Airfoil Trailing-Edge Noise Prediction using Physics-Informed Neural Networks

Kaining Yuan
Department of Computer Science and Engineering
Aspiring Scholars Directed Research Program
Fremont, CA, USA
0009-0007-4633-8794

Sathvik Malla
Department of Computer Science and Engineering
Aspiring Scholars Directed Research Program
Fremont, CA, USA
0009-0009-9835-0677

Varun Ramachandran
Department of Computer Science and Engineering
Aspiring Scholars Directed Research Program
Fremont, CA, USA
0009-0007-5567-110X

Vinamra Singhal
Department of Computer Science and Engineering
Aspiring Scholars Directed Research Program
Fremont, CA, USA
0009-0004-1700-8604

***Abstract*—Predicting airfoil trailing-edge noise accurately is crucial for sound-aware aircraft design, but conventional approaches still rely on time-intensive and expensive Computational Fluid Dynamics (CFD) simulations. This work introduces a Physics-Informed Neural Network (PINN) framework combined with transfer learning (TL), trained on two aeroacoustic datasets from NASA and the University of Bristol, to produce accurate and physically consistent sound pressure level (SPL) predictions across various airfoil shapes. The key input parameters are frequency, angle of attack, chord length, free-stream velocity, radius of microphones from the trailing-edge, Reynolds number, and Strouhal number, plus a compact spline-based airfoil-shape encoding. Compared with a transfer-learning Deep Neural Network (TL-DNN), the TL-PINN achieves better accuracy and generalization, especially under extrapolation, achieving test RMSE below 2 dB. This approach offers a fast and scalable alternative to CFD-based aeroacoustic simulations.**

***Keywords*—Physics-Informed Neural Networks, Aeroacoustics, Airfoil Noise**

I. Introduction

Quiet supersonic travel for civilian use has been a popular topic for many years, and several organizations have already been working toward that goal by developing airplane prototypes. Despite efforts, no commercial supersonic airliners have been made available to the public due to the Federal Aviation Administration (FAA) ban, enacted because of the disruptive sonic booms they would create. The aeroacoustic noise generated by the airfoils pose a critical design consideration for aircraft. Quiet supersonic aircraft such as the X-59 need to be designed using a very inefficient process involving Computational Fluid Dynamics (CFD) simulations to measure noise output of the aircraft. As shown, currently, airfoil noise prediction has relied on CFD simulations, coupled with acoustic solvers such as the Ffowcs Williams-Hawkings equation (FW-H) [1]. Despite its strengths in modeling fluid behavior, the drawbacks of CFD can outweigh these as it can also often be sensitive to meshing strategy, turbulence modeling, and boundary conditions, particularly near the trailing edge where noise is predominantly generated.

Semi-empirical models such as the Brooks-Pope-Marcolini (BPM) model [2] are a lower cost alternative, but their assumptions (canonical NACA sections, attached flow, fixed observer geometry) make them less accurate for modern high-lift shapes or strongly separated regimes.

In recent years, machine learning (ML) has emerged as a viable alternative to noise prediction due to its computational efficiency and ability to learn complex nonlinear patterns from data. For example, Redonnet et al. [3] trained a 14-layer deep neural network (DNN) on 1503 NASA spectra and achieved a 1.5–2.5 dB root mean squared error (RMSE) on unseen configurations. Remarkably, the model still maintained 1.9 dB RMSE when trained on just 50% of the data, outperforming BPM predictions by roughly 60% and delivering predictions in milliseconds. Similarly, Rastgoo et al. [4] demonstrated that gradient-boosted decision trees (GBDTs) could also be effective. Their hybrid CatBoost-AOA model achieved an R^2 of 0.9706 and reduced test RMSE from 2.185 dB to 1.198 dB, outperforming eleven other optimization techniques.

However, purely data-driven neural networks often lack physical interpretability [5], generalizing poorly outside the training process, and in scenarios with limited or noisy data. This challenge highlights the value of PINNs. Introduced by Raissi et al. [6], PINNs represent a significant shift in modeling physical systems by incorporating known physical laws such as conservation equations or partial differential equations (PDEs) directly into the training process. This results in models that can yield physically-consistent predictions even

with sparse data [7].

This paper presents a physics-informed deep learning framework to predict far-field SPL of different airfoils under varying subsonic flow conditions using a combination of two datasets. The developed PINN framework provides a robust foundation for future work in supersonic noise prediction. The model utilizes a fully connected neural network architecture in PyTorch [8], trained with both mean squared error loss and custom physics-informed loss functions.

