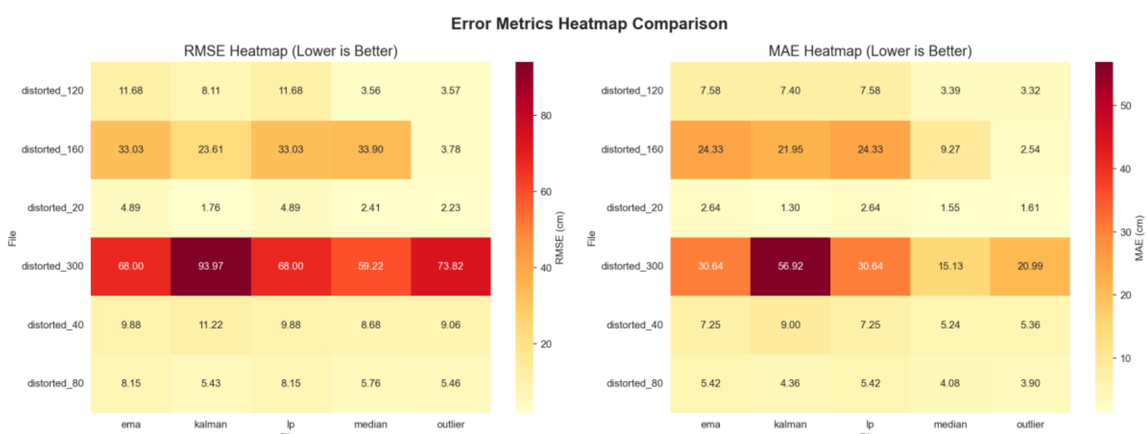
5) Kalman



Filter performance summary:

RMSE (Root Mean Square Error) measures how far the filtered signal is from the real or reference measurement on average. In this project, a lower RMSE means the filter was able to clean the sensor noise while keeping the motion data close to the true movement. It highlights how accurately each filter preserves the actual motion without large errors.

MAE (Mean Absolute Error) shows the average difference between the filtered and real signal in the same units (like cm or deg/s). It tells how much the filter typically deviates from the real value. A smaller MAE means smoother and more reliable readings overall.

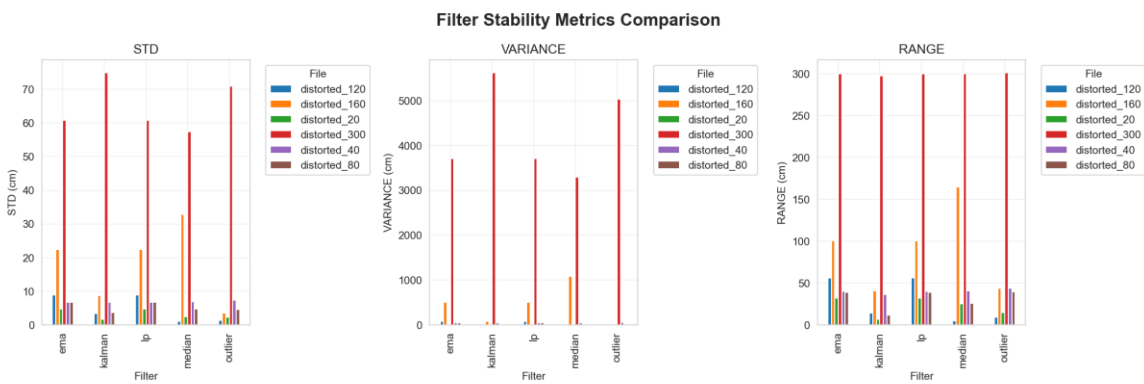


Here we can see a comparison of filter stability across different distorted datasets using three metrics, **Standard Deviation (STD)**, **Variance**, and **Range**.

