A Novel Approach for Hull Hydrodynamic Surrogate Models with Convolutional Neural Networks (CNNs)

MARINE 2023

Austin Shaeffer*† and Chi Yang†

* Ship and Submarine Concept Design Tools Branch (823)
Naval Surface Warfare Center – Carderock Division (NSWC-CD)
9500 MacArthur Blvd
Bethesda, Maryland 20817, USA

e-mail: austin.k.shaeffer.civ@us.navy.mil

† Department of Physics and Astronomy
George Mason University (GMU)
4400 University Dr
Fairfax, Virginia 22030, USA

Email: cyang@gmu.edu

ABSTRACT

A novel approach for using artificial neural networks (ANNs) in the hull design problem is considered. Convolutional Neural Networks (CNNs), which are primary used for image recognition tasks and other models that require spatial reasoning, are used to model the resistance of a hull form. The CNNs are given the (X, Y, Z) value of the hull control point coordinates instead of the traditional RGBA (Red, Blue, Green, Alpha) values of image pixels. The CNNs are trained to regress the non-dimensional coefficient of resistance of the hull form. This new method for generating surrogate models is compared to conventional surrogate model methods as a function of regression error and training sample size. This sensitivity shows that this new approach for generating surrogate models of hull form performance is comparable to existing methods but offers more functionality and flexibility. Primarily the CNNs offer an efficient computation of the partial derivate of the movement of each control point to the objective function, which is the coefficient of resistance in this study. Figure 1 is a heat map of these partial derivatives relating the suggested movement in the y-direction (beam) of each of the control points on the series-64 hull form to the coefficient of resistance at a Froude number of 0.28. This heat map enables the naval architect to be better inserted into the hull form optimization process resulting in more successful use of optimization tools in terms of improving the outcomes from automated hull optimization software. In addition, the efficient development of partial derivatives from the surrogate model supports the use of direct searching optimization methods. Finally, this paper investigates the memory of the CNNs and their ability to learn the entire design space and be used to optimize a subset of the design space. A small study is conducted training the CNNs on a larger, more variable design space and then using this CNN to optimize the hull with a subset of the trained design variables. Multiple methods for training and re-training the CNNs are tested for effectiveness with initial results suggesting this could be a functional use case for CNNs to improve hull form optimization efficiency.

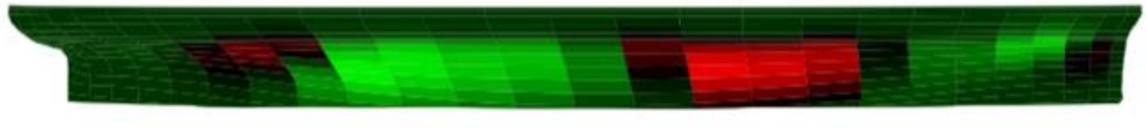

Figure 1. Series 64 Hull Form Control Point Modification Heatmap, Froude Number = 0.28

Keywords: Resistance; Hull Form; Surrogate Model; Artificial Neural Networks; Convolutional Neural Network; Optimization.

NOMENCLATURE

ANN Artificial Neural Network
CNN Convolutional Neural Network
DV Design Variables
FFN Feed Forward Network
GMU George Mason University
ONR Office of Naval Research
RBF Radial Basis Function
RBG Red, Blue, Green
SSF Steady Ship Flow

1. INTRODUCTION

Hull form performance is extremely important to the effectiveness of a ship design. In addition, reducing the hull form resistance is an extremely effective way to reduce the total operating cost of a ship. This study exclusively considers the hull optimization of ship resistance for the purposes of considering a new surrogate model methodology. Hydrodynamic optimization is most effective when the evaluation of the objective function can be done quickly. However, predicting hull performance with computational fluid dynamics (CFD) software is relatively computationally expensive and difficult to automate. The state-of-the-art solution to reducing the cost of hull performance evaluation is to utilize a surrogate model that can regress the objective function.

Surrogate models are developed with training data to approximate the objective functions used in optimization based on a pre-defined set of input variables. The fidelity of these approximations is crucial to the success of the optimization process. However, there is an expected and acceptable amount of error that can come from surrogate models since it is an approximation. A recent emphasis in data analytics has been in the use of Artificial Neural Networks (ANNs) which are often used as surrogate models. In the typical application as regression models, simple Feed Forward Networks (FFNs) are functionally very similar to other surrogate model methodologies. A previous study by the authors considered the use of these simple ANNs as surrogate models and the sensitivity to sample size (Shaeffer and Yang, Application of Machine Learning to Early-Stage Hull Form Design) (Shaeffer, Application of Artificial Neural Networks to Early-Stage Hull Form Design).

Convolutional Neural Networks (CNNs) are a more complex type of ANNs that are designed to capture spatial information and are often used in image recognition. The use of CNNs as regression models is less common than FFNs but a first principles dissection of the ship resistance problem lends itself well to the spatial reasoning the CNNs are designed to capture. Based on the authors' previous work there are many benefits to teaching an ANN about hull form hydrodynamics problems. The motivation for this research is to prove that CNNs can perform adequately well compared to the conventional surrogate model methods. In addition, the CNN offers more functionality and flexibility than conventional methods based on the same partial derivative method used in the authors' previous paper on FFNs (Shaeffer and Yang, Application of Machine Learning to Early-Stage Hull Form Design) (Shaeffer, Application of Artificial Neural Networks to Early-Stage Hull Form Design).

1.1 Hull Form Optimization

The Hull Form design problem has been described in many of the papers referenced here. For the purpose of clarification, terminology, and general completion of information, a summary is presented in this section. The software used during this study is SimDShip which is a hull form optimization tool developed by Chi

Yang's group at George Mason University (GMU) which was sponsored by the Office of Naval Research (ONR).

Hull form optimization problems require three main components. The first is the objective function which is typically the hydrodynamic performance of the hull. Common objective functions are to reduce drag at specific speed(s) or to reduce ship motions in a seaway. This paper only considers the reduction of resistance in calm water as the objective functions. The second requirement for a hull optimization problem is design variables. Design variables are aspects of the hull form that will be changed such as length or fullness. In this study the design variables are the y-direction location of control points in the Non-Uniform Rational B-Spline (NURBS) surface that represents the hull. The third requirement for a hull optimization problem is the design constraints. Typical design constraints are to maintain a design displacement, avoid becoming prohibitively expensive to construct, and not change hull geometry around constraining ship features such as a sonar dome.

1.2 Surrogate Models

The primary driver of this research is to determine the applicability of Convolutional Neural Networks (CNNs) for approximating the ship calm water resistance problem. The applicability of the CNNs as a surrogate model will be measured as a comparison to two other surrogate model methods with three primary metrics:

1. Reduced Error from Approximation
2. Reduced Training Time & Cost
3. Flexibility & Functionality

The two other surrogate modelling methods, Radial Basis Function (RBF) and Feed Forward Neural Networks are also considered and discussed in this section.

A surrogate model is any method that relates an input variable(s) to output variable(s). This relationship between inputs and outputs is determined via training process with a set of training data. The training process for all of the surrogate models considered focuses on optimizing a set of model weights to achieve a desired accuracy. Generally, all surrogate models rely on the same basic steps to be created:

1. Generate a set of data that is reprehensive of the problem being trained
2. Train the model with the dataset
3. Validate the model

The primary difference between the surrogate methods is the fundamental way the models relate the inputs and out. In addition, the CNN uses a different set of input variables than the FFN and RBF methods, which leads to the added flexibility and functionality.

1.2.1 Radial Basis Function (RBF)

The Radial Basis Function (RBF) based surrogate models represent a conventional method for hull form optimization. There is significant literature on the use and success of the RBF based surrogate models in the hull optimization problem as well as the use of other conventional surrogate model methods such as the Kriging model (F. Huang, L. Wang and C. Yang) (Wang, Huang and Yang) (Huang and Yang, Hull Form optimization of a Cargo Ship for Reduced Drag). For the purposes of this study only RBF based models are considered as conventional surrogate models. The mathematical details of the RBF model are explained in detail in (Huang, Wang and Yang, Hull Form Optimization for Reduced Drag ad Improved Seakeeping Using a Surrogate Based Method) (Huang and Yang, Hull Form optimization of a Cargo Ship for Reduced Drag).

For the purposes of this study all conventional surrogate models in the hull optimization problem are generalized with two main similarities. The first similarity is that the conventional models use the design variables as the inputs for the approximate model. For the purposes of just approximating the objective function in an optimization problem, this is an appropriate and efficient setup of the model. However, only including the design variables in the model has the consequence of needing to develop a new approximate model anytime the design variables change.

The second similarity is that all models deemed conventional use weights within a system of equations. The other two surrogate model methods used in this study train weights that are used within a sequential model that has user-defined architecture. This topology of the FFNs and CNNs are consequential to the accuracy of the models and represent a risk to model robustness and prediction that Kriging and RBF methods do not need to consider.

1.2.2 Feed Forward Network (FFN)

The Feed Forward Networks (FFNs) represent the simplest type of artificial neural networks (ANNs) and consist of an input layer, hidden layers, and output layer. These layers represent the quintessential topology that makes up a neural network. Figure 2 shows an example topology for a Feed Forward Network designed for regression. Each of the white circles represents a node that poses an activation function. The activation function is determined by the values fed into it via the weighted connections to the previous layer. The value of the weight on the connections is what is trained in an ANN.

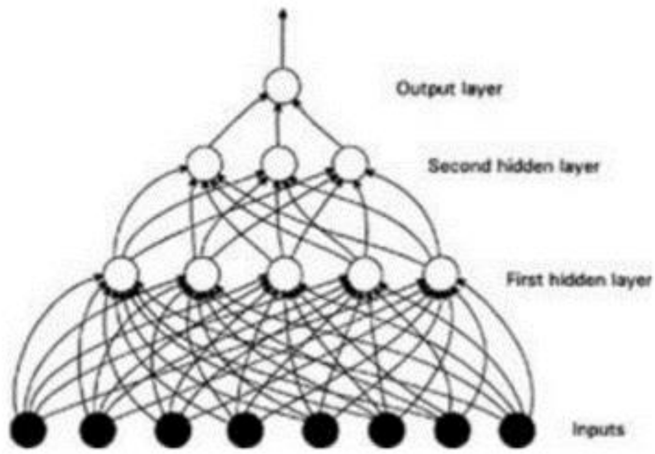

Figure 2. Neural Net Topology Example

There is a more detailed description of Feed Forward Networks and ANNs in general in the authors' previous work. In addition, there is a sensitivity study presented in that technical paper that shows limited differences in accuracy in relation to sample size and topology. However, there is significant increases to the cost of training as sample size increases (Shaeffer and Yang, Application of Machine Learning to Early-Stage Hull Form Design) (Shaeffer, Application of Artificial Neural Networks to Early-Stage Hull Form Design).