II. Methodology

A. Experimental Dataset

Two publicly available aeroacoustic datasets from the NASA Langley Research Center [9] and the University of Bristol [10] were used to train the models. While separately obtained, both datasets share compatible input features, making a parallel comparison of turbulent boundary layer trailing edge (TBL-TE) noise possible. Throughout this paper, the two datasets will be referred to as the NASA dataset and Bristol dataset, respectively.

Developed by Brooks et al. [2], the NASA dataset measurements were made in the Quiet Flow Facility with NACA 0012 airfoils tested at many chord lengths (0.0254–0.3048 m), free-stream velocities (up to 71.3 m/s), and angles of attack (up to 25°). The tripping of the boundary layer was employed to isolate TBL-TE noise. Far-field microphones recorded SPL spectra, partially corrected for installation effects such as background noise, flow downwash, and refraction. The dataset includes 106 tripped configurations with 1503 data points, covering both broadband and tonal noise behavior.

The Bristol dataset covers four NACA series airfoils: NACA 0012, NACA 0024, NACA 16506, and NACA 16616 [10]. Anechoic wind tunnel measurements had been performed, which included far-field microphones measurements of surface pressure and SPL spectra. These tests were conducted across a wide range of angles of attack, including pre-stall and post-stall regimes. The dataset provides diversity in airfoil contour and noise dynamics, including stall features.

Although more features were included in the NASA and Bristol datasets, only features that were either directly measured or closely derivable from both datasets were used. By aligning both datasets to a shared schema of frequency (f), angle of attack (α), chord length (c), free-stream velocity (U), and Radius of Microphones from the Trailing-edge (r_e), feature compatibility was ensured. From these, two additional parameters were calculated: the Reynolds number, which accounts for viscous flow effect, and the Strouhal number, which normalizes frequency-based noise behavior across varying flow scales.

B. Airfoil Parameterization

Since the datasets included a variety of airfoils, it was essential for the models to be able to distinguish between them, as shape variations in each airfoil can significantly alter noise generation mechanisms. To achieve generalization across different airfoils, a polynomial-based parameterization strategy capturing the airfoil geometries was used, since inputting hundreds of coordinates would make training very inefficient.

The geometric data for each airfoil were initially provided as a set of discrete (x, y) coordinates that define the airfoil surface. These coordinates were normalized so that the chord length (c) spans from 0 to 1. The leading edge of the airfoil was first identified, which corresponded to the minimum x coordinate, then the airfoil was segmented into upper and lower surfaces. Each surface was independently fitted using a 7th-degree polynomial spline, enabling continuous representations of the airfoil profile. Let the upper and lower surfaces be described by functions $f_u(x)$ and $f_l(x)$, respectively:

$$f_u(x) = \sum_{i=0}^{7} a_i x^i \tag{1}$$

$$f_l(x) = \sum_{i=0}^{7} b_i x^i \tag{2}$$

where a_i and b_i are the polynomial coefficients obtained via least squares fitting. The fitted splines closely approximate the surface contours. These 16 coefficients (8 per surface) were concatenated into a single feature vector and passed as input to the models. Spline fits for the four airfoils used in this study were NACA 0012, 0024, 16506, and 16616, shown in Fig. 1.

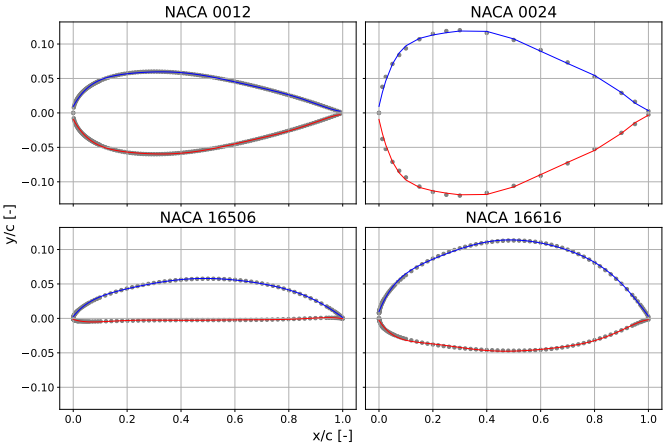

Fig. 1: Spline fits of four airfoils, where the gray dots represent coordinates and the blue and red curves are upper and lower splines respectively.

