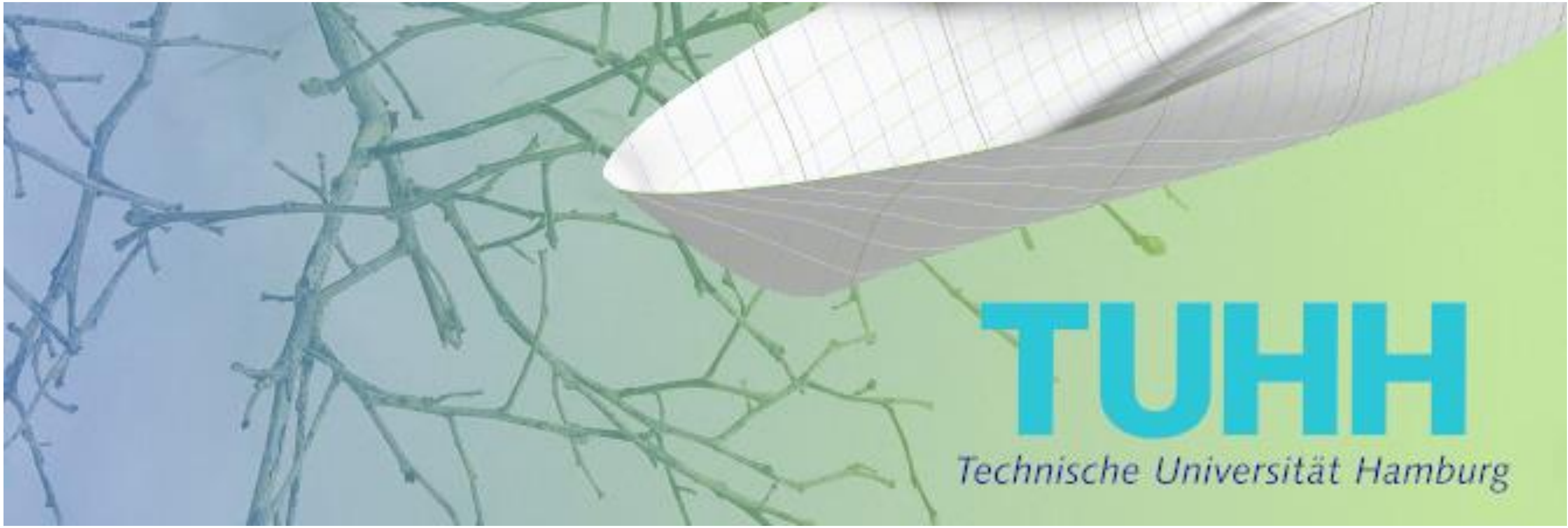
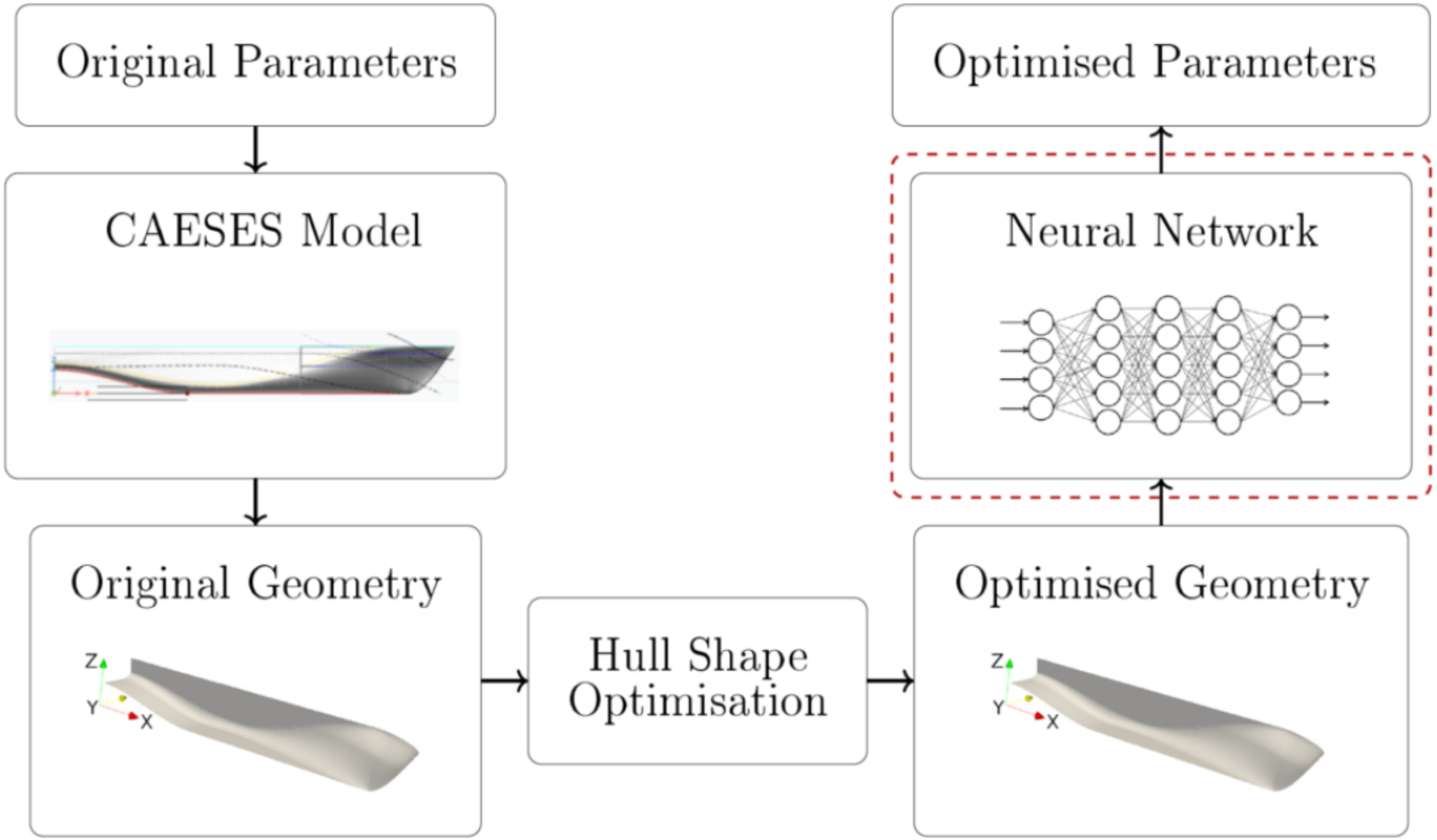


Geometry to Parameter Mapping Based on Neural Networks

Johanna Serr, M.Sc. Student at TUHH


Original Parameters
CAESES Model
Original Geometry
Hull Shape Optimisation
Optimised Geometry
Neural Network
Optimised Parameters

Research Question

How to extract descriptive parameters
from a hull geometry?

