

# Neural Network Time Series Prediction of Ship Motion

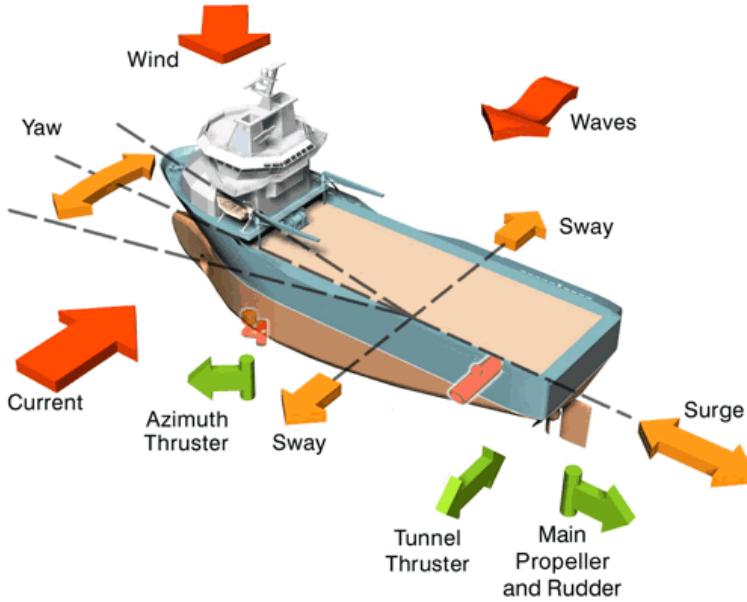
Bikram Kawan

May 2016

NTNU i Aalesund

Supervisors: Guoyuan Li & Hao Wang

# Motivations



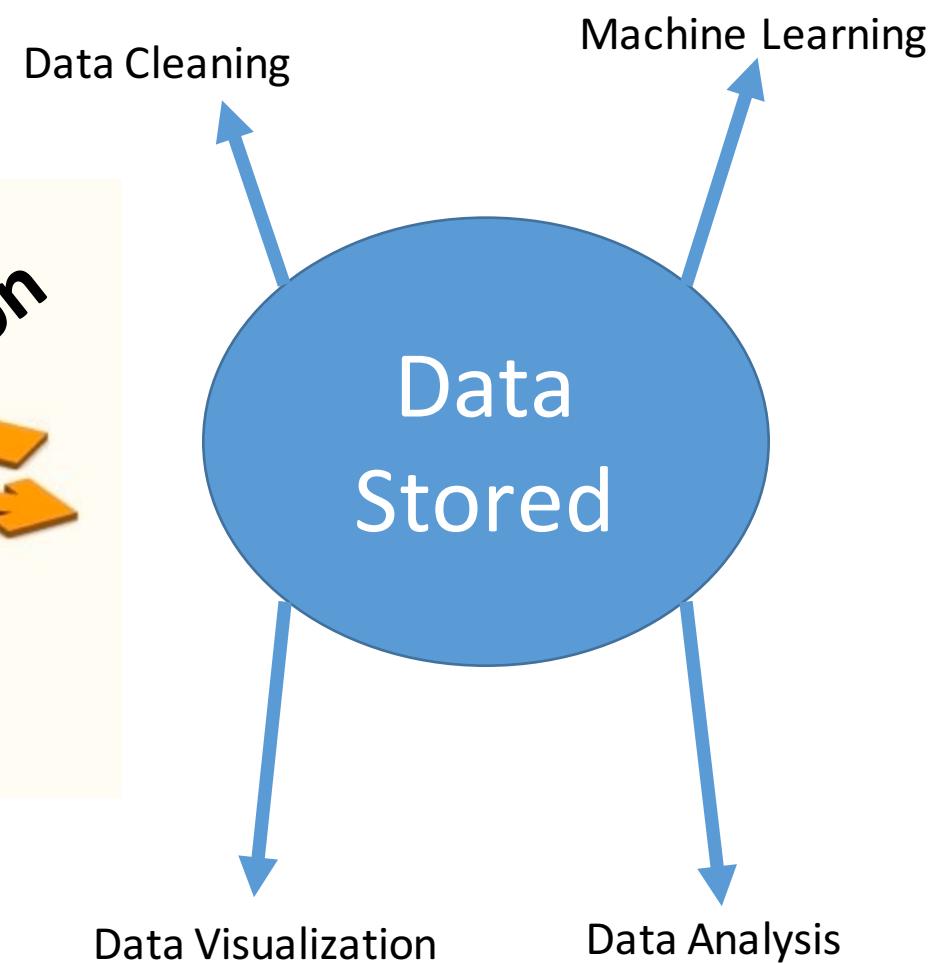
[1]

Dynamics of Ship



[2]

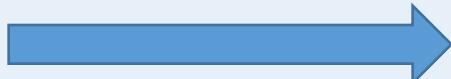
Decision Support



Monetize Data

# GOAL

Time Series Data



Ship Motion Prediction



Multi-Step Prediction

Read real-time  
sensor data



Online Prediction

Raw Data



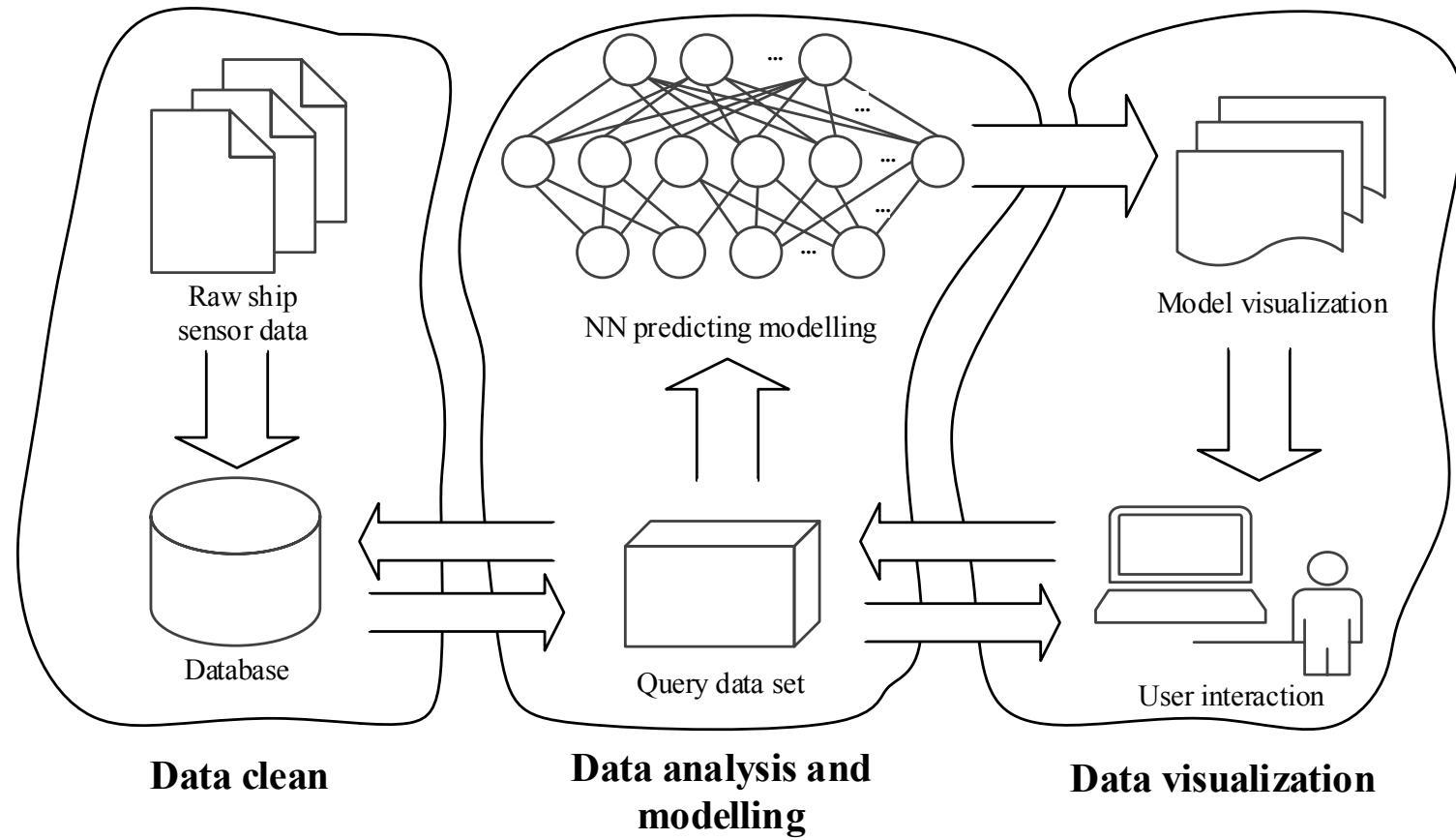
Cleaned Data

Main →

Contributions

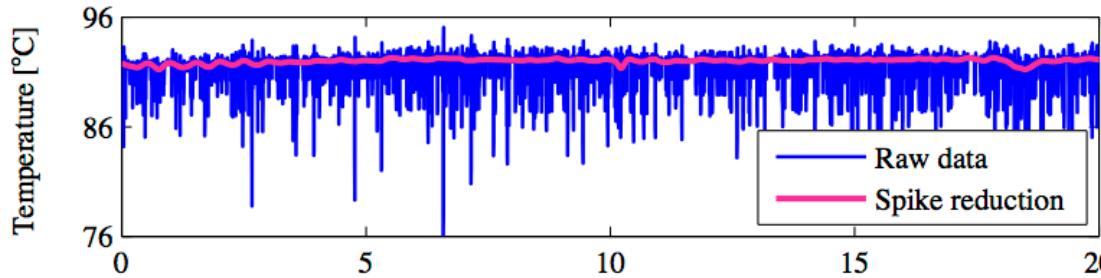
Investigate and Compare  
Different Learning Process

