of the self-localization of AUV is presented in (Petrich, Brown, Pentzer, & Sustersic, 2018) which is a robust technique for localization.

2.2: Fusion Algorithms for Underwater Localization

For underwater localization using multi-sensor fusion (MSF), various methods are discussed in (Pan & Wu, 2016) (Tan et al., 2011) (Leonard & Bahr, 2016) (Paull et al., 2013).

2.2.1: Kalman Filter

Kalman Filter (KF) is a stochastic filtering based state estimating algorithm that comprises prediction and estimation stages.

Figure-2.4: Working principle of kalman filter

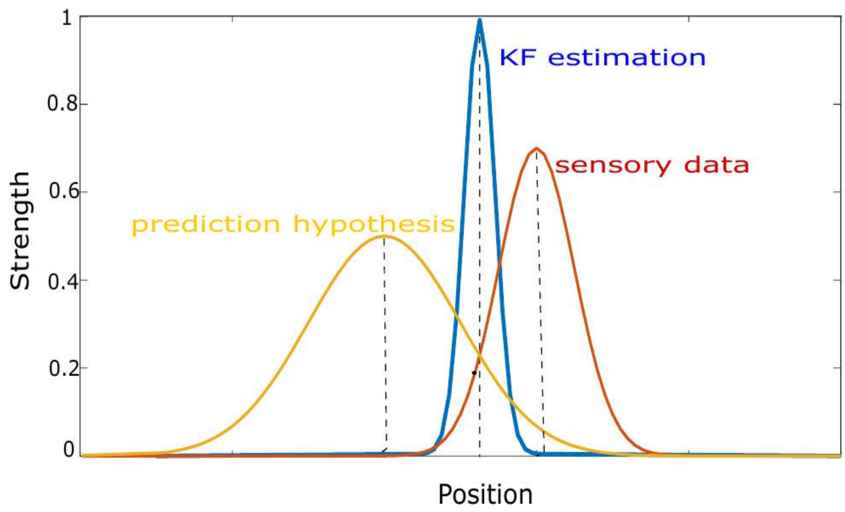
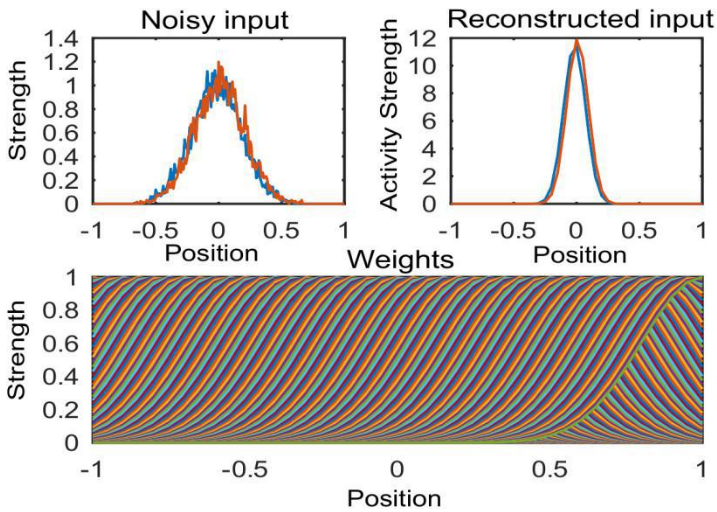


Figure 2.4 is showing the general working of the Kalman filter in which the filter gives the hypothesis of location by combining prediction hypothesis of filter and measurements of sensory data. In (Karras & Kyriakopoulos, 2007), Kalman Filter is used to fusing inertial and visual positioning sensory information for an approximation of location from a fixed earth reference but results are not satisfactory for deep water because of the addition of non-gaussian noise. A chronological linear state estimator performs poorly in presence of non-linear motion equations of the underwater environment. The underwater environment cannot be modeled using linear concepts due to which prediction hypothesis cannot be accurate and there will be no overlapping of the output of the Kalman filter.