- Mapping function f:

$$f(point\ set) = parameters$$

Why using Neural Networks (NN) for this problem?

- Task: Pattern recognition, Regression problem
- Easy and inexpensive generation of data

Table of Contents

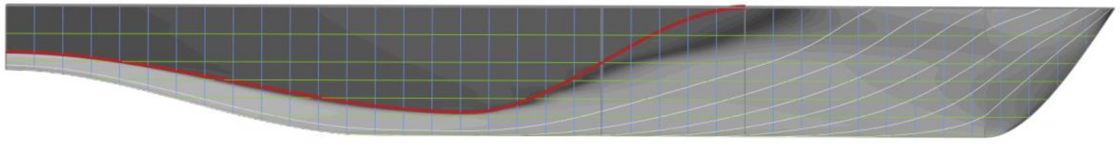
1. Creating Dataset Using CAESES

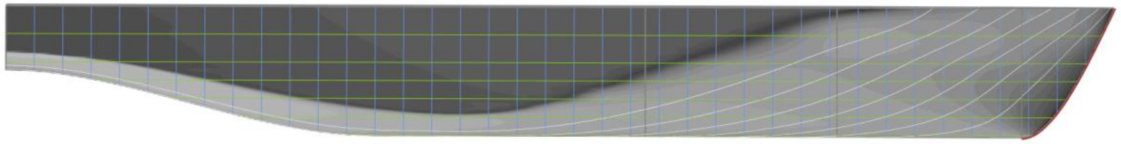
1. Creating Hull Geometries

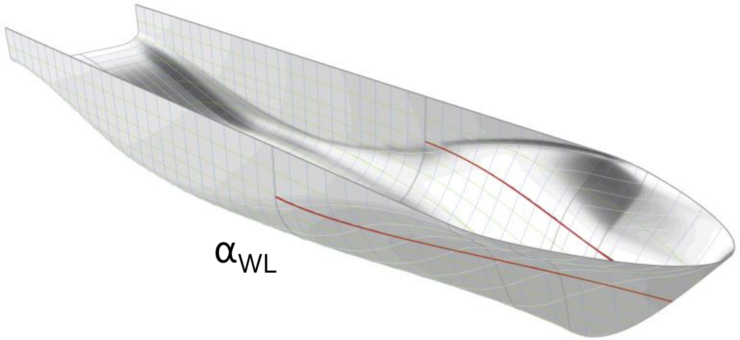
- Investigation of 2 different hull models
- Defining different types of parameters to be varied
  - ➢ length, angle and coordinates
- Creation of 4000 unique geometries with constant bounding box

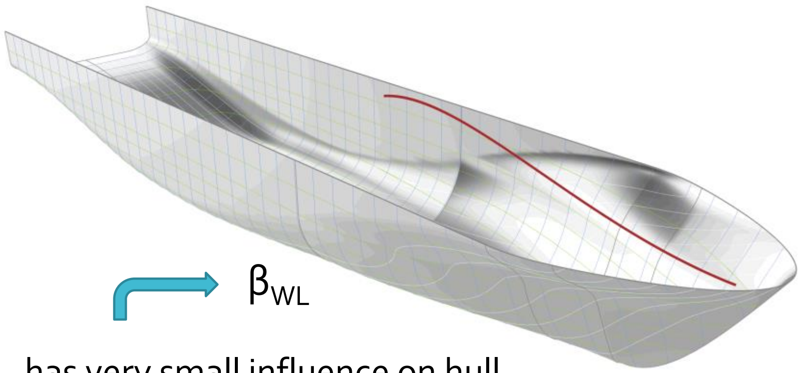
1. Hull model #1[1]

Name	Parameter	Type
Length of Flat Side	L_{FOS}	Length
Length of Stem	L_{stem}	Length
Waterline Entrance Angle	α_{WL}	Angle
Frame Gradient at the Waterline Distribution	β_{WL}	Angle





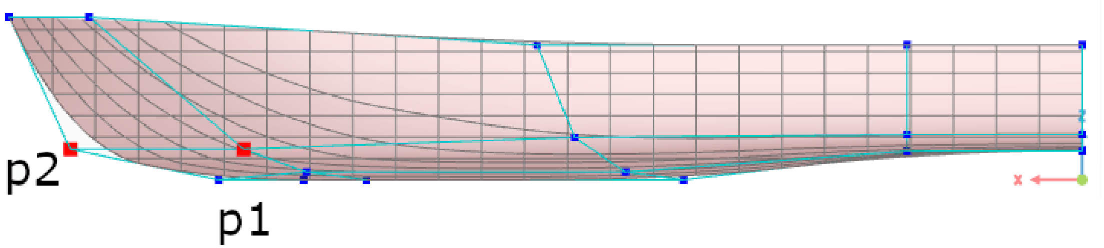


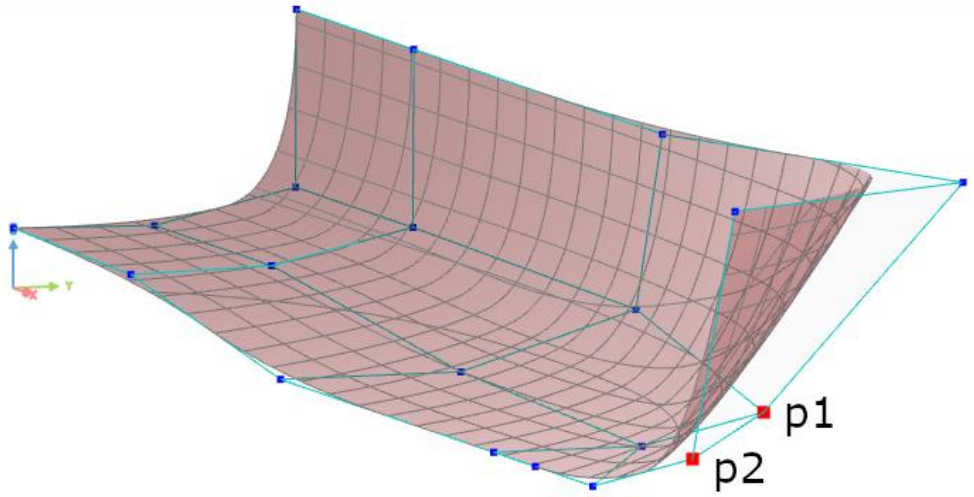


Variation of β_{WL} has very small influence on hull shape compared to the other parameters

1. Hull model #2

Name	Parameter	Type
X-Coordinate	$p1_X$	Coordinate
Y-Coordinate	$p1_Y$	Coordinate
Z-Coordinate	$p1_Z$	Coordinate
X-Coordinate	$P2_x$	Coordinate
Z-Coordinate	$p2_z$	Coordinate





1. Extracting Point Sets

- Uniform and random distribution of points on surface of geometry
- Make use of symmetry
- Total number of points: 1024
- Data further prepared for training (scaled, permutations of order)

Example of point cloud

2. Challenges of Using Point Sets as Input Data

2. Image versus Point Set

IMAGE:

255	133	105	255
255	105	105	255
255	201	105	255
255	255	105	255

$$\longrightarrow \begin{bmatrix} 255 & 133 & 105 & 255 \\ 255 & 105 & 105 & 255 \\ 255 & 201 & 105 & 255 \\ 255 & 255 & 105 & 255 \end{bmatrix}$$

POINT SET:

$\longrightarrow$?

