Integrating Fourier Neural Operators into High-Fidelity Helicopter Flight Simulation for Real-Time Urban Wind Prediction

Maximilian Dauner[1], Michael Kurz[2], Gudrun Socher[1], Alexander Knoll[2]

[1]Munich Center for Digital Sciences and AI
[2]Hochschule München University of Applied Sciences
{maximilian.dauner0, michael.kurz, gudrun.socher, alexander.knoll}@hm.edu

Abstract

High-fidelity helicopter flight simulators are essential for preparing pilots for complex and hazardous environments, yet realistic urban wind dynamics are difficult to reproduce in real time when relying on precomputed computational fluid dynamics (CFD) data. We present the first integration of a Fourier Neural Operator (FNO) into a Level D full flight simulator for real-time, physics-based urban wind field generation. Trained on high-resolution urban flow simulations, the FNO predicts one-minute-averaged 3D wind fields that dynamically adapt to flight state and location, replacing static wind inputs in the simulator pipeline. Turbulence levels are computed from the predictions and injected directly into the simulation loop. Professional pilots evaluated the system in an urban scenario and reported that it reproduced wind effects they would expect, such as turbulence and directional changes when landing behind buildings. They highlighted its value for less experienced pilots to develop wind awareness and for realistic training in critical operations, including offshore platform landings.

Introduction

Many problems in science and engineering can be formulated as partial differential equations (PDEs) and solved numerically. Among the most widely used discretization schemes are the finite-element method (FEM) (Larson and Bengzon 2013), the finite-volume method (FVM) (Moukalled, Mangani, and Darwish 2016), and the finite-difference method (FDM) (MITCHELL and Griffiths 1980). These techniques underpin the simulation of various physical phenomena, including structural and material mechanics, incompressible and compressible flows, numerical weather prediction, and heat conduction, at industrial and research scales (Liu, Li, and Park 2021; Würth et al. 2024). Despite continuing advances in high-performance computing (HPC) hardware, a pronounced trade-off persists. Coarser spatial or temporal discretizations shorten runtimes and reduce computational cost, but at the expense of accuracy. In contrast, finer meshes or smaller time steps improve fidelity while incurring substantially higher cost. This dilemma is especially acute when real-time or near-real-time predictions are required. Reducing wall-clock time to seconds with state-of-the-art solvers either degrades accuracy or demands prohibitive HPC resources (Ashton et al. 2024).

In parallel with hardware developments, deep-learning methods have advanced significantly, offering large speed-ups, lower operational cost, and competitive accuracy for PDE problems (Würth et al. 2024; Qin et al. 2025). Current research on neural solvers falls into three broad categories:
Hybrid models accelerate conventional solvers by replacing the most expensive algorithmic kernels with learned surrogates or by correcting discretization error (Um et al. 2020). While highly effective when tailored to a specific PDE, they often generalize poorly to new equations or boundary conditions (Tran et al. 2023).
Physics-informed models, typified by Physics-Informed Neural Networks (PINNs) (Raissi, Perdikaris, and Karniadakis 2019; Cuomo et al. 2022), embed the governing PDEs or conservation laws directly into the loss function. By enforcing physical consistency during training, these models achieve higher accuracy in data-scarce settings (Qin et al. 2025; Würth et al. 2024; Ashton et al. 2024).
Fully data-driven models avoid explicit physical constraints and learn directly from large collections of historical simulations or measurements. Although they require more training data than physics-informed approaches, they avoid the simplifying assumptions inherent in many traditional models and can capture complex real-world dynamics (Qin et al. 2025; Ashton et al. 2024; Li et al. 2021).

Within the fully data-driven class of neural network models, the Fourier Neural Operator (FNO) (Li et al. 2021) has emerged as a state-of-the-art architecture for efficiently solving PDEs. FNOs learn resolution-invariant operators capable of generalizing across diverse PDE instances and support zero-shot super-resolution. Consequently, a model trained on coarse spatial discretizations can generalize to finer resolutions without additional training, providing significant computational advantages and flexibility (Li et al. 2021). These features make FNOs particularly advantageous for real-time simulation tasks where fast inference and adaptability to varying conditions are essential. Recently, FNOs have been successfully employed in related flow physics applications, ranging from city-scale and pedestrian-level wind prediction (Qin et al. 2025; Peng et al. 2024; Chen et al. 2025) to high Reynolds number turbulence closure (Wang

et al. 2024; Luo et al. 2024) and accelerated atmospheric forecasting from Weather Research and Forecasting (WRF) model data (Costa Rocha et al. 2025).