PC/BC-DIM can perform computations with probability distributions when input is a probability function. Weights are the elementary components so every specific input can be reconstructed from these weights. PC/BC-DIM can reconstruct noiseless signals. Figure 3.2 shows that how a noisy causes reconstruction through the PC/BC-DIM neural network.

Figure-3.2: PC/BC-DIM noisy input reconstruction



Sensory measurements of an underwater robot are just discrete values of position and they are mixed with an abrupt and non-gaussian type of noises. Weights are considered as a modeled system that is why input stimulus decides the nature of weights of the network. PC/BC-DIM network weights are set intuitively and less noisy reconstructed results are obtained using noisy sensory data. The input of sensors can be encoded in various formats and to encode them into probability density function one-dimensional gaussian equation is presented in equation 3.4 below.

$$x = \exp\left(\frac{(ranges - center)^2}{-2\sigma^2}\right) \tag{3.4}$$

A single dimensional gaussian encoded input is used to set a weight vector in W weight matrix until the training of weights is completed. For a trained network encoded input is processed from the network for some number of iterations. For a medium-sized network, 25 numbers of iterations are enough to produce a stable reconstructed input. Reconstructed input can be decoded back with the help of following equation:

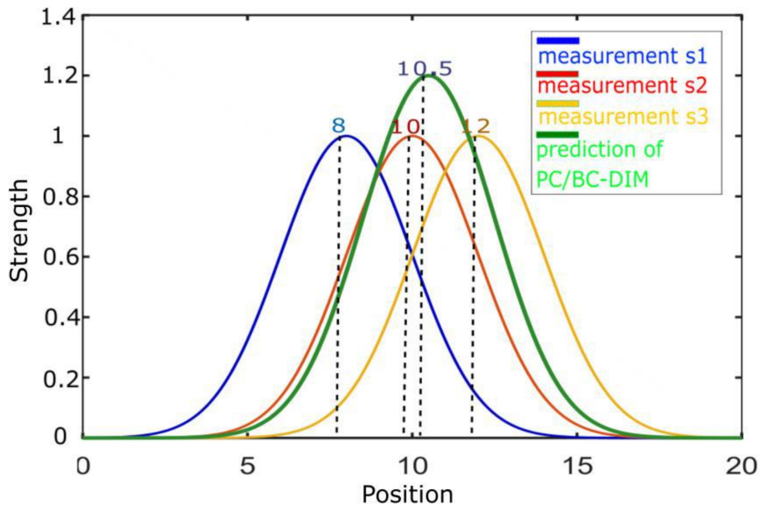
$$\mu = \frac{\sum_i z_i x_i}{\sum_i z_i} \tag{3.5}$$

where μ is the mean value of probability density function (PDF) where zi is the activation of neuron I and si is a receptive field (RF) of neuron i. Similarly, the variance can be calculated using the equation below

$$\sigma^2 = \frac{\sum_i z_i (s_i - \mu)}{\sum_i z_i} \tag{3.6}$$

The filter can combine the likelihood of the prior to calculate the posterior probability. Fusion or integration of multiple causes is another ability of the filter.

Figure-3.3: PC/BC-DIM working principle



3.1.1: Training of Weights

In a neural network, the main part is always training the weights. In PC/BC-DIM input makes the weights so weights are like the dictionary for all possible inputs. One way to train the weights is to store non-noisy different types of inputs in a matrix form but for underwater localization, weights are intuitively set for convenience to observe the performance of the network. Suppose there are 4 types of input and they are named as IMU, DVL, USBL and DGPS. The weights for all sensors are shown in the 3.4 figure. In simulations, all weights are concatenated like all sensors and reconstructed input r is obtained.s

3.1.2: Multisensory Data Fusion

In real-time experiments, there are always multiple sources of information about the same sensory stimulus. This sensory information can be obtained from the same sensory modality or different sensory modalities for the same task. The purpose of multisensory fusion is to determine a single estimate for a different type of information obtained from different sensor modalities. For example, human performance in cue integration is optimal as it can use different sensory information for the same task by considering the reliability of every cue. PC/BC-DIM has the ability to optimally integrate these sensory modalities. Figure 3.5 is demonstrating the integration of two sensory cues that are present at a different location. Their weights, actual input, prediction neuron and reconstructed input is mentioned.