2. Property of Point Sets[2]

Unordered set of 3D points :

$$\{\, P_i = (x\ y\ z)_i \ \in \mathbb{R} \mid i = 1, 2, \ldots, N\}$$

Requirements for the Neural Network:

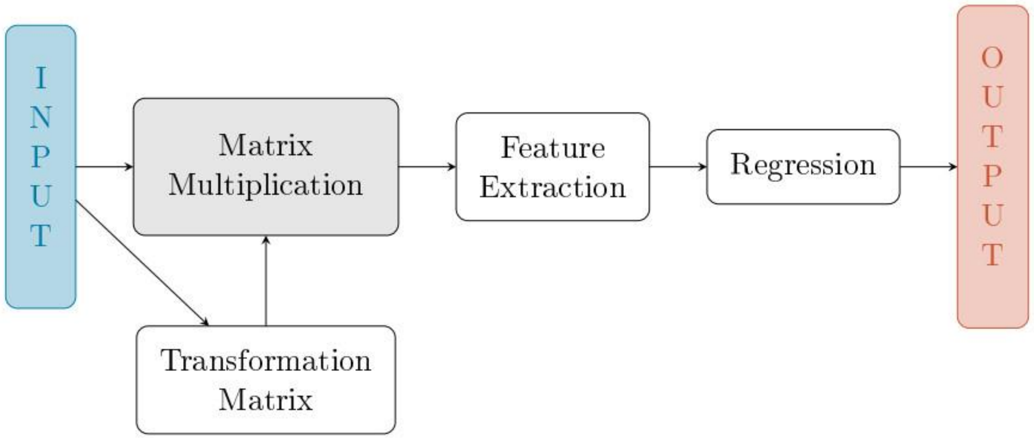
- Invariance to N! permutations of input sets

3. Neural Network Architecture

3. Basic Idea

Challenge problems when using point sets as input data

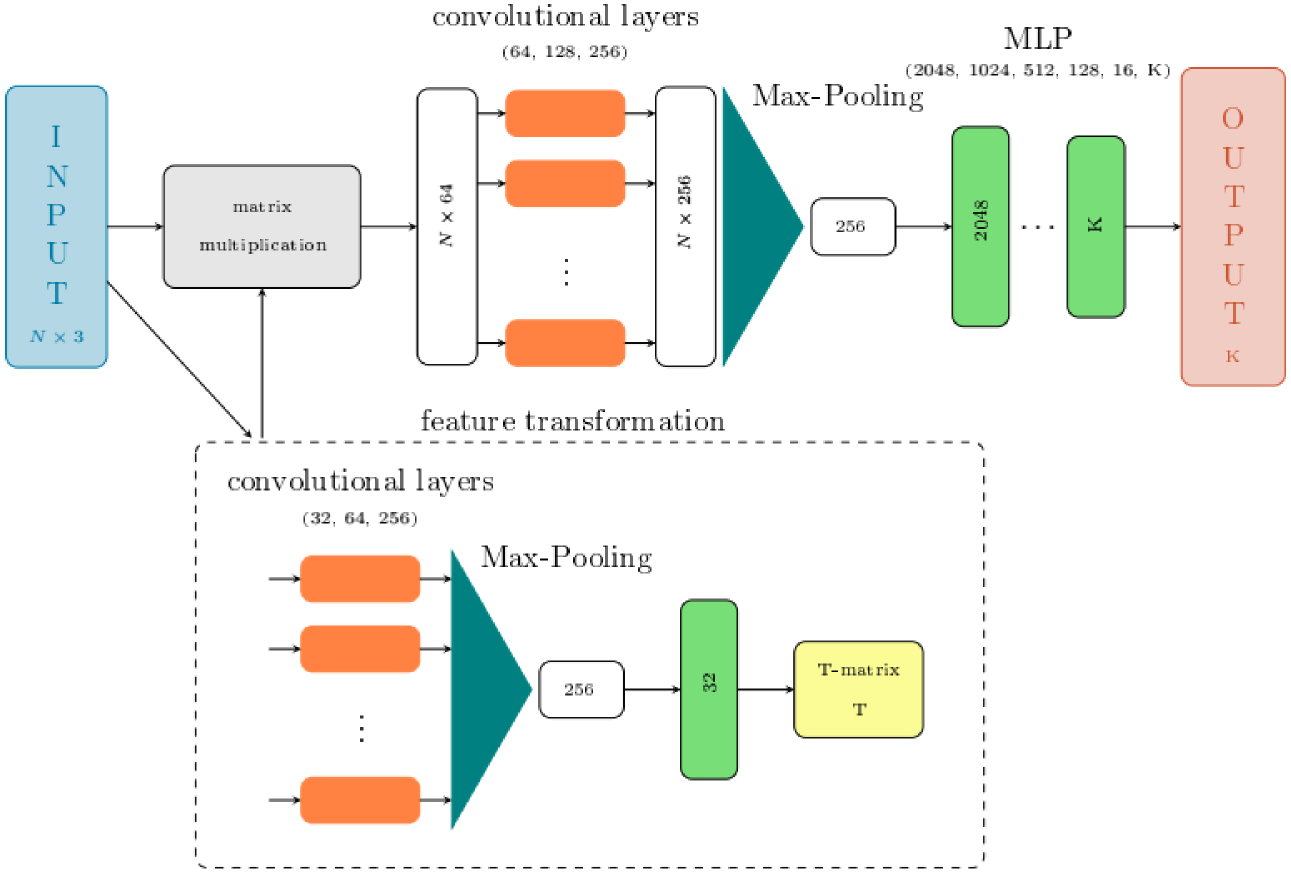
- Make input data invariant by performing matrix multiplication:



Outline of the NN Architecture, inspired by *pointNet*[2]

- Implementation of NN and training carried out using Keras[3] and Tensorflow[4]