Besides those applications a compelling deployment of FNOs would be their integration into flight simulators for real-time prediction and simulation of realistic wind conditions, thereby enhancing pilot training with dynamically evolving atmospheric scenarios. Pilot training requires exposure to challenging scenarios that are either too hazardous or expensive to replicate with real aircraft. Modern flight simulators offer a safe, controlled and cost-effective environment to practice such critical situations. Integrating advanced AI-driven wind prediction models, such as FNO, into these simulators significantly enhances the realism and educational value of training exercises. Research indicates that realistic simulator-based training significantly reduces learning curves in actual flight conditions, improves pilot proficiency, and allows trainees to safely practice emergency maneuvers, adverse weather scenarios, or system malfunctions that are otherwise impractical or unsafe in real-world conditions (Sabadas et al. 2024; White et al. 2021; Rantanen and Talleur 2005). A critical factor influencing the realism of flight simulation is the accurate representation of environmental conditions such as wind, turbulence, and weather phenomena. Recent research efforts have emphasized integrating high-fidelity atmospheric models into helicopter simulators, enabling pilots to experience and respond to dynamic and realistic wind conditions in real-time. Leveraging pre-computed CFD flow fields, these studies stream meter-scale, four-dimensional wind grids into the pilot loop, using a PALM LES solver over complex terrain (Liu et al. 2022), turbine-wake CFD for offshore operations (Štrbac et al. 2022), and high-resolution terrain resolved CFD for canyon environments (Watson et al. 2025). While integrating precomputed CFD-based wind fields into flight simulators can reproduce coherent gust structures, increase pilot control workload, and enhance training realism, these approaches rely on replaying pre-generated scenarios. Due to the high computational cost of CFD simulations, wind fields must be generated in advance and cannot be dynamically updated during runtime. As a result, such methods are limited to a fixed set of wind and terrain scenarios, reducing their ability to capture the variability and complexity of real-world environments.

To address these limitations, we propose the integration of an FNO into the flight simulator pipeline for real-time wind field generation. In contrast to traditional methods that rely on precomputed CFD data, our approach generates realistic urban wind fields in real time, enabling dynamic adaptation to changing flight conditions. This allows helicopter pilots to experience a wider range of landing scenarios under varied and physically plausible wind conditions, improving both the variability and realism of training exercises. To the best of our knowledge, this is the first work to integrate an FNO, or any other machine learning model, into a flight simulator for real-time, physics-based wind simulation in pilot training.

Overall System Architecture

Figure 1 depicts the runtime architecture for integrating neural, real-time wind-field generation into a Level D full flight simulator (Reiser Simulation and Training, H145). A Factorized Fourier Neural Operator (F–FNO) (Tran et al. 2023) forecasts minute-mean 3D wind fields autoregressively, rolling out $\hat{W}_{t+1} = f_\theta(\hat{W}_t)$ from an initial field W_{t_0}. Because each $\hat{W}_t$ denotes a one-minute average, a background service publishes the current field once per minute for all downstream components.

The simulator streams aircraft state via the User Datagram Protocol (UDP)—position (longitude, latitude, altitude; WGS–84), attitude (roll, pitch, yaw), and velocity in the North–East–Down (NED) frame. The aircraft position is mapped to the Cartesian grid used by the F–FNO. The wind vector $\mathbf{w} = (u, v, w)$ is then sampled at the helicopter's location from the latest field, and both $\mathbf{w}$ and a discrete turbulence level $\tau \in \{0, 1, 2, 3\}$ (none to severe), computed from wind-field features (e.g., magnitude, directional variability, and up-/downwash), are returned to the simulator via UDP.

The same urban geometry used for model training is embedded in the simulator to ensure geometric consistency between the prediction domain and the visual/physics environment. Implementation details on coordinate transforms, training, and turbulence estimation are provided in subsequent sections.