Compared to directly inputting the coordinates, which can introduce high dimensionality and noise, this polynomial parameterization method captures the key features while reducing the complexity and making it easier for the model to interpret [11].

C. Deep Neural Network (DNN)

A fully connected supervised Deep Neural Network (DNN) was implemented due its ability to approximate the complex mapping between various inputs and the resulting far-field SPL [12]. The input vector consisted of multiple physical and geometric features, including: Frequency (Hz), Angle of

Attack (°), Chord Length (m), Free-stream Velocity (m/s), Microphone Array Radius from Trailing Edge (m), Reynolds Number, and Strouhal Number.

The output was the predicted SPL (in dB) at the observation location. The base model was trained only on the NASA dataset and comprised of a primary branch that processed the seven physical input features and an auxiliary branch that processed the airfoil geometry, parameterized using two 7th-degree polynomials derived from spline fits to the upper and lower surface coordinates. The primary branch consisted of three fully connected layers with 64, 32, and 16 neurons, respectively, each followed by ReLU activation. The airfoil branch was comprised of two hidden layers with 16 and 5 neurons. The outputs of both branches were concatenated and passed through a final two-layer fully connected network to produce the SPL prediction, and the dataset was randomly partitioned 80% / 20% into training and testing sets.

To ensure generalization and training stability, batch normalization was employed within the network. Mean Squared error (MSE) was used as the loss function, where K represents the number of data points and $SPL_k^{[\mathrm{TRUE}]}$ and $SPL_k^{[\mathrm{PRED}]}$ represent the actual and predicted SPL.

$$\mathrm{MSE} = \frac{1}{K}\sum_{k=1}^{K}\left(SPL_k^{[\mathrm{TRUE}]} - SPL_k^{[\mathrm{PRED}]}\right)^2 \tag{3}$$

The network was optimized using the Adam optimizer with a learning rate of 0.001. Training was conducted using mini-batches of size 64 and a maximum of 10,000 epochs. To further improve performance, transfer learning was implemented. The primary branch of the base model was frozen and a new network was initialized for the University of Bristol dataset. Only the airfoil branch and the final fully connected layers were trained (10,000 epochs), allowing the model to use the learned flow representations while adapting to new geometric and acoustic patterns. This approach significantly reduced overfitting and allowed the model to converge faster. Once trained, the models were assessed through the Root Mean Squared error (RMSE).

D. Physics-Informed Neural Network (PINN)

Physics-Informed Neural Networks (PINNs) were applied to improve the SPL prediction from airfoils by embedding known physics laws as constraints. So instead of merely reducing the difference between predicted and observed data, PINNs also minimize the residuals of the governing physics equations, which ensures that predictions remain physically meaningful. The PINN has the same architecture and training/testing process as the DNN described in the previous section, but with the addition of physics-based loss functions.

The combined loss of a PINN is calculated as the sum of the data-driven loss and each individual physics-based constraint:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \mathcal{L}_{\text{PINN}_1} + \mathcal{L}_{\text{PINN}_2} \tag{4}$$

where $\mathcal{L}_{\text{total}}$ and $\mathcal{L}_{\text{data}}$ are the total loss and MSE loss in Equation (3), respectively.

The first PINN loss term was the constraint that acoustic energy must be less than or equal to kinetic energy ($AE \leq KE$). Acoustic energy is not an independent source of energy, and it comes directly from the kinetic energy of the flow. This makes it physically impossible for the acoustic energy to be greater than the available kinetic energy, as that would violate the law of energy conservation. The model was penalized when it over-predicted the SPL, as it could yield ($AE > KE$). To incorporate this idea into a loss function, the torch.ReLU($AE-KE$) function was used. Since $AE-KE$ is normally negative or zero, ReLU would equal to 0, implying no penalty. However, when $AE > KE$, ReLU would be positive, penalizing the model.