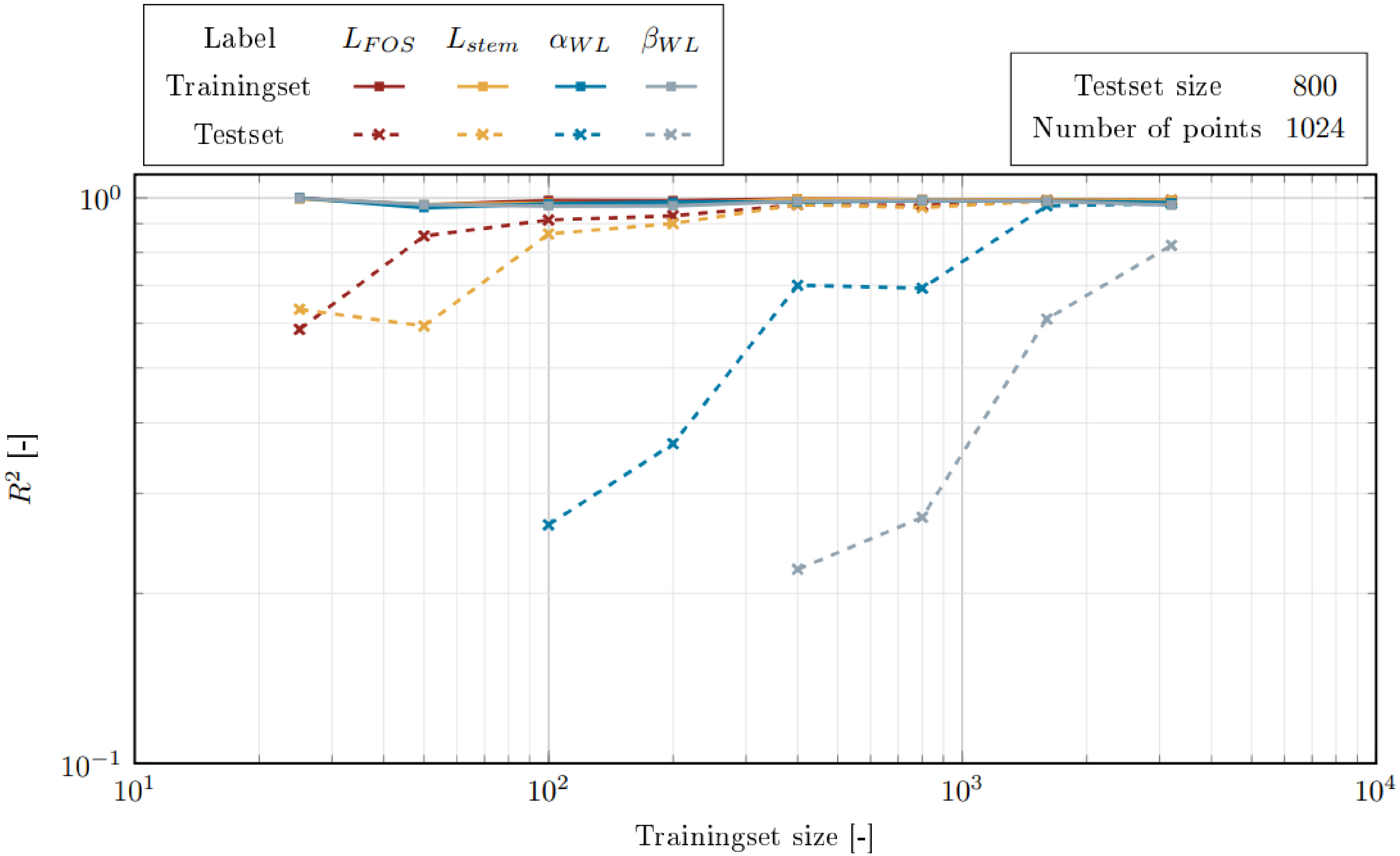
3. Neural Network Architecture



3. Performance Diagnostic

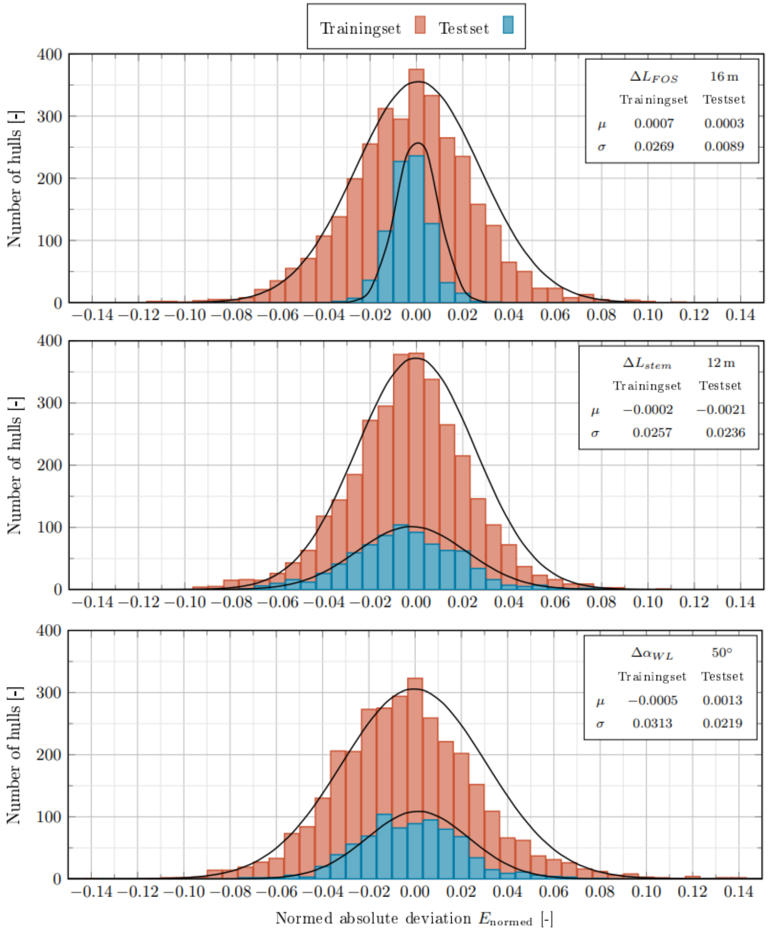
- Goal: Ability of Generalisation
- Split dataset into training- and testset
- Answer the question: How much training data is required?
    - Learning Curves