Flight Simulator Environment

The experiments are conducted on a high-fidelity, Level D full flight simulator for the H145 helicopter by Reiser Simulation and Training GmbH. Level D certification, as defined by EASA and FAA regulations, denotes the highest standard of realism in flight simulation. The device provides six-degrees-of-freedom (6-DoF) motion, high-fidelity visual systems, and a fully replicated cockpit, enabling pilot training under realistic and dynamic conditions, including system malfunctions, adverse weather (e.g., strong winds), and emergency procedures.

The software architecture is modular and organized into three main simulation components (MSCs):

- **SimShip MSC**: flight dynamics, motion system, cockpit interfaces, control loading, aural cues, and visual output.
- **World MSC**: environment-level data, including weather, air traffic, and navigation databases.
- **Simulator MSC**: terrain rendering, projector pipelines, and external data feeds.

In the proposed setup, conventional replay-based wind inputs are replaced by F–FNO-based dynamic 3D wind fields that are updated at runtime and sampled at the aircraft position each frame. The minute-mean wind fields are published once per minute and injected into the simulator via the World MSC, while visualizations are presented through the Out-the-Window (OTW) rendering system.

To enhance training, the simulator can overlay hazardous wind regions derived from the predicted fields as volumetric keep-out, referred to as geofences. These are rendered in real time using Unreal Engine 5 and, when a virtual-reality

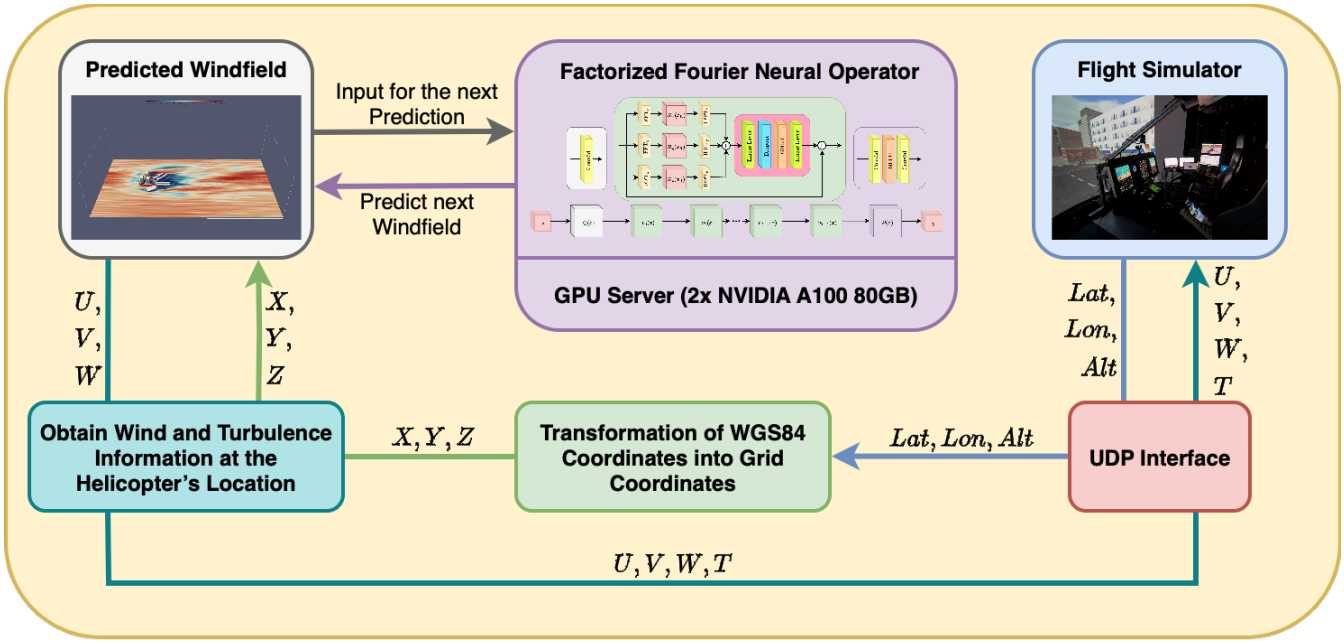

Figure 1: Runtime architecture of the proposed system. A F–FNO forecasts one-minute mean wind fields autoregressively, which are streamed into the Level D helicopter flight simulator in real time. Aircraft state is transmitted via UDP, mapped to the F–FNO grid, and used to sample local wind vectors and turbulence levels. The outputs are injected back into the simulator to ensure geometric consistency and realistic pilot-in-the-loop wind dynamics.