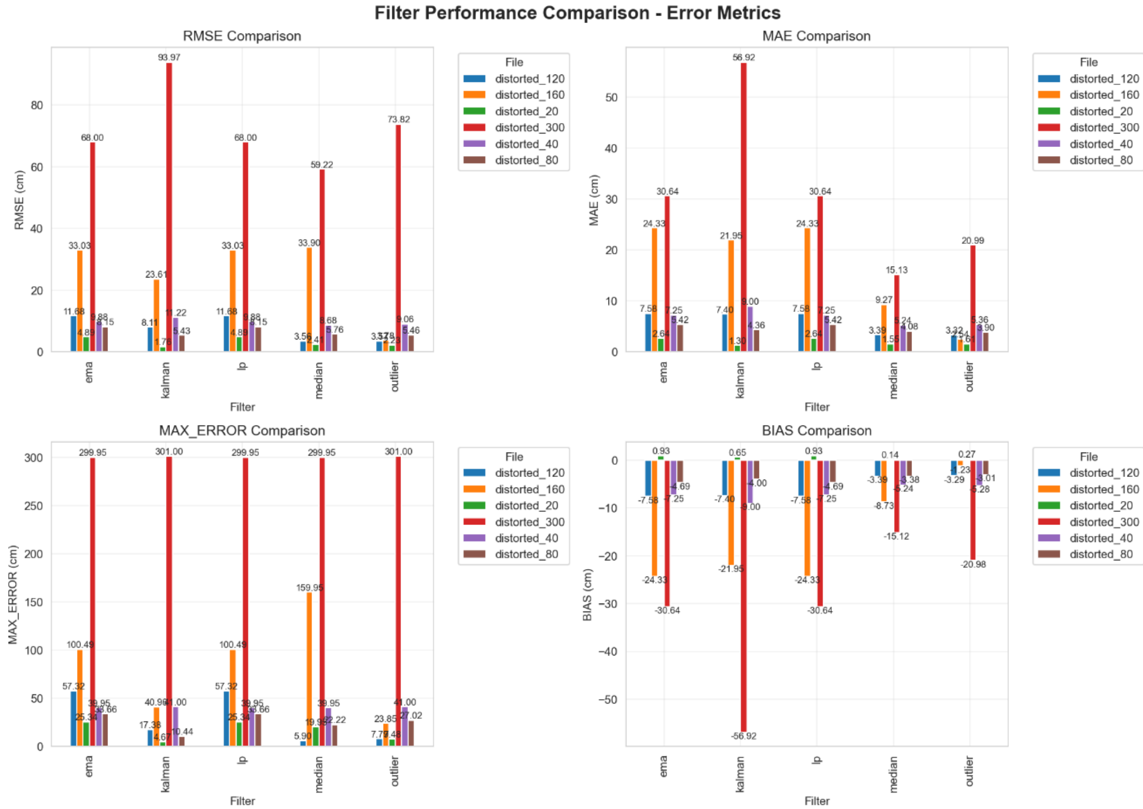
In the **STD** plot (left), we observe how much each filter's output fluctuates around its mean value. Lower bars mean smoother, more stable results. The outlier filter shows the lowest STD, indicating strong noise suppression.

The **Variance** plot (middle) tells the same story but on a squared scale, emphasizing how much the signal's energy varies. Again, outlier and median filters have the least variance, confirming they effectively reduce signal instability.

The **Range** plot (right) shows the span between the minimum and maximum filtered values. A smaller range indicates better suppression of extreme readings. The outlier and median filters clearly produce the most consistent and bounded outputs compared to others.



Error Metrics across all runs:



Filter Ranking according to the Error Metrics above:

Here we can see the overall filter ranking generated from all the error metrics combined. The script first calculates several error measures for each filter (such as RMSE, MAE, bias, and max error) across all datasets. Then, for each metric, filters are ranked the filter with the lowest average error gets rank 1, and higher errors get worse ranks. These ranks are then averaged across all metrics to produce one final “average rank” per filter.