3. Learning Curves

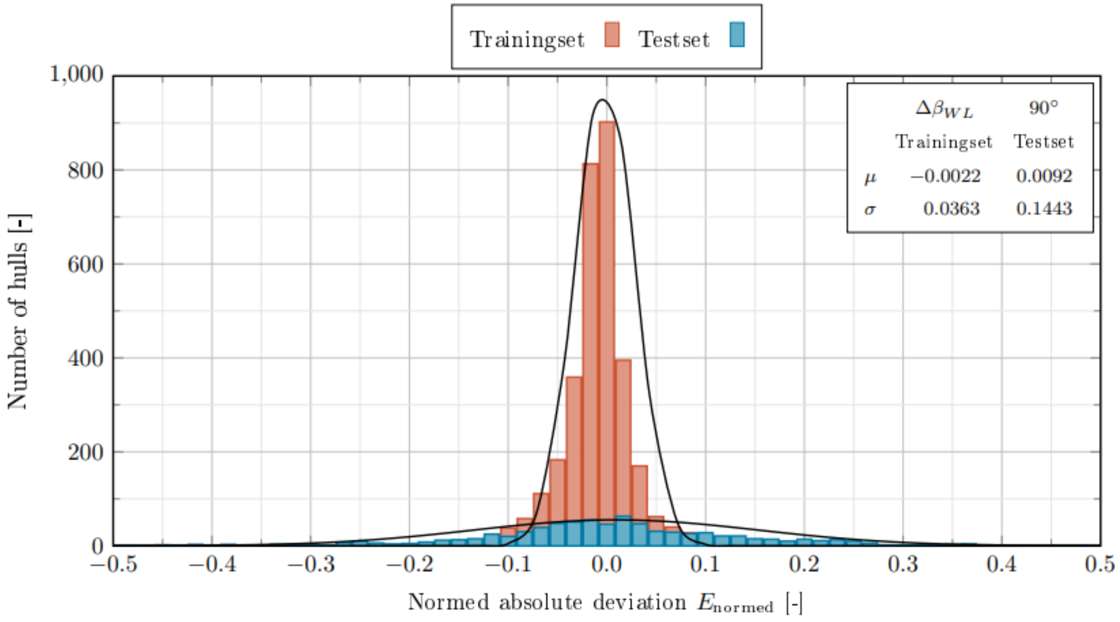


4. Results

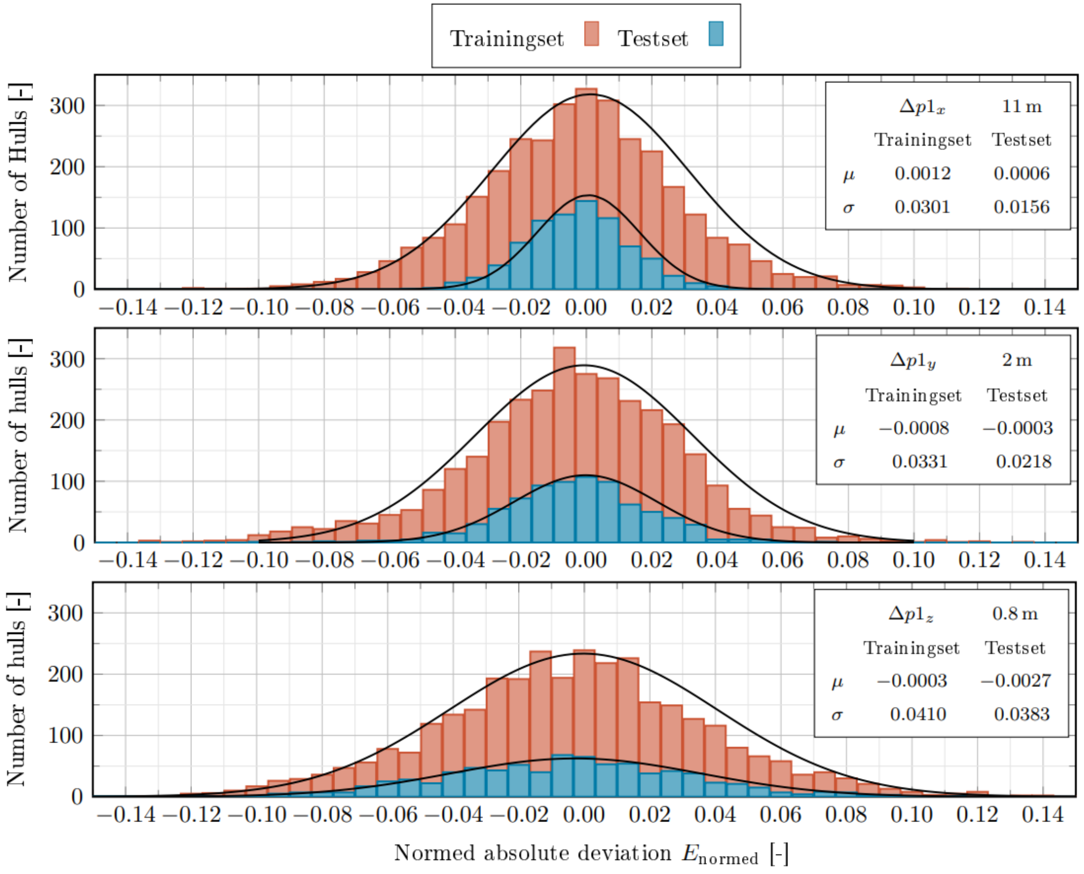
4. Histogram – Hull Model #1



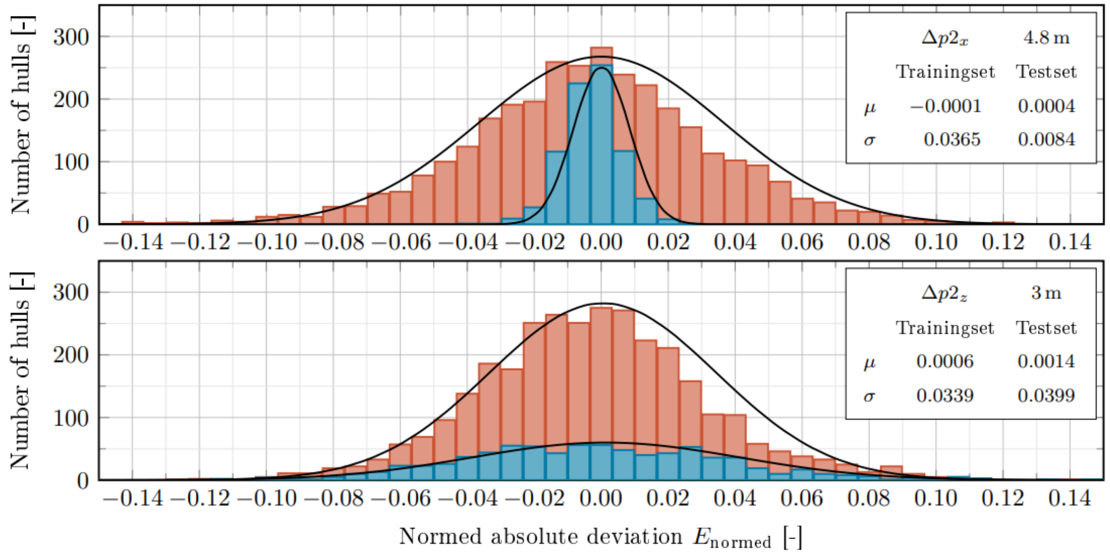
$$E_{normed} = \frac{x_{True} - x_{Pred}}{x_{max} - x_{min}}$$