# System Structure

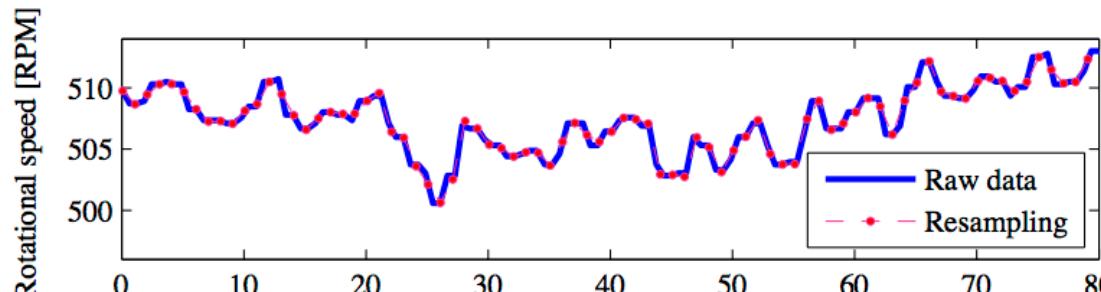


# Data Cleaning

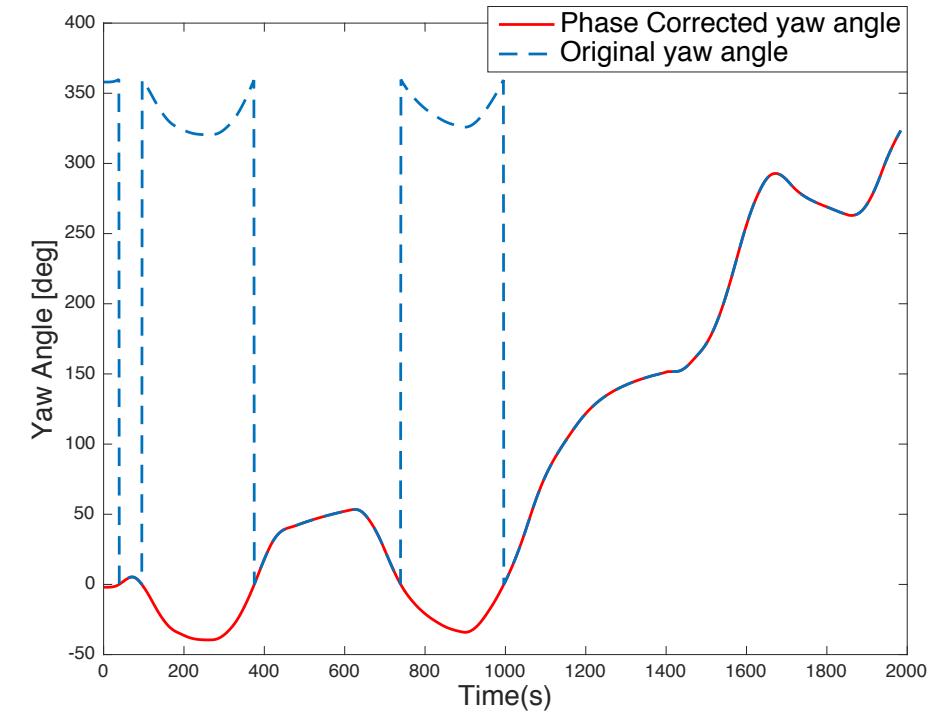
## Noise Removing



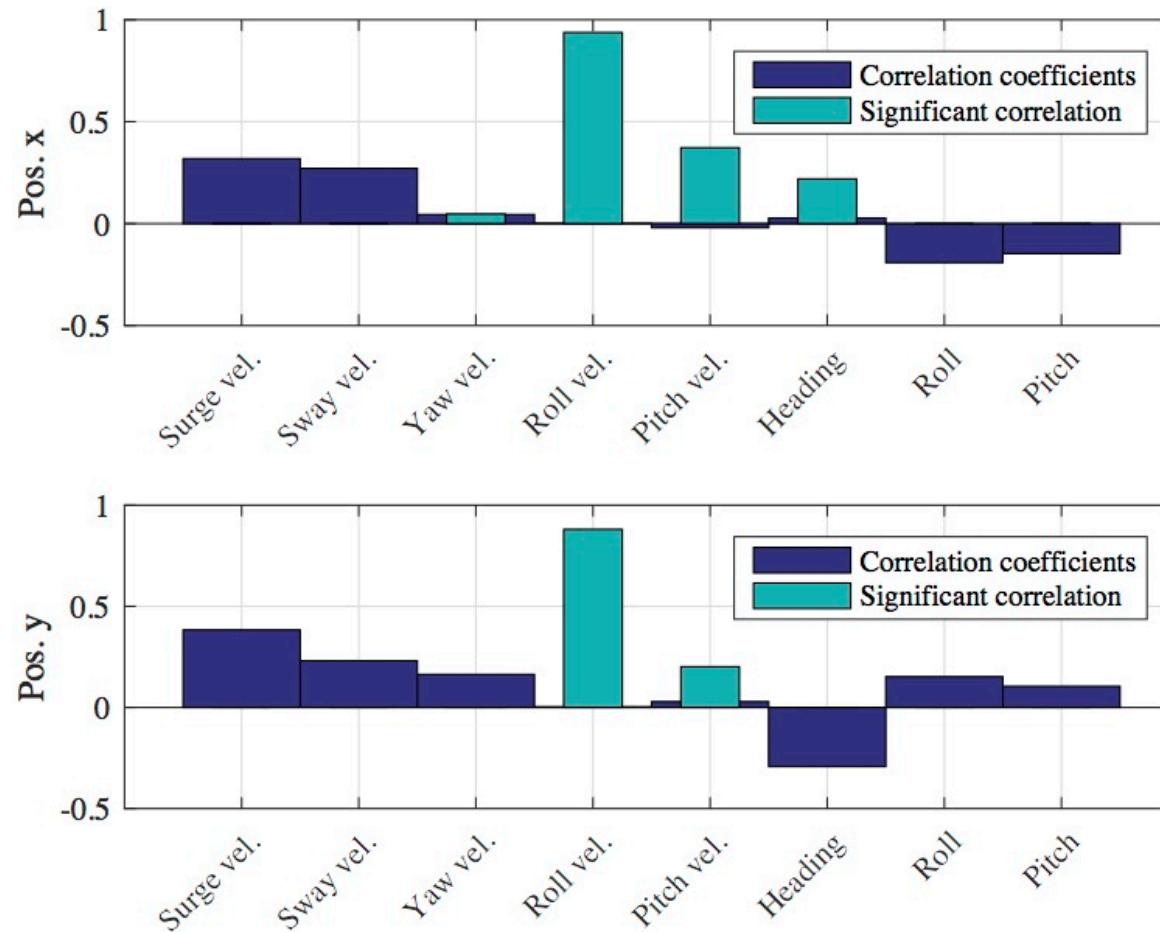
## Resampling



## Phase Correction



# Data Analysis



## Correlation Coefficients (R)



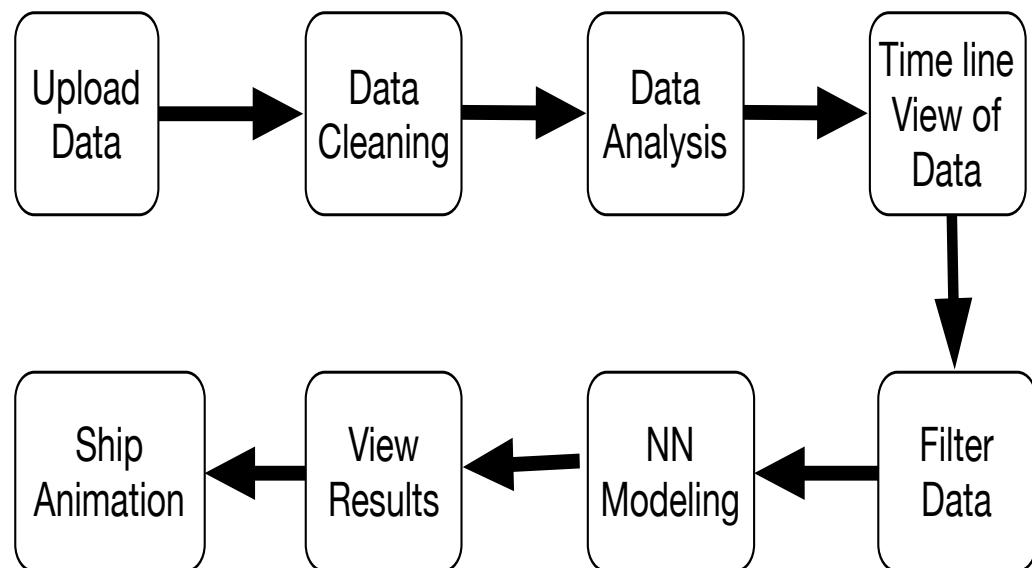
- Strength of association between two variables
- Range between -1 to +1

## Significant Correlation (P)

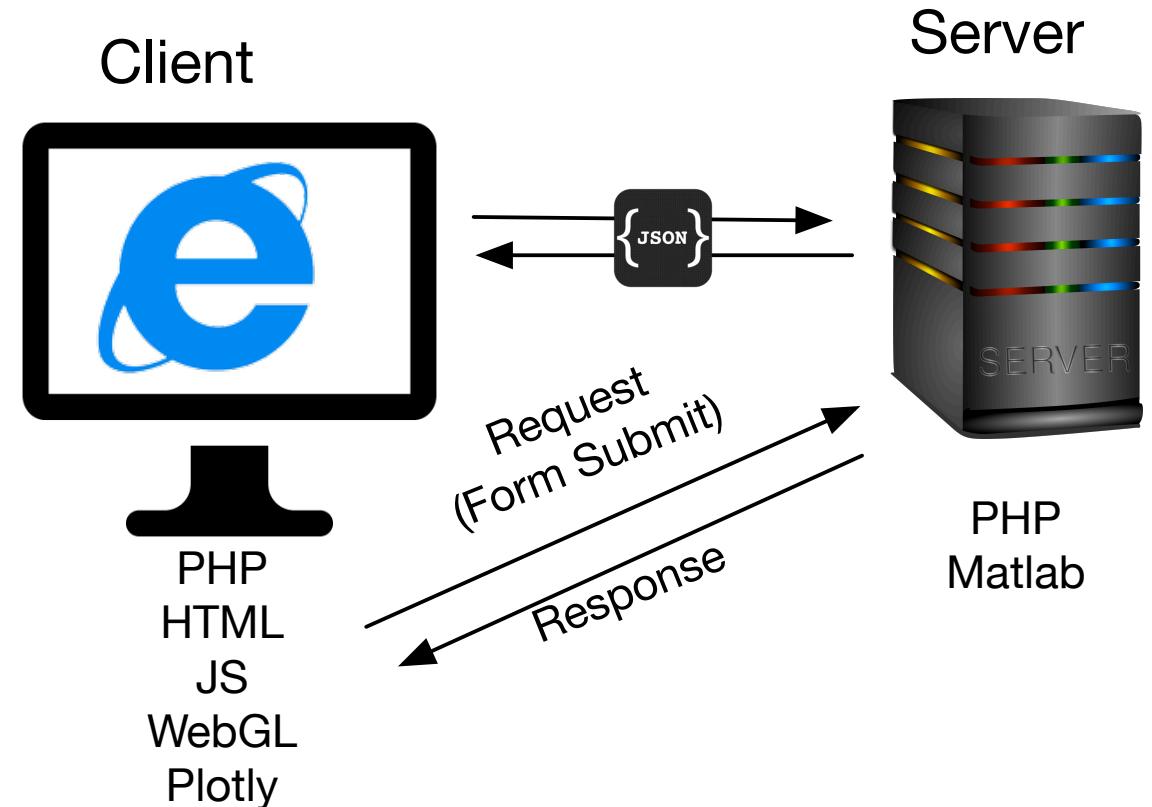


- Significance of Results
- Hypothesis Testing
- $P \leq 0.05$  – Strong Evidence
- $P > 0.05$  – Weak Evidence

