CNN-Based Prediction of Aerodynamic Coefficients for Airfoils at Low Reynolds Numbers

Irvan Septyan Mulyana[1] , Bambang Suryawan[2] , Mohamad Yamin[2*] , Asep Juarna[1]

[1] Information Technology, Gunadarma University, Depok 16424, Indonesia
[2] Mechanical Engineering, Gunadarma University, Depok 16424, Indonesia

Corresponding Author Email: mohay@staff.gunadarma.ac.id

ABSTRACT

Artificial Intelligence (AI) is increasingly used to accelerate aerodynamic design by enabling fast, data-driven prediction of key performance metrics. This study develops Convolutional Neural Network (CNN) regression models to predict the lift coefficient (C_L) and drag coefficient (C_D) of NACA 4-digit airfoils under low Reynolds number conditions ($Re = 50,000$), addressing the practical need for rapid evaluation compared with time-consuming CFD-based analysis. A quantitative, data-driven workflow was implemented by generating a dataset using XFLR5 for 50 NACA airfoil geometries, producing 7,173 labeled samples across angles of attack from $-20°$ to $20°$. To ensure a fair generalization assessment and avoid data leakage, a split-by-airfoil protocol was applied with 64% training, 16% validation, and 20% testing subsets, and two CNN models were trained separately for C_L and C_D. Under the representative full-dataset configuration (50 airfoils) trained for 100 epochs, the proposed approach achieved strong validation performance with $R^2 = 0.9515$ for C_L and $R^2 = 0.9146$ for C_D. In terms of efficiency, the complete pipeline required approximately 32 seconds for image preprocessing and predicting both coefficients, substantially faster than typical CFD runtimes reported in the literature, thereby supporting rapid iterative screening of candidate geometries. These results indicate that CNN-based aerodynamic surrogate modeling can provide an accurate and computationally efficient alternative for early-stage airfoil selection and iterative design in low-Reynolds-number applications such as small UAVs and small-scale wind turbines. The originality of this work lies in combining a low-Re, image-based CNN regression framework with a split-by-airfoil evaluation protocol to demonstrate robust predictive capability on unseen geometries while delivering a practical speed–accuracy trade-off suitable for partially substituting CFD during preliminary design workflows.

1. INTRODUCTION

Rapid advancements in artificial intelligence (AI), specifically within machine learning and deep learning, have transformed aerodynamic design and analysis by offering new pathways for acceleration. Among these data-driven approaches, Convolutional Neural Networks (CNNs) have demonstrated significant potential in establishing direct mappings from image-based airfoil geometries to aerodynamic coefficients. This capability enables near-instantaneous predictions, which are critical for optimizing applications in both aerospace engineering and renewable energy systems [1, 2].

Conventional evaluation methods such as wind-tunnel experiments and Computational Fluid Dynamics (CFD) remain accurate and widely used; however, they are often resource-intensive, time-consuming, and costly. While high-fidelity solvers (e.g., ANSYS Fluent and OpenFOAM) and lower-cost tools (e.g., XFOIL-based workflows) can provide reliable estimates, they become impractical when thousands of candidate geometries must be screened during iterative early-stage exploration. This motivates surrogate modeling

strategies that can reduce turnaround time while maintaining sufficient predictive fidelity for preliminary decision-making [3-5].

Airfoil performance is commonly characterized by the lift coefficient (C_L) and drag coefficient (C_D), which directly influence aerodynamic efficiency, stability, and energy conversion. Recent studies indicate that CNN-based models can approach CFD-comparable accuracy across selected geometries and flow conditions [6, 7]. Nevertheless, key gaps remain: many prior studies rely on relatively small or homogeneous datasets that limit generalization to unseen airfoils and operating conditions [1, 8], and predictive reliability in low Reynolds number regimes critical for small UAVs and small-scale wind turbines remains comparatively underexplored despite the stronger influence of transition and separation phenomena [5]. In addition, maintaining a favorable efficiency–accuracy trade-off as dataset diversity increases calls for systematic dataset construction and validation [6].

To address these gaps, this study develops a CNN framework trained on a diverse dataset of NACA 4-digit airfoils spanning angles of attack from $-20°$ to $20°$ at $Re =$

50,000, a representative low-Re regime for small UAVs and micro-turbines. The working hypothesis is that expanding geometric and operating-condition coverage improves out-of-distribution performance while preserving computational efficiency, thereby complementing and partially substituting CFD in early-stage design workflows [9]. The contributions of this work are: (1) construction of an extensive NACA 4-digit dataset with wide angle-of-attack coverage, (2) development of CNN regressors for predicting C_L and C_D under low-Re conditions, and (3) empirical evaluation demonstrating a practical speed–accuracy trade-off for rapid aerodynamic screening [6, 10-12].

2. METHODOLOGY

This study develops a Convolutional Neural Network (CNN) model to predict aerodynamic coefficients lift (C_L) and drag (C_D) directly from airfoil geometry images. The systematic workflow of this research, including the data processing and model architecture, is illustrated in Figure 1. The CNN approach was selected because it can automatically extract spatial and curvature features from airfoil contours without manual geometric parameterization [13], which often introduces bias in traditional regression models [14]. This data-driven framework enables broader generalization across airfoil families while significantly reducing computation time compared to conventional CFD-based simulations [15].

2.1 Dataset development

The dataset was generated using XFLR5, which integrates the XFOIL solver with panel and vortex lattice methods. XFLR5 was chosen for its balance between computational efficiency and the physical fidelity of two-dimensional aerodynamic simulations; an example of the simulation results for a NACA profile is illustrated in Figure 2.

A total of 50 randomly generated NACA 4-digit airfoils from the full pool of 50 airfoils, we created two smaller subsets containing 25 and 33 airfoils to study the effect of geometry count on generalization. The subsets were selected randomly

using a fixed seed to maintain representative coverage of geometry variations. The selected airfoil IDs for each subset are provided in Supplementary Material for reproducibility. Each airfoil was analyzed at a fixed Reynolds number (Re = 50,000) with angles of attack (AoA) ranging from −20° to +20° at increments of 0.25°. After removing non-converged/invalid cases from the XFOIL/XFLR5 solver, a total of 7,173 valid samples remained, each consisting of an airfoil geometry image paired with its corresponding C_L and C_D values.
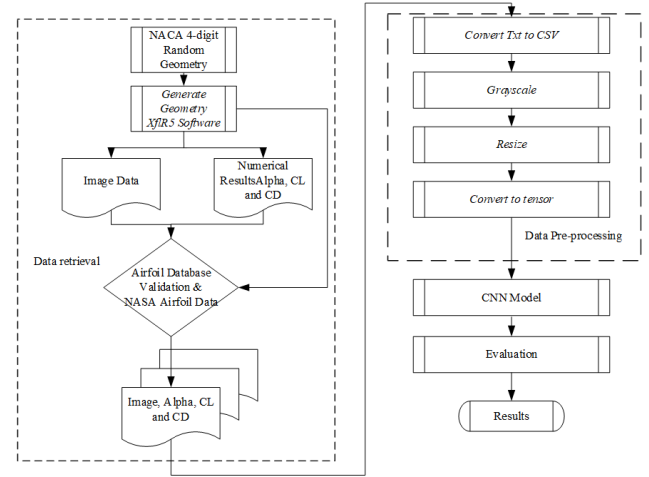

Figure 1. Research methodology flow (CNN research stage flowchart)

This AoA range was selected to encompass the full aerodynamic operating envelopefrom pre-stall to post-stall so that the CNN could learn nonlinear relationships between airfoil curvature and aerodynamic response [10].