1.2.3 Convolutional Neural Network (FFN)

Convolutional Neural Networks (CNNs) are a more sophisticated type of ANN that is designed to capture spatial information. The characteristic component of a CNN is the convolutional layers which uses a kernel to capture the relationship of information spatially as depicted in **Figure 3**. Typically CNNs are used with images and the spatial information that the kernel is reading in the numerical Red, Blue, and Green (RBG) values of individual pixels.

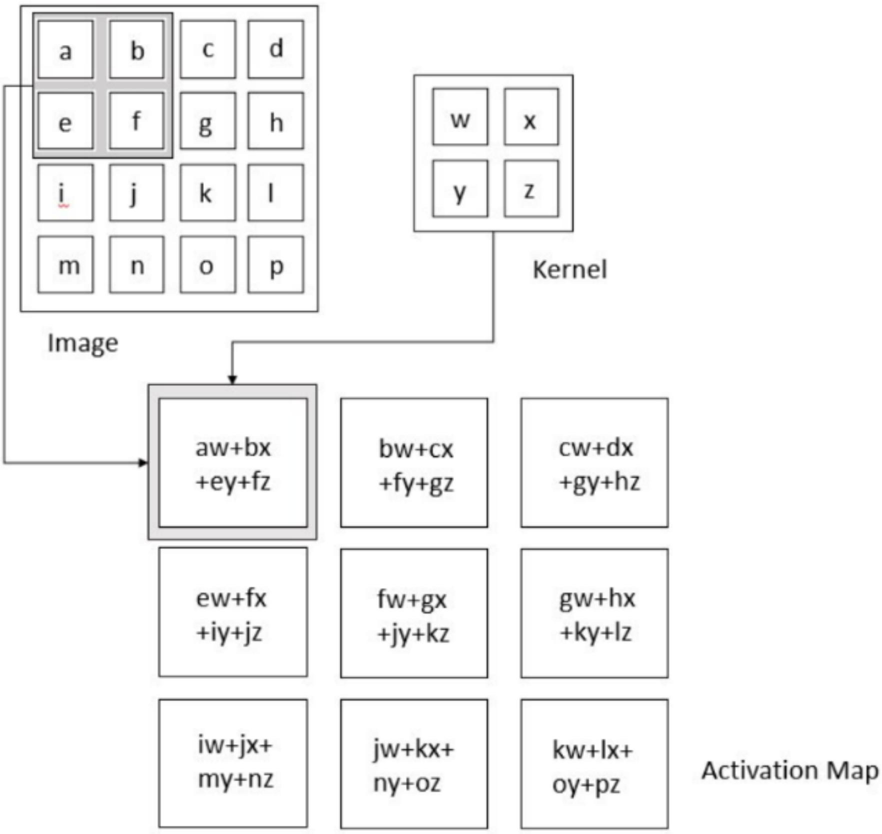

Figure 3. Convolutional Layer Example

Primarily CNNs are used with image classification tasks and the convolutional layers will feed into pooling and flattening layers that are designed to compress the information from the convolutional layers into a format that can be fed to a FFN. An example of a typical CNN topology is shown in **Figure 4**. (5)

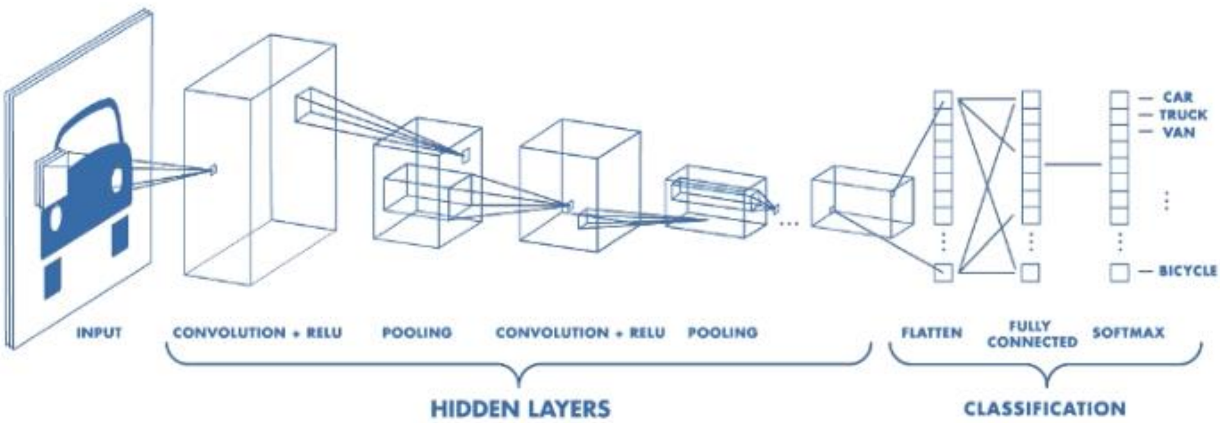

Figure 4. CNN Example Topology

Beyond the inclusion of convolutional, pooling, and flattening layers there is no significant difference between the fundamental workings of a CNN and FFN. The training algorithm is the same except there are typically significantly more training parameters, weights, in a CNN than FFN. Notably, it is less common for CNNs to be used as a regression model than as a classification model.

2. METHODOLOGY

The main objective of this study was to identify how a CNN based surrogate model would compare to the conventional surrogate modelling strategies. A major component of the accuracy of any surrogate model is the goodness of the training dataset. So in order to measure the difference in accuracy versus cost, a sensitivity study with respect to sample size was conducted.

The requirement for training sample size is often a function of the number of design variables for conventional surrogate models. Another aspect of this sample size sensitivity study will include how more design variables impacts sample size sensitivity.

Previous work by the authors found that topology of FFNs did not have significant impact on the accuracy of the regression. Based on this research, it was determined that CNN topology was not going to be varied systemically in this study. Based on the success of previous works and fact the FFNs share fundamental similarities with CNNs and RBF based models, FFNs are included in this study.

Another objective of this study is to determine the secondary benefits that could be obtained by using a CNN as the surrogate model. One benefit of the CNN surrogate model as it is implemented in this study is that there is no requirement to create a new model if the design variables change. For this reason, an aspect of this study is determining how good a CNN surrogate model can be at approximating multiple design variable setups. As a part of this study, the same CNN will be trained and retrained with different design variable setups.

2.1 Dataset Development

Based on previous work with SimDShip and an abundance of research available, the Series 64 hull form was chosen as the base hull in this study. Two different design variable setups were used in this study – a 4 design variable and 9 design variable setups, as illustrated in Figure 5. In both setups all of the NURBS control points along the edge of the hull surface were set as fixed (blue) and the remaining interior NURBS control points (yellow) were set to move dependent on the design variables (cyan). All design variables are set to be movable in the y direction (port/starboard) with a minimum and maximum constraint of approximately ±50% of the original y value.

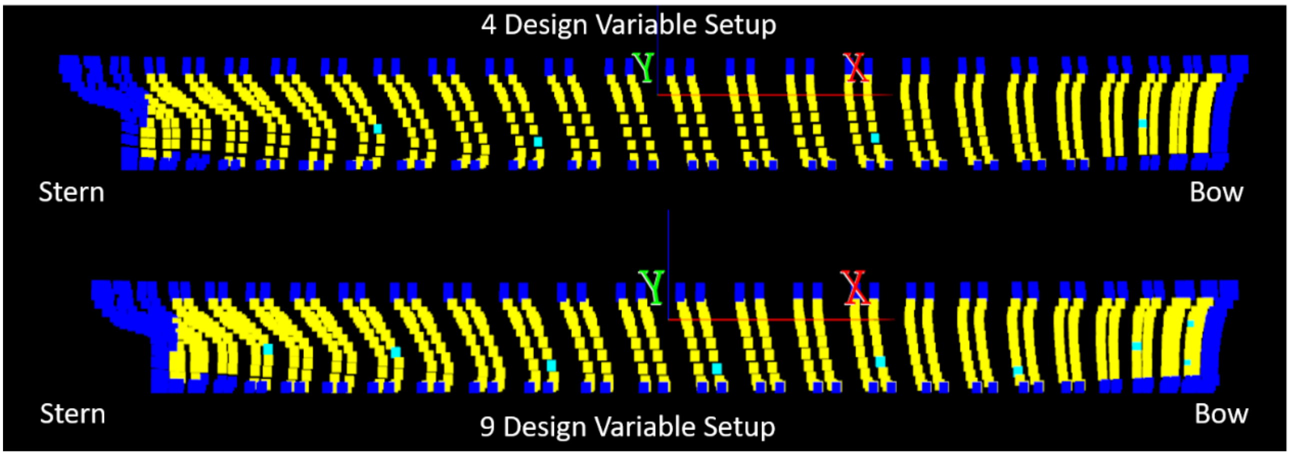

Figure 5. SimDShip Design Variables for the 4 and 9 Design Variable Cases

The datasets have 3 objective functions that have surrogate models. The first is a cost function that approximates the displacement of the hulls. One of the constraints of the optimization is to not change displacement by more than 1%. The other two objective functions are for the resistance at a Froude number of 0.28 and 0.35.

Three sample sizes were used for each design variable setup per the values in Table 1. The samples were created using a Latin Hypercube method within SimDShip. The conventional surrogate model generally requires approximately 10 samples per design variable. In addition to the training datasets a test dataset was created for each design variable setup with 10 test samples for the 4 design variable setup and 25 for the 9 design variable setup.