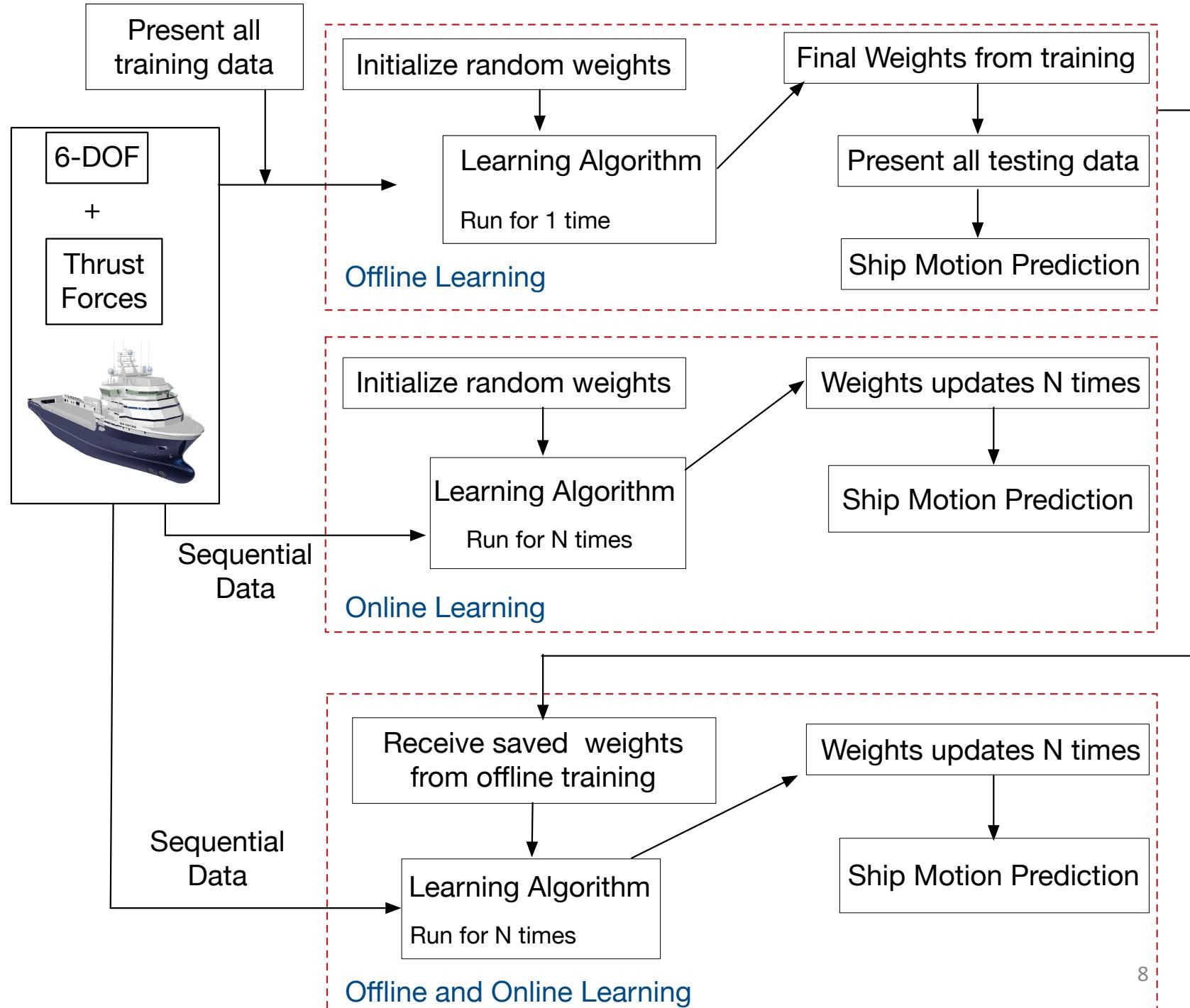
# Data Visualization



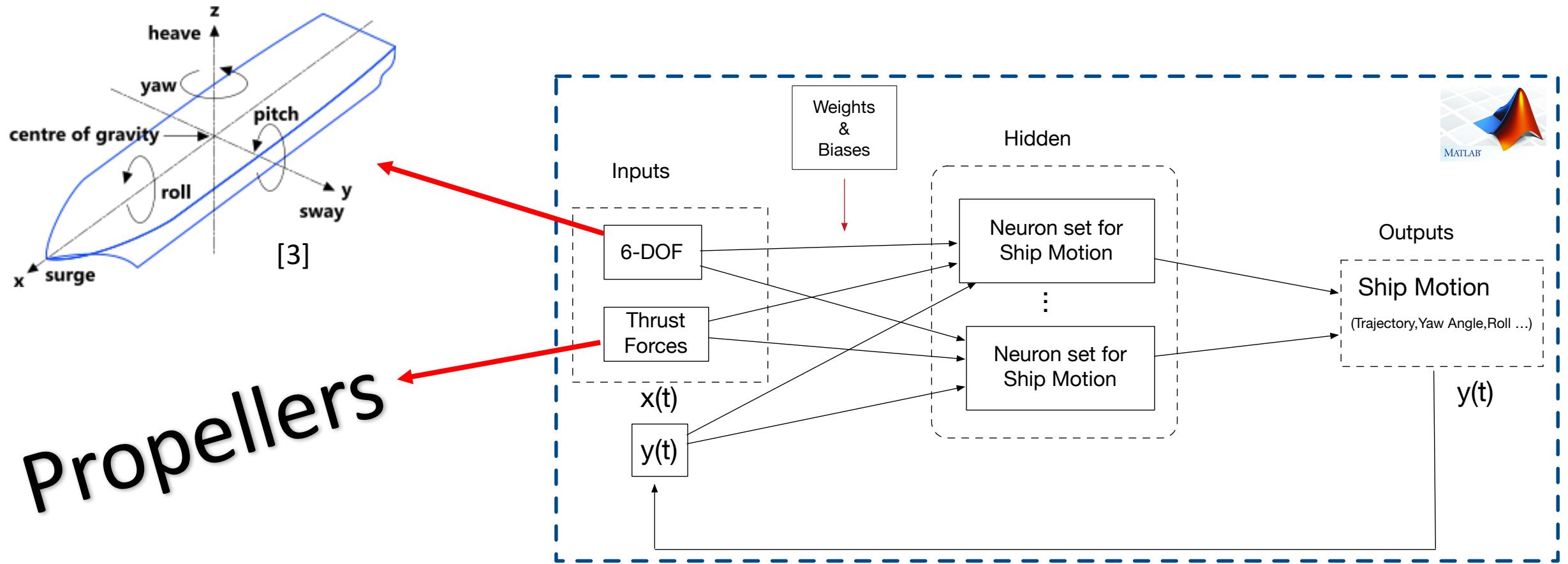
Data Visualization Procedure



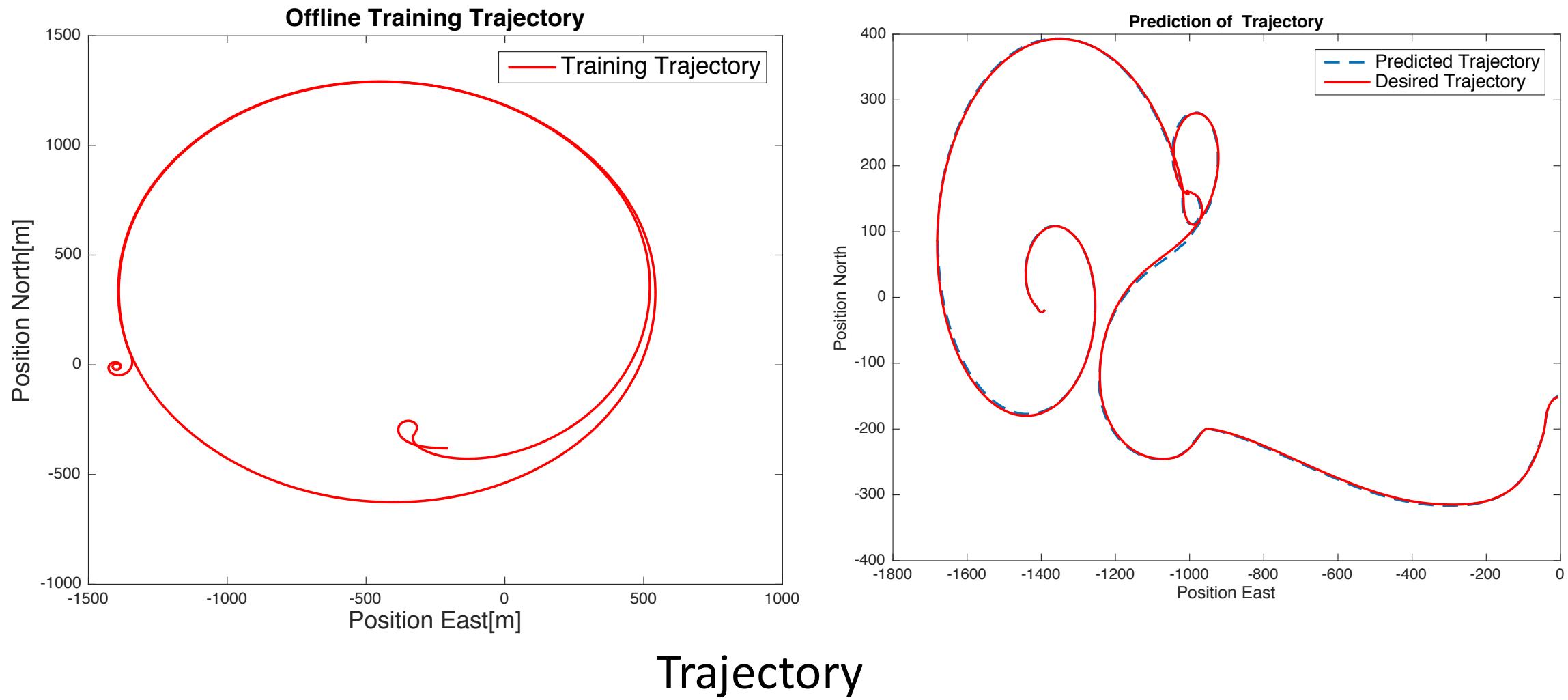
# Ship Motion Prediction



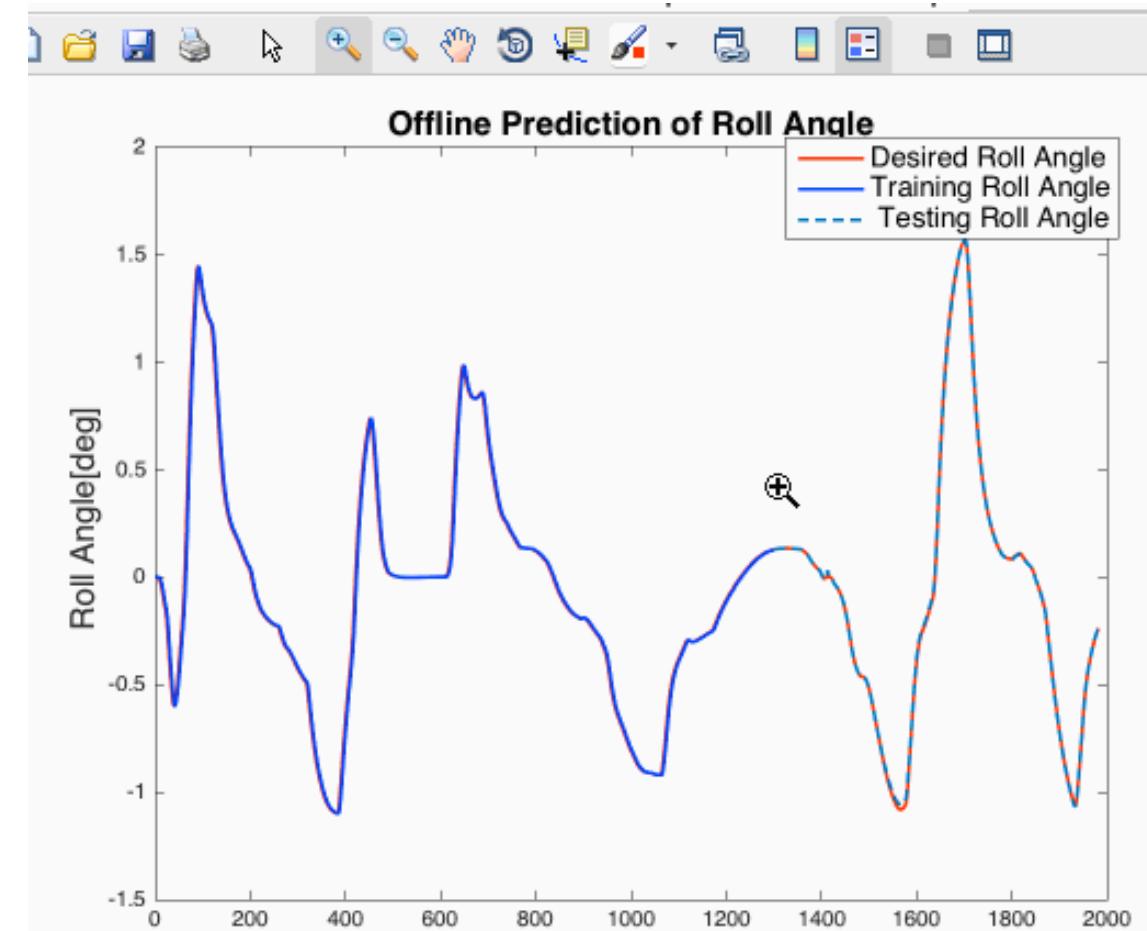
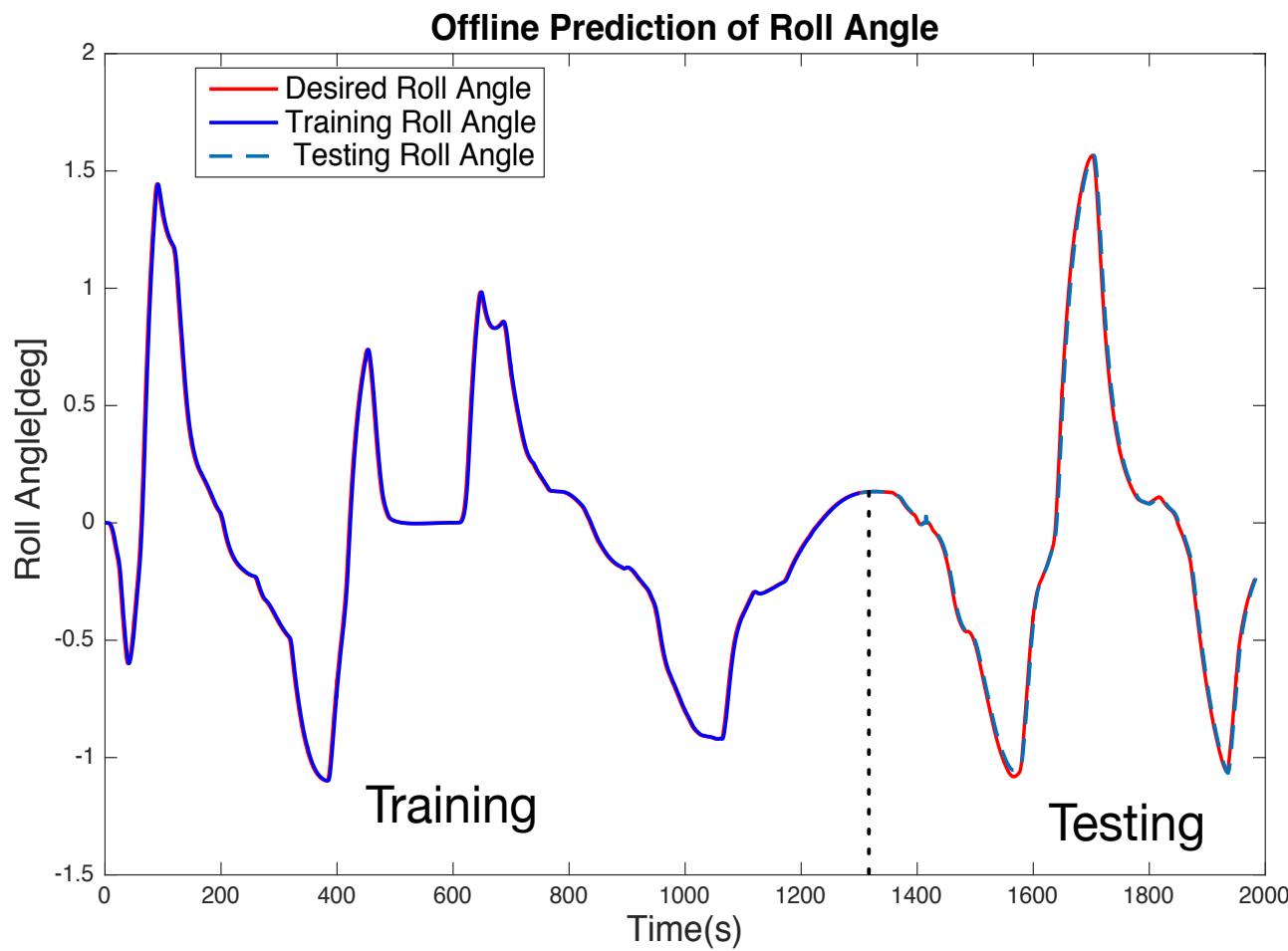
# Network diagram