The second PINN loss term was directly inspired by two key observations made by Hutcheson et al. [13] in their study of incident turbulence interaction noise. The first was when the abscissa was converted from raw frequency to the Strouhal number, the spectra obtained at different free-stream Mach numbers align horizontally. The second was after shifting every spectrum downward by $10\log_{10} M^6$ (i.e. subtracting a U_∞^6 dependence in decibels, where U_∞ is the free-stream velocity), and the three Mach curves in each figure also collapsed vertically into each other. These two re-scalings condensed flow-speed–dependent spectra into a single curve. In the present work, this qualitative criterion was turned into a quantitative loss function. First, the raw frequency was made non-dimensional by forming the Strouhal number. Simultaneously, the predicted SPL was releveled by subtracting a velocity-dependent offset $10\log_{10} U_\infty/c$. Within every angle-of-attack bucket present in the mini-batch the Strouhal axis was divided into ten equally spaced centers, and for each center the algorithm collects all samples whose Strouhal lies inside a narrow $\pm 3\%$ band, and all points in such a band should share the same SPL, accordingly the sample variance in that band was zero. The loss sums those variances over all populated bands and averages them, producing a single scalar that was strictly positive whenever the predicted spectra fail to collapse. Minimizing this quantity forces the network to reproduce the horizontal and vertical overlap that Hutcheson et al. demonstrated.

E. Transfer Learning

To improve generalization across the two different datasets, transfer learning was employed using a base DNN and a base PINN pre-trained on the NASA dataset. These two base models worked by extracting physical relationships from the NASA dataset, which were later implemented to enhance the new model's performance on the Bristol dataset.

Before training the transfer-learning DNN (TL-DNN) and the transfer-learning PINN (TL-PINN), the Bristol dataset was normalized using scikit-learn's StandardScaler() function [14]. The architecture of transfer learning was to reuse the original input branch of the base models trained on the NASA dataset and to freeze it so that it did not update during training of the new TL-DNN and TL-PINN on the Bristol dataset. After the frozen branch and the new branch containing airfoil

geometry parameters were combined into a fusion branch (a concatenation of both branches), the normalized SPL value was regressed. Freezing the original input branch mitigated overfitting and accelerated convergence, which aligned with the findings of other transfer-learning studies [15], [16].

F. Results and Discussion

Multiple experiments were conducted to evaluate the performance of all four models. Fig. 2 shows the training and testing loss curves for the TL-DNN and TL-PINN.

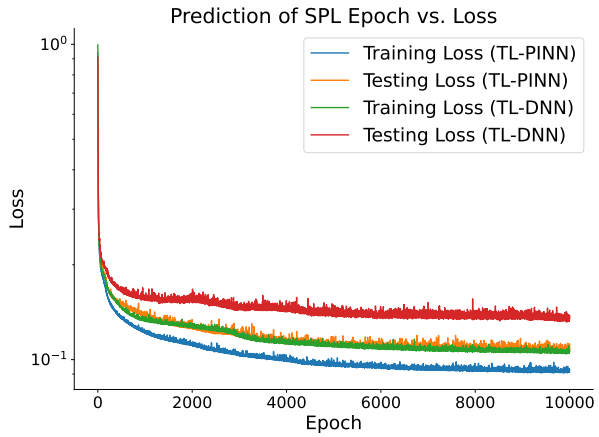

Fig. 2: Training and testing loss curves for TL-PINN and TL-DNN over 10,000 epochs.

In Fig. 2, both models rapidly drive down their losses in the first few hundred epochs and then taper off, but TL-PINN consistently attains a lower floor (reaching around 0.09 train loss and around 0.10 test loss), whereas TL-DNN levels out around 0.10 and 0.13, respectively. This indicates that embedding the physics constraints via PINN not only accelerates convergence but also yields better generalization to unseen data.

Fig. 3 is a scatter plot of the sound pressure level (SPL) as a function of frequency ($0 - 4000Hz$) at an angle of attack (AoA) of $0°$ and $20°$. The actual SPL values are shown as as black x, while the predictions of the TL-DNN and TL-PINN are plotted in blue and orange, respectively. Both models accurately capture the overall decay of SPL versus frequency and track the true data within a few dB across the spectrum.

Fig. 4 illustrates how purely data-driven TL-DNN compares to TL-PINN when asked to predict SPL beyond its training range. The last two frequency values were intentionally hidden during training, and both models were trained on SPL spectra from $0.25\ kHz$ to approximately $2\ kHz$ (solid lines) and then evaluated up to 4 kHz (dashed lines). In the training regime both models tracked the true SPL (black ×) with comparable accuracy. However, once forced to extrapolate, the TL-DNN (blue dashed) developed unphysical oscillations and significant bias, whereas the TL-PINN (orange dashed) retained the expected smooth, monotonic roll-off and stayed much closer to the measurements across all four foil shapes.