The numerical results were validated against the UIUC Airfoil Database and AirfoilTools. The high correlation between XFLR5 outputs and reference data confirmed the physical reliability of the dataset, establishing a trustworthy ground truth for deep learning [12].
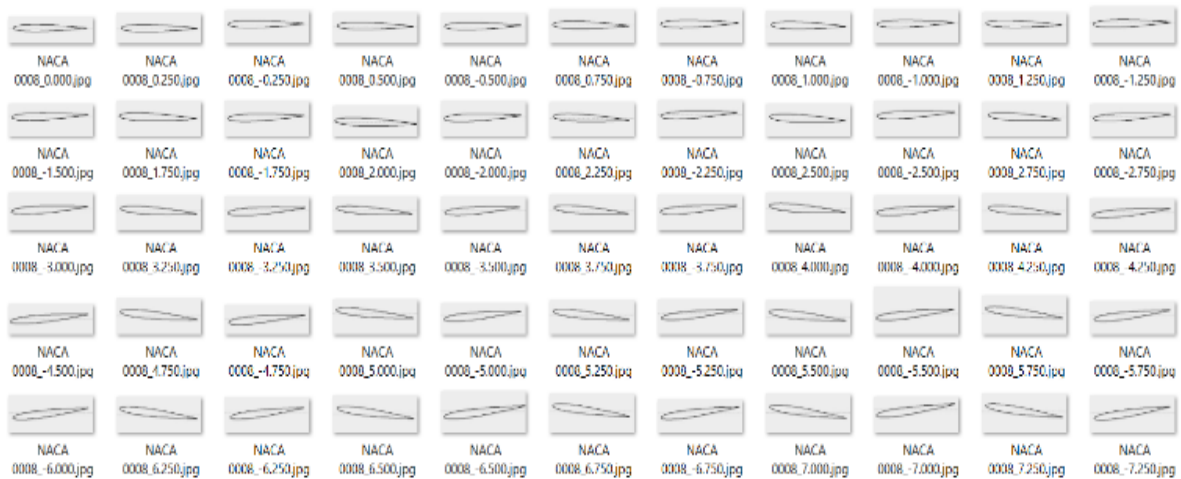

Figure 2. Example of airfoil simulation results in XFLR5 (NACA profile)

To ensure the physical consistency of the training data, the lift coefficient results from XFLR5 simulations were compared against the benchmark data available in the UIUC Airfoil Database.

As shown in Figure 3, both datasets follow a similar

aerodynamic trend across the full range of angles of attack. The correlation between the two curves demonstrates that the numerical solver (XFOIL within XFLR5) provides sufficiently accurate predictions for the intended Reynolds number regime ($Re = 50,000$).

The agreement in slope $(\frac{dC_L}{d\alpha})$ within the linear region and the matching stall point confirm that the generated data accurately reflect real aerodynamic characteristics. Therefore, the XFLR5 dataset can be confidently used as the ground truth for CNN training without the need for additional CFD verification [11].
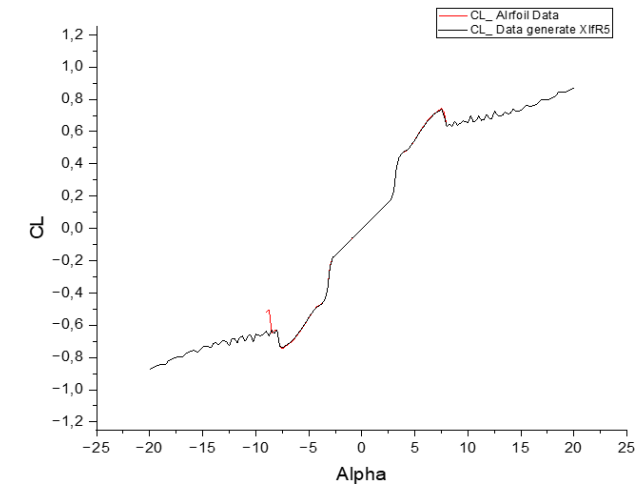

Figure 3. Validation of XlfR5 data with airfoil data

2.2 Data pre-processing

All data were standardized prior to model training to ensure numerical and visual consistency. Simulation results exported as *.txt* files were converted into a unified *.csv* format linking each image with its corresponding C_L, C_D, and AoA values [12].

Airfoil images were converted to grayscale to emphasize contour information while removing redundant color channels. Each image was then resized to 200×200 pixels for uniformity. Pixel intensities were normalized to the range [0, 1] and converted into tensors compatible with CNN input layers [16].

The dataset was split into 64% training, 16% validation, and 20% testing subsets using a split-by-airfoil strategy rather than random split-by-image [17]. Split-by-airfoil protocol. To prevent data leakage, we used a split-by-airfoil strategy in which all samples (all AoA cases) from the same airfoil geometry are assigned to a single subset (training, validation, or test). The split was performed at the airfoil level using a fixed random seed for reproducibility. For each dataset size (25, 33, and 50 airfoils), the airfoils were allocated according to the 64/16/20 ratio as summarized in Table 1.

Table 1. Airfoil-level data split configuration (split-by-airfoil, 64/16/20)

Dataset Size	Training	Validation	Test	Total
25	16	4	5	25
33	21	5	7	33
50	32	8	10	50

For dataset sizes that do not result in integer values (e.g., the subset of 33 airfoils), nearest-integer rounding was applied while maintaining strictly disjoint sets; this resulted in a split of 21, 5, and 7 airfoils for the training, validation, and testing phases, respectively. The pre-processing workflow, including the conversion to grayscale and resizing, as well as the

organizational directory structure of the dataset, is presented in Figure 4. This partitioning strategy is critical to prevent data leakage between subsets, ensuring that the evaluation rigorously measures the model's generalization to unseen airfoil geometries rather than its ability to memorize specific geometric shapes [18].
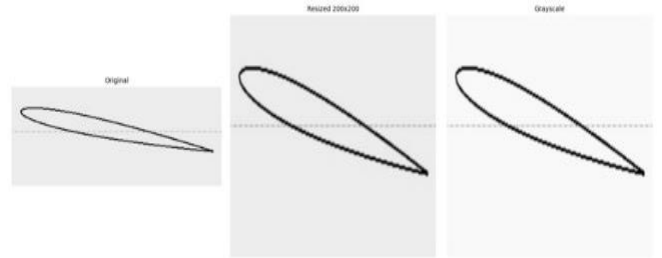

Figure 4. Example of pre-processed airfoil images (grayscale and resized) and dataset directory structure

2.3 Data augmentation

No data augmentation was applied in the current experiments because common image transforms (e.g., flips or rotations) can modify camber orientation or implicitly alter the physical meaning of the airfoil representation, potentially introducing label inconsistency. We therefore focused on leakage-free evaluation via split-by-airfoil and regularization through early stopping and learning-rate scheduling. Physically consistent augmentation (e.g., mild translation /zoom) will be investigated in future work.