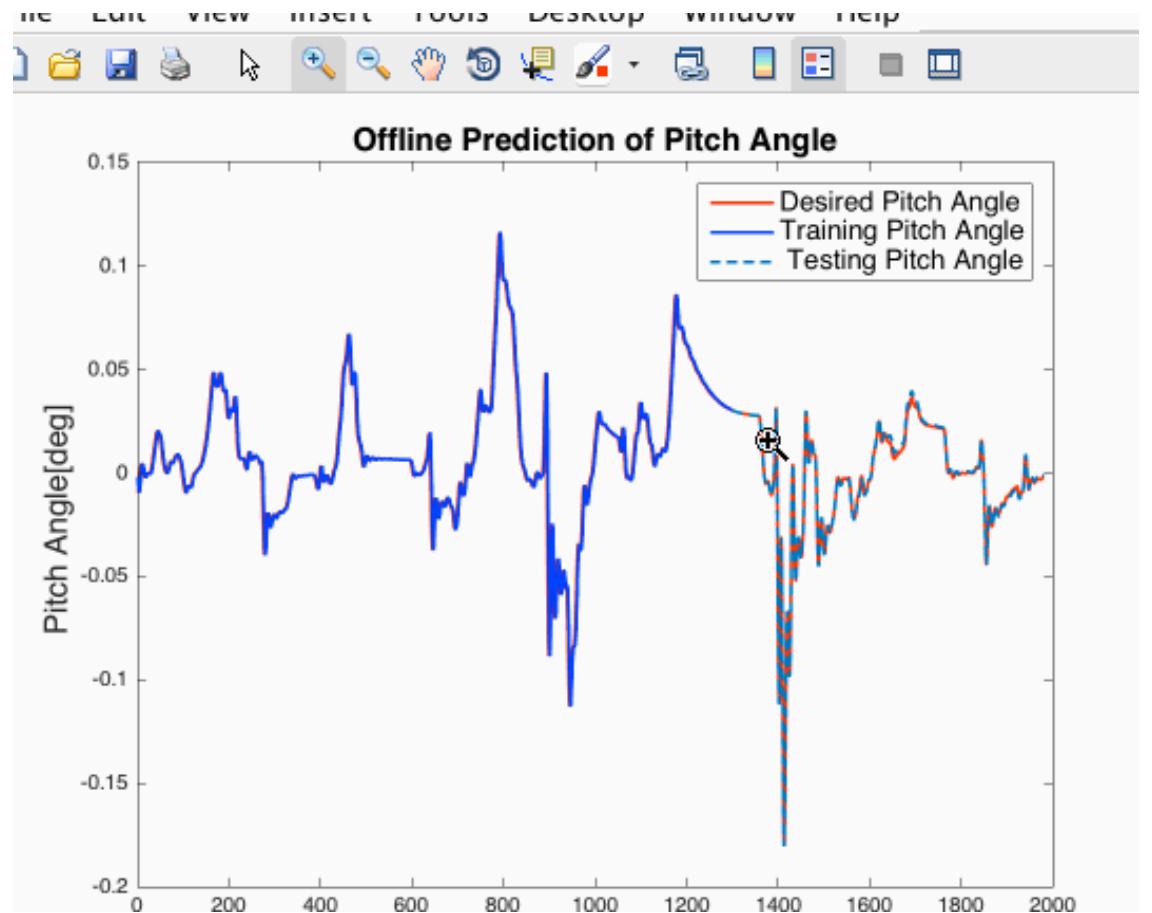
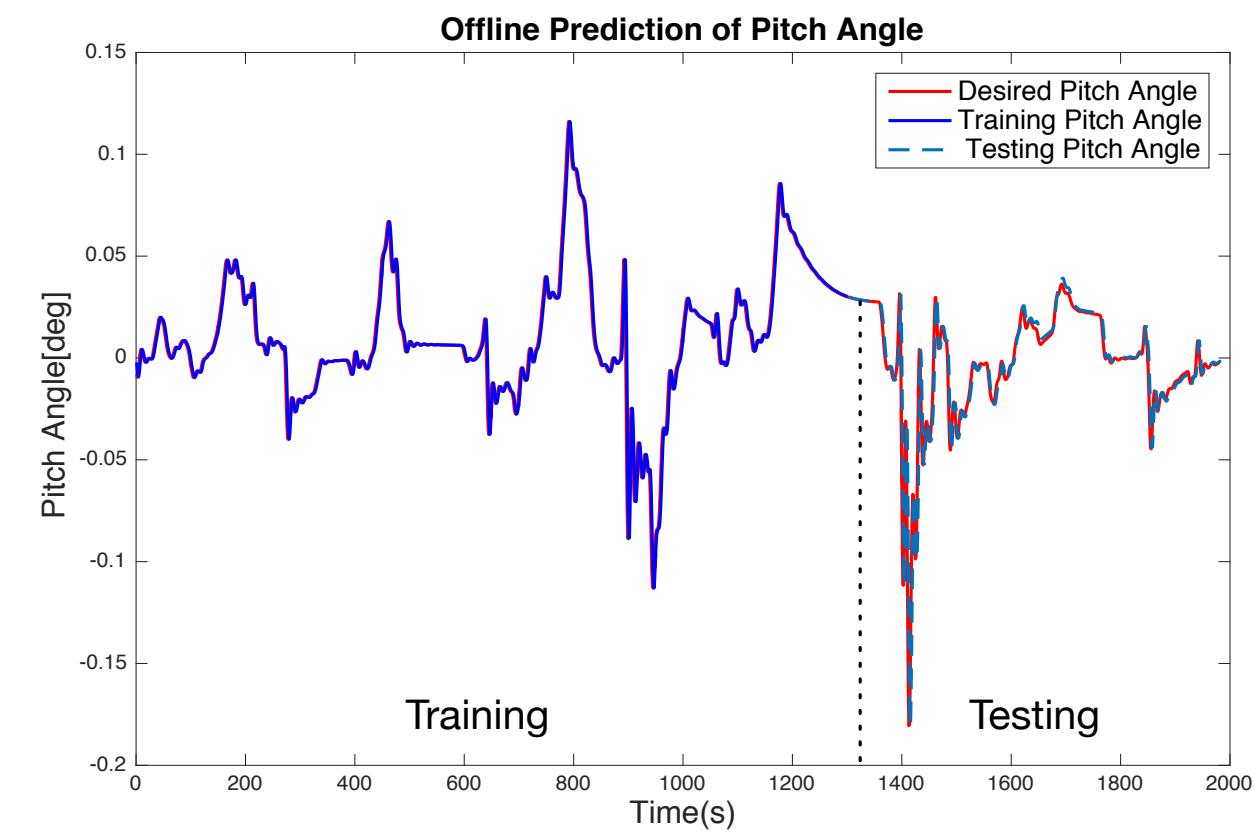
# Offline Prediction Results