Table 1. Training Sample Sizes

	Small	**Medium**	**Large**
4 DV	20	40	100
9 DV	100	250	500

2.2 Radial Basis Function Based Surrogate Model

As shown in Figure 6, there are four main steps to build a surrogate model in SimDShip.

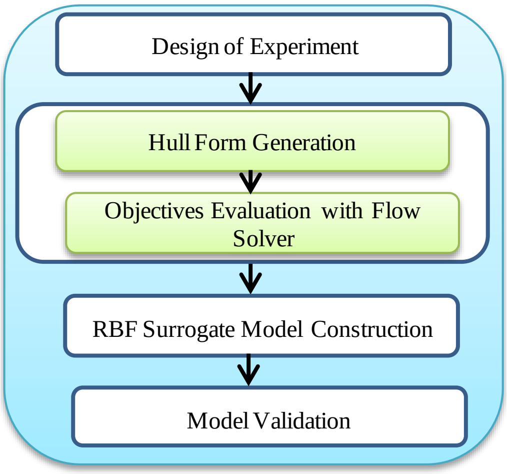

Figure 6. Flowchart of the RBF Surrogate Model Construction

The design of experiments (DOE) is a strategy for selecting sample points in the design space that aims at maximizing the amount of information acquired (Giunta, Wojtkiewicz and Eldred). There are a number of DOE techniques in the literature, such as factorial designs (Giunta, Wojtkiewicz and Eldred), Latin hypercube sampling (LHS) (McKay, Beckman and Conover). In SimDShip, LHS technique is used to generate sample points in the design space. Once the sample points in the design space are generated, the candidate hull forms are produced using the surface modification tool within SimDShip (Chi and Huang). In the present study, various samples shown in Table 1 are generated for 4 design variable case and 9 design variable case, respectively.

A practical steady ship flow (SSF) solver based on the Neumann-Michell (Noblesse, Huang and Yang; Yang, Kim and Delhommeau) potential flow theory is used to evaluate the resistance for the construction of the surrogate model. In addition to evaluating the resistance at two given Froude numbers (0.28 & 0.35) for each hull form within a given sample, the displacement of each hull form is also calculated and recorded. Three RBF-based surrogate models can then be constructed for a given sample to predict the displacement and resistance at two Froude numbers (0.28 & 0.35), respectively. The cross validations for these surrogate models are performed afterwards. The detailed formulation for RBF-based surrogate model can be found from author's previous work (F. Huang, L. Wang and C. Yang).

2.3 Feed Forward Network (FFN) Model

The simple FFN that was used for this study consists of 3 hidden layers and batch normalization and dropout layers in between to improve training. The plot of the architecture for the FFN of the 9 DV problem is shown in **Figure 7**. Notably, the Architecture for the FFN is dependent on the design variables.

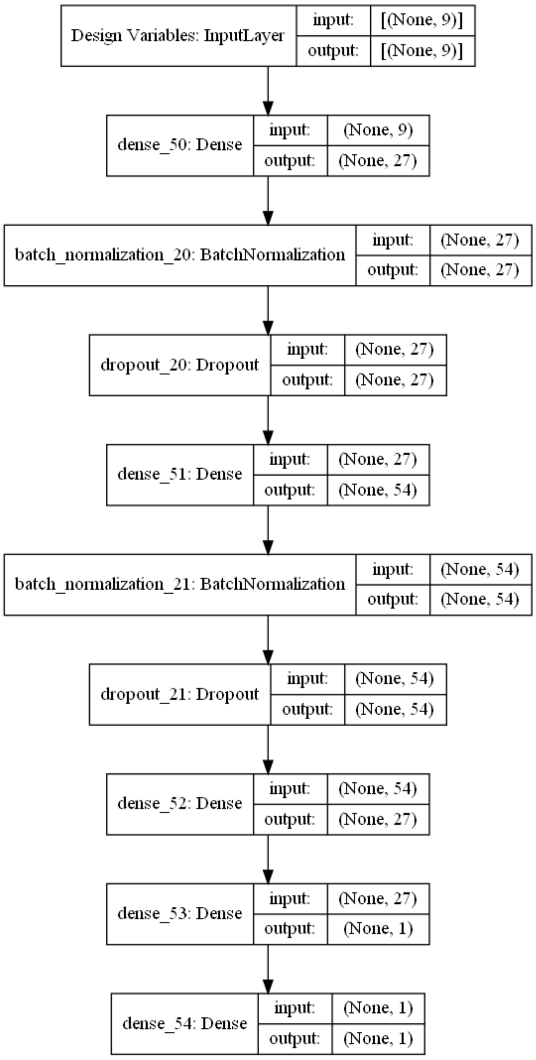

Figure 7. FFN Architecture from TensorFlow

2.4 Convolutional Neural Network (CNN) Model

CNNs are usually used with images in which the pixels' RBG values are input into the ANN. In order to input the hull form geometry into a CNN, the NURBS control points were used instead of pixels and the (X, Y, Z) values of each point were used instead of RBG values. The architecture of the CNN is shown in **Figure 8** in TensorFlow format.

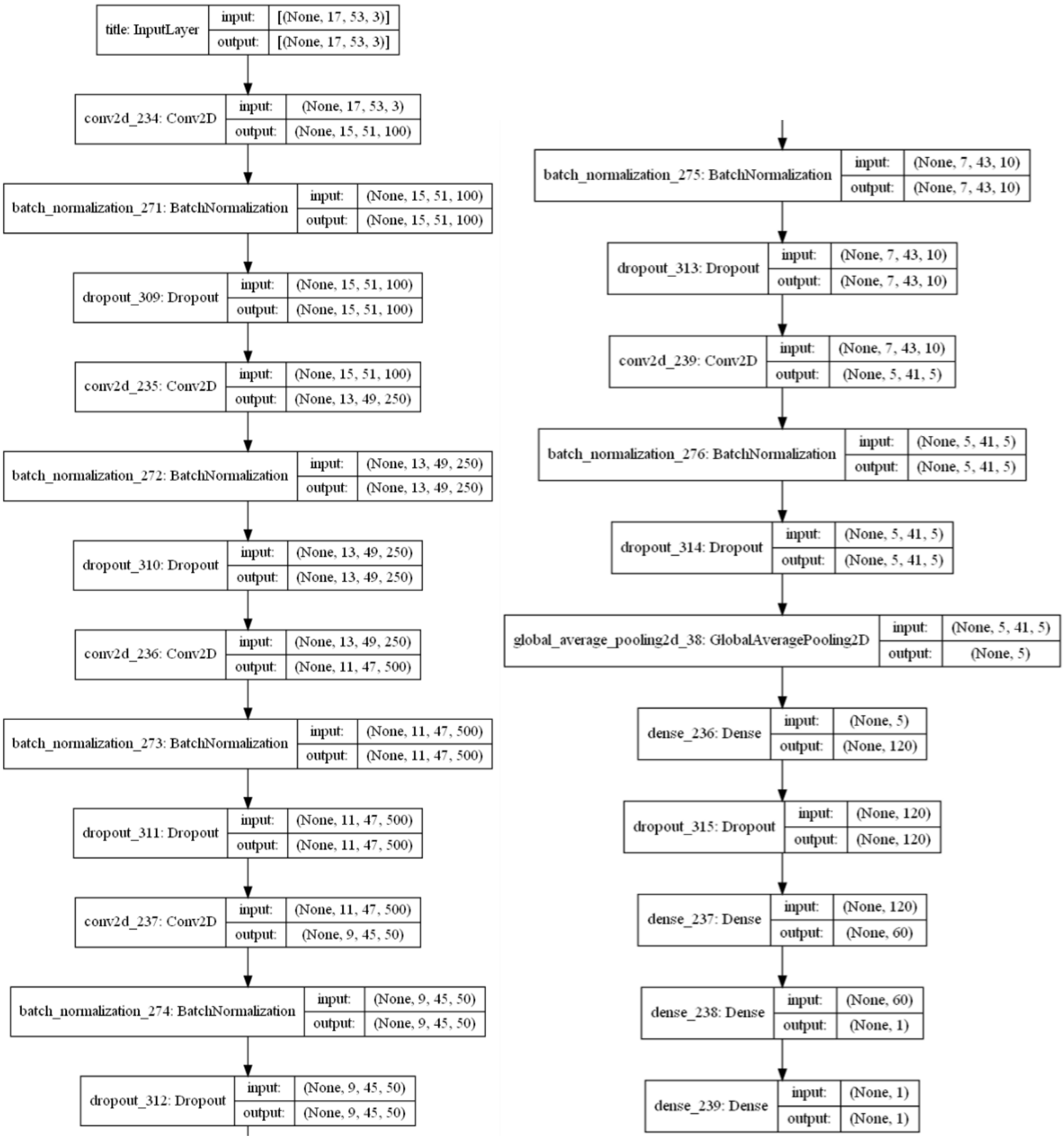

Figure 8. CNN Architecture from TensorFlow

2.5 Long Term Memory Strategies

The purpose of long term memory strategies is to determine how effective a CNN can be when applied to a set of design variables it was not originally trained on. This study will consider three strategies and all of them will be tested on both the 4 design variable and 9 design variable setups.

The first strategy represents the best case scenario in which the CNN can be trained on a single set of data that covers many design variables and be effectively used for any subset of those design variables. In this study, the large dataset is the 9 design variable set and the subset is the 4 design variable sets. This is the preferred strategy since it requires the least amount of user input and the least amount of training.

The second strategy is slightly less desirable, in which it requires some retraining. This strategy is to initially train on the large dataset (9 design variable) and then retrain with just the small dataset (4 design variable). The possible problem with this strategy is that the CNNs can experience memory loss. This strategy will only be effective if the CNN remains acceptably accurate for the original training dataset. In addition, this method requires more training than the first strategy.

The last strategy is the most costly and requires the retraining of all of the training data combined. There is still some benefit if this strategy proves successful in which the CNN would be able to remember and recall past design variable setups easily. In addition, there are some time savings from training an existing CNN as opposed to training a new CNN.

3. RESULTS

All of the results will be in the form of a cross validation plot with the actual resistance predicted with SSF in SimDShip on the X-axis and the approximate model prediction on the Y-Axis. The resistance values shown in the plot are normalized values of the total drag. The dashed line on the plots represents no error from the approximate model.