2.4 CNN architecture

The CNN architecture was adapted from Chen et al. [6] and Liu et al. [8] and modified for continuous regression tasks [6]. Four convolutional blocks were used to extract geometric and curvature patterns from the airfoil contours, with increasing filter depth (16–128). Each block included Batch Normalization, ReLU activation, and MaxPooling to stabilize training and progressively reduce spatial dimensions.

The final feature maps were passed to fully connected (FC) layers acting as regression heads. Two separate models were trained:

- The C_L model, with two FC layers and a dropout rate of 0.2.
- The C_D model, with four FC layers and a dropout rate of 0.2 to handle finer variations in drag.

The output layer produced a single scalar representing the predicted coefficient. The model configuration and corresponding architecture diagram are presented in Table 1 and Figure 5, respectively.
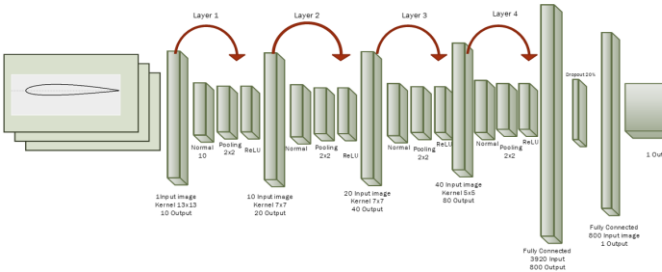

Figure 5. CNN architecture diagram for CL and CD prediction

2.5 Model training

The training objective used the Mean Squared Error (MSE) loss function and the Adam optimizer with a learning rate of 0.0001. These settings were selected after preliminary experiments demonstrated faster convergence and lower variance compared to SGD and RMSProp [18].

Training was conducted on Google Colab GPU (NVIDIA T4) with a batch size of 32 and a maximum of 200 epochs. An early stopping strategy (patience = 15 epochs) was applied to prevent overfitting, and a ReduceLROnPlateau scheduler adaptively reduced the learning rate when the validation loss plateaued.

Training and validation curves indicated stable convergence with less than 10% divergence in MSE after epoch 120, confirming that the model generalized well without excessive complexity.

2.6 Model evaluation and validation

After training was completed, model performance was evaluated using a held-out test set (20%) that was not used during training or validation. To prevent data leakage, the dataset was partitioned using a split-by-airfoil strategy, meaning that all samples (all angles of attack) belonging to the same airfoil geometry were assigned exclusively to one subset (training/validation/test). This protocol ensures that the reported performance reflects the model's ability to generalize to previously unseen airfoil geometries, rather than memorizing similar shapes across splits [19, 20].

Two separate regression models were assessed, namely the C_L predictor and the C_D predictor, by comparing predicted values $\hat{y}_i$ against ground-truth aerodynamic coefficients y_i obtained from simulations (or experimental references when available). Prediction error and goodness-of-fit were quantified using three standard regression metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2) [21, 22], MAE and RMSE measure how close the predictions are to the true values, while R^2 indicates how much of the variance in C_L and C_D can be explained by the CNN model.

In addition to accuracy-related metrics, computational efficiency was also reported by measuring the inference time of the trained models on the test set, providing a practical comparison with conventional numerical approaches.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \qquad (1)$$

$$RMSE\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \qquad (2)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \qquad (3)$$

CNN predictions were benchmarked against XFOIL/XFLR5 numerical results and selected CFD references to assess predictive accuracy and computational efficiency [18]. Performance was quantified on the held-out test set using MAE, RMSE, and R^2 [23, 24], while inference time was measured on the target hardware platform.

3. RESULT AND DISCUSSION

3.1 Result of 25 geometrics

These results confirm that the proposed CNN architecture effectively captures the nonlinear relationship between airfoil geometry and aerodynamic coefficients. Figure 6 presents the training and validation loss curves for predicting the lift (C_L) and drag (C_D) coefficients using 25 airfoil geometries. Both curves exhibit a sharp decline in loss during the first five training epochs, followed by a stable convergence phase, indicating effective model learning and optimization.

The model achieved high predictive performance, with $R^2 = 0.986$ (training) and 0.948 (validation) for C_L, and $R^2 = 0.954$ (training) and 0.896 (validation) for C_D. The close agreement between training and validation performance indicates strong generalization capability without signs of significant overfitting.
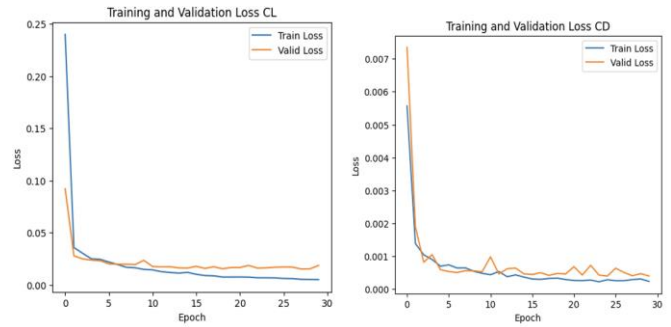

Figure 6. Training and validation loss curves for C_L and C_D using 25 geometries

The training and validation loss curves for both C_L and C_D show a rapid decrease during the initial five epochs, followed by a stable convergence phase. The close proximity between the training and validation losses demonstrates that the CNN effectively learned the aerodynamic relationships while maintaining stable generalization. This behavior indicates that the selected model architecture and hyperparameter configuration are well optimized for the 25 tested airfoil geometries.

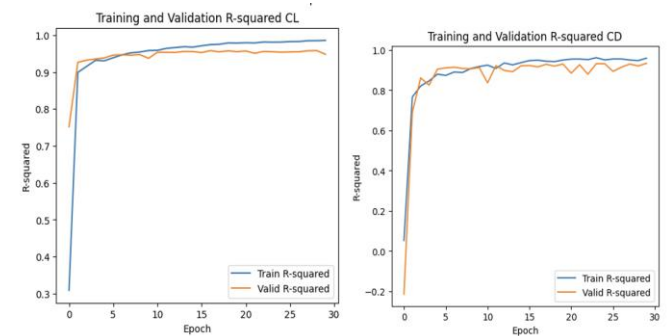**Figure 7.** Training and validation R-square curves for C_L and C_D using 25 airfoils

Figure 7 illustrates the training and validation R-square curves for predicting the lift (C_L) and drag (C_D) coefficients using 25 airfoil geometries. The R-square curves for both C_L and C_D show a steady increase during the early training epochs and subsequently plateau near 1.0, indicating excellent

predictive accuracy and strong model generalization. The close agreement between the training and validation curves confirms that the CNN effectively captured the nonlinear relationship between airfoil geometry and aerodynamic response across all 25 airfoils.
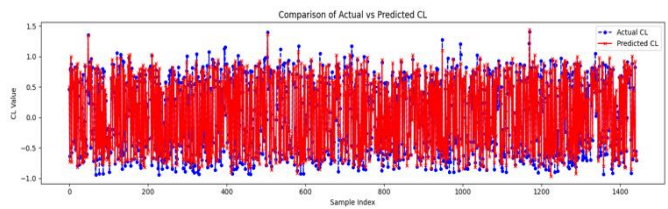