# Offline Prediction Results

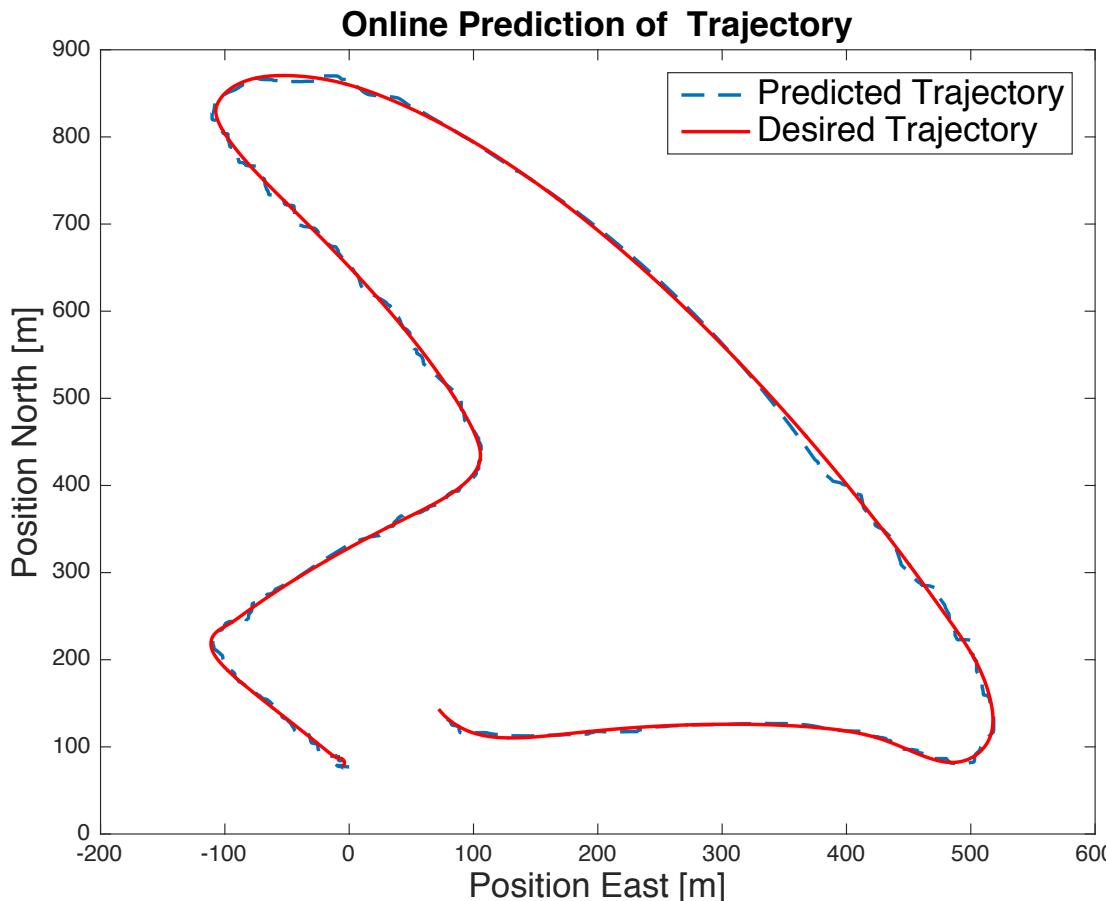


# Offline Prediction Results

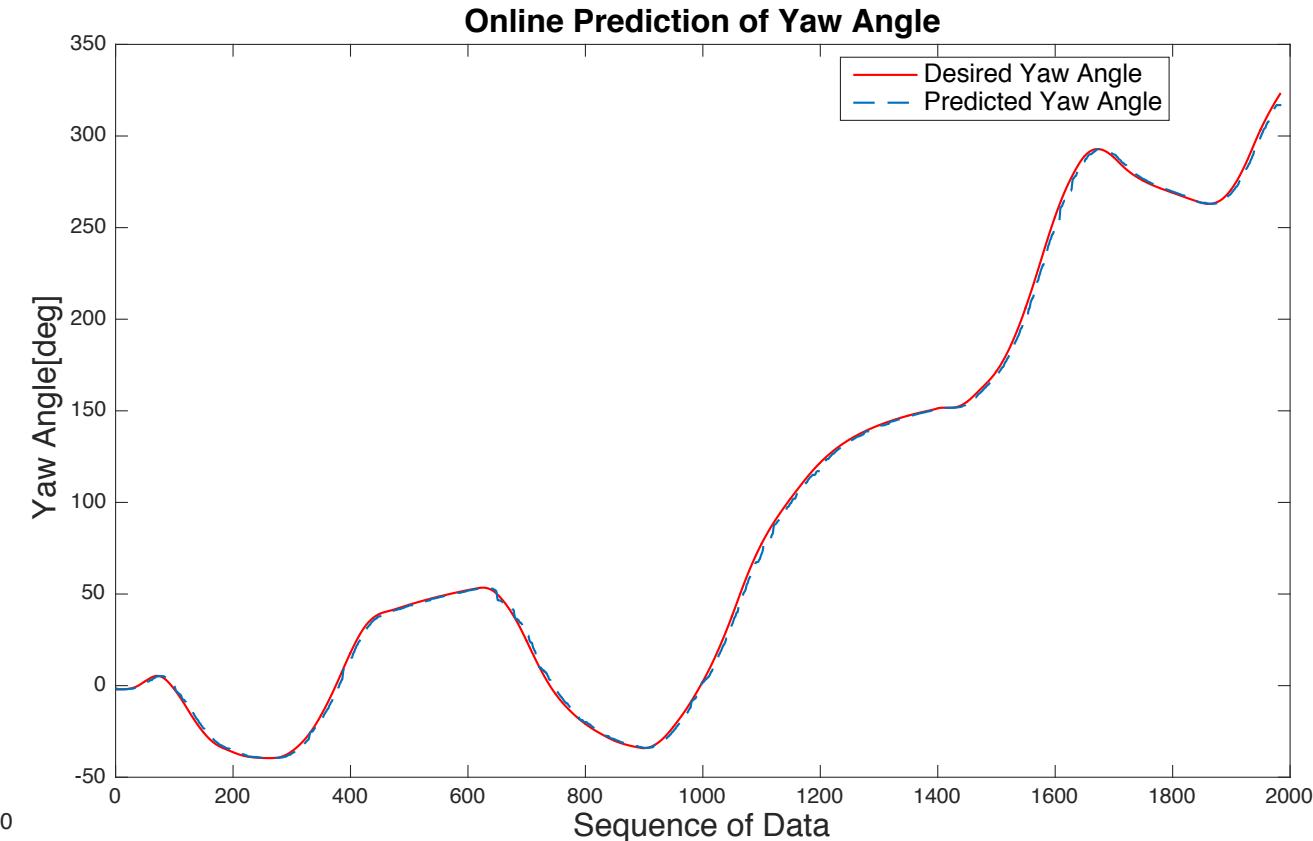


Pitch Angle

# Online Prediction Results

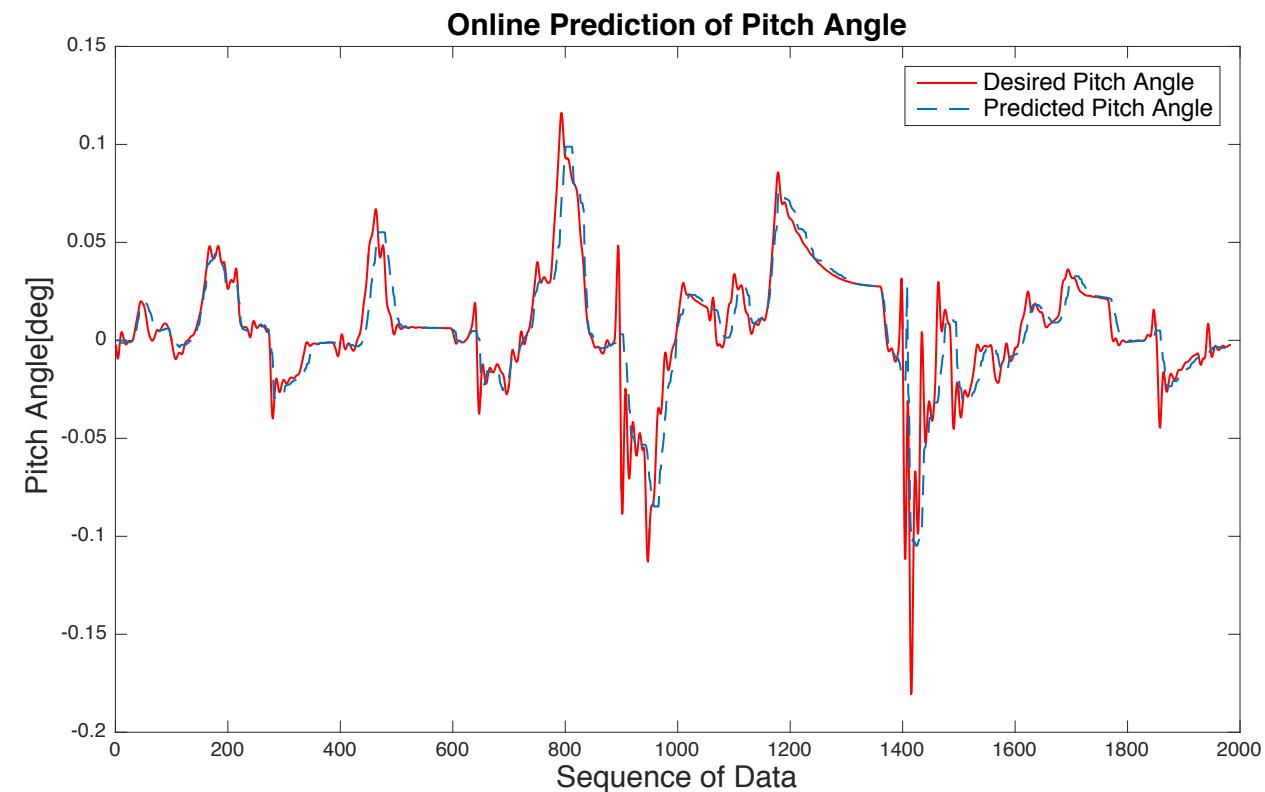


Trajectory

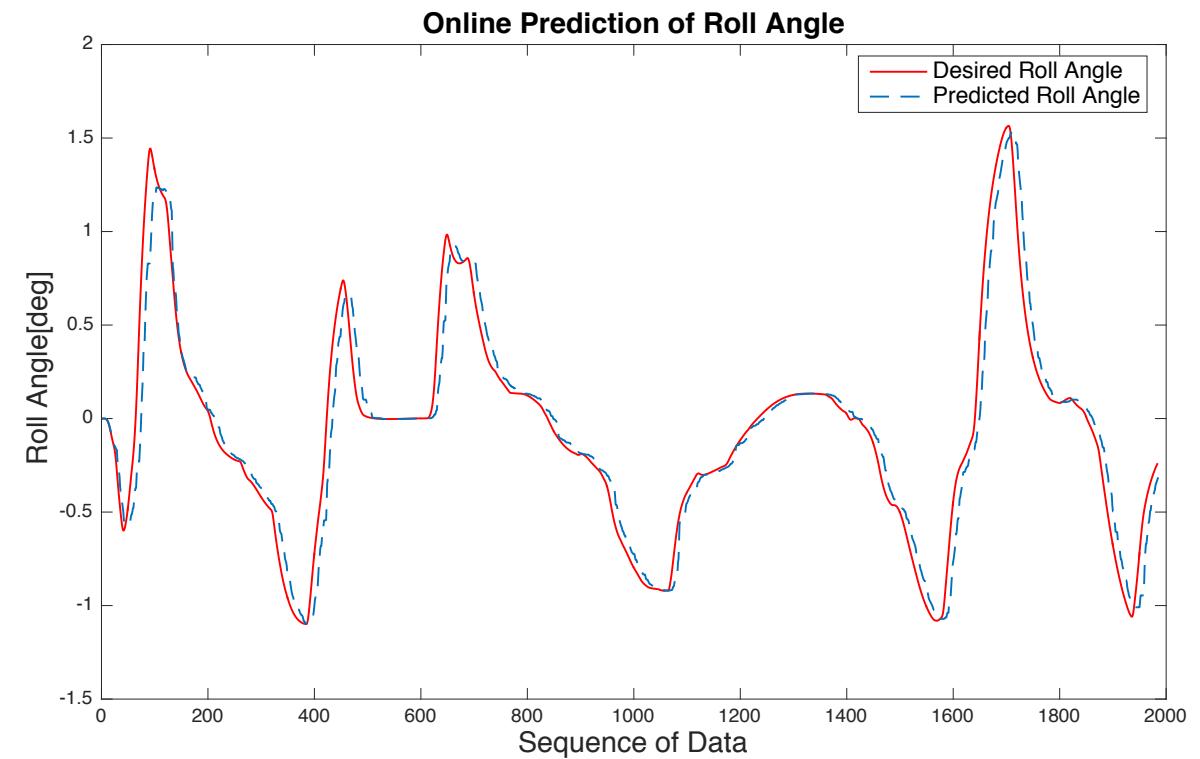


Yaw Angle

# Online Prediction Results

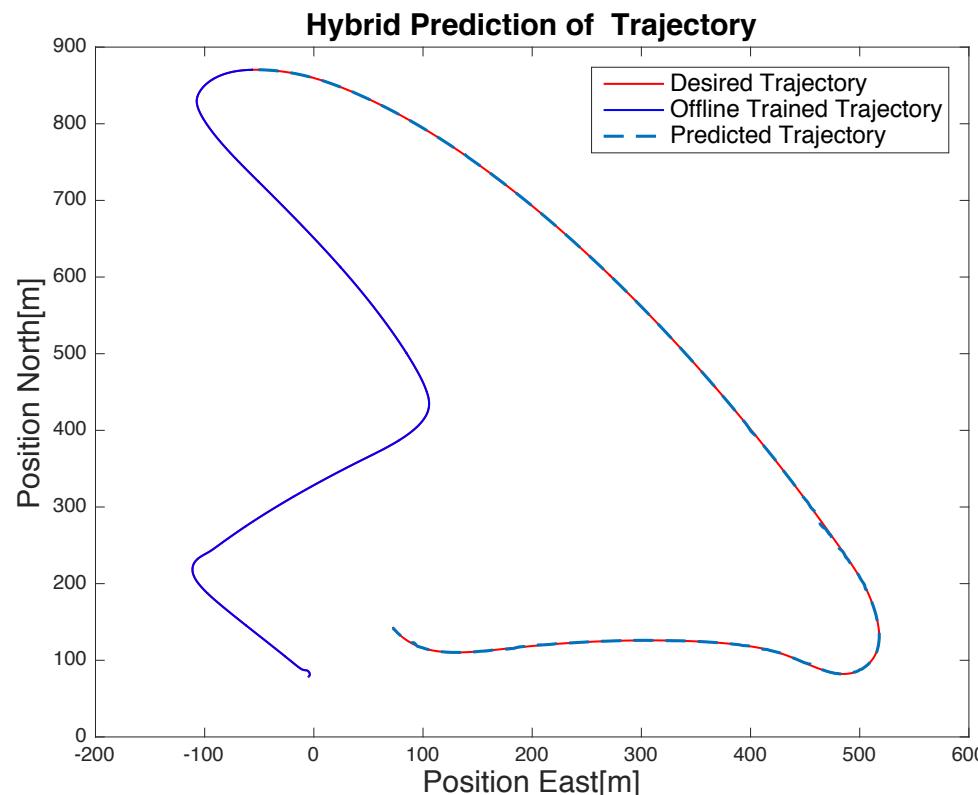


Pitch Angle

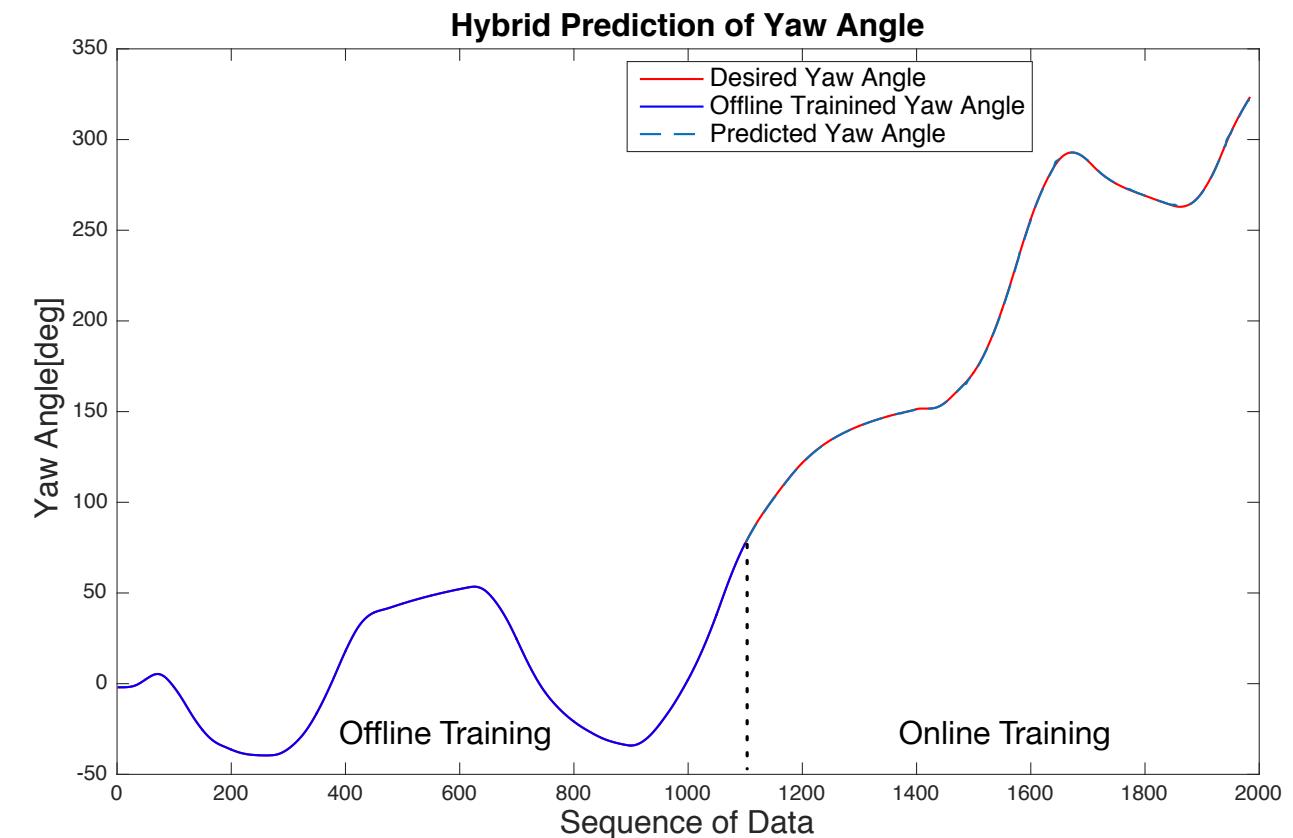


Roll Angle

# Hybrid Prediction Results

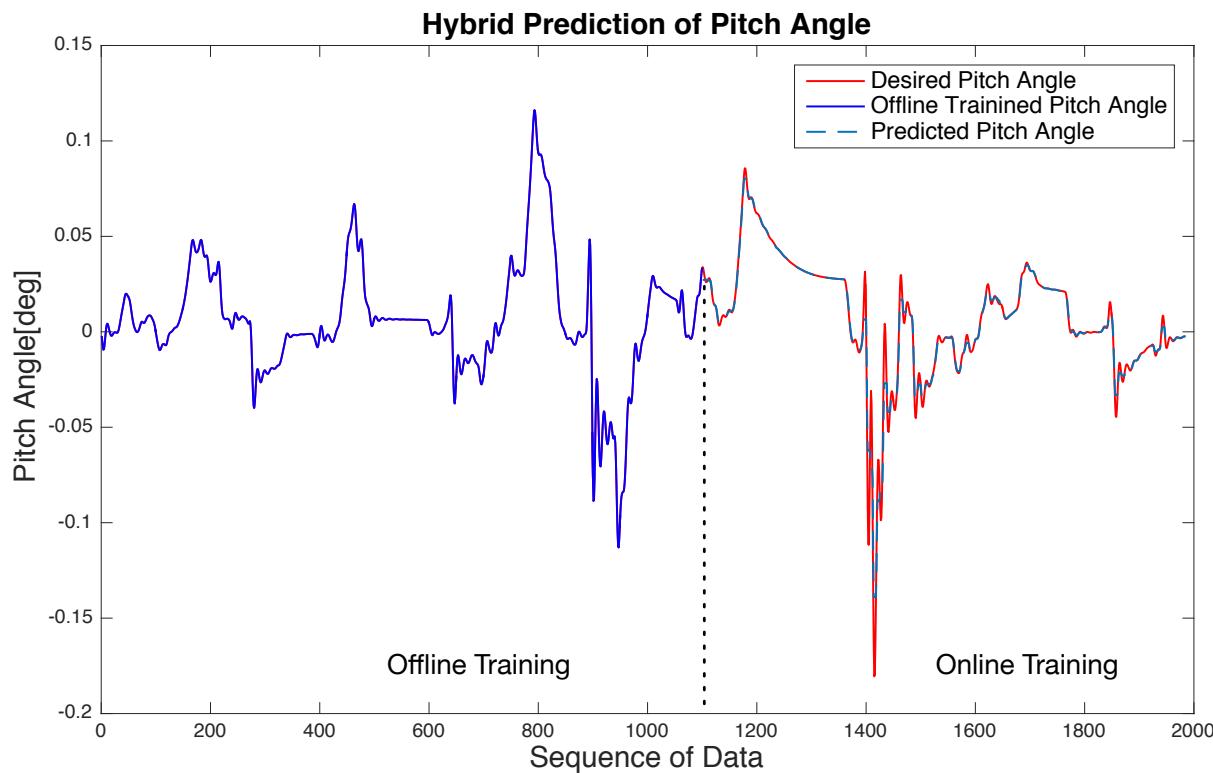


Trajectory