4. Histogram – Hull Model #2

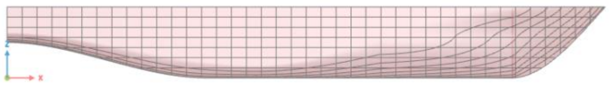


$$E_{normed} = \frac{x_{True} - x_{Pred}}{x_{max} - x_{min}}$$

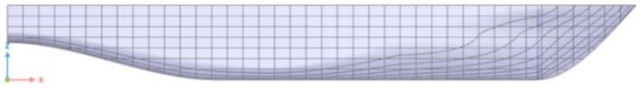


4. Example – Hull Model #1

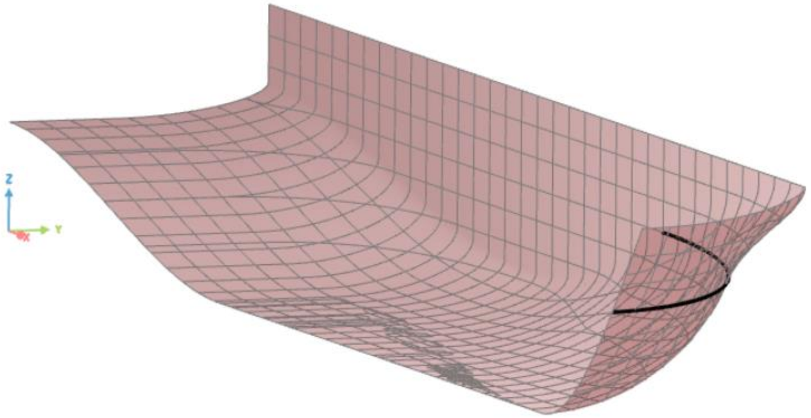
Parameter	True Value	Prediction	x_{True} - x_{Pred}
L_{FOS}	50,8029m	50,7768m	0,0261m
L_{stem}	9,3808m	9,5047m	-0,1239m
α_{WL}	80,7325°	83,6001°	-2,8676°
β_{WL}	43,7752°	50,7448°	-6,9696°



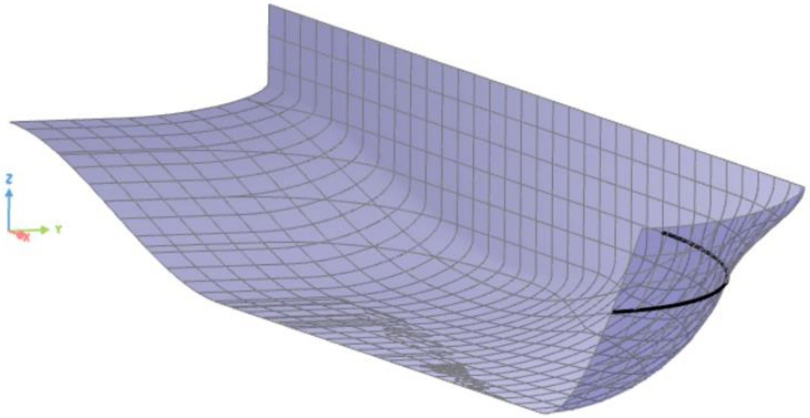
(a) Hull generated using true values



(b) Hull generated using predicted values



(e) Hull generated using true values



(f) Hull generated using predicted values

5. Summary

5. Summary

- NN model able to take point sets as input data and extract features
- NN model able to extract parameters from hull
  - Different types of parameters
- Results depend on influence of parameter on the hull shape (-> β_{WL})
  - Small influence -> problem more complex

Ideas for future research:

- Choose non-uniform distribution of points on surface
  - Higher resolution of relevant regions
- Investigate problem using more appropriate error metrics (difference in volume, ...)

Questions?

Sources

[1] Friendship System AG. *Creating a Fast Monohull Vessel.* url: https://www.caeses.com/

[2] C.R.Qi et al. *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*. 2017

[3] F. Chollet et al. *Keras*. url: https://www.keras.io

[4] M.Abadi et al. Tensorflow: Large-Scale Machine Learning on Heterogeneous Systems. url: https://www.tensorflow.org

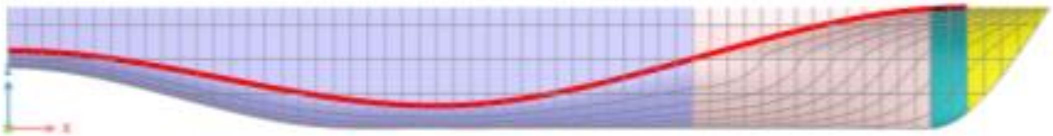
Contact Information

Johanna Serr
johanna.michaele.serr@tuhh.de
https://github.com/bielerich/geom2parNet