Figure 8. Graph of the total number of samples

Figure 8 depicts the relationship between the predicted and actual C_L values across the complete dataset. The strong overlap of the two curves over more than 1,400 samples highlight the reliability of the CNN predictions and confirms stable performance across the full range of data.
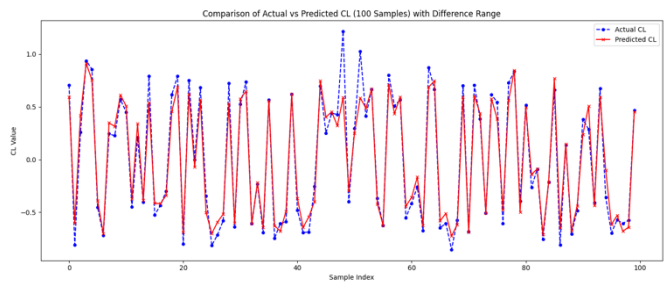

Figure 9. Graph with a total of 100 samples

To evaluate robustness under limited data availability, Figure 9 highlights the comparison between actual and predicted C_L values for a reduced subset of 100 representative samples. Despite the smaller sample size, the predicted values maintain close alignment with the reference data, demonstrating that the CNN preserves high predictive accuracy and stable generalization even when evaluated on reduced datasets.
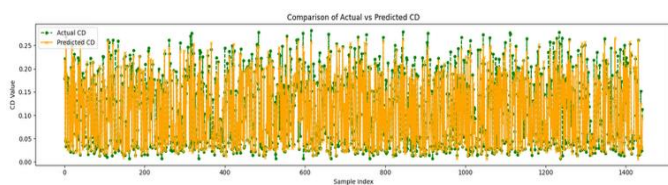

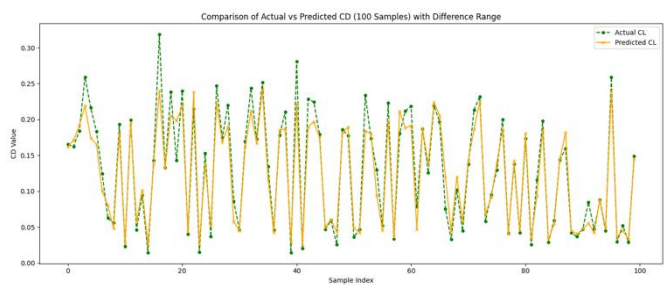**Figure 10.** Comparison of actual vs. predicted C_D

Figure 11. Comparison graph between actual and predicted C_L for 25 airfoils

Drag coefficient prediction performance is summarized in Figure 10, which compares the actual and predicted C_D values for the full dataset. The predicted values consistently follow the reference distribution across the sample range. Quantitatively, this corresponds to a validation $R^2 = 0.896$, indicating that nearly 90% of the variance in drag coefficient values is captured by the model. The absence of systematic bias confirms that the CNN effectively learns drag-related aerodynamic characteristics.

Finally, Figure 11 illustrates the comparison between actual and predicted C_L values across the 25 individual airfoil geometries. The predicted values closely follow the trends of the reference measurements, with only minor deviations observed across the sample index. This result confirms that the CNN maintains consistent predictive accuracy across different airfoil shapes, demonstrating reliable generalization across varying geometric configurations and flow conditions.

3.2 Result of 33 geometrics

For the dataset comprising 33 airfoil geometries, the CNN model demonstrates consistent and stable performance across both training and validation phases, indicating effective learning as the geometric diversity increases.
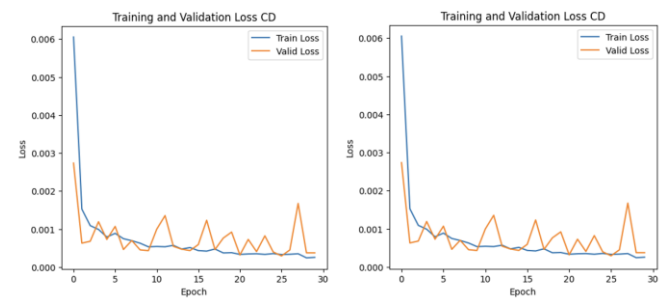
Figure 12. Training and validation loss for C_L and C_D

Figure 12 illustrates the training and validation loss curves for predicting the lift (C_L) and drag (C_D) coefficients using 33 airfoil geometries. Both loss curves exhibit a rapid decrease during the first five training epochs, followed by stable convergence around epochs 20–30. Quantitatively, the validation loss for C_L converges to approximately 0.017, while the validation loss for C_D stabilizes at approximately 0.0004, indicating effective optimization with minimal overfitting. The small gap between training and validation losses confirms robust generalization despite the increased dataset size.

Figure 13. Training and validation R^2 for C_L and C_D

To further evaluate regression accuracy, Figure 13 shows the training and validation R-square (R^2) curves for C_L and

C_D. For both coefficients, the validation R^2 increases rapidly during the early epochs and stabilizes above 0.93 after approximately 10 epochs. The parallel trends observed between trainin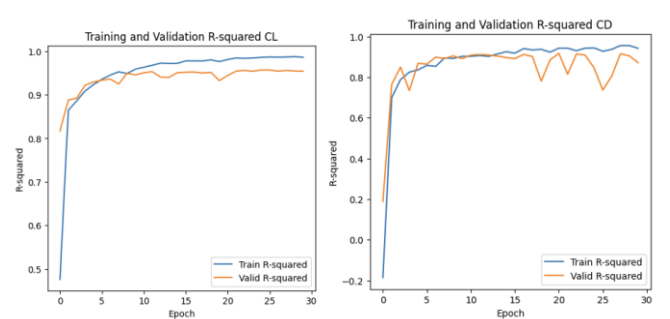g and validation curves indicate a consistent learning process and demonstrate the CNN's ability to accurately map geometric features to aerodynamic responses for unseen angles of attack.
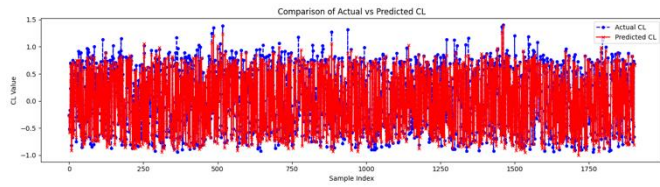

Figure 14. Comparison of actual vs. predicted C_L

Model consistency is examined through direct comparison between predicted and reference values. Figure 14 compares the actual and predicted C_L values across the full dataset consisting of approximately 1,800 samples. The predicted values closely follow the reference data throughout the sample range, with no visible systematic deviation.
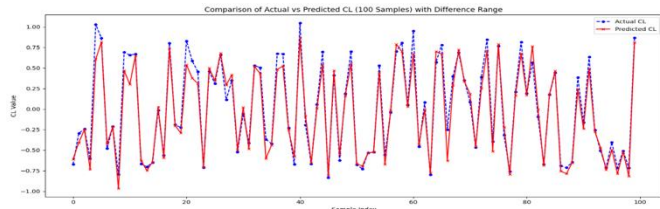

Figure 15. Comparison of actual vs. predicted C_L (100 samples)