Figure-3.4: Trained weights of multiple sensors

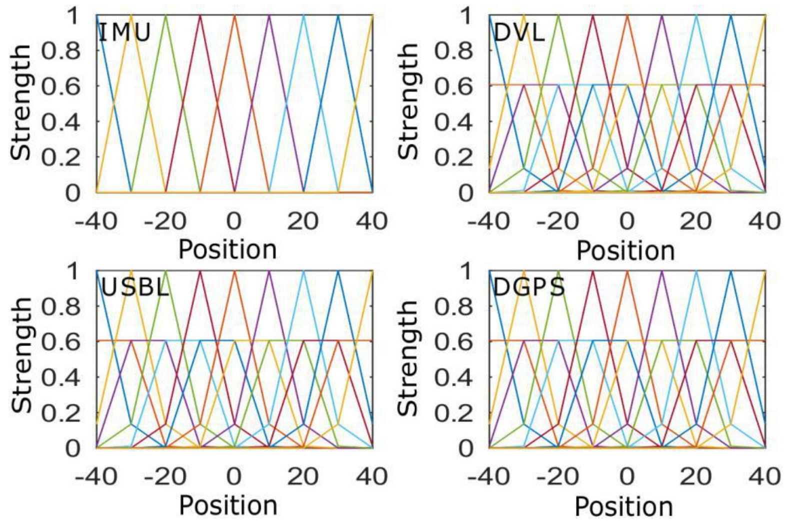


Figure-3.5: Fusion of two types of sensory inputs

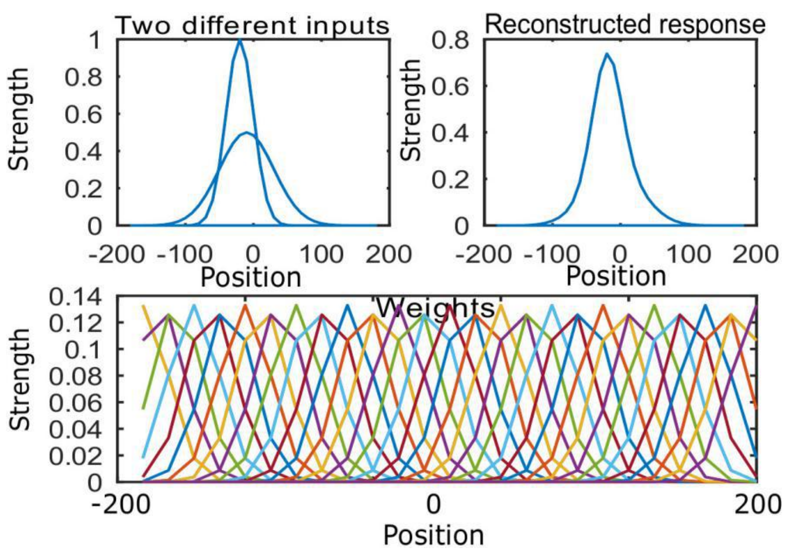
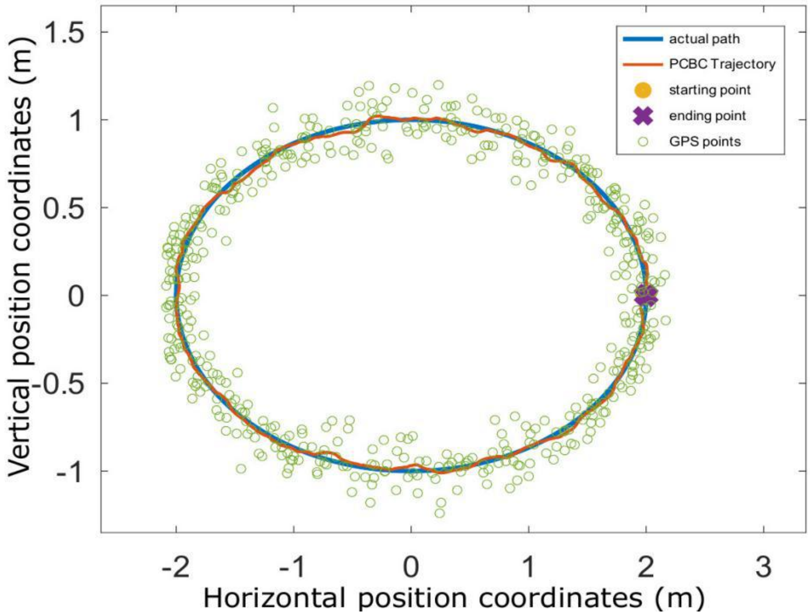
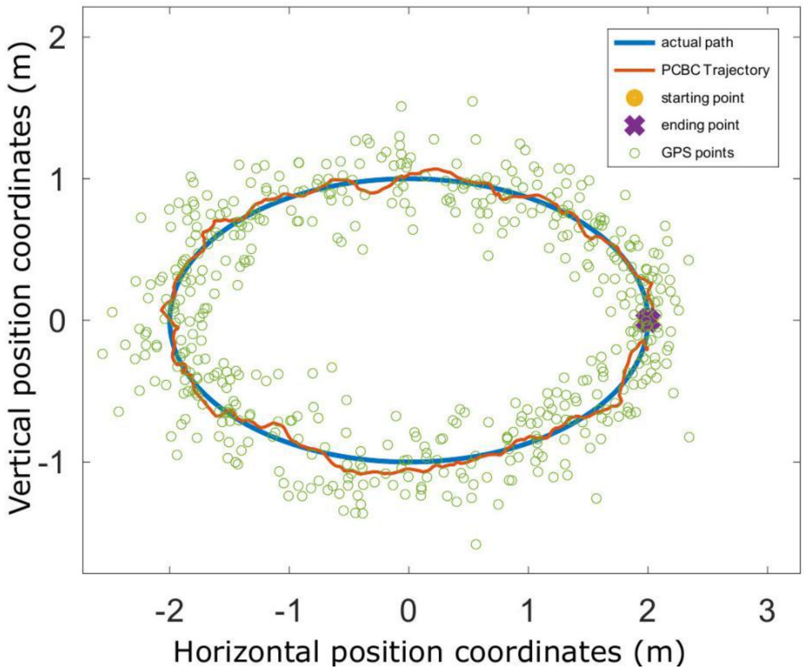