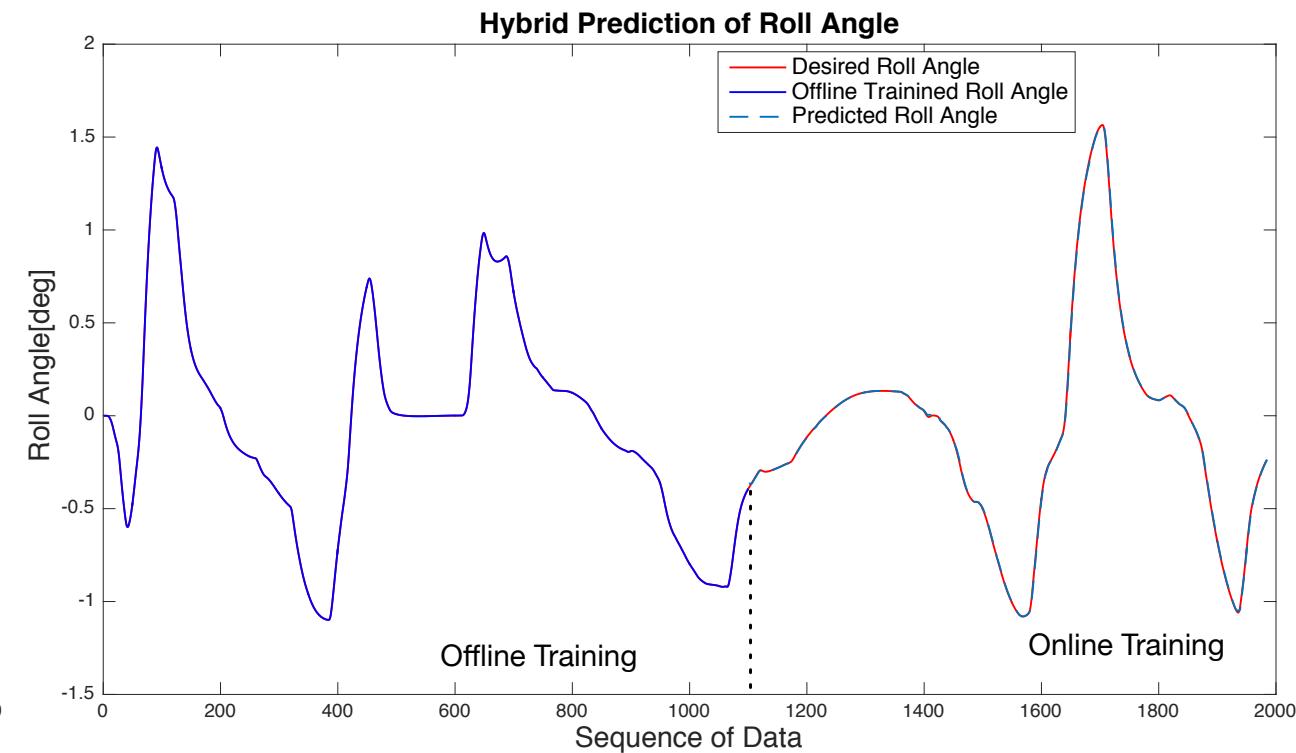


Yaw Angle

# Hybrid Prediction Results



Pitch Angle



Roll Angle

# Network Performance

MSE	Trajectory [m]	Yaw Angle [deg]	Pitch Angle [deg]	Roll Angle [deg]	Sway Velocity [m/s]	Surge Velocity [m/s]	Yaw Velocity [deg/s]
Offline	1.1	12	1.1e-06	3e-05	2.1e-04	2.9e-05	3.7e-04
Online	0.78	3.3	9.3e-05	0.0063	1.2e-04	0.0151	0.0042
Hybrid Prediction	0.022	0.0347	9e-08	4.4e-07	3.1e-07	1.9e-04	1.8e-05

MSE is small - so model is able to predict arbitrarily close to the targets

# Multi-step Prediction

Why ?