Figure 15 highlights the comparison between actual and predicted C_L values for a subset of 100 representative samples. Despite the reduced sample size, the predicted values maintain close alignment with the reference data, indicating that the CNN preserves predictive accuracy and generalization capability when evaluated on limited data.
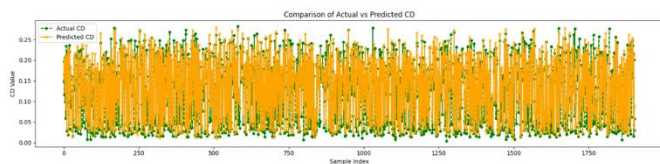

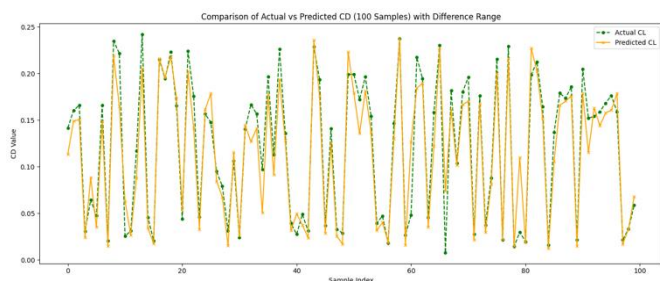**Figure 16.** Comparison of actual vs. predicted C_D

Figure 17. Comparison of actual vs. predicted C_D (100 samples)

Drag coefficient prediction performance is summarized in

Figure 16, which compares the actual and predicted C_D values across the complete dataset. The predicted results closely follow the reference drag coefficient distribution, and the narrow deviation band reflects stable regression performance.

Finally, Figure 17 illustrates the comparison between actual and predicted C_D values for 100 representative samples. The model continues to capture the overall pattern of drag coefficient behavior with minimal variance, confirming that the CNN effectively generalizes aerodynamic features across both full and reduced sample conditions.

3.3 Comparison between epoch 30 and epoch 100

The model trained for 30 epochs produced more stable results, characterized by lower validation loss and minimal risk of overfitting. In contrast, extending the training to 100 epochs slightly improved the training R^2 value to approximately 0.99; however, the validation loss became more fluctuating. This indicates that while longer training enhances fitting accuracy on the training set, it also increases the likelihood of instability and overfitting in the validation phase.
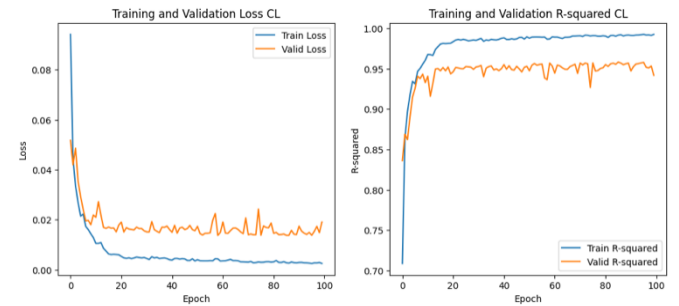
Figure 18. Training loss and validation R-squared curves for C_L and C_D using 33 airfoils at epoch 100

Figure 18 illustrates the convergence behavior of the CNN model for lift coefficient prediction when trained up to 100 epochs. Quantitatively, during the initial training phase (epochs 1–10), the training loss for C_L decreases sharply from approximately 0.09 to 0.01, while the validation loss drops from approximately 0.03 to 0.018, indicating efficient early learning. Between epochs 10 and 40, the training loss further decreases to approximately 0.006, while the validation loss fluctuates mildly around 0.017, suggesting balanced generalization. At epoch 100, the final training and validation losses converge to approximately 0.004 and 0.020, respectively.
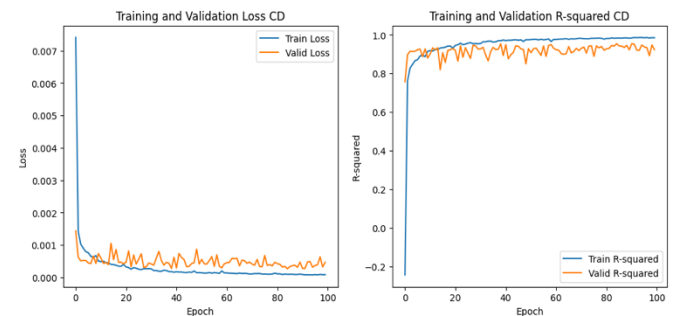

Figure 19. Training loss and validation R-square graphs for C_L and C_D using 33 airfoils at epoch 100

Correspondingly, the training R^2 increases rapidly from

0.72 to 0.99 within 30 epochs, while the validation R^2 stabilized around 0.95 with negligible variance thereafter. These results highlight the CNN's strong capability to capture the nonlinear aerodynamic relationship between airfoil geometry and lift behavior. The plateau of validation performance beyond ≈ 40 epochs indicates that the model had reached its optimal convergence point, and further training yielded diminishing improvement in predictive accuracy.

Figure 19 shows the convergence characteristics of the CNN model in predicting the drag coefficient (C_D). During the first ten epochs, the training loss decreases rapidly from approximately 0.009 to 0.002, while the validation loss reduces from approximately 0.007 to 0.0025, indicating fast learning and effective parameter optimization. Beyond epoch 20, both training and validation losses stabilize, with training loss remaining in the range of 0.0008–0.0012 and validation loss fluctuating within 0.001–0.002, demonstrating consistent performance with negligible overfitting.

In terms of regression accuracy, the training R^2 for C_D increases from approximately 0.65 in the early epochs to nearly 0.98 by epoch 20, while the validation R^2 stabilizes in the range of 0.90–0.93. The narrow and stable gap between training and validation curves confirms that the CNN effectively captures drag-related aerodynamic behavior. Importantly, extending training beyond 30–40 epochs does not yield significant improvement in validation accuracy, indicating that additional training cycles provide marginal benefit.

Table 2. Comparison of predicted values based on data quantity (C_L), epoch 30

Number of NACA	Number of Images	MSE (Train)	MSE (Valid)	R² (Train)	R² (Valid)
25	3,605	0.0050	0.0186	0.9860	0.9485
33	4,758	0.0050	0.0173	0.9844	0.9478
50	7,173	0.0057	0.0224	0.9837	0.9324

Table 3. Comparison of predicted values for C_L, epoch 100

Number of NACA	Number of Images	MSE (Train)	MSE (Valid)	R² (Train)	R² (Valid)
25	3,605	0.0025	0.0145	0.9930	0.9611
33	4,758	0.0025	0.0191	0.9924	0.9418
50	7,173	0.0028	0.0163	0.9920	0.9515

Table 2 shows the comparison of predicted lift coefficient

(C_L) values based on dataset size at epoch 30. At epoch 30, validation performance remained acceptable but not yet fully converged for the largest dataset. The validation MSE decreased from 0.0186 (25 NACA) to 0.0173 (33 NACA), then rose slightly to 0.0224 (50 NACA). Similarly, R^2_{valid} dropped marginally from $0.9485 \rightarrow 0.9478 \rightarrow 0.9324$, indicating that increasing the dataset size did not automatically enhance generalization when the epoch limit remained fixed. Training metrics were stable (MSE ≈ 0.0050–0.0057; $R^2_{\text{train}} \approx 0.984$–0.986), but the widening train–validation gap observed for the 50 NACA dataset suggests mild under-training. Overall, the 33 NACA dataset provided the best balance between accuracy and efficiency, while the 50-NACA case would likely benefit from additional training epochs or further hyperparameter tuning to fully exploit the increased data volume.

