**Method:**

**Data Information:**

In this experiment, the vehicle travelled for 61 minutes to cover 693 meters of distance. The method is proposed for estimation of 2D location of vehicle on the surface of water. The location is estimated using low rate absolute positioning data and high rate relative positioning data. In open environment, DGPS is very accurate above the surface of water so it is treated as ground truth with its 3662 positions. For the absolute location estimation 1450 positions of Ultrashort baseline (USBL) are measured and for relative position measurement 19992 grayscale images of 16 bit is taken by SONAR. SONAR images are converted to useful odometry data to find change in angle and to get change in images frames. Equation is presented to measure positional change.





Where theta is current angle or heading of vehicle, x and y are obtained from odometry data. Compass has measured 4357 heading angles. When compass sensory data is not available then partial theta of odometry data is used to correct the angle with the following equation and each angle is wrapped to 2pi using octave mapping package.



**PC/BC-DIM Filter Code:**

**Training of Weights:**

Ranges of -14 to 14 with step size of 1 are selected and centers are also initialized with same values as ranges. Weights for USBL and Odometry data are set intuitively. For USBL weights gaussians are produced at each center having standard deviation of 0.5 and each has the size equal to range. For odometry weights 0.5 of standard deviation of gaussian is set with same procedure. Weights of USBL and odometry are concatenated to get a single weight W which is normalized with its sum. V is normalized transpose of W.

**Inputs for PC/BC-DIM:**

Compass, Odometery and USBL are three main sensors which are used for estimation of location. As reference the USBL position is used and when USBL data not available at particular instant then Odometry data is used but Odometry data is depended on accurate heading angle. Considering this in first if condition compass data is used to get an absolute theta which is very important for accurate positional change. Odometry if statement use this theta to calculate the change in position. When compass data is not available then partial theta from odometry is accumulated to absolute theta of campass to get more accurate angle. In third if condition the USBL data is fetch which is not dependent on any other parameter like odometry is dependant on theta but USBL measure abrupt positions due to delaying in measurement of arriving signal or corrupted signal

**Encoding for Inputs:**

. Odometry and USBL data is first encoded to gaussian format and then concatenated to a single input. This input is processed from PC/BC-DIM neural network and then decoded to give reconstructed position. Odometery is a relative positioning data and measure difference with respect to previous position so every time Odometry data is accumulated with previous decoded position. A customized strategy in adopted to encode USBL position in which change of USBL position is encoded. This Change of USBL position is not with respect to previous USBL position but it is difference of current USBL position and previously decoded position when USBL data was available. If difference of two consecutive USBL is taken then residual error sustains which throughout produces lagging estimation of position in trajectory.



**Decoding of Reconstructed input:**

After processing of PC/BC-DIM reconstructed and fused input is produced which is further decoded using below equation to get single point location. 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.



**How Data is stored to make trajectory:**

As described in input information that odometry is accumulated and for USBL change is used. But how the trajectory is made to understand it the complete for loop has to understand. There are 3 main cases.

1. USBL available:

If USBL value is present then it is fused with accumulated odometry data. Current position is stored in a “TempSavePos” variable and last decoded usbl position is updated. Decoded position is set to zero.

1. USBL not present:

If USBL value is not presented then encoded values are initialized with zeros and concatenated with accumulated odometry data. Odometry is accomulated with decoded position.

**TABLE for Understanding:** (idea values)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Actual position | USBL data   1. reference | ODOM data   1. reference | TempSavePos = T..S..P +decod | Decoded Pos | Usbl Change | Accomulated  ODOM | GlobalTraj  T..S..P+Decoded |
| **1** | **1** | **1** | **1** | **1** | **1-0=1** | **0+1=1** | **1** |
| 2 | // | 1 | 1 | 2 | // | 0+1=1 | 1+1=2 |
| 3 | // | 1 | 1 | 3 | // | 1+1=2 | 1+2=3 |
| 4 | // | 1 | 1 | 4 | // | 2+1=3 | 1+3=4 |
| **5** | **5** | **1** | **1+4=5** | **4** | **5-1=4** | **3+1=4** | **1+4=5** |
| 6 | // | 1 | 5 | 1 | // | 0+1=1 | 5+1=6 |
| 7 | // | 1 | 5 | 2 | // | 1+1=2 | 5+2=7 |
| 8 | // | 1 | 5 | 3 | // | 2+1=3 | 5+3=8 |
| **9** | **9** | **1** | **5+4=9** | **4** | **9-5=4** | **3+1=4** | **5+4=9** |
| 10 | // | 1 | 9 | 1 | // | 0+1=1 | 9+1=10 |

Exactly this approach is implemented in code.

.