Decision Support for Captain

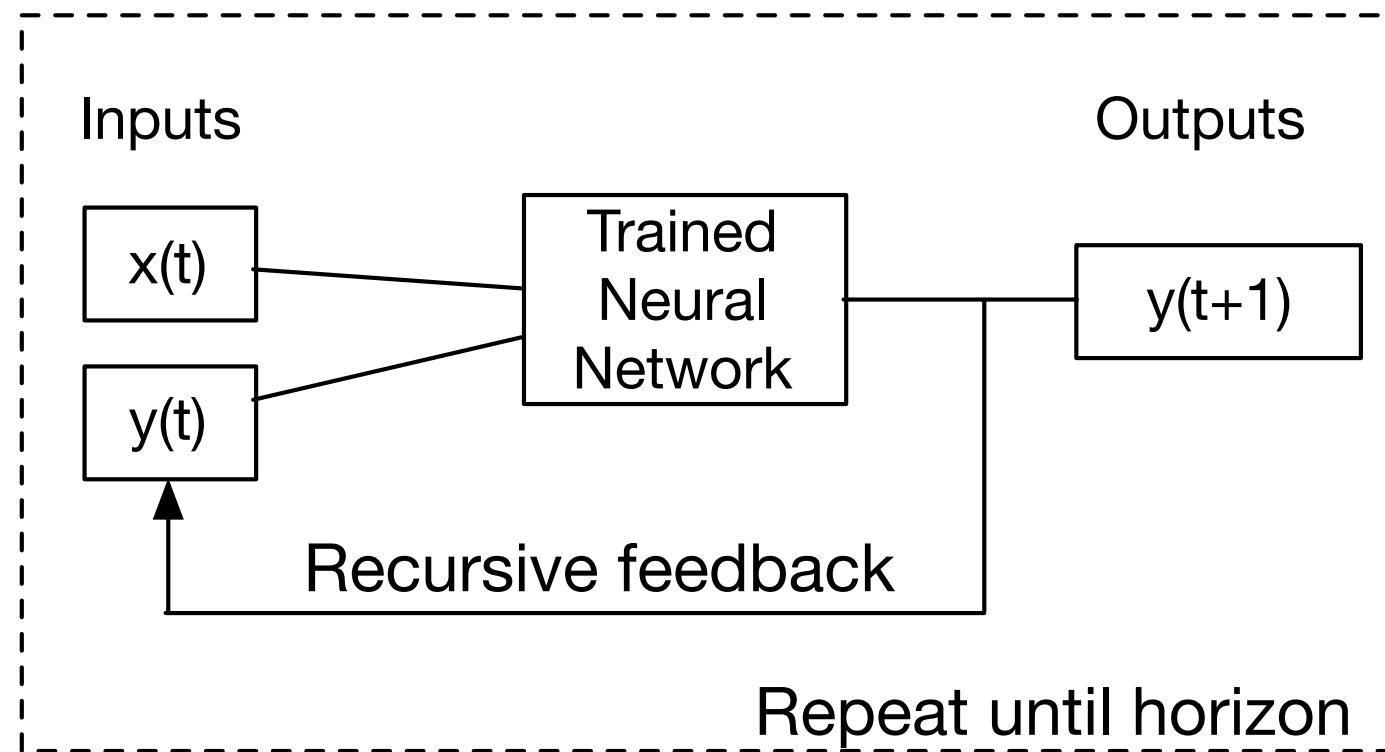
Early Actions

Example – Car Parking

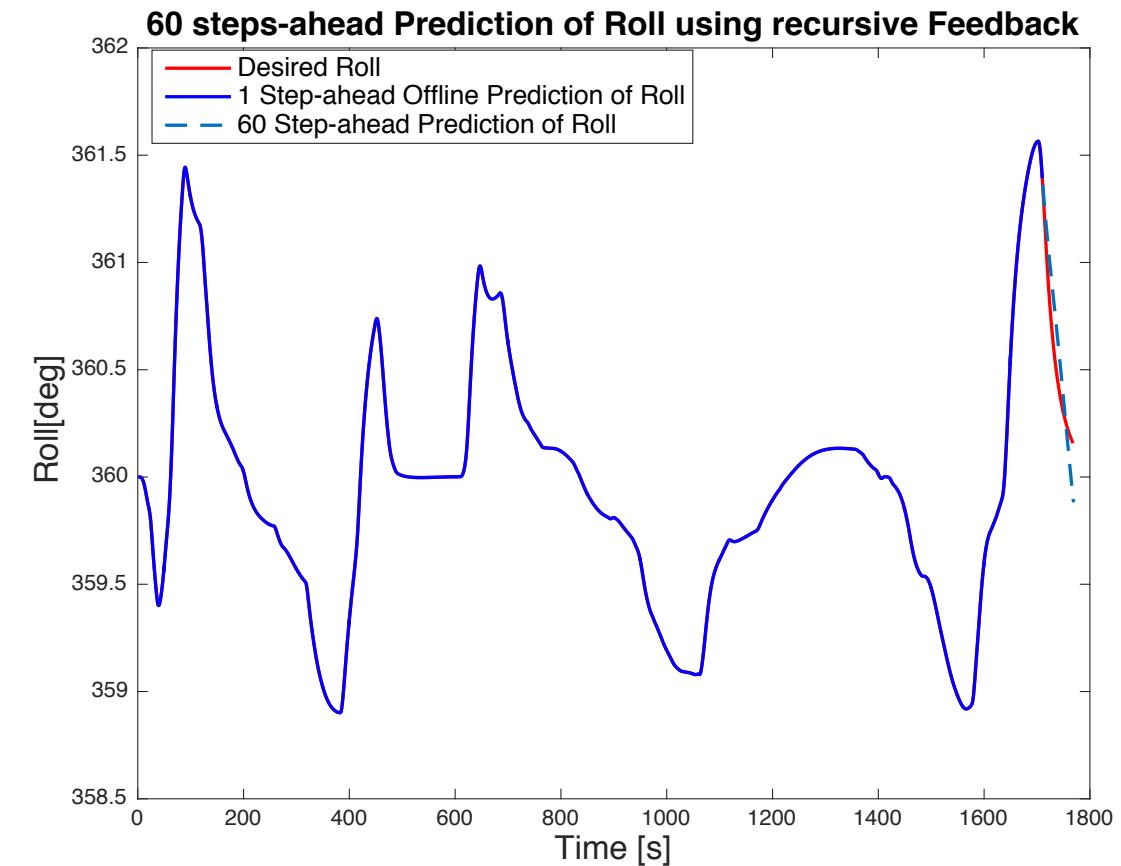
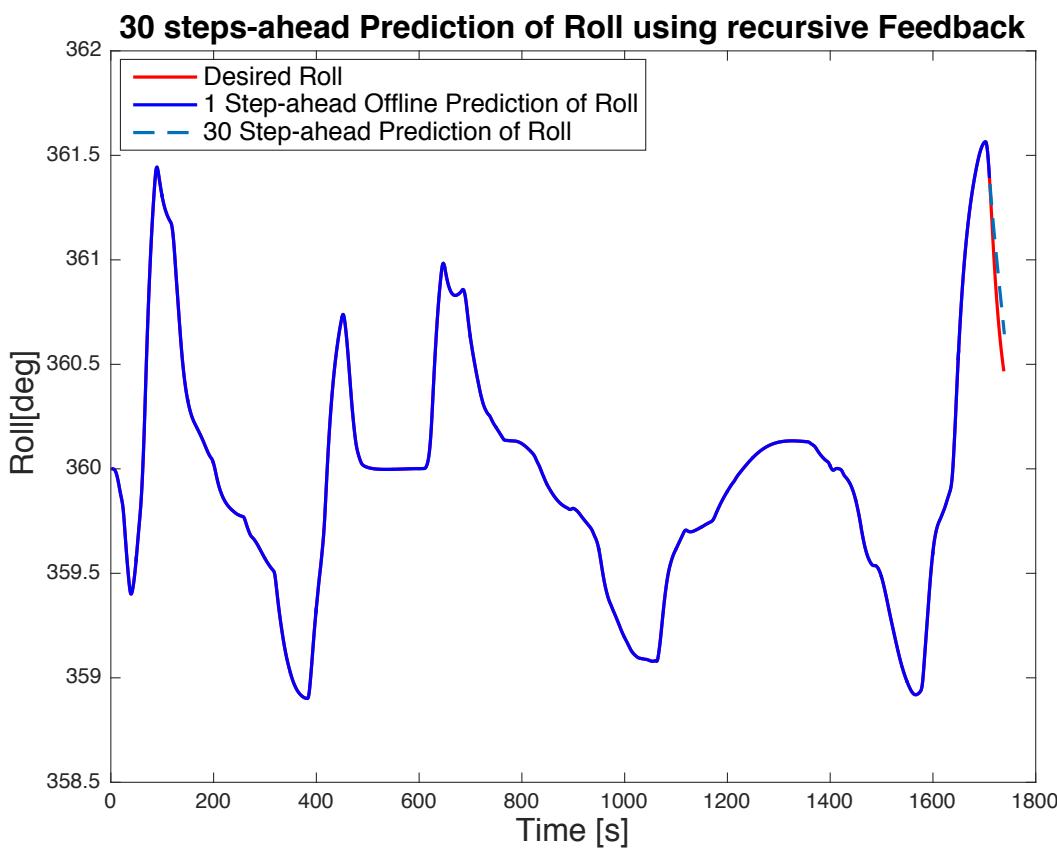


[4]