3.1 Sample Size Sensitivity Study

For each sample size there are 9 plots, 3 plots for each of the objective function variables: displacement, resistance at a Froude number of 0.28, and resistance at a Froude number of 0.35. Each of these three plots are shown for each of the 3 surrogate model types. Each plot shows the cross validation plot and the Mean Square Error (MSE) for the test data. The resulting plots are in Figure 9 - Figure 14.

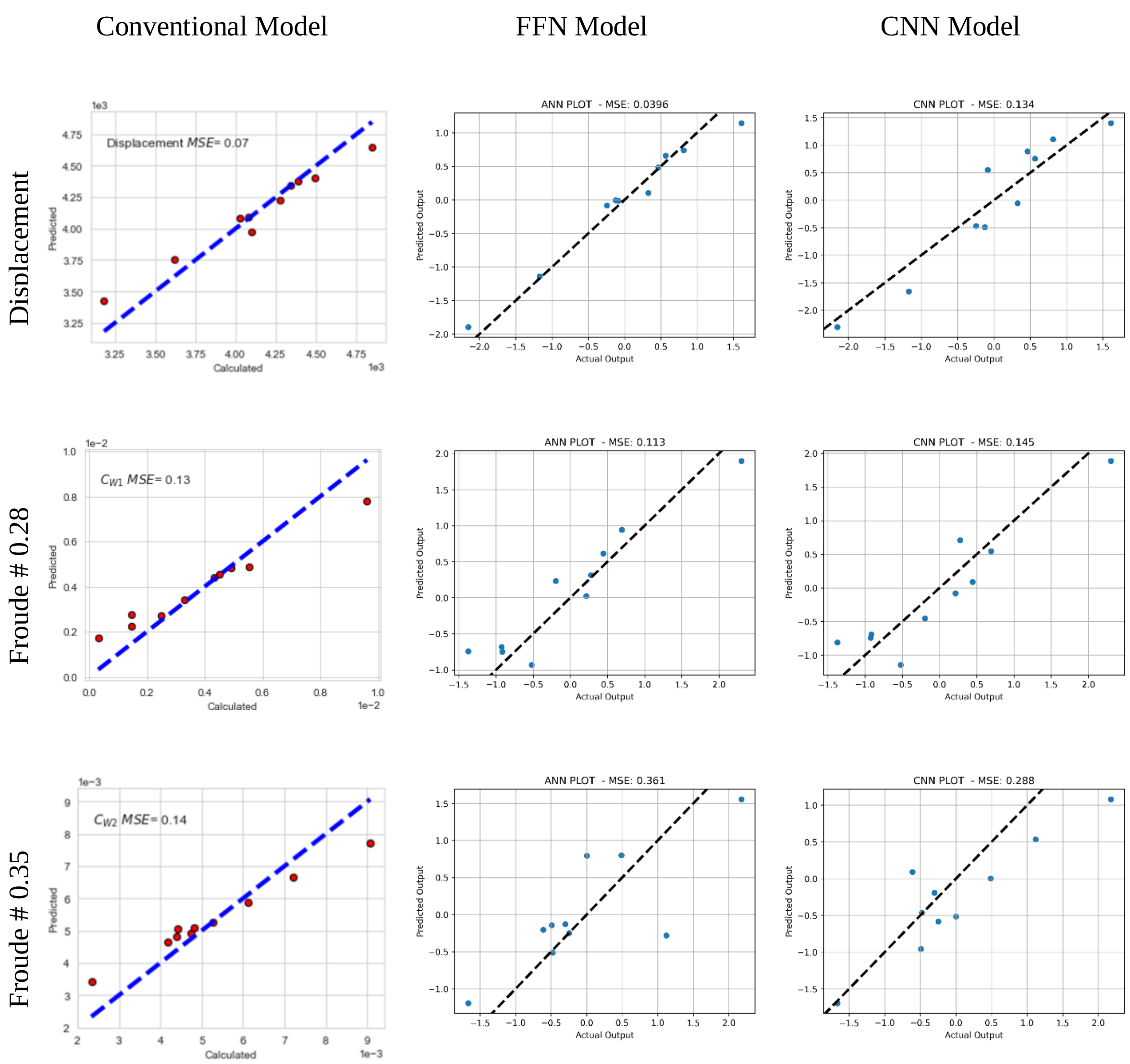

Figure 9. 4 Design Variable - 20 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

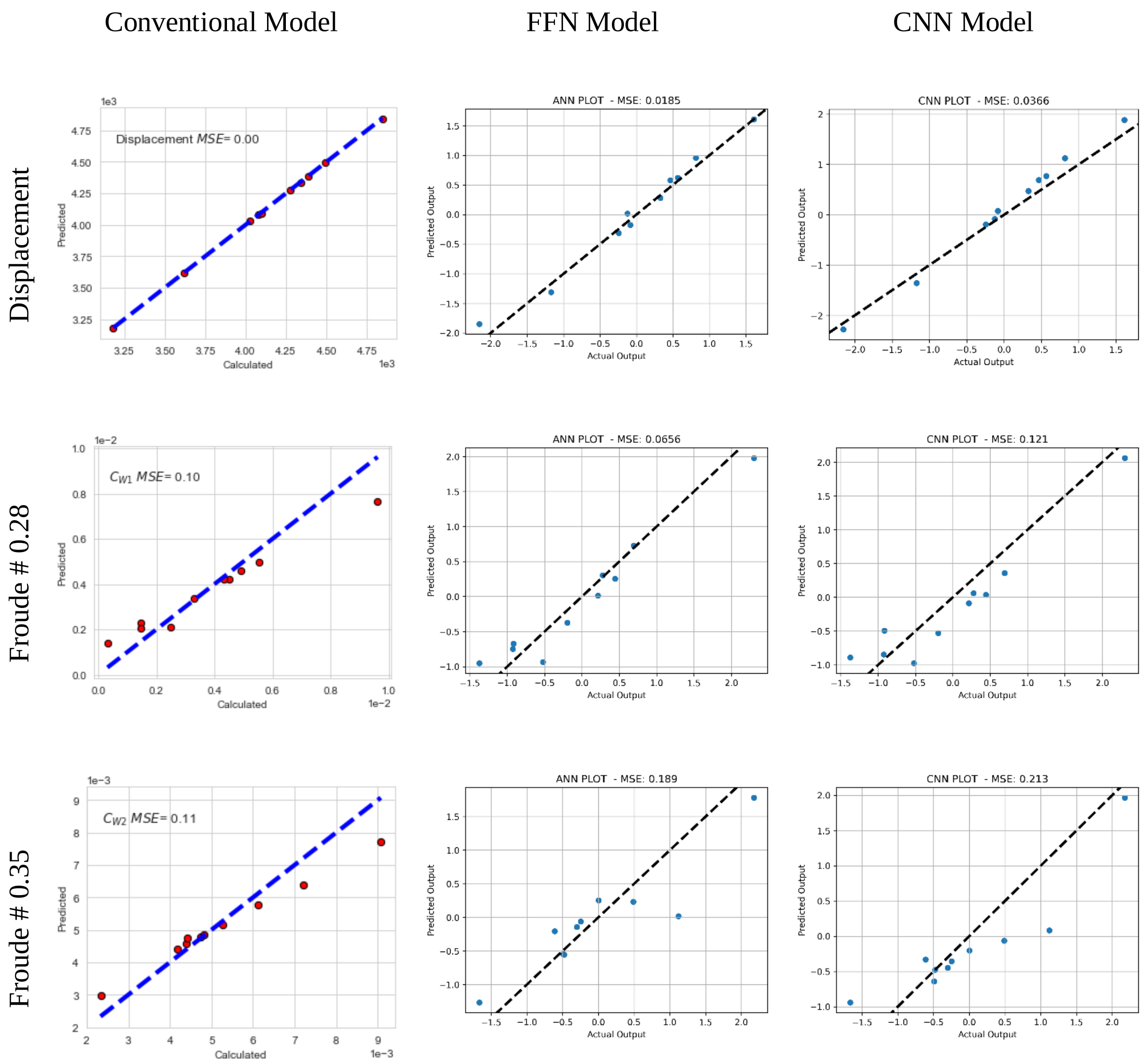

Figure 10. 4 Design Variable - 40 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN

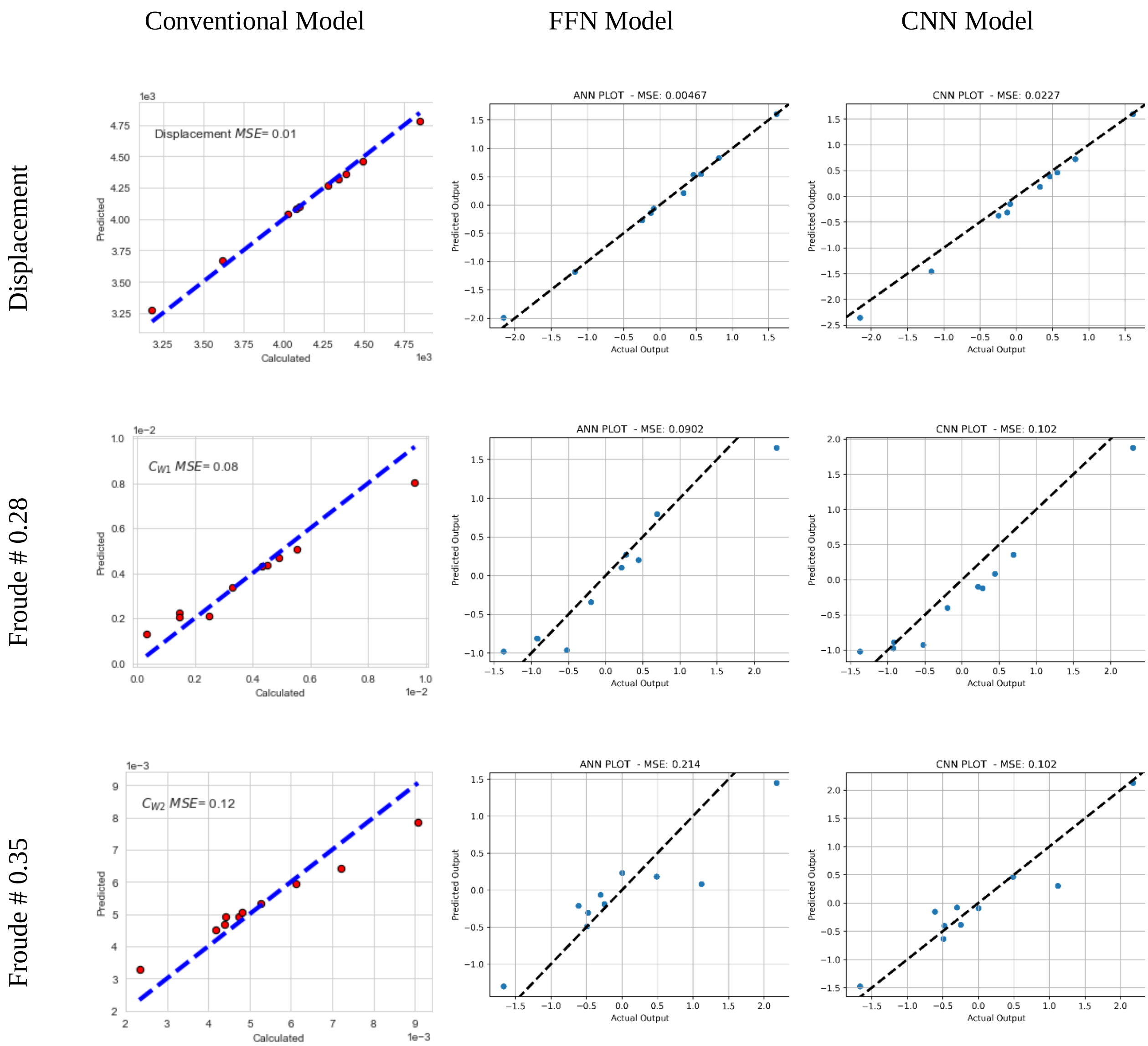

Figure 11. 4 Design Variable - 100 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

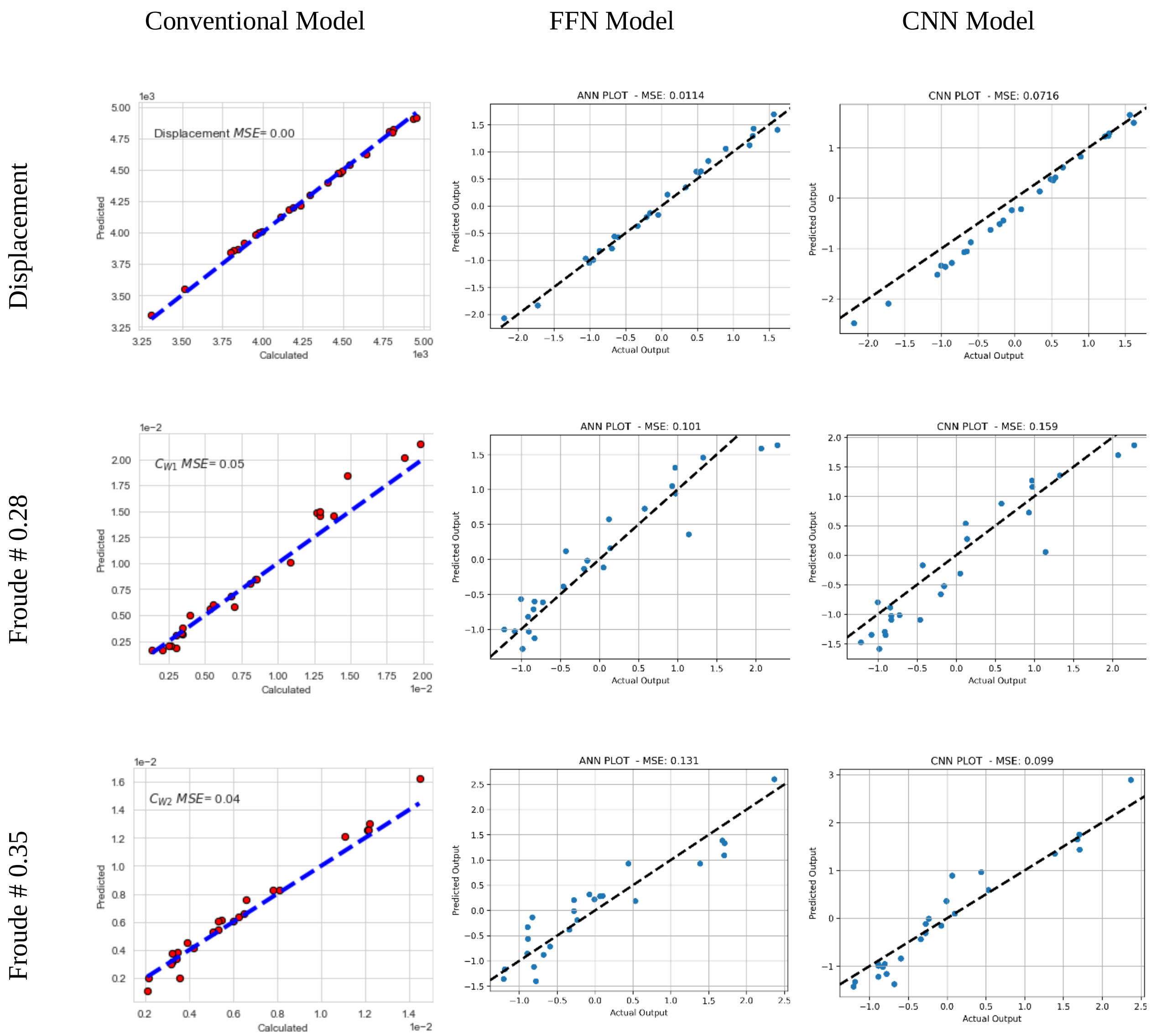

Figure 12. 9 Design Variable - 100 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

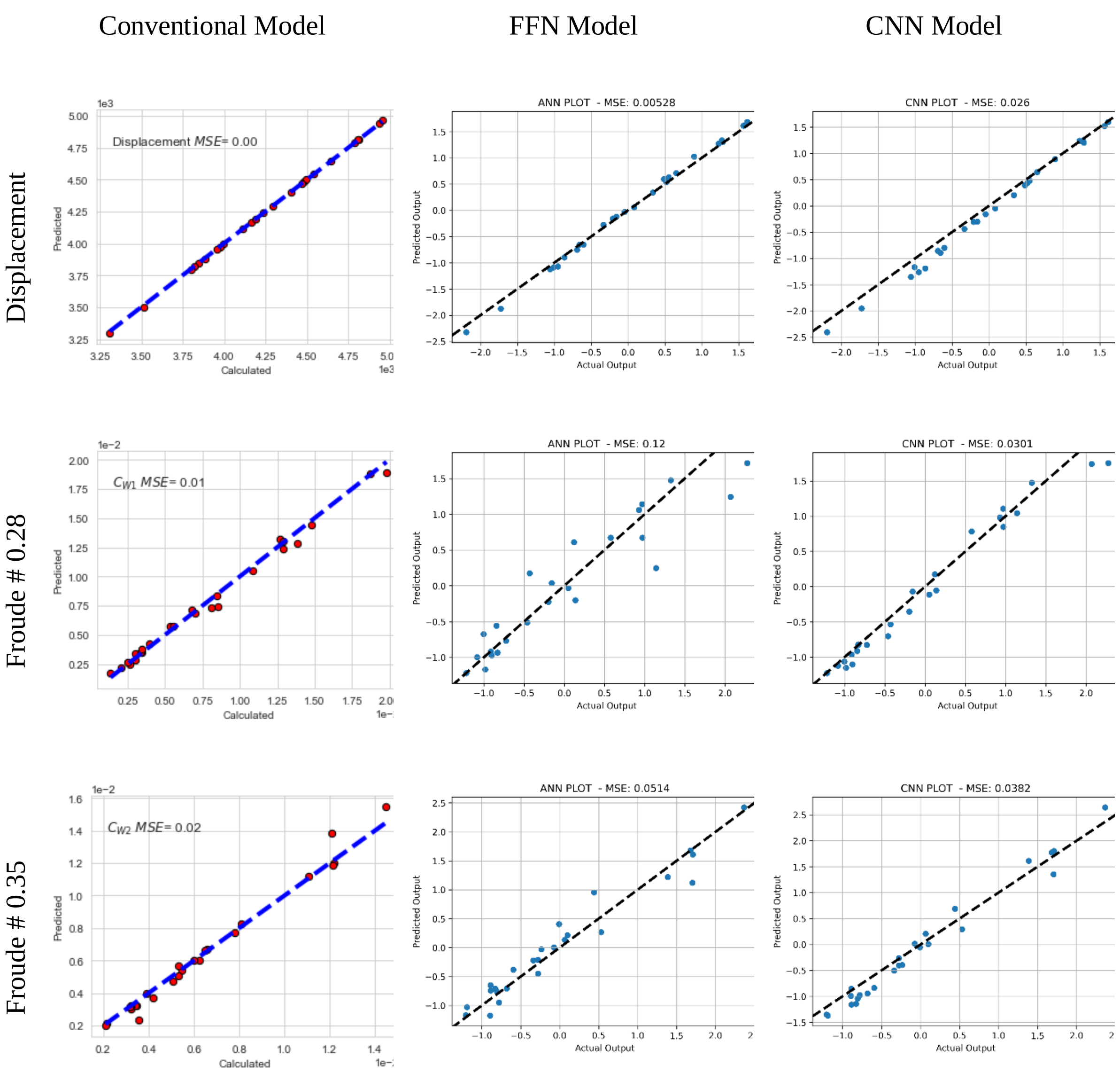

Figure 13. 9 Design Variable - 250 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