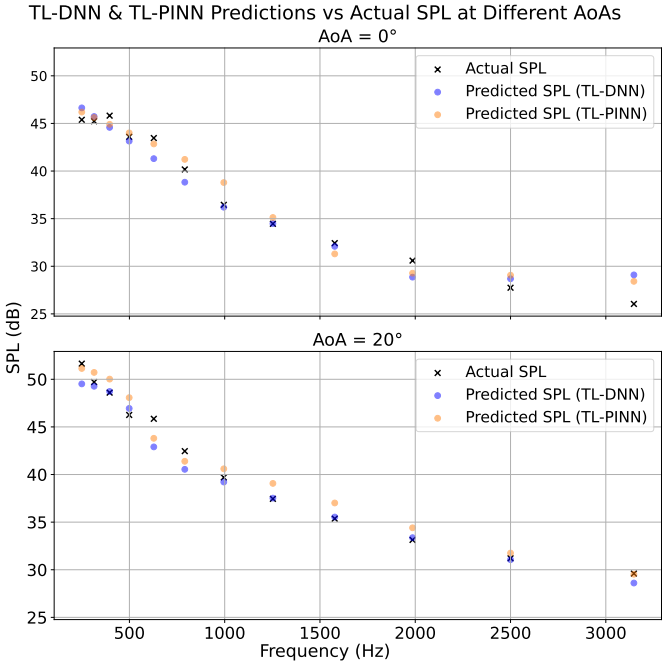

Fig. 3: TL-DNN & TL-PINN SPL Predictions vs. Actual SPL across Frequency at Angles of Attack of $0°$ and $20°$.

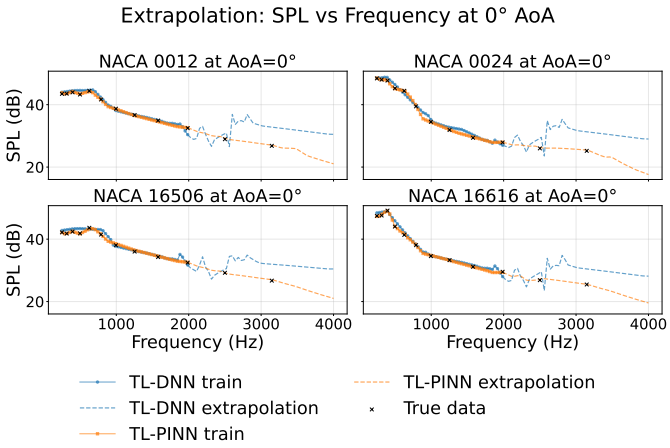

Fig. 4: Extrapolation of TL-DNN vs. TL-PINN in predicting SPL spectra at $0°$ AoA.

This is because when two frequency values were held out of the training data, the TL-DNN's purely data-driven nature lacks the ability to infer SPL behavior in those gaps. As a result, its predictions at the withheld frequencies deviate substantially from the true values, showing worse accuracy than in Fig. 3 when the two values were not hidden. This behavior confirmed that embedding the governing physics into the neural network enhances the models' generalization ability.

Fig. 5 is a 2D heatmap of SPL as a function of frequency ($0 - 4000\ Hz$) and angle of attack ($-5°$ to $30°$) with dashed white lines marking the edges of the training domain (AoA = $25°$, $f = 3147\ Hz$). The TL-PINN shows coherent and

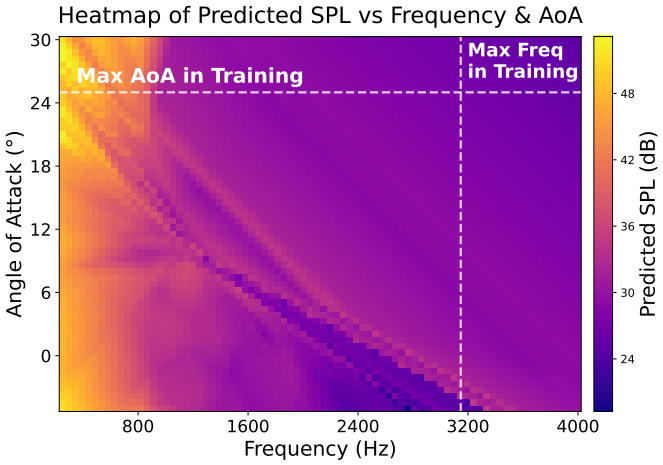

Fig. 5: 2D heatmap of TL-PINN predicted SPL as a function of frequency ($0-4000\ Hz$) and angle of attack ($-5°$ to $30°$), with dashed white lines marking the training domain limits.