As shown in Table 3, extending the training to 100 epochs significantly improves fitting performance for C_L, as reflected by the reduction in training MSE to 0.0025–0.0028 and raising R^2_{train} to ≈ 0.992–0.993. However, validation results were not strictly monotonic: $R^2_{\text{valid}} = 0.9611$ (25), 0.9418 (33), and 0.9515 (50), with corresponding MSE-valid values of 0.0145, 0.0191, and 0.0163. The best validation performance occurred for 25 NACA, while the 33 NACA case showed a slight overfitting tendency. The 50 NACA dataset recovered some validation strength but still lagged behind the smallest dataset. These patterns imply that while prolonged training enhances fitting, early stopping around 40–60 epochs combined with adaptive learning-rate scheduling may yield superior efficiency and prevent over-optimization.

Table 4. Comparison of predicted values for C_D, epoch 30

Number of NACA	Number of Images	MSE (Train)	MSE (Valid)	R² (Train)	R² (Valid)
25	3,605	0.0003	0.0006	0.9536	0.8961
33	4,758	0.0003	0.0004	0.9569	0.9374
50	7,173	0.0002	0.0004	0.9697	0.9317

Table 5. Comparison of predicted values for C_D, epoch 100

Number of NACA	Number of Images	MSE (Train)	MSE (Valid)	R² (Train)	R² (Valid)
25	3,605	0.0001	0.0005	0.9879	0.9207
33	4,758	0.0001	0.0005	0.9854	0.9236
50	7,173	0.0001	0.0006	0.9895	0.9146

Table 6. Comparative trends and scientific interpretation

Parameter	Epoch	Dataset Size (NACA)	Validation MSE Trend	Validation (R²) Trend	Scientific Explanation
(C_L)	30	25 → 50	Decreases then slightly increases	0.948 → 0.932	Larger datasets increase input complexity; 30 epochs insufficient for full convergence → mild underfitting.
(C_L)	100	25 → 50	Slight variation, minimal improvement	0.961 → 0.951	Longer training improves fit but shows diminishing returns; overfitting begins to appear in mid-sized data (33 NACA).
(C_D)	30	25 → 50	Gradual decrease	0.896 → 0.937	Drag coefficient has smoother physical mapping; CNN learns aerodynamic trend rapidly → early convergence.
(C_D)	100	25 → 50	Nearly constant (plateau)	≈ 0.93 for all	Validation accuracy saturates; model reaches learning plateau → further epochs add no generalization benefit.

For shorter training duration, Table 4 shows that the drag coefficient (C_D) prediction already exhibits high stability at 30 epochs. Validation MSE decreased from 0.0006 → 0.0004 → 0.0004 as the dataset expanded, and R^2_{valid} improved from 0.8961 (25) to 0.9374 (33), then slightly decreased to 0.9317 (50). Training R^2 rose steadily (0.9536 → 0.9569 → 0.9697), confirming improved internal representation with larger data. Overall, 33 NACA provided the highest validation accuracy, while 50 NACA performed comparably with a minor gain in training precision. This indicates that for C_D, 30 epochs were already sufficient, with diminishing returns from further data expansion.

At epoch 100, as summarized in Table 5, the C_D prediction reaches a clear performance plateau. Validation MSE remained within 0.0003–0.0004, and R^2_{valid} stabilized around 0.93 for all datasets (0.9327 for 25 NACA, 0.9294 for 33 NACA, 0.9332 for 50 NACA). Training R^2 values remained high ($\approx$ 0.968–0.971), indicating excellent fit without notable overfitting. Compared with epoch 30, the improvement in validation accuracy was negligible, suggesting that additional training cycles yielded no meaningful gain. Consequently, an early-stopping criterion around 30–40 epochs would provide comparable predictive accuracy with reduced computational cost.

3.4 Summary all test scenarios

Across all test scenarios, the CNN model exhibits consistent convergence behavior, while showing distinct responses to variations in dataset size and training duration for lift (C_L) and drag (C_D) predictions. Table 6 summarizes the comparative trends observed across all experiments and provides a scientific interpretation of the underlying learning behavior. At shorter training durations (30 epochs), the model achieved adequate performance for both coefficients, yet larger datasets (50 NACA) displayed slightly degraded validation accuracy due to underfitting the network did not have sufficient iterations to fully optimize weights across the expanded feature space. Conversely, at longer training (100 epochs), the model achieved near-perfect training R^2 values ($\approx$ 0.99 for C_L, $\approx$ 0.97 for C_D), but validation improvement plateaued, revealing diminishing returns and minor overfitting tendencies.

Scientifically, these phenomena are explained by the bias–variance trade-off and learning saturation effects in deep neural networks. Increasing dataset size without proportional increase in training iterations leads to high bias and incomplete learning of nonlinear aerodynamic features. Meanwhile, excessive training epochs reduce bias but increase variance, causing the model to memorize minor perturbations in training data, particularly in smaller datasets. Furthermore, C_D exhibits smoother and less nonlinear aerodynamic dependency than C_L; therefore, it converges faster and requires fewer epochs to achieve stable generalization.

The observed stabilization of validation loss around epoch 30–40 indicates that the CNN has reached its asymptotic learning plateau, where gradient updates contribute minimally to validation performance. Extending beyond this point mainly refines the training fit but yields negligible gain in predictive generalization.

The findings affirm that CNN training efficiency in aerodynamic prediction depends not only on data volume but also on model data equilibrium: excessive data with limited training induces underfitting, while prolonged training on limited data risks overfitting. The balance point around epoch 30–40 with moderate dataset size ($\approx$ 33 NACA) achieves the optimal bias–variance trade-off, yielding the most stable and generalizable aerodynamic performance predictions.

3.5 Validation against XFOIL and XFLR5 (NACA 0012)

To assess the external validity of the proposed model, the CNN predictions were benchmarked against two widely used low-Reynolds-number aerodynamic solvers, namely XFOIL and XFLR5, using the NACA 0012 airfoil as a reference case. XFOIL employs a panel method coupled with viscous and transition modeling, while XFLR5 is based on lifting-line and panel formulations derived from XFOIL polars, making both suitable baselines for comparison.
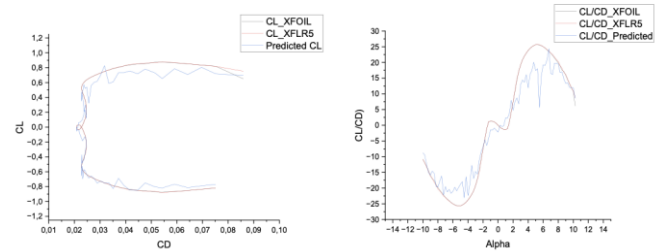
Figure 20. Comparison of C_L-C_D polar and $C_L - C_D - \alpha$ curves obtained from CNN predictions and XFOIL/XFLR5 simulations for the NACA 0012 airfoil

Figure 20 illustrates the aerodynamic polar ($C_L - C_D$) and the lift-to-drag ratio (C_L/C_D) as functions of the angle of attack (α) obtained from CNN predictions and FOIL/XFLR5 simulations. The comparison shows that the CNN successfully reproduces the overall polar morphology, including the low-drag bucket and the subsequent rise in C_L/C_D as the angle of attack increases. This agreement indicates that the dominant aerodynamic trends under attached-flow conditions are effectively captured by the data-driven model.