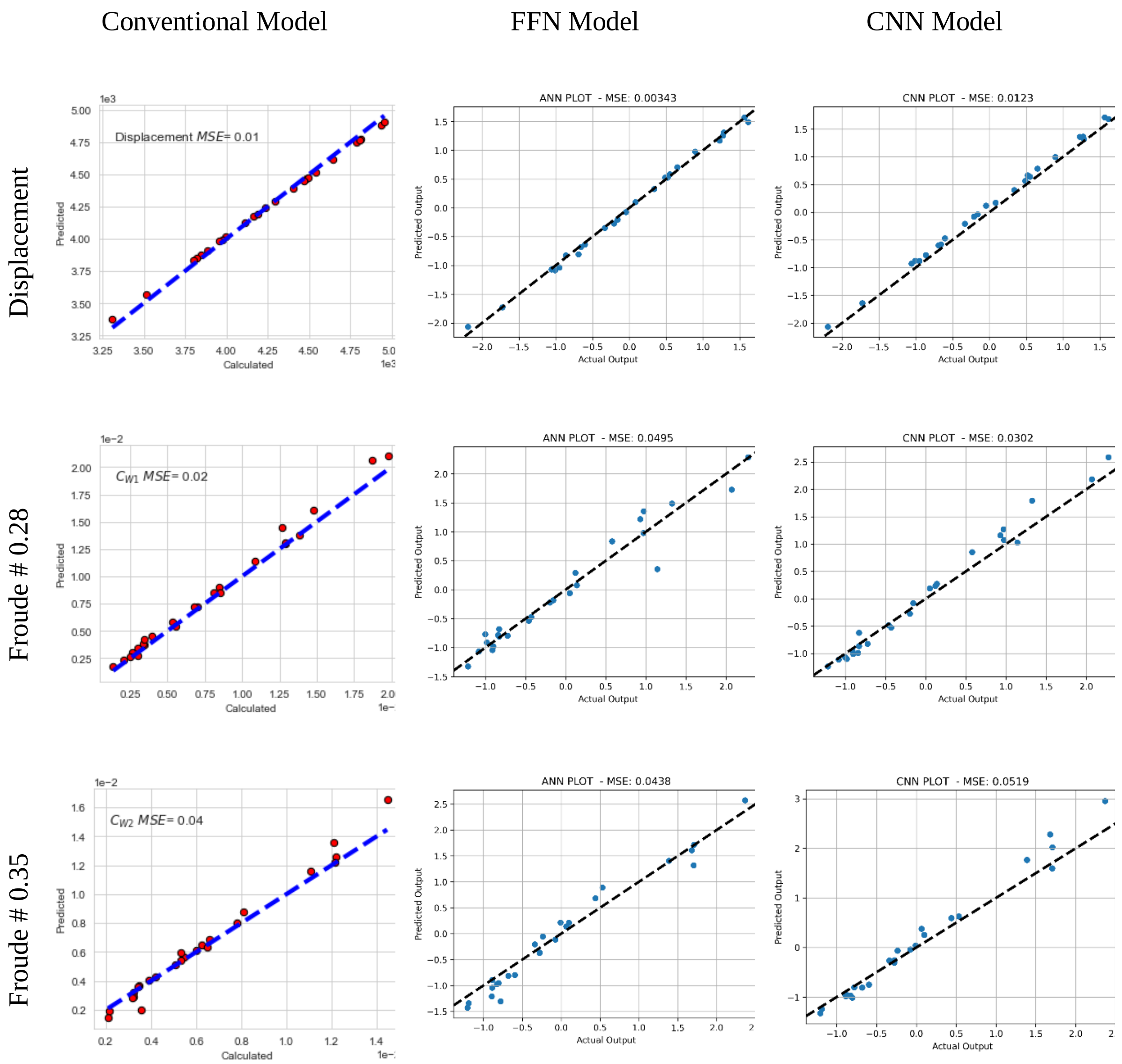

Figure 14. 9 Design Variable - 500 Samples Training - Actual Output (X-axis) equal to the objective function labelled per row calculated in SimDShip. The objective function in the first row is dimensional and the objective function in both ANNs are normalized. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

The results generally show that all of the surrogate models, including the conventional model, are sensitive to the training sample size. The minimum sample size required for LHS depends on several factors, including the number of design variables, the desired level of accuracy, and the complexity of the distribution being sampled. In general, as the number of design variables increases, the required sample size also increases to maintain an acceptable level of accuracy. There is no one-size-fits-all answer to the sample size question since the minimum sample size required for LHS with N design variables can vary depending on the specific application and the requirements of the problem. However, a common rule of thumb used in practice is to have at least 5 to 10 times the number of design variables as the sample size. For example, for a 9 design variable problem, a minimum sample size of 45 to 90 would be recommended. However, this is only a rough guideline and more samples may be required for some applications to achieve the desired level of accuracy. Based on authors' previous research, the sample size is about 10 times the number of design variables for constructing the RBF-based surrogate model for evaluating resistance.

The FFN and CNN models both show this expected performance however the CNN is notably more erroneous at smaller sample sizes suggesting a greater sensitivity to sample size than the other two models.

Considering the CNN has an order of magnitude more trainable parameters in the model, it is conceivably reasonable that the CNN would require large sample sizes. Once the sample sizes exceeded the general rule of 10 times the number of design variables the performance of the three surrogate models is reasonably comparable. There does not appear to be any serious loss of regression performance when using the CNNs or the FFNs.

3.2 Long Term Memory Study

Based on the results in the previous section, it is proven that the CNN models can offer the same level of fidelity as the conventional models but the CNNs also offer other advantages that make them more useful than the conventional counterparts. A major proposed benefit of the CNN based surrogate model is the potential for the surrogate model to learn the design problem over-time and retain memory of previous study cases for the improvement of future studies. The conventional surrogate models are used in an existing process that essentially requires the generation of an entirely new surrogate model for each new hull form design case. For example, if an additional control point is selected to be added as a design variable, all previous hull form design optimizations would be unusable and an entirely new set of data points would need to be generated and a new model trained.

Three strategies are considered for using CNNs to develop long term memory. The first strategy is to train a surrogate model on a single, reasonably large, and varying dataset to cover all possible configurations of the design space. The second strategy is to train the CNN-based model sequentially on every new set of data, with surrogate model learning the new set of data and ideally retaining the old information. The last strategy is to train the CNN on all possible data points, concatenating every dataset into a new master dataset and continuously training the CNN on an increasingly larger dataset. The first strategy represents the ideal example because it is the most computationally and user-in-the-loop efficient. The second strategy is a possible compromise if strategy one fails to provide good results but this strategy will likely be prone to what is call catastrophic forgetting. Catastrophic forgetting is a known performance of ANNs when they are retrained on new datasets and this performance generally shows a complete loss of all previous information. The final strategy is the most comprehensive but also the least-efficient, but still offers more value than the conventional surrogate models. Figure 15 through Figure 18 show the regression accuracy of CNNs tested with three different strategies for using this type of surrogate model to develop long term memory. The plot are identical to the previous section but the tables instead show the accuracy of the same model on two different sets of test data.

CNN Performance on 4DV 20 Sample Test | CNN Performance on 9DV 25 Sample Test

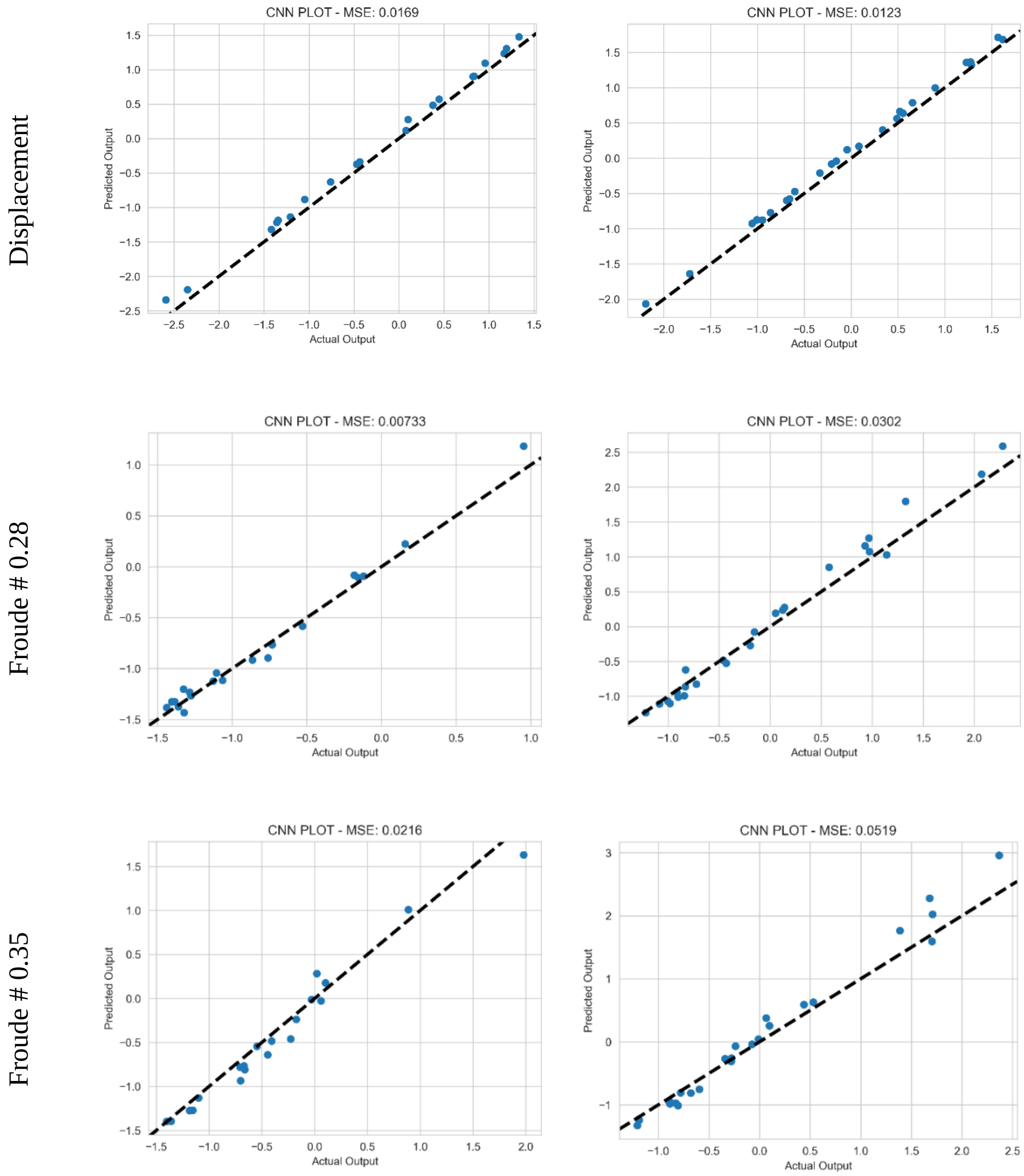

Figure 15. Long-Term Memory Study - CNN Trained with 9DV, 500 Sample Dataset - Actual Output (X-axis) equal to the normalized objective function labelled per row calculated in SimDShip. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