Figure-4.1: PC/BC-DIM outcome for noisy positioning points



With the experiment, it is observed that by increasing the number of positioning points the trajectory of the PC/BC-DIM neural network improves.

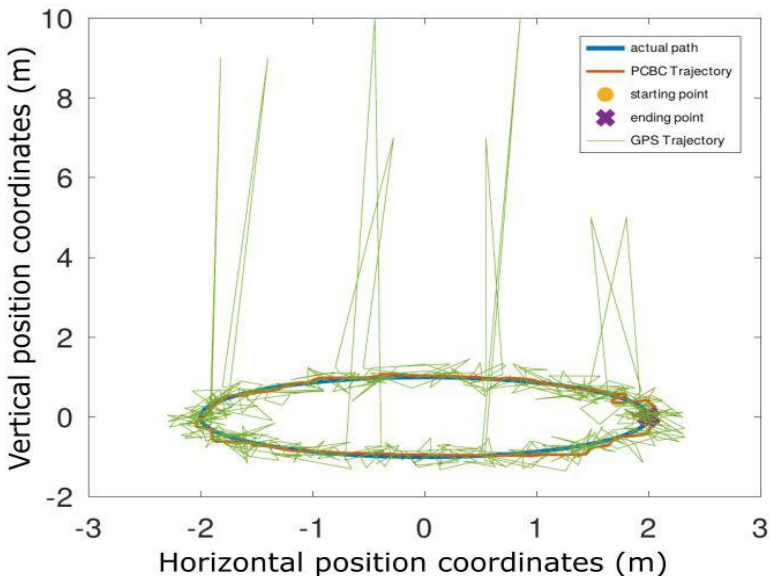
Figure-4.2: PC/BC-DIM outcome for 2 times noisy positioning points