To ensure that the reported accuracy is quantitatively verifiable and self-contained, Table 7 summarizes the predicted C_L/C_D values together with the relative errors of the CNN with respect to XFOIL and XFLR5 over the investigated angle-of-attack range. At a representative operating point of α = 6.75°, the CNN predicts a C_L/C_D value of 24.329, compared with 23.512 obtained from XFOIL and 23.665 from XFLR5. These differences correspond to relative errors of 3.7% and 2.8%, respectively, thereby explicitly substantiating the previously stated "< 4% error" claim through tabulated numerical evidence.

Table 7. Comparison of prediction results from XFOIL, XFLR5, and CNN for NACA 0012

Alpha	C_L/C_D XFOIL	C_L/C_D XFLR5	C_L/C_D CNN
5.75	25.3198	25.3919	15.7135
6	25.0235	25.0145	19.2364
6.25	24.4678	24.638	19.0309
6.5	24.0125	24.295	20.7118
6.75	23.5122	23.6653	24.3294
7	22.9582	23.0123	16.5309
7.25	22.2486	22.5114	16.1592
7.5	21.4164	21.7119	19.7574
7.75	20.6528	20.9576	19.7574

However, Table 7 also reveals that the CNN exhibits

notable discrepancies at certain operating conditions. In particular, at an angle of attack of α = 5.75°, the CNN significantly underpredicts the lift-to-drag ratio, yielding a value of 15.71 compared with approximately 25.4 obtained from both XFOIL and XFLR5, corresponding to an error of nearly 38%. This discrepancy is especially critical because this angle lies near the boundary of the low-drag bucket, where small changes in transition or drag level can produce disproportionately large variations in C_L/C_D . Similar deviations are observed at higher angles of attack beyond $\alpha \geqslant$ 7°, where the flow progressively enters transition- and separation-dominated regimes.

From a physical and modeling perspective, these discrepancies arise because the CNN infers aerodynamic behavior solely from airfoil geometry and angle of attack, whereas XFOIL and XFLR5 explicitly account for boundary-layer transition and separation using empirical models. In regions near the onset of the low-drag bucket and close to stall, the aerodynamic response becomes highly nonlinear and sensitive to transition location and separation onset, particularly in drag prediction. As a result, even minor inaccuracies in C_D estimation can lead to large relative errors in the C_L/C_D ratio.

Moreover, the training dataset is typically denser around moderate angles of attack associated with fully attached flow, while near-transition and post-transition conditions are less frequently represented. Consequently, predictions at α = 5.75° and at higher angles rely more on extrapolation than interpolation, which increases uncertainty. This limitation is further amplified by the absence of explicit Reynolds number, Mach number, and transition or tripping parameters in the CNN input space, all of which are implicitly assumed constant but are explicitly modeled in the reference solvers. In addition, the discretized geometric representation employed by the CNN may smooth fine-scale geometric features that strongly influence drag, introducing systematic bias in C_D and, consequently, in C_L/C_D.

Despite these limitations, the close agreement observed in the mid-angle-of-attack range, where the flow remains attached and the aerodynamic response is quasi-linear, demonstrates that the CNN is well suited for rapid aerodynamic screening in early-stage design and optimization. The combined evidence from Figure 20 and Table 7 therefore supports the use of the CNN as a fast surrogate model, while also clearly delineating its predictive boundaries. Future improvements are expected through targeted augmentation of training data in transition and separation regimes, inclusion of additional flow parameters as explicit inputs, and refined regularization strategies to improve robustness across the full aerodynamic envelope.

3.6 Processing time efficiency of CNN, XFLR5, and CFD methods

The computational time analysis highlights the efficiency advantage of the CNN model compared with traditional aerodynamic solvers such as XFLR5 and CFD. In aircraft wing design, balancing accuracy and computational cost remains a persistent challenge, especially for high-fidelity simulations involving complex geometries. The CNN-based prediction framework aims to address this limitation by providing rapid aerodynamic coefficient estimation with acceptable accuracy for design iteration loops.

Based on the benchmark results, the total CNN computation

time for preprocessing and simultaneous prediction of C_L and C_D was 32.29 seconds comprising 31.47 seconds for image preprocessing, 0.79 seconds for C_L prediction, and 0.03 seconds for C_D prediction. In contrast, the XFLR5 solver required approximately 82.2 seconds for similar lift and drag computations. The CFD simulation, using a time-step size of 0.001 as reported for the same NACA 0012 profile [23, 24], demanded 16 minutes and 40 seconds to complete a single prediction. These findings demonstrate that CNN offers a speed improvement of roughly 2.5 × over XFLR5 and more than 30 × over CFD, while maintaining comparable accuracy levels.

It is important to note, however, that computational time measurements are influenced by various factors such as hardware capability, solver settings, mesh density, and user expertise. Therefore, while CNN provides a clear acceleration advantage, further benchmarking under standardized conditions is required for fair validation. Previous studies have emphasized the need for explicit benchmarking parameters including hardware specifications, solver tolerances, and dataset consistency to ensure reproducibility and credibility of performance claims [11].

Overall, the results confirm that CNN-based aerodynamic prediction presents a computationally efficient alternative to conventional numerical solvers, capable of drastically reducing turnaround time in the early design phase of aircraft development without compromising prediction reliability.

4. CONCLUSIONS

This study establishes a CNN-based framework as a fast and reliable surrogate model for low-Reynolds-number airfoil aerodynamic screening, with particular emphasis on predicting the lift-to-drag ratio (C_L/C_D) of NACA 4-digit airfoils at Re = 50,000. Rather than serving as a direct replacement for high-fidelity solvers, the proposed approach is positioned as an efficient pre-screening tool that captures the dominant aerodynamic trends required in early-stage design and optimization.

The principal contribution of this work lies in demonstrating that a geometry-driven CNN can learn the nonlinear mapping between airfoil shape and aerodynamic performance under low-Reynolds-number conditions while maintaining strong generalization across multiple airfoil subsets. The results show that accurate prediction of both lift and drag coefficients is achievable within a fraction of the computational cost associated with conventional CFD or panel-based solvers. This enables rapid exploration of design spaces that would otherwise be computationally prohibitive, thereby accelerating preliminary aerodynamic assessment and decision-making.

Beyond predictive accuracy, the study highlights important insights into the data–training equilibrium governing neural-network-based aerodynamic models. The analysis reveals that model performance depends not only on dataset size but also on an appropriate balance between training duration and geometric diversity, with early stopping emerging as a critical factor for preventing overfitting while preserving generalization. These findings provide practical guidance for deploying deep-learning surrogates in aerodynamic applications where data availability and computational resources are constrained.

While the proposed framework demonstrates strong performance in attached-flow regimes, the validation against

XFOIL and XFLR5 also clarifies its current limitations in transition-sensitive and near-stall conditions. This transparent identification of predictive boundaries reinforces the suitability of the CNN as a screening-level model rather than a high-fidelity solver.