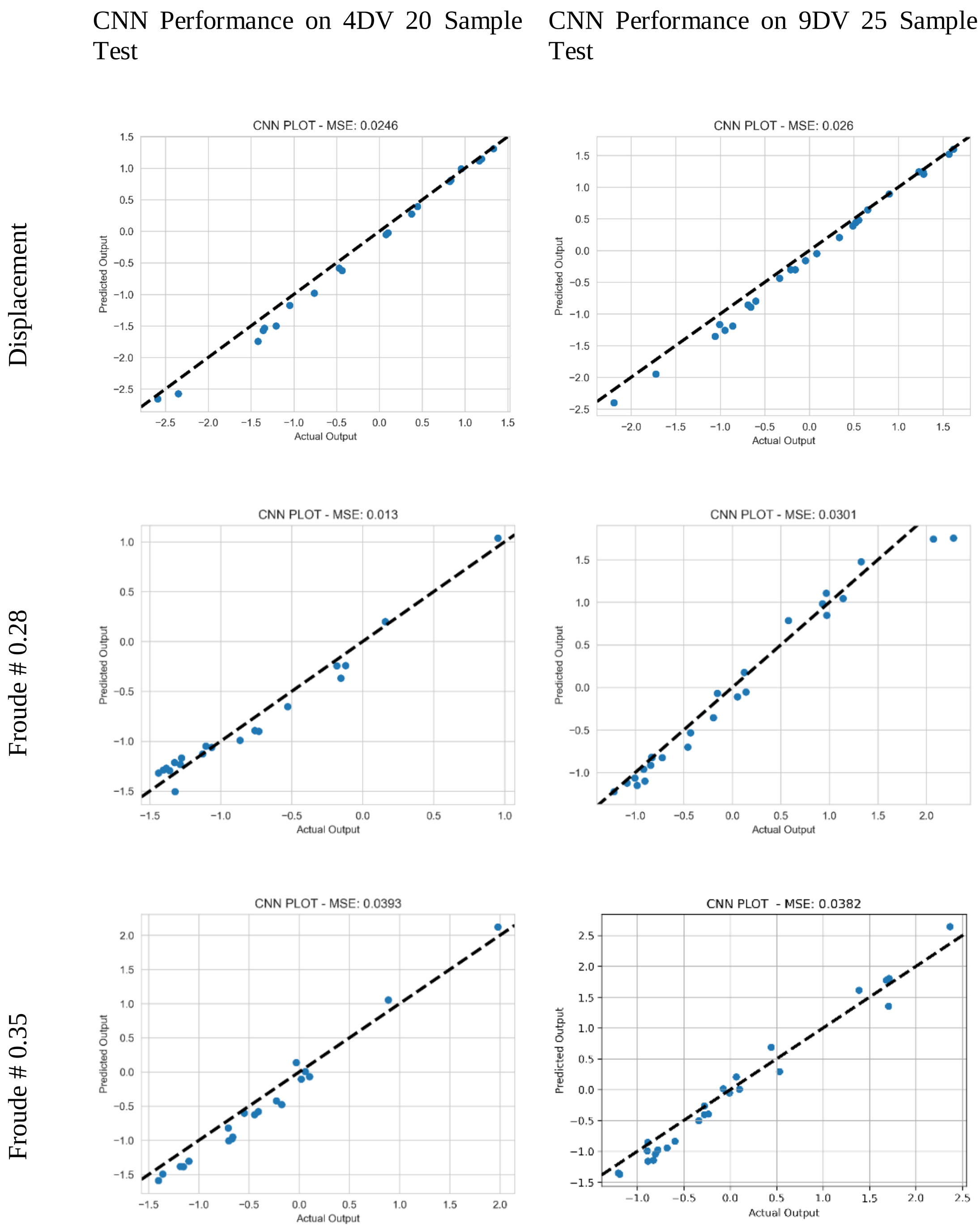

Figure 16. Long-Term Memory Study - CNN Trained with 9DV, 250 Sample Dataset - Actual Output (X-axis) equal to the normalized objective function labelled per row calculated in SimDShip. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

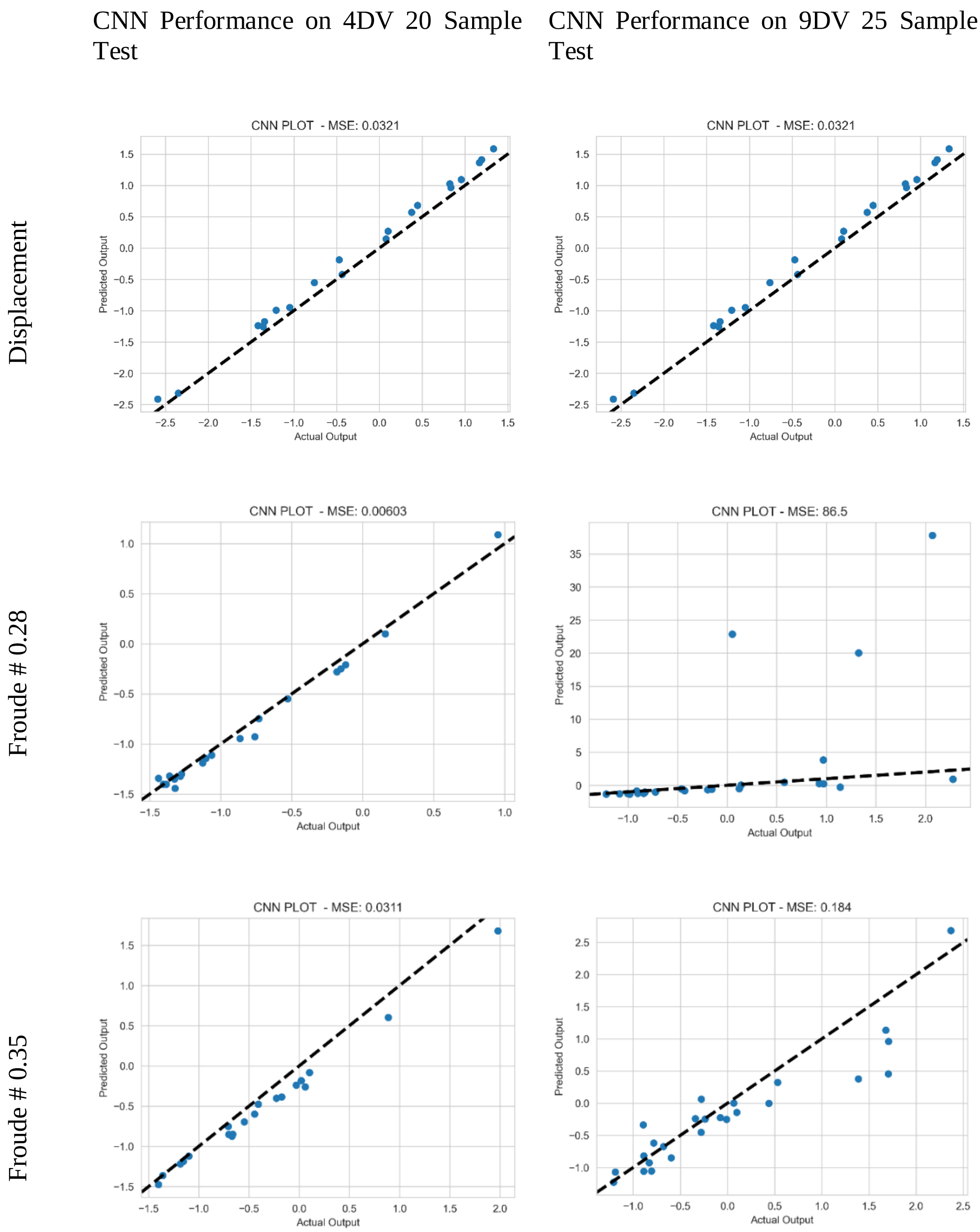

Figure 17. Long-Term Memory Study - CNN Trained with 9DV 500 Sample Dataset and 4DV 100 Sample Dataset Sequentially - Actual Output (X-axis) equal to the normalized objective function labelled per row calculated in SimDShip. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