(VR) headset is employed, appear directly in the pilot's field of view to support planning and approach-phase situational awareness.

The framework further includes a custom geofencing and hazard-navigation module driven by the F–FNO predictions. Helicopter state is received via the UDP interface; hazard volumes are computed from wind-field features, encoded as STL geometry, streamed to the renderer, and displayed as dynamic 3D polygons. The module currently operates as a standalone visual cue and is designed for future integration as a cockpit overlay to improve pilot awareness.

Neural Network Based Wind Simulation

To integrate realistic urban wind dynamics into flight simulators, a neural operator framework is employed to learn resolution-agnostic mappings from urban geometries to three-dimensional wind fields. The next subsection outlines the architecture of the neural network, designed for efficient spectral learning in 3D domains as well as the training setup based on simulated wind fields in an urban environment, which provides the high-fidelity data necessary for robust and geometry-aware predictions.

Operator-Learning Setup and Architecture

In this work, urban wind forecasting is framed as operator learning. Instead of predicting point values on a fixed mesh, the approach learns a resolution-invariant map between function spaces that can be evaluated in real time inside a pilot-in-the-loop simulator. Let $D \subset \mathbb{R}^d$ be a bounded spatial domain and consider separable Banach spaces $\mathcal{A} = \mathcal{A}(D;\mathbb{R}^{d_a})$ (inputs) and $\mathcal{U} = \mathcal{U}(D;\mathbb{R}^{d_u})$ (outputs). Given i.i.d. samples $(a_j, u_j)j = 1^N$ with $u_j = G^\dagger(a_j) + \xi_j$, where $G^\dagger : \mathcal{A} \to \mathcal{U}$ is the unknown target operator and ξ_j denotes simulation noise, a parametric operator $G\theta : \mathcal{A} \to \mathcal{U}$ is learned. Discretization invariance is enforced by evaluating inputs and outputs at arbitrary point sets $D_n = x_i{}_{i=1}^n \subset D$, enabling the same model to serve different grids and cropped subdomains at runtime.

The operator G_θ is instantiated with a FNO (Li et al. 2021), which implements nonlocal coupling via spectral (frequency-domain) convolution and therefore generalizes across mesh resolutions. Concretely, let $v_t \in \mathbb{R}^{n\times H}$ be the hidden field at depth t on a discretization of size n with hidden width H. A standard FNO layer updates

$$v_{t+1}(x) = \sigma\Big(W\, v_t(x) \;+\; \mathcal{F}^{-1}\big(R_\phi \odot \mathcal{F}v_t\big)(x)\Big), \qquad (1)$$

where $W : \mathbb{R}^H \to \mathbb{R}^H$ is a pointwise linear map, σ is a nonlinearity, $\mathcal{F}/\mathcal{F}^{-1}$ denote the (discrete) Fourier transform/inverse Fourier transform, R_ϕ are learnable complex spectral weights restricted to the top M modes per axis, and $\odot$ denotes channelwise multiplication in frequency space.

To scale to 3D flow while maintaining real-time inference, the F–FNO (Tran et al. 2023) is employed. F–FNO factorizes the spectral operator along each spatial axis to reduce parameters and memory from $\mathcal{O}(H^2M^d)$ to $\mathcal{O}(H^2Md)$ (with spatial dimension d, hidden width H, and retained modes M):

$$(\mathcal{K}_\phi v_t)(x) = \sum_{i=1}^{d} \mathcal{F}_i^{-1}\Big(R_{\phi,i} \odot \mathcal{F}_i v_t\Big)(x), \qquad (2)$$