4.1.2: Non-Gaussian or Abrupt Noise Addition:

Abrupt noise is the common problem of the ultrashort baseline sensor and it is observed that this noise is overcome by PC/BC-DIM using a defined approach. To demonstrate this problem random high noise is added at different indexes of noisy GPS signals. Range and centers are set -2 to 2 with the step size of 0.1 while sigma of inertial and global input is set as 1. It is already mentioned that when there is no inertial sensory data available then the last decoded position of the global sensor is used instead of it. Figure 4.3 is showing the abrupt noise at different indexes of the y_gps axis. In the described approach if the change goes beyond the range then it is ignored by the neural network and previous decoded input is used instead of noisy values as GPS is showing in its trajectory.

Figure-4.3: PC/BC-DIM outcome for non-gaussian or abrupt noise



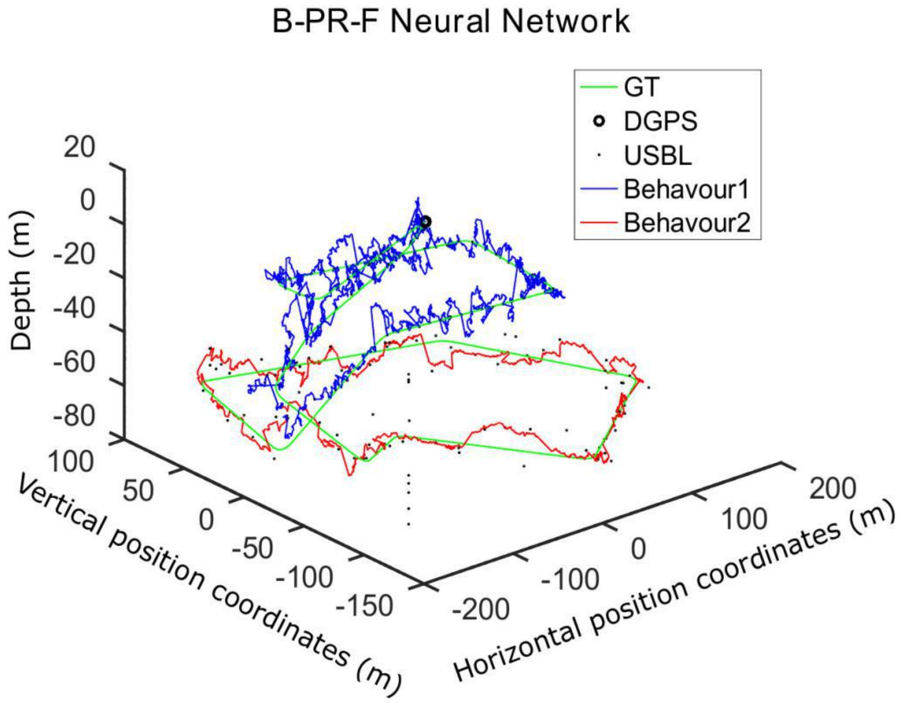
4.2: Simulation Data Comparison with a B-PR-F Neural Network:

Behaviour Prediction Reliability Fusion (B-PR-F) proved that it is better than Kalman filter, Extended Kalman filter and Monte Carlo methods while our proposed PC/BC-DIM neural network produces more accurate results than the B-PR-F neural network. We used the same sensory data and the same simulation environment for testing the PC/BC-DIM neural network. IMU is highly noisy sensory data and using only IMU with USBL and DGPS the proposed PC/BC-DIM neural network produces better results than 4 sensory data of B-PR-F neural network. Similarly, individual DVL with USBL and DGPS produces better approximation. A detailed comparison is presented in sections below.