CNN Performance on 4DV 20 Sample Test　　CNN Performance on 9DV 25 Sample Test

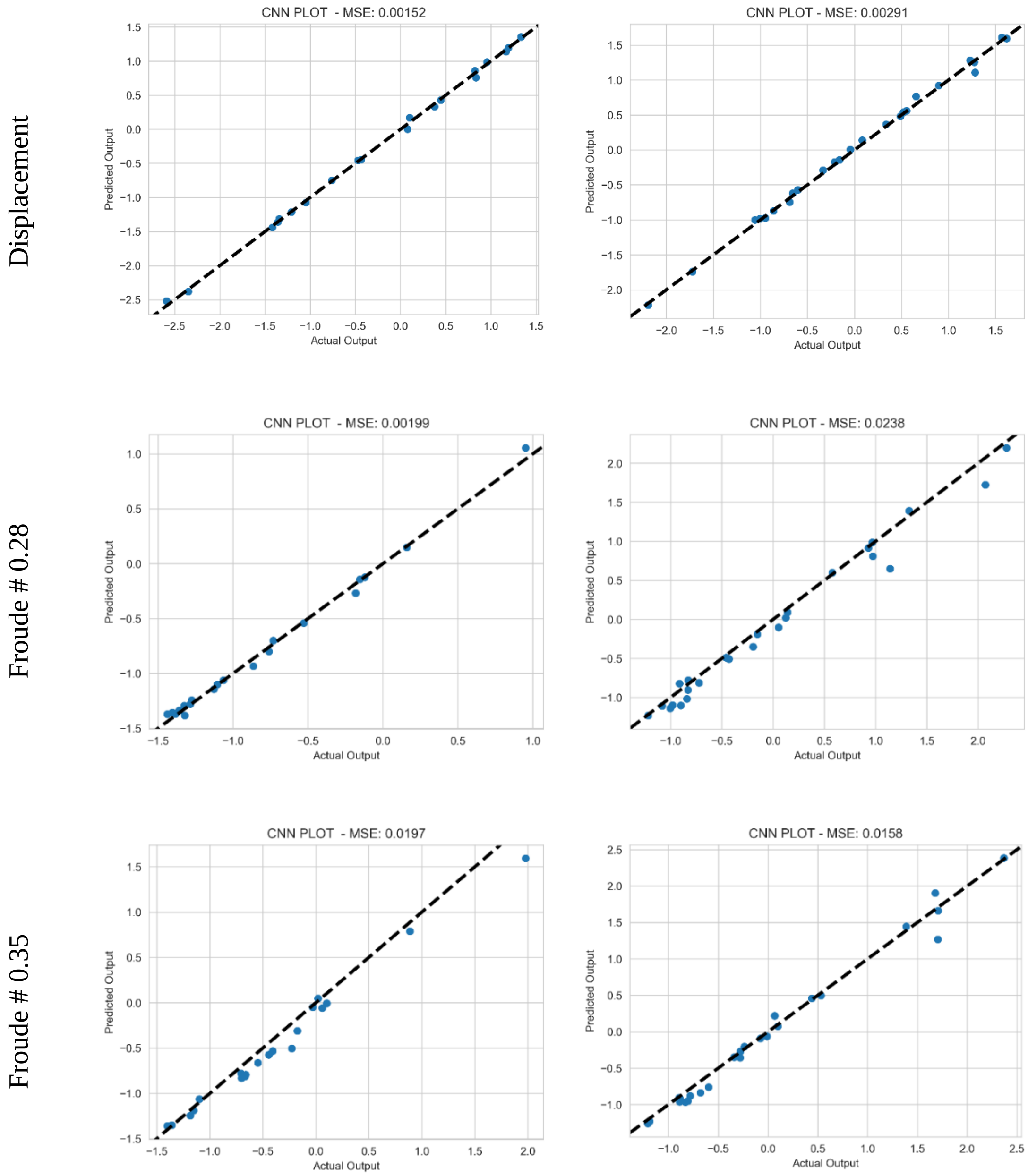

Figure 18. Long-Term Memory Study - CNN Trained with 9DV 500 Sample Dataset and 4DV 100 Sample Dataset Together - Actual Output (X-axis) equal to the normalized objective function labelled per row calculated in SimDShip. Predicted Output (Y-axis) equal to the respective predicted objective function from the ANN.

Figure 15 shows the results of the first strategy discussed in the section on long term memories. In this strategy, the CNN model was only trained on the 500 samples of 9 design variable (DV) and then tests on both the 4 DV and 9 DV test data. The ideal results would show good agreement with both the 9 and 4 DV test datasets. The results for the 9 DV test data show good accuracy as expected. The results for the 4 DV test data are comparable to the 9 DV test data which is an excellent sign for the feasibility of using strategy 1 to develop a comprehensive CNN-based surrogate model. Figure 16 shows the same strategy but trained with the 250 sample 9 DV dataset and the results are comparable those shown in Figure 15 with 500 sample data. This is a positive sign that this strategy is not sensitive to sample size as long as the minimum is met as discussed in the previous section.

Figure 17 shows the results for the second strategy, in which the CNN model was initially trained on the 9 DV training data, and then retrained on only the 4 DV training data. The displacement prediction performs well for both the 9 and 4 DV test data, however the resistance approximations show unacceptable error for the 9 DV test data. This result suggests the training strategy currently used in this study is causing severe memory loss known as catastrophic forgetting. Future efforts should consider some strategies that are used to mitigate catastrophic forgetting within the CNN architecture and training process.

Figure 18 shows the results of training the CNN model with the 9 DV data initially and then retraining briefly with the combined data and randomized date of the 4 and 9 DV training datasets. Because the CNN is already trained on the 9DV significantly fewer training iterations are required for the model to reach a steady-state with the combined dataset. This strategy shows relatively good accuracy for both the 4 DV and 9 DV test datasets suggesting that the CNN is capable of learning both at the same time. The accuracy of this strategy is not surprising but is important as an option in future studies.

3.3 CNN Model Flexibility and Functionality

The CNN surrogate models also offer other benefits not measured in this study. One primary benefit is that the partial derivative of the input layer can be easily computed via the method used in the previous work by these authors (Shaeffer, Application of Artificial Neural Networks to Early-Stage Hull Form Design) (Shaeffer and Yang, Application of Machine Learning to Early-Stage Hull Form Design). The partial derivative of the input layer of the CNN model is equivalent to predicting the relationship of every control point location to the objective function.

A primary driver in pursuing this outcome is that hull form NURBS surfaces are typically made of 100s of control points, each of which can be specified to move in 3 dimensions and two directions in each dimension. This results in a number of possible variations that is difficult to prioritize what the preferred design variables should be in a hull form optimization without prior experience. By taking the gradient of the control points to the objective function, the prioritization of the control points can be identified with little to no user input.

Figure 19 shows the graphic output of the gradient of the control point Y locations of the base hull to the resistance at a Froude # of 0.28. The red control points represent points that will reduce drag when the Y value is decreased and the green control points will reduce drag when the Y value is increased. The brighter red or green a control point is, the more that control point will impact the objective function.

Figure 19. Partial Derivative of control points Y location with respect to resistance

In addition to providing feedback to the naval architect executing hull optimization, the partial derivative of the control points with respect to the objective function can also be used with a direct search algorithm. Conventional hull optimization typically relies on indirect searching algorithms that use a sophisticated guess-and-check method to fund optimum solution. Direct search algorithms require the input of the gradient of the problem but can be more efficient and/or effective than indirect counterparts.

4. CONCLUSIONS

The CNN-based surrogate model proved to be a comparable surrogate model to existing conventional methods when the sample size is adequately large. In addition, the CNN models offer more functionality and flexibility than existing surrogate models. Primarily the CNNs were proven to be able to learn sequentially the entire design space in order to support a functionality similar to long term memory in the human brain. The long term memory capabilities of the CNN models provide a solution for existing short-comings of the conventional model in regards to the need to recreate all new data and retrain a new surrogate model if the design variables change. This is a critical component of making the hull form optimization process more forgiving to the changing understanding of the naval architect and requirements of the hull form. In addition, the CNN based model provides a direct means for computing partial derivatives of design variables with respect to objective functions. These partial derivatives can be used to inform the naval architect what design variables should be prioritized and in what directions. The partial derivatives can also be used to iteratively step through a gradient based optimization algorithms such as gradient descent.

Future applications of these findings will ideally be implemented into the existing software that supports hull form optimization. A recommended use case of the CNN based model is to have the design software perform an initial study without user input on every possible design variable with a Monte Carlo or Latin Hypercube sampling method. An adequately large and varying study could be used to support all down-stream optimization studies without the need for analyzing any new hull designs with CFD. This process makes it feasible for higher-fidelity resistance analysis tools to be used as well because there would be no need for the hull optimization program to support fast, real-time resistance analysis. In addition, a comprehensive CNN model would be extremely effective for informing the naval architect which design variables to include in the hull optimization prior to any human-in-the-loop activity.

ACKNOWLEDGEMENTS

Thank you to the U.S. Navy for making this research possible. This research would not have been possible without the tools and resources sponsored by HPCMP CREATE-Ships. The execution of this work in this timeline would not have been possible without the support of ONR through the In-House Lab Independent Research (ILIR) program at Naval Surface Warfare Center – Carderock Division (NSWCCD). The authors also would like to thank the support of ONR grant N00014-17-1-2691 under Ms. Kelly Cooper to develop the computational tool, SimDShip, which was used to generate all training and testing datasets in this research.

REFERENCES

Chi, Yang and Fuxin Huang. "An Overview of Simulation-based Hydrodynamic Design of Ship Hull Forms." *Journal of Hydrodynamics* (2016): 947-960.

Giunta, Anthony, Steven Wojtkiewicz and Michael Eldred. "Overview of modern design of experi-ments methods for computational simulations." *41st Aerospace Sciences Meeting and Exhibit*. Reno, Navada: American Institute of Aeronatuics and Astronautics, 2004.

Huang, Fuxin and Chi Yang. "Hull Form optimization of a Cargo Ship for Reduced Drag." *Journal of Hydrodyamics* (2016): 173-183.

Huang, Fuxin, et al. "Hull form optimization of a TriSWACH for reduced drag." *In 13th International Conference on Fast Sea Transportation FAST 2015*. Washington DC, 2015.

Huang, Fuxin, L. Wang and Chi Yang. "Hull Form Optimization for Reduced Drag ad Improved Seakeeping Using a Surrogate Based Method." *The Twenty-fifth International Offshore and Polar Engineering Conference*. Big Island, Hawaii, 2015.

McKay, M. D., R. J. Beckman and W. J. Conover. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code." *Technometrics* 12 March 2012: 55-61.

Noblesse, Francis, Fuxin Huang and Chi Yang. "The Neumann–Michell Theory of Ship Waves." *Journal of Engineering Mathematics* (2012): 51-71.

Shaeffer, Austin and Chi Yang. "Application of Machine Learning to Early-Stage Hull Form Design." *SNAME 2020 Transactions* (2020).

Shaeffer, Austin. "Application of Artificial Neural Networks to Early-Stage Hull Form Design." Phd Dissertation. 2023.

Wang, Lijue, et al. "Hydrodynamic Optimization of a Wedge Hull." *In 13th International Conference on Fast Sea Transportation-FAST 2015*. Washington DC, 2015.

Yang, Chi, et al. "The Neumann-Kelvin and Neumann-Michell Linear Models of Steady Flow about a Ship." *Proceedings of the 12th International Congress of the International Maritime Association of the Mediterranean (IMAM'2007)*. Varna, Bulgaria, 2007.