Figure 2: Full flight simulator architecture and visualization pipeline. The Level D H145 simulator provides six degrees of freedom motion, high-fidelity out-the-window (OTW) visuals, and a fully replicated cockpit. Hazardous wind regions are depicted as volumetric geofences, rendered in real time via OTW and optional VR (Unreal Engine 5). The pilot's headset visualization supports operational awareness and avoidance of critical areas. Aircraft state is received via UDP; hazard volumes are derived from wind-field features, encoded as STL, and streamed for display, furnishing the pilot with an additional source of turbulence information.

with $R_{\phi,i} \in \mathbb{C}^{M\times H\times H}$. Each operator block applies this factorized spectral update followed by a two-layer feed-forward transformation with post-activation residual connection for stability in deep stacks:

$$v_{t+1}(x) = v_t(x) + \sigma\Big(W_2\,\sigma\big(W_1\,(\mathcal{K}_\phi v_t)(x)\big)\Big). \qquad (3)$$

A lifting layer $\mathcal{P} : \mathbb{R}^{d_a} \to \mathbb{R}^H$ maps inputs into the hidden width; T operator blocks (Eq. 3) produce v_T, which is projected to the output via $\mathcal{Q} : \mathbb{R}^H \to \mathbb{R}^{d_u}$. Minute-averaged 3D wind fields are rolled out autoregressively, $\hat{W}t+1 = f\theta(\hat{W}_t)$, and the results are published to the flight-simulation pipeline each minute.

Dataset and Training

The urban wind dataset of Rienecker, Hildebrand, and Pfifer (2023), generated with the PALM (Maronga et al. 2020) large-eddy simulation (LES) system (Zhiyin 2015), is used. The scene represents a canonical European city block with eight buildings: three residential towers (50 m), four terraced houses (20 m), and one supermarket (15 m), designed to capture typical local wind effects around structures. The computational domain spans $1125 \times 600 \times 250$ m on an equidistant Cartesian grid with $\Delta x = \Delta y = \Delta z = 2.5$ m ($450\times240\times100$ cells). The same urban geometry is embedded in the simulator at identical scale, enabling a one-to-one mapping from the model grid to the simulator world coordinates and sampling of predicted winds at the helicopter's current position.

A single westerly inflow direction is simulated at three representative free-stream speeds, 6.5, 8.3, and 9.9 m/s, approximating the minimum, mean, and maximum values from a one-year record at 10 m height in Dresden, Germany. Wind fields are stored as one-minute averages and serve as training targets for the F–FNO.

Each training sample contains the minute-mean 3D velocity components (u, v, w), a binary occupancy mask for buildings/terrain, and normalized coordinates $(x, y, z) \in [0, 1]^3$ to facilitate geometry-aware and resolution-agnostic learning. Mini-batches draw random subvolumes of size $64 \times 64 \times 32$ to accelerate training. Data augmentation includes rotations about the vertical (yaw) axis and small additive Gaussian noise on input channels to improve robustness to heading changes and rollout imperfections.

During training, the model receives a mini-batch of augmented subvolumes and is trained to autoregressively predict the next $H = 5$ time steps for each sample. Optimization is performed using AdamW with a cosine learning-rate schedule and gradient clipping. The loss balances pointwise accuracy, spectral fidelity, and approximate incompressibil-

ity over the rollout horizon:

$$\begin{aligned}\mathcal{L} = &\|\hat{W}_{t+1} - W_{t+1}\|_2^2 \\ &+ \lambda_{\text{spec}} \big\|\mathcal{F}(\hat{W}_{t+1}) - \mathcal{F}(W_{t+1})\big\|_2^2 \\ &+ \lambda_{\text{div}} \|\nabla \cdot \hat{W}_{t+1}\|_2^2,\end{aligned} \tag{4}$$

where the spectral term promotes preservation of coherent structures at retained modes and the divergence penalty discourages spurious compressibility. After convergence, the composite validation loss stabilized at $\mathcal{L} \approx 0.094$, comprising a pointwise MSE component of 0.058, a weighted spectral component of 0.028, and a weighted divergence penalty of 0.008. The pointwise MSE accounts for the largest portion of the total loss, as it directly quantifies deviations from the LES reference snapshots. The spectral term contributes moderately by promoting the preservation of coherent flow structures without overpowering the optimization dynamics.