smooth variation of SPL beyond both boundaries, free from any indication of discontinuities or unphysical oscillations in the extrapolated regions. A deep diagonal hump forms from approximately (AoA = $25°$, $f = 700\ Hz$) through to (AoA = $0°$, $f = 2800\ Hz$), characteristic of tonal noise response due to laminar boundary-layer instability and vortex shedding. Frequency–AoA variability observed is in the pattern of an inverse Strouhal scaling trend with increasing flow separation at higher angles.

TABLE I: RMSE values for the DNN, PINN, TL-DNN, and TL-PINN

	DNN	PINN	TL-DNN	TL-PINN
Train RMSE (dB)	6.963	3.359	1.974	1.889
Test RMSE (dB)	6.682	3.792	2.087	1.865

Table I illustrates the RMSE values across all models, with lower values indicating better model performance. Both transfer-learning counterparts perform better than the base models themselves, demonstrating improved accuracy due to transfer learning. Both PINNs achieved lower RMSE values than their DNN equivalents, demonstrating better performance and highlighting improved predictions due to the physics-informed loss functions.

III. Conclusion

This work demonstrates a PINN-based framework for predicting airfoil trailing-edge noise by embedding physics-based constraints and spline-derived airfoil geometry into the learning process. The TL-PINN was trained on the NASA and Bristol datasets using transfer learning, reaching RMSE below 2 dB over broad ranges of input regimes, comparable to the accuracy achieved by the CatBoost models developed by Rastgoo et al. [4] or the DNN developed by Redonnet et al. [3]. The TL-PINN captures flow-dependent acoustic behavior even in data-sparse regimes. This scalable and simulation-free method lays the groundwork for efficient, noise-aware airfoil designs.

This framework can be integrated into broader aerodynamic optimization pipelines or early-stage airfoil screening tools. These results suggest that PINNs have the potential to be well-suited for fast iteration in real-world design scenarios, where the cost and duration of traditional CFD simulations remain major challenges.

Acknowledgment

We would like to express our sincere gratitude to Chris DeGrendele for his guidance and mentorship, which were instrumental to the research.

References

[1] J. E. Ffowcs Williams, D. L. Hawkings, and M. J. Lighthill, "Sound generation by turbulence and surfaces in arbitrary motion," pp. 321–342, 1969.

[2] T. F. Brooks, D. S. Pope, and M. A. Marcolini, "Airfoil self-noise and prediction," Tech. Rep., 1989.

[3] S. Redonnet, T. Bose, A. Seth, and L. K. Li, "Airfoil self-noise prediction using deep neural networks," *Engineering Analysis with Boundary Elements*, vol. 159, pp. 180–191, 2024.

[4] A. Rastgoo and H. Khajavi, "A novel study on forecasting the airfoil self-noise, using a hybrid model based on the combination of catboost and arithmetic optimization algorithm," *Expert Systems with Applications*, vol. 229, p. 120576, 2023.

[5] M. Kohler and A. Krzyzak, "Over-parametrized deep neural networks do not generalize well," 2020.

[6] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[7] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris, "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data," *Journal of Computational Physics*, vol. 394, pp. 56–81, 2019.

[8] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," 2019. [Online]. Available: https://arxiv.org/abs/1912.01703

[9] P. D. Brooks, Thomas and M. Marcolini, "Airfoil Self-Noise," UCI Machine Learning Repository, 1989, DOI: https://doi.org/10.24432/C5VW2C.

[10] J. Branch, N. Zang, and M. Azarpeyvand, "Aeroacoustic data for four naca series aerofoils," 2023. [Online]. Available: https://data.bris.ac.uk/data/dataset/30c8q39wi1mxf205flnun3l5rl/

[11] W. Chen, K. Chiu, and M. Fuge, "Airfoil design parameterization and optimization using bézier generative adversarial networks," 2020.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436–444, May 2015.

[13] F. V. Hutcheson, T. F. Brooks, and D. J. Stead, "Measurement of the noise resulting from the interaction of turbulence with a lifting surface," pp. 675–700, 2012.

[14] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[15] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014.

[16] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.