# Multi-step Prediction - Process

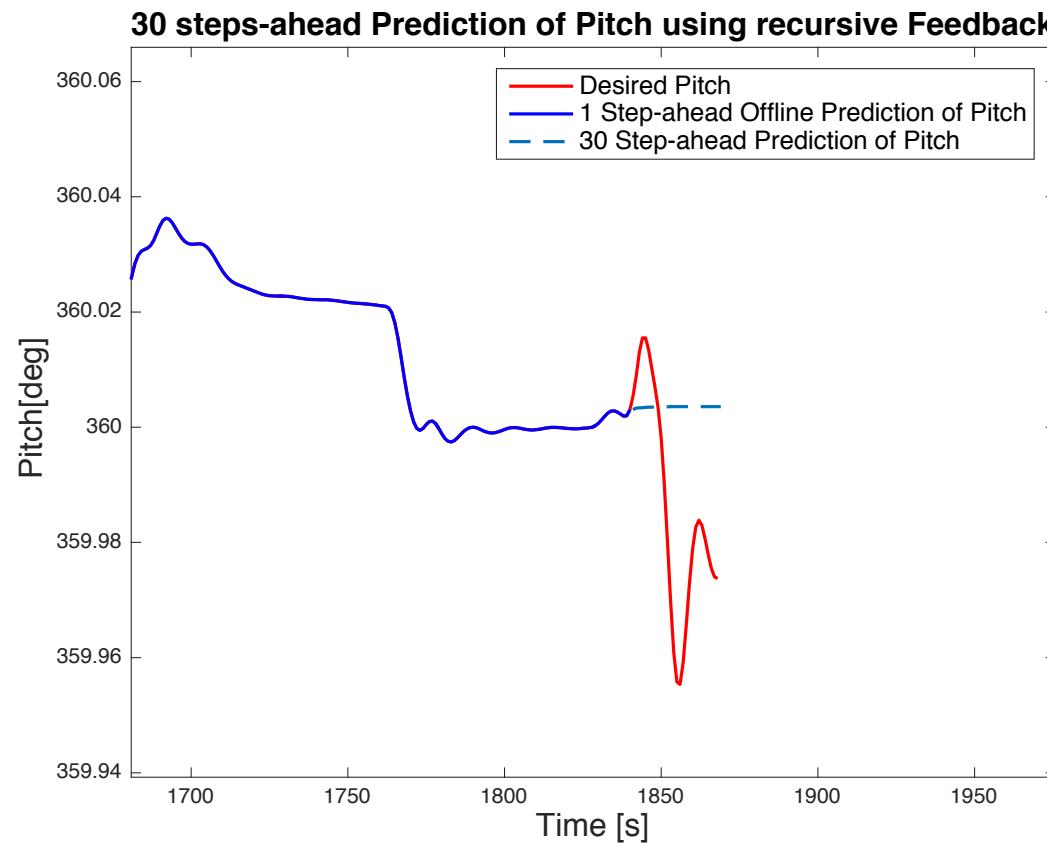


# Multi-step Prediction Results

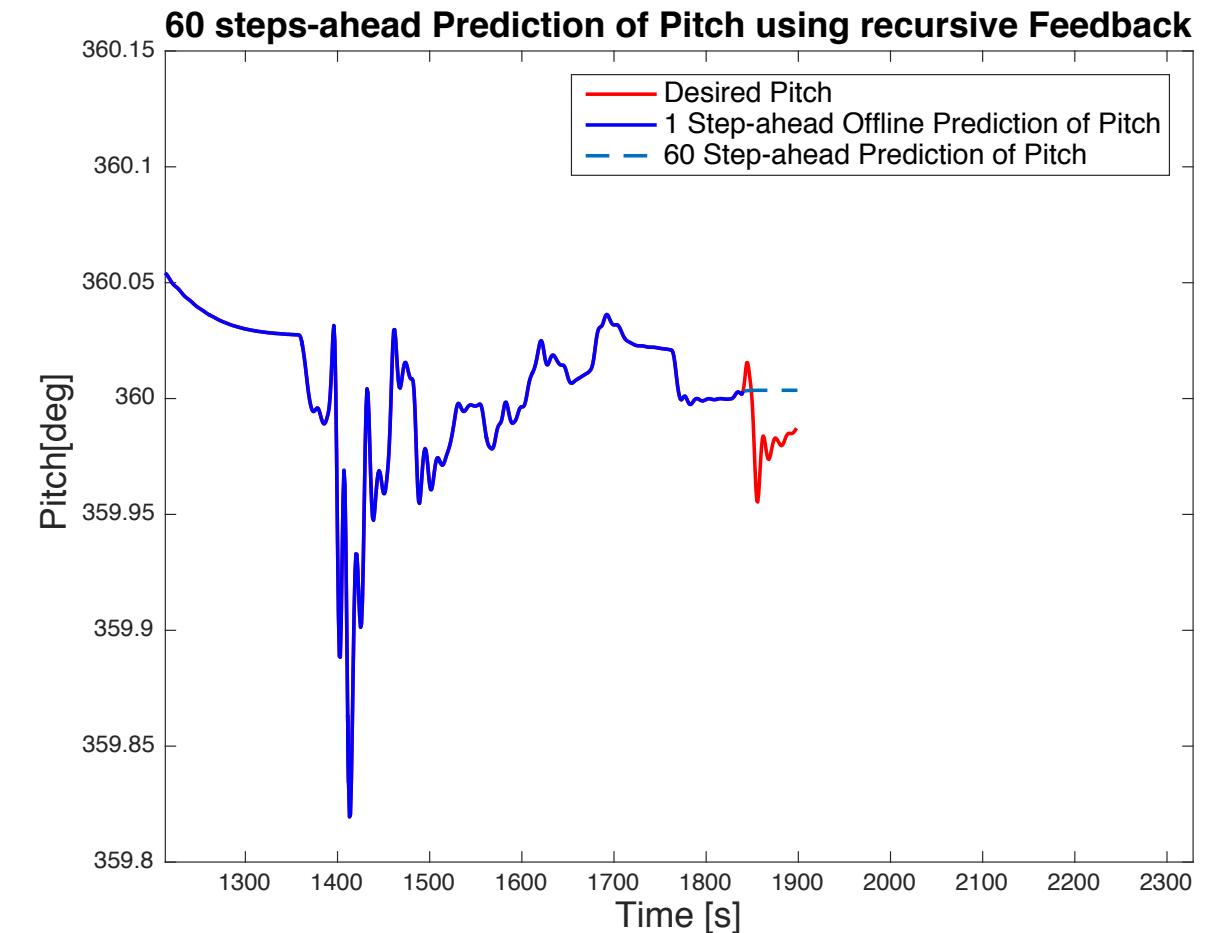


Roll Angle

# Multi-step Prediction Results

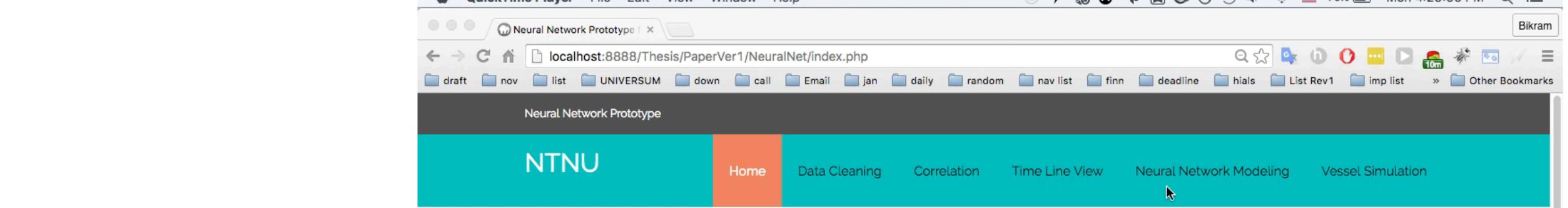


Pitch Angle



# Conclusions

- ❖ Learning Process
  - Offline
  - Online
  - Hybrid
- ❖ Multi-Step Prediction
- ❖ Prototype Application



Neural Network Prototype

localhost:8888/Thesis/PaperVer1/NeuralNet/index.php

NTNU

Home Data Cleaning Correlation Time Line View Neural Network Modeling Vessel Simulation

# Prototype Application

Application - <http://bit.ly/1U9DtcH>  
Demo Video - <http://bit.ly/24nrjmb>



## Welcome To Our Neural Network Prototype

This application will illustrate how we can implement Neural Network for the prediction of ship behaviour.

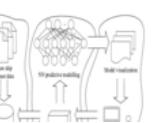
### About Our Application

This is a demonstration for the Paper entitled "Analysis and Modeling of Sensor Data for Ship Motion Prediction".

[Read More](#)

### Our Frame Work

Click Image to see large Size



### Follow The Step By Step Procedure.

# Future Work

- Consider the control forces and the environmental forces
- Kalman Filter to compare results
- Recurrent network, Radial Basis function
- Simulated annealing and Genetic Algorithm to seek minimum gradient

# Thank You !

# References

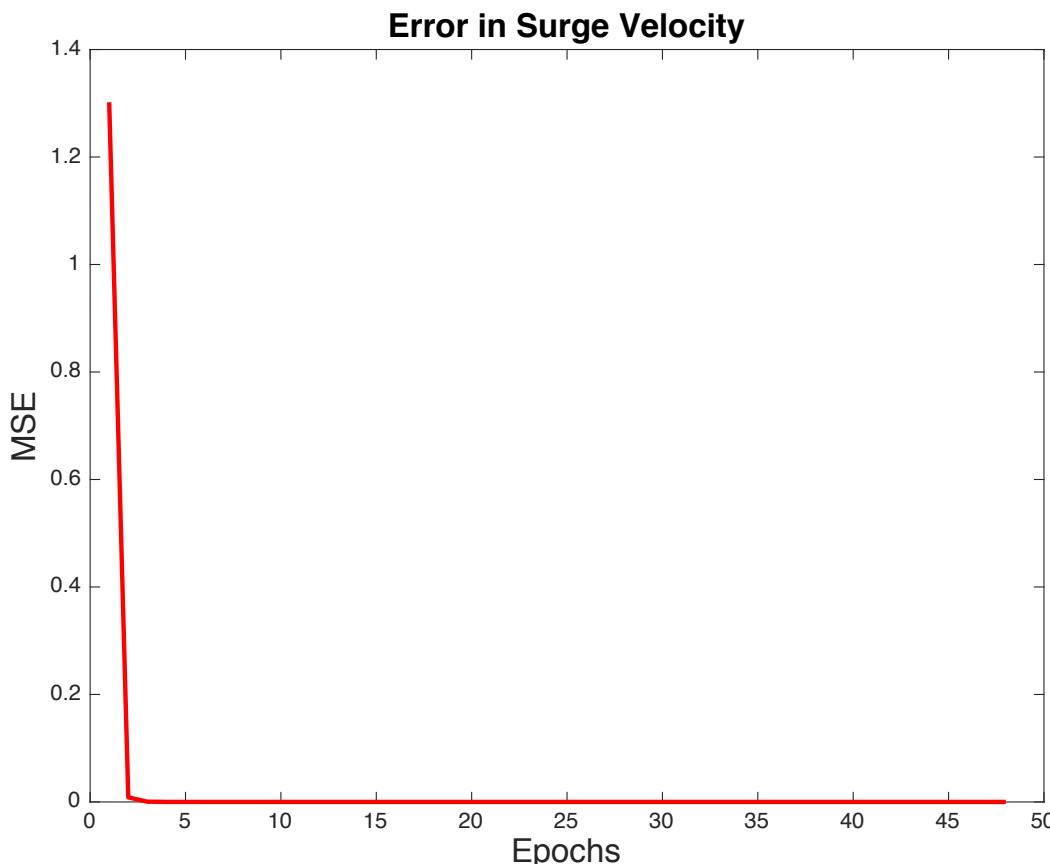
1. [http://www.rigzone.com/training/insight.asp?insight\\_id=342](http://www.rigzone.com/training/insight.asp?insight_id=342)
2. <http://wintlogix.com/business-intelligence/>
3. <http://calqlata.com/productpages/00059-help.html>
4. <https://i.ytimg.com/vi/V06d1co4pk8/maxresdefault.jpg>

# Neural Network Configuration

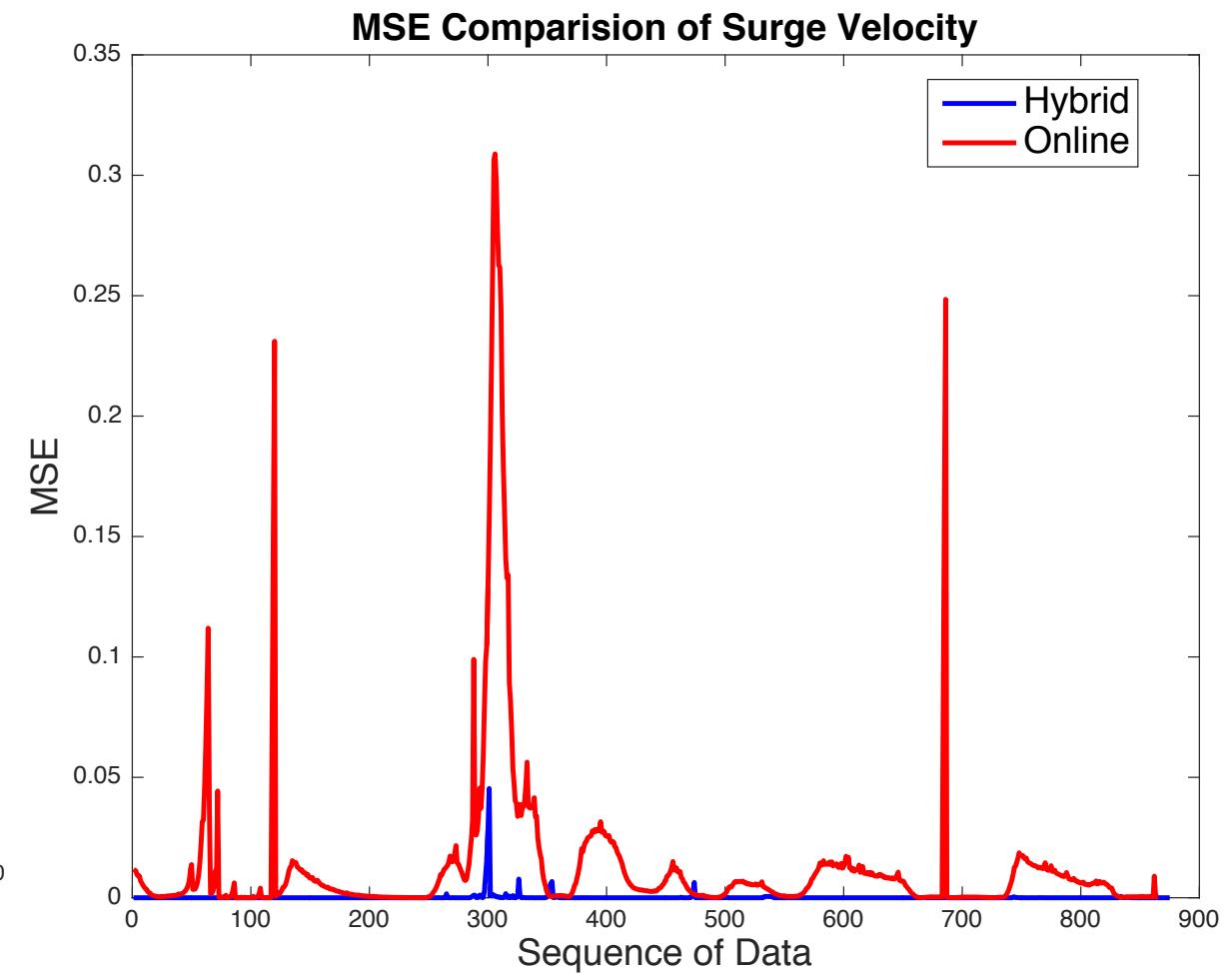
<b>Number of inputs attributes</b>	<b>28</b>
<b>Number of hidden neurons</b>	10
<b>Number of outputs attributes</b>	2 <sup>#</sup> or 1 <sup>¥</sup>
<b>Time Delay</b>	1 time step
<b>ANN architecture</b>	NARX
<b>Training Algorithm</b>	Levenberg-Marquardt
<b>Number of Epochs</b>	1000
<b>Minimum performance gradient</b>	1e-7
<b>Network Performance measure</b>	Mean Squared Error(MSE)
<b>Initial adaptive value (mu)</b>	0.001
<b>Initial Weights</b>	Random values*
<b>Adaptive decrease factor</b>	0.1
<b>Adaptive increase factor</b>	10
<b>Activation function in hidden layer</b>	Hyperbolic tangent sigmoid transfer function
<b>Activation function in output layer</b>	Linear transfer function

# Neural Network Performance

		Trajectory	Yaw Angle	Pitch Angle	Roll Angle	Sway Velocity	Surge Velocity	Yaw Velocity
OFFLINE	MSE	1.1	12	1.1e-06	3e-05	2.1e-04	2.9e-05	3.7e-04
	epochs	1000	459	13	300	99	117	125
	min. gradient	2.8e-3	9.9e-8	6.8e-8	5.8e-05	1.1e-4	4.2e-4	2.6e-4
	total time	497s	150s	18s	323s	104s	25s	128s
	Learning algorithm	0.27	1e-08	1e-09	1e-08	1e-06	1e-07	1e-07
ONLINE	MSE	0.78	3.3	9.3e-05	0.0063	1.2e-04	0.0151	0.0042
	epochs	6	6	5	12	5	5	3
	total time	426s	379s	303s	197	214s	315s	178
	mean time	0.25s	0.19s	0.18s	0.21s	0.2s	0.16s	0.27s
	min. gradient	0.02e-04	0.02	0.03e-04	0.01	2.2e-04	0.013	0.01
HYBRID	MSE	0.022	0.0347	9e-08	4.4e-07	3.1e-07	1.9e-04	1.8e-05
	epochs	6	3	3	4	4	7	4
	total time	141s	183s	94s	107s	133s	212s	151s
	mean time	0.16s	0.21s	0.11s	0.12s	0.15s	0.24s	0.17s
	min. gradient	2.1e-05	2.2e-05	1e-07	7.2e-07	3.6e-07	1.8e-05	2.9e-06



Offline Learning



Online and Hybrid Learning