Future work will focus on several targeted extensions to enhance robustness and applicability. These include incorporating Reynolds number and Mach number as explicit input features, augmenting the training dataset in transition and separation-dominated regimes, and extending validation to non-NACA airfoil families to assess generalization beyond parametric shape classes. Additional benchmarking against experimental data and high-resolution CFD will further strengthen confidence in real-world deployment.

Overall, this study demonstrates that CNN-based surrogate modeling offers a computationally efficient and physically informed pathway for low-Reynolds-number airfoil evaluation, bridging the gap between rapid design screening and high-fidelity aerodynamic analysis.

ACKNOWLEDGMENT

REFERENCES

[1] Bhatnagar, S., Afshar, Y., Pan, S., Duraisamy, K., Kaushik, S. (2019). Prediction of aerodynamic flow fields using convolutional neural networks. Computational Mechanics, 64(2): 525-545. https://doi.org/10.1007/s00466-019-01740-0

[2] Zhang, Y., Sung, W.J., Mavris, D.N. (2018). Application of convolutional neural network to predict airfoil lift coefficient. In 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, Florida, p. 1903. https://doi.org/10.2514/6.2018-1903

[3] Dhileep, K., Kumar, D., Ghosh, S., Faruque Ali, S.A.A. (2020). Numerical study of camber morphing in NACA0012 airfoil. In AIAA Aviation 2020 Forum, p. 2781. https://doi.org/10.2514/6.2020-2781

[4] Dhileep, K., Kumar, D., Vigneswar, P.G., Soni, P., Ghosh, S., Ali, S.F., Arockiarajan, A. (2022). Aerodynamic study of single corrugated variable-camber morphing aerofoil concept. The Aeronautical Journal, 126(1296): 316-344. https://doi.org/10.1017/aer.2021.71

[5] Moin, H., Khan, H.Z.I., Mobeen, S., Riaz, J. (2022). Airfoil's aerodynamic coefficients prediction using artificial neural network. In 2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, pp. 175-182. https://doi.org/10.1109/IBCAST54850.2022.9990112

[6] Chen, H., He, L., Qian, W., Wang, S. (2020). Multiple aerodynamic coefficient prediction of airfoils using a convolutional neural network. Symmetry, 12(4): 544. https://doi.org/10.3390/sym12040544

[7] Liu, D., Chen, D., Li, Q., Xu, X., Peng, X. (2015). Investigation on the correlation of CFD and EFD results for a supercritical wing. International Journal of Heat and Technology, 33(3): 19-26. http://doi.org/10.18280/ijht.330303

[8] Liu, W., Fang, J., Rolfo, S., Moulinec, C., Emerson, D.R. (2021). An iterative machine-learning framework for RANS turbulence modeling. International Journal of Heat and Fluid Flow, 90: 108822. https://doi.org/10.1016/j.ijheatfluidflow.2021.108822

[9] Nugraha, A.H., Suryady, S. (2023). Information system for making PC wire heads with pneumatic systems in heading machine. International Journal Science and Technology, 2(2): 61-65.

[10] Setiyanto, K., Indriyani, A.D. (2025). Comparison of machine learning and deep learning in shopee review sentiment analysis. International Journal Science and Technology, 4(2): 152–167. https://doi.org/10.56127/ijst.v4i2.2225

[11] Alzahrani, M.K., Shapoval, A., Chen, Z., Rahman, S.S. (2023). Pore-GNN: A graph neural network-based framework for predicting flow properties of porous media from micro-CT images. Advances in Geo-Energy Research, 10(1): 39-55. https://doi.org/10.46690/ager.2023.10.05

[12] Chowdhury, I.A. (2024). State-of-the-art CFD simulation: A review of techniques, validation methods, and application scenarios. Journal of Recent Trends in Mechanics, 9: 45-53. https://doi.org/10.46610/JoRTM.2024.v09i02.005

[13] Suryady, S., Soerowirdjo, B., Sari, S.P., Ernastuti. (2025). Impact of combining RGB and grayscale images on hotspot detection in solar panels using Inception Resnet V2 architecture. Ingénierie des Systèmes d'Information, 30(4): 1043-1056. https://doi.org/10.18280/isi.300420

[14] Corcione, S., Cuozzo, C., Cusati, V., De Marco, A., Page, J., Zanon, A. (2019). AI-driven surrogate modeling for iced tailplane aerodynamic prediction. Aerospace Science and Technology, 168(Part E): 111102. https://doi.org/10.1016/j.ast.2025.111102

[15] Muchlis, A., Wibowo, E.P., Irawan, R., Afzeri, A. (2025). YOLOv8 and ResNet-50 based real-time fabric defect detection and quality grading system. Informatica, 49(19): 271-284. https://doi.org/10.31449/inf.v49i19.10031

[16] Nemati, M., Jahangirian, A. (2023). A data-driven machine learning approach for turbulent flow field prediction based on direct computational fluid dynamics database. Journal of Applied Fluid Mechanics, 17(1): 60-74. https://doi.org/10.47176/jafm.17.1.2109

[17] Inada, Y., Li, C. (2023). Optimized wing design of tandem-wing aircraft using microbial genetic algorithm and aerodynamic performance analysis software XFLR5. In Asia-Pacific International Symposium on Aerospace Technology, pp. 1611-1621. https://doi.org/10.1007/978-981-97-4010-9_125

[18] Chen, C.J., Zhang, Z. (2020). Grid: A Python package for field plot phenotyping using aerial images. Remote Sensing, 12(11): 1697. https://doi.org/10.3390/rs12111697

[19] Vrigazova, B. (2021). The proportion for splitting data into training and test set for the bootstrap in classification problems. Business Systems Research: International Journal of the Society for Advancing Innovation and Research in Economy, 12(1): 228-242.

https://doi.org/10.2478/bsrj-2021-0015

[20] Kongsgard, K.W., Nordbotten, N.A., Mancini, F., Haakseth, R., Engelstad, P.E. (2017). Data leakage prevention for secure cross-domain information exchange. IEEE Communications Magazine, 55(10): 37-43. https://doi.org/10.1109/MCOM.2017.1700235

[21] Thuerey, N., Weißenow, K., Prantl, L., Hu, X. (2020). Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows. AIAA Journal, 58(1): 25-36. https://doi.org/10.2514/1.J058291

[22] Zhou, T., Tian, Y., Liao, H., Zhuo, Z. (2023). Computational simulation of molecular separation in liquid phase using membrane systems: Combination of computational fluid dynamics and machine learning.

Case Studies in Thermal Engineering, 44: 102845. https://doi.org/10.1016/j.csite.2023.102845

[23] Jia, Z., Ai, Z., Yang, X., Mak, C.M., Wong, H.M. (2023). Towards an accurate CFD prediction of airflow and dispersion through face mask. Building and Environment, 229: 109932. https://doi.org/10.1016/j.buildenv.2022.109932

[24] Thongnoi, P., Chandra-Ambhorn, W., Chalermsinsuwan, B., Wattananusorn, S., Wongpromrat, P., Bumrungthaichaichan, E. (2024). RANS equation-based gas cyclone separator CFD simulation: An appropriate time step size. Chemical Engineering Transactions, 113: 649-654. https://doi.org/10.3303/CET24113109