4.2.1: PC/BC-DIM and B-PR-F Comparison:

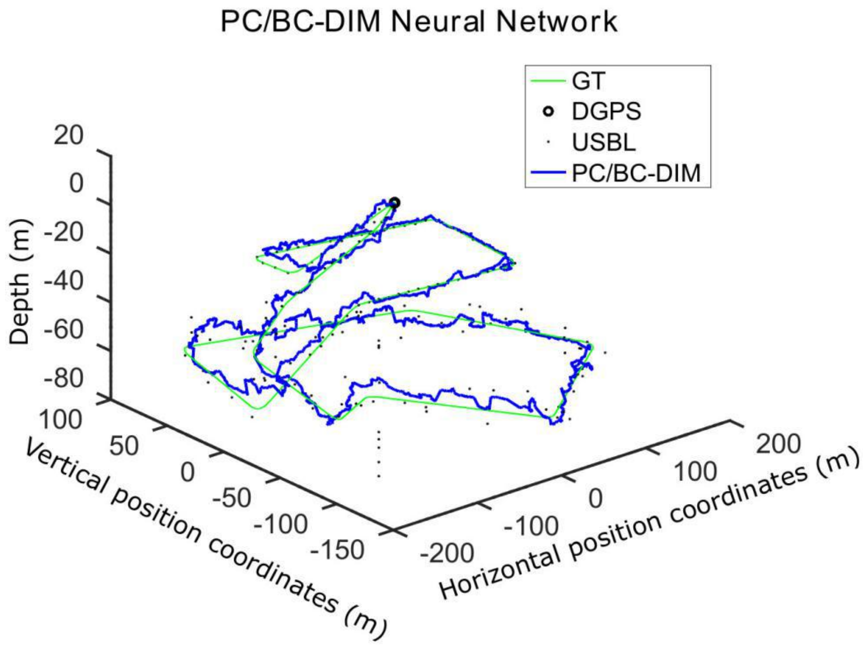
The simulations are conducted in an open-source software named as GNU Octave. Before making a comparison, some information about the data is required. For comparison purposes, it is necessary to use the same sensory data for both filters. Detailed information of sensory data is available in appendix A of (Chame et al., 2018) in which a ground truth (GT) is the actual noise-free trajectory. Sensory data of simulations include IMU, DVL, USBL, DGPS and Altimeter. Altimeter sensor is not used as a proposed filter gives optimal location without any manual switching. IMU data is simulated by twice the differentiating and adding noise to it so integrating the measurements diverge from the original trajectory. Similarly, DVL is one time differentiated and noise is added that is why it is relatively accurate than IMU. From reference GT trajectory with the equal time difference, various points are selected and in those selected points two types of noise are added to achieve USBL like sensory data. Noise 1 is random noise at all points which has a higher scale in deep water and noise 2 is highly non-gaussian abrupt values at some indexes. DGPS works fine above the surface of the water so for a few seconds similar GT values are selected for DGPS sensory data.

Figure-4.4: B-PR-F neural network results



Range of -40 to 40 with the difference of 5 is selected and centers are set using the same configuration. Sigma of DGPS is set equal to step size 5, sigma of USBL is selected 8.5 while both of dead-reckoning method based sensors (DVL and IMU) have 10 of sigma to achieve optimal position estimation. Iterations for PC/BC-DIM are set as 35.

Figure-4.5: PC/BC-DIM neural network results



Above figures 4.4 and 4.5 are demonstrating the visual difference between two filters using the same sensory data. PC/BC-DIM is returning more converging and smooth results. B-PR-F has two behavior of deep and near-surface while PC/BC-DIM has an optimal attitude without any manual switching of any sensor. The green trajectory is showing ground truth data. Statistical comparison gives more clear differences. Table 4.1 shows the differences

Table-4.1: PC/BC-DIM and B-PR-F Simulation Comparison

Filter	Mean Sq Error	std Error	RMS error	Time of execution(s)
B-PR-F	51.078	2.0788	6.8385	15.612
PC/BC-DIM	18.790	1.761	3.9618	20.351