Backup

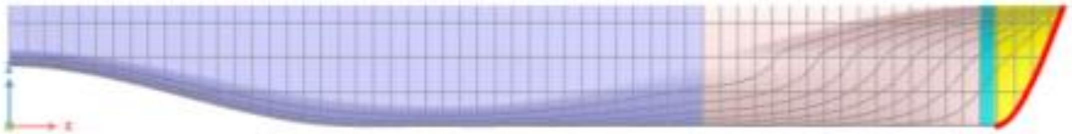
1. Hull Model #1 - L_{FOS}

$L_{FOS} = 40\,\mathrm{m}$



$L_{FOS} = 56\,\mathrm{m}$

1. Hull Model #1 - L_{stem}



$L_{stem} = 3\,\mathrm{m}$

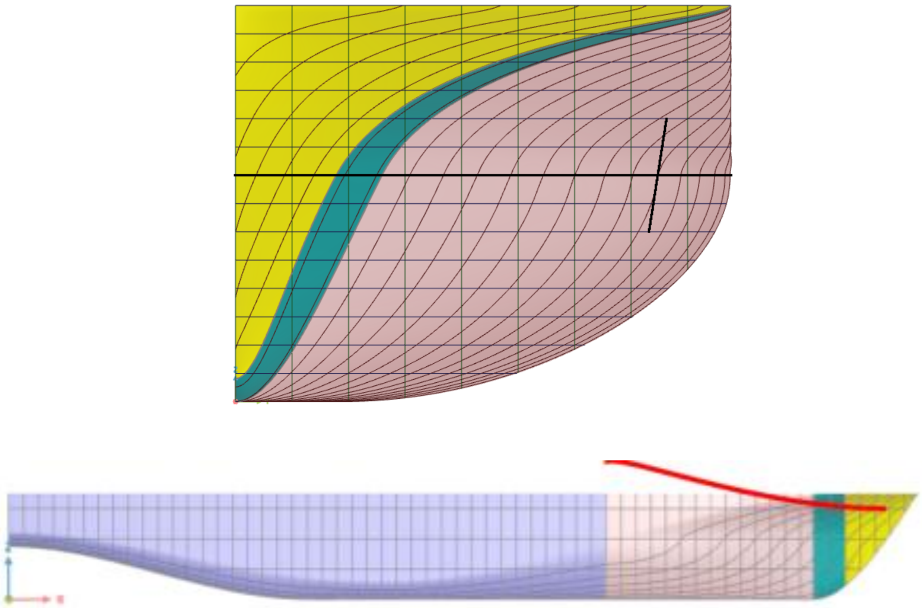
$L_{stem} = 15\,\mathrm{m}$

1. Hull Model #1 - α_{WL}

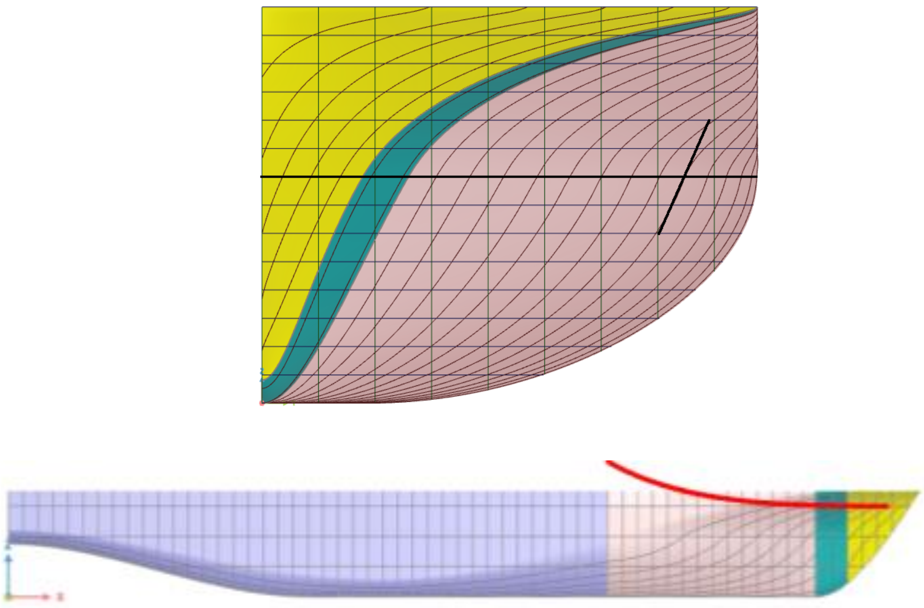
$\alpha_{WL} = 40°$

$\alpha_{WL} = 90°$

1. Hull Model #1 - β_{WL}

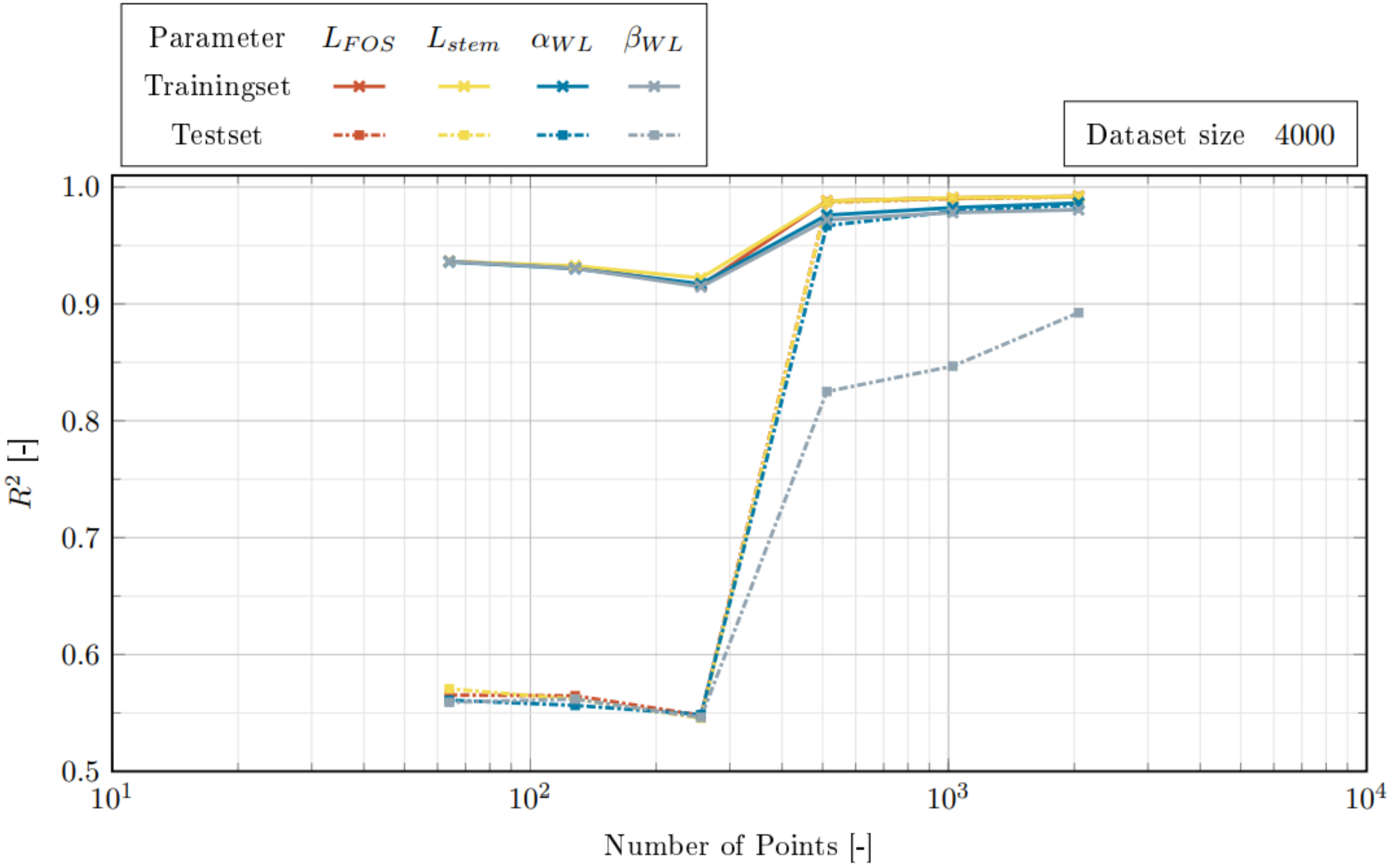


$\beta_{WL} = 30^\circ$

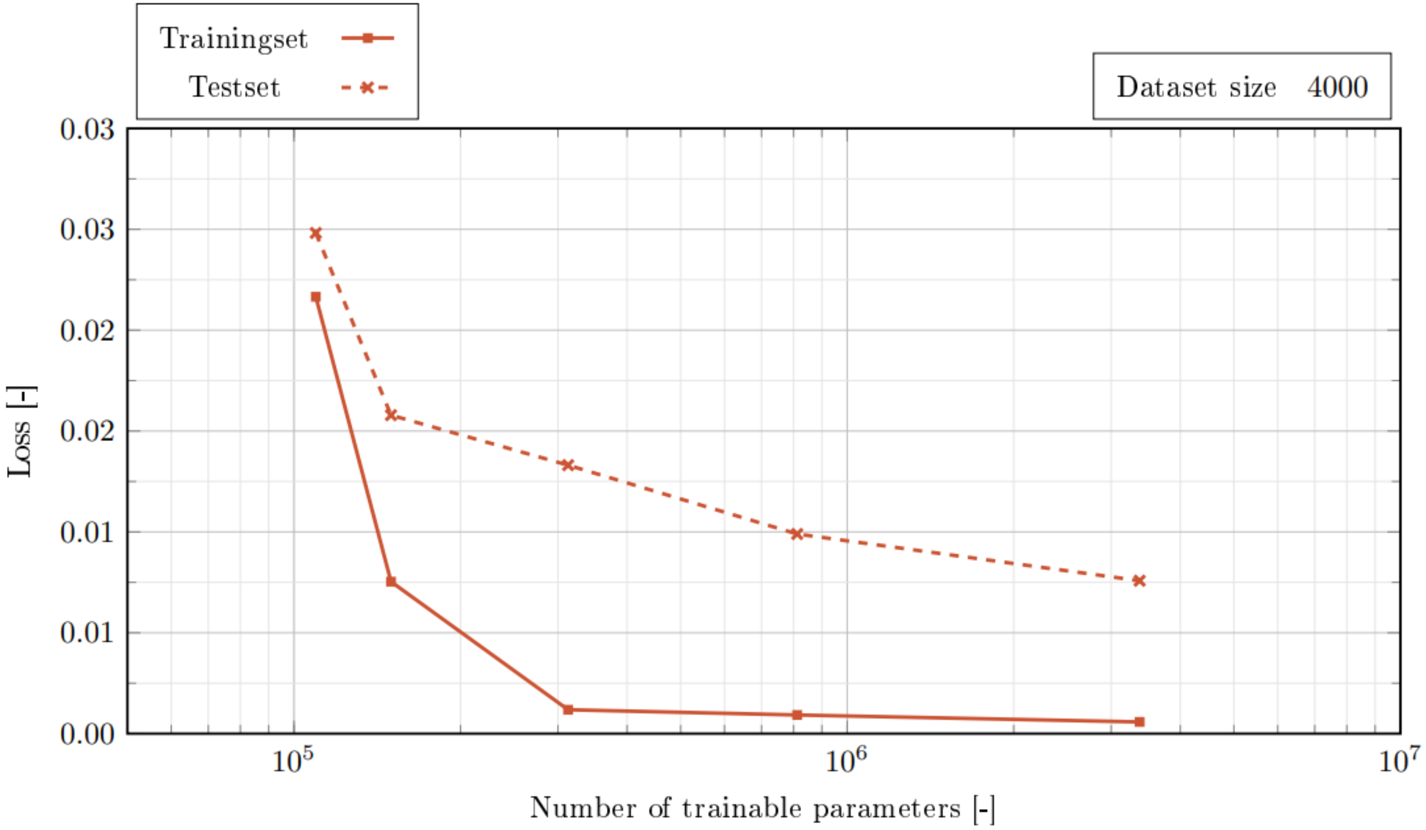


$\beta_{WL} = 120^\circ$

3. Total Number of Points in Point Set

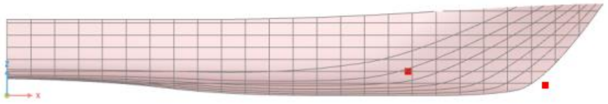