Real-Time Integration of Learned Wind Predictions into the Flight Simulator

At inference, a single F–FNO step autoregressively produces the next minute-mean wind field, $\hat{W}_{t+1} = f_\theta(\hat{W}_t)$, starting from an initial field W_0. In deployment, inference runs on a GPU server equipped with two NVIDIA A100 GPUs (80 GB each). A background process publishes the current wind field once per minute, ensuring that the wind information sampled at the helicopter's position always corresponds to the latest minute and evolves in real time.

To seamlessly couple the predicted wind fields with the pilot-in-the-loop simulator, geodetic coordinates are transformed to the network grid in three stages. First, the simulator's WGS–84 latitude, longitude, and altitude are converted to Earth-Centered, Earth-Fixed (ECEF) coordinates. Second, ECEF vectors are mapped to a local East–North–Up (ENU) frame centered at a designated reference point within the simulated urban area. Third, the ENU coordinates are rotated according to the simulator heading and scaled into the equidistant Cartesian grid of the F–FNO. This transformation establishes a consistent correspondence between the aircraft's position in the simulation and the discretized prediction domain, enabling wind vectors to be queried and injected into the simulator in real time.

The simulator publishes aircraft state via a predefined UDP interface, including latitude, longitude, altitude, velocity, and attitude (roll, pitch, yaw). This information is transformed into the F–FNO grid coordinates, the local wind vector $\mathbf{w} = (u, v, w)$ is sampled from the latest published field, and $\mathbf{w}$ is returned to the simulator via UDP.

In addition to the 3D wind vector, the system computes a discrete turbulence level $\tau \in \{0, 1, 2, 3\}$ (none to severe) based on wind-field features relevant to rotorcraft operations, including turbulence intensity, gust amplitude, wind shear, up- and downdrafts, vorticity, and absolute wind speed. Each feature is thresholded against safety-critical values derived from regulatory and empirical sources (Wolfson et al. 1994; Federal Aviation Administration 2008) and adjusted to aircraft-specific characteristics (H145), such as mass, maneuverability, handling qualities, and available energy margin. After each minute-mean wind field is published, the algorithm evaluates these criteria across the grid to obtain τ at all locations. During the simulation, the current wind vector and the corresponding turbulence level at the helicopter's position are forwarded via UDP and applied within the simulator's environment model.

Pilot-in-the-Loop Flight Test Scenarios and Evaluation

To assess the realism of the setup with the integrated F–FNO-predicted wind fields in the full flight simulator, multiple helicopter pilots conducted test sessions. Several flight scenarios were designed in which pilots typically operate in urban environments during medical emergency missions. These included landings behind and on top of buildings, where complex wind conditions often arise. The scenarios were executed in the simulator environment with the integration of the F–FNO-based wind field predictions.

To complement the real-time wind field simulation, a detection and visualization pipeline was implemented. This pipeline identifies hazardous regions in terms of critical wind conditions and represents them as three-dimensional geofences. For the calculation of turbulence-related hazard factors, the relevant features are combined into a unified mask, which is then transformed into geofences. The process involves surface extraction using marching cubes, clustering with DBSCAN, and hull construction via α-shapes:

$$\mathcal{P}_k = \alpha\text{-Shape}\big(\mathcal{V}_k,\ \alpha\big), \tag{5}$$

The resulting volumes are streamed into the simulator environment and overlaid in real time, providing pilots with intuitive awareness of hazardous wind regions and enabling training for safe navigation in complex urban conditions. All test pilots flew multiple routes and landing approaches under varying wind conditions. During the flights, they also evaluated different visualization strategies to support recognition of critical wind field scenarios.

After completing the flights, all test pilots filled out a questionnaire regarding the realism of the simulated wind as well as the usefulness of the visualization techniques. They unanimously rated the F–FNO-simulated wind as highly realistic. The pilots agreed that the vortices and turbulence experienced during the landings matched what they would expect in real-world conditions. Typical wind phenomena behind buildings, which strongly influence landing maneuvers, were realistically captured and integrated into the simulation by the F–FNO. The pilots further emphasized that such realistically generated wind fields are highly valuable for training difficult landing scenarios, such as offshore platforms or rooftop hospital helipads. In particular, they noted that younger and less experienced pilots could significantly benefit from exposure to these complex wind conditions during training.