Figure-4.6: Error comparison for both neural networks

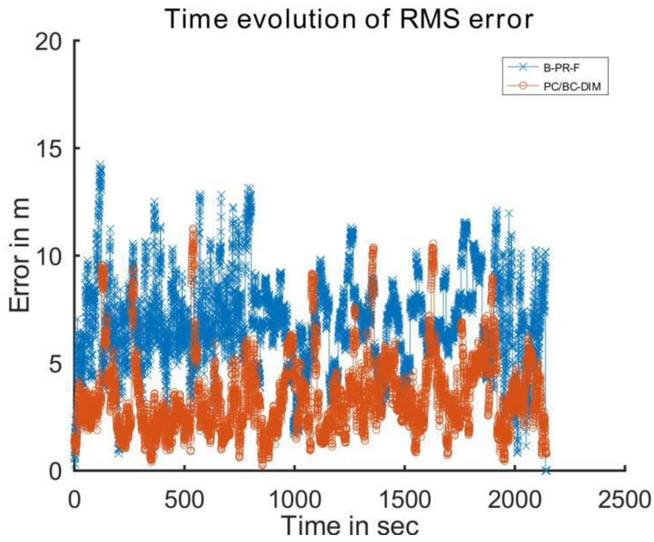
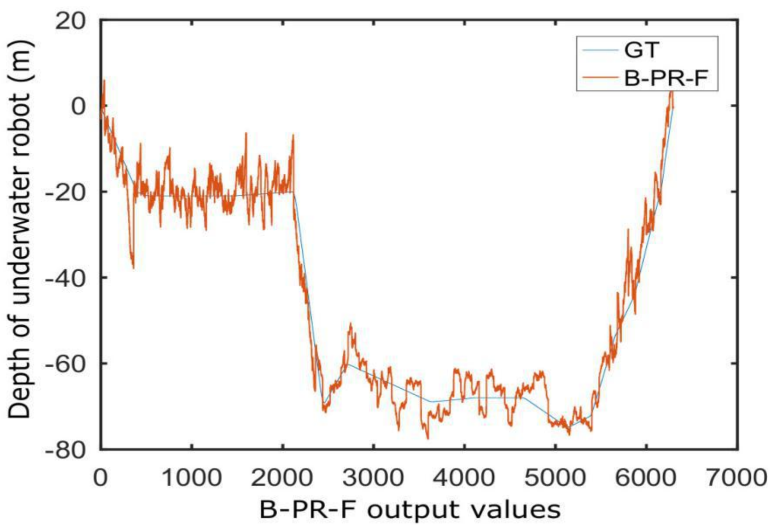


Figure 4.6 is representing the difference between the Root Mean Square (RMS) error of both neural networks. The spikes in PC/BC-DIM can be improved further by increasing the iteration but a minor computation cost will increase..

Figure-4.7: Depth coordinate of B-PR-F neural network



For more clear visualization figure 4.7 and figure 4.8 are representing depth coordinate comparison with the GT depth coordinate.