3. Number of Trainable Parameters

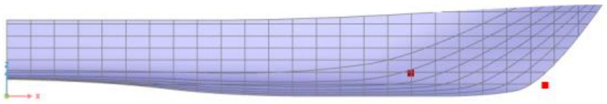


4. Example – hull model #2

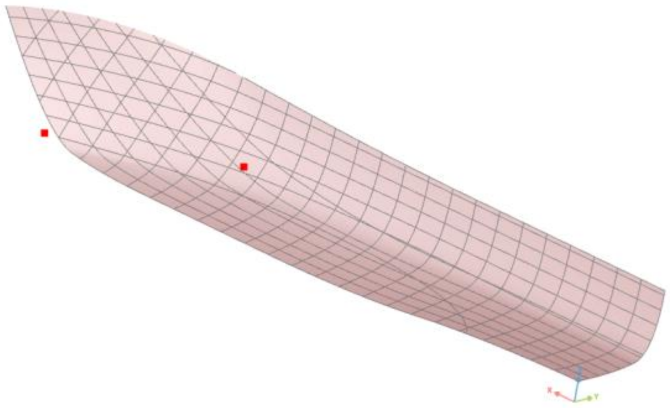
Parameter	True Value	Prediction	x_{True} - x_{Pred}
$p1_x$	16,8198m	16,9657m	-0,1459m
$p1_x$	2,8811m	2,9327m	-0,0516m
$p1_x$	0,9975m	0,9561m	0,0414m
$p2_x$	22,5793m	22,6124m	0,0331m
$p2_z$	0,4279m	0,4514m	-0,0235m



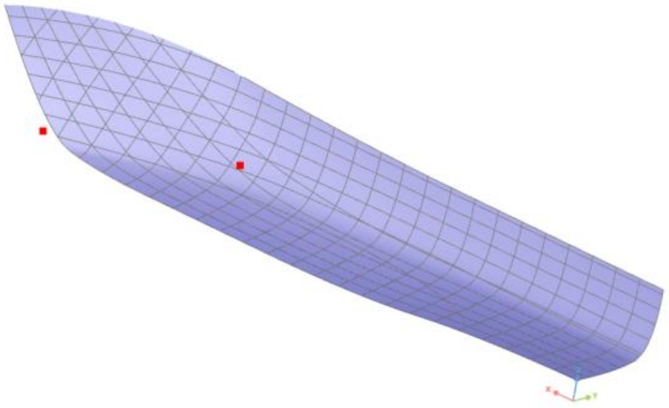
(a) Hull generated using true values



(b) Hull generated using predicted values



(c) Hull generated using true values



(d) Hull generated using predicted values

4. Rotation – hull model #1