Figure-4.8: Depth coordinate of PC/BC-DIM neural network

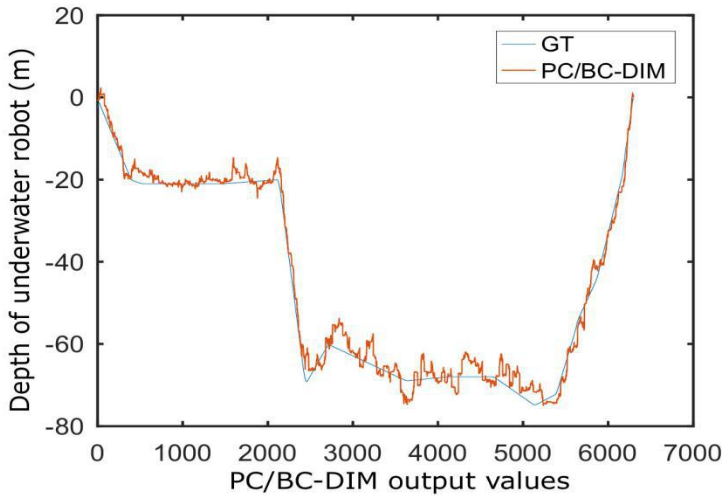
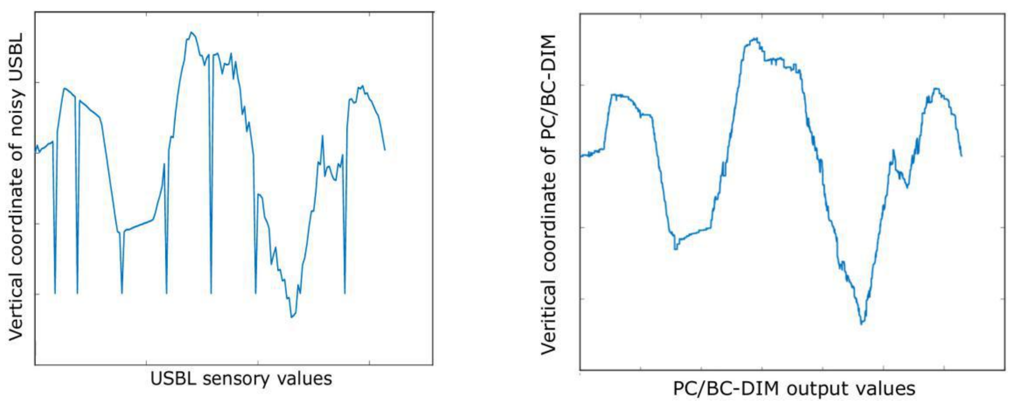


Figure 4.8 is showing less noisy depth coordinate of the PC/BC-DIM neural network as compared to the B-PR-F neural network. Both neural networks are removing noise of the USBL sensor but PC/BC-DIM is not only eliminating non-gaussian noise but it is also producing smooth and less noisy trajectory as in figure 4.9 presented.

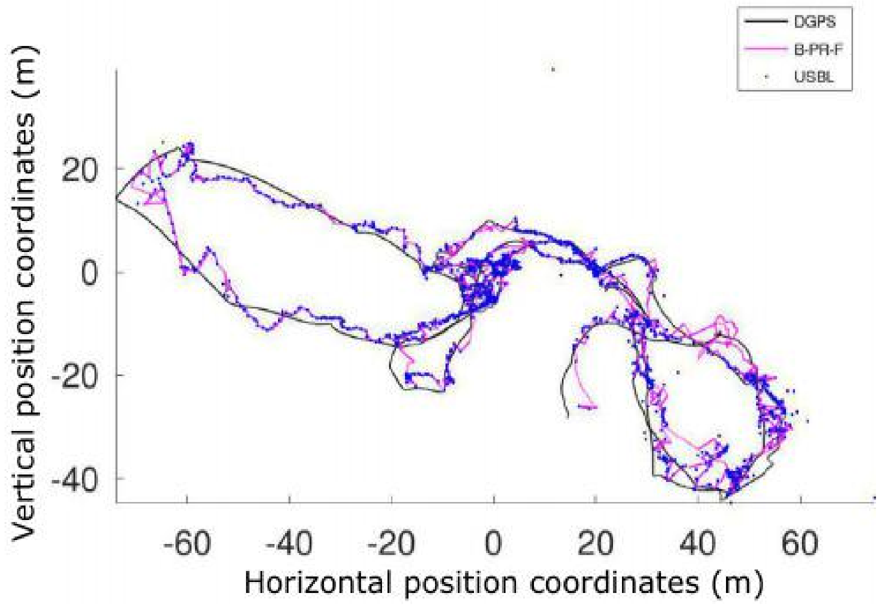
Figure-4.9: Noisy USBL vs PC/BC-DIM results for vertical coordinate



4.2.2: All Sensors B-PR-F and without IMU PC/BC-DIM Comparison:

PC/BC-DIM can give better results without using the IMU sensor and by relying only on the DVL sensor. USBL, DGPS and DVL can produce better results than B-PR-F filter in above

Figure-4.10: B-PR-F position estimation



The plotting of experimental results provides a clear difference between the B-PR-F and PC/BC-DIM neural network estimations.

Figure-4.11: PC/BC-DIM position estimation