The continuous in-flight visualization of critical turbulence was, however, perceived as distracting. The pilots suggested that the visualization is more useful either before the flight, or shortly before landing as part of planning and situational preparation. They also highlighted its potential in the

training of novice pilots. In contrast, a permanent visualization via a head-mounted display was considered distracting and insufficiently precise for real-time flight assistance.

Conclusion, Limitations and Future Work

This paper presented the first integration of an FNO-based network, specifically a Factorized Fourier Neural Operator, into a certified Level D helicopter flight simulator for real-time, physics-based wind field generation. The proposed approach replaces static or precomputed CFD inputs with dynamically evolving 3D wind predictions, which are sampled at the helicopter's position and injected into the simulation. Professional pilots evaluated the system in urban landing scenarios and unanimously confirmed the realism of the generated wind fields, highlighting that turbulence and vortices behind buildings matched their operational expectations. They emphasized the value of such simulations for training, especially in difficult scenarios such as rooftop hospital landings or offshore platforms, and noted the particular benefit for younger or less experienced pilots. While continuous in-flight visualization of hazardous regions was considered distracting, the pilots agreed that selective visualization before or during approach can support situational awareness and enhance training effectiveness.

Despite these encouraging results, several limitations remain. First, although the F–FNO architecture is inherently resolution-invariant and can, in principle, generalize across grids and domains, the present training dataset is restricted to a single stylized urban neighborhood. As a result, the generalizability to diverse and realistic city geometries is limited, and arbitrary urban areas worldwide cannot yet be simulated without retraining. Second, the dataset contains only three wind inflow speeds and a single dominant direction, which constrains the variability of predicted wind scenarios. Third, because the training data consist of one-minute averages, the neural operator reproduces only minute-mean wind fields. Higher temporal resolutions, which are crucial for capturing sub-minute gusts and transient turbulence, are not possible with the current data.

Beyond the documented data constraints, several additional factors currently limit deployment and generalizability. First, the presented system does not yet incorporate explicit uncertainty quantification or out-of-distribution detection. When faced with urban layouts, inflow conditions, or operating regimes that differ substantially from the training corpus, F–FNO may yield smooth but physically implausible wind fields. This risk is addressed in the present prototype by restricting usage to validated scenarios and by providing a fallback to a static wind model outside the trained domain. Second, the pilot-in-the-loop evaluation is preliminary: a limited number of professional helicopter pilots and scenario types were considered, and their assessments were collected using mainly subjective ratings. The reported realism should therefore be interpreted as an initial feasibility indication rather than a comprehensive validation across aircraft types, weather conditions and training curricula. Third, hazardous region visualization is provided as a research prototype rather than an operational aid. Continuous VR display of geofences was perceived as distracting by test pilots, indicating that these visualizations are more suitable for preflight and approach preparation than for persistent cockpit overlays in safety-critical operations.

These limitations directly inform our roadmap for future work. Over the next six months, we plan to conduct additional pilot-in-the-loop test sessions to extend feedback on usability, identify shortcomings in visualization strategies, and refine integration into operational training. In parallel, the training dataset is being expanded to include multiple realistic and heterogeneous urban neighborhoods, with a broader range of inflow directions and wind speeds. A key objective is to achieve finer temporal resolution by moving beyond minute-averaged data, thereby enabling the neural operator to reproduce short-term gust dynamics critical for rotorcraft safety. Taken together, these steps establish a clear trajectory towards integrating FNO-based wind field generation as a default component of flight simulators, enabling pilots to systematically train under realistic, data-driven wind conditions across a wide variety of operational scenarios. Ultimately, this work lays the foundation for next-generation pilot training systems where realistic, data-driven wind dynamics become a standard feature of flight simulators rather than an experimental add-on.

References

Ashton, N.; Maddix, D. C.; Gundry, S.; and Shabestari, P. M. 2024. AhmedML: High-Fidelity Computational Fluid Dynamics Dataset for Incompressible, Low-Speed Bluff Body Aerodynamics. arXiv:2407.20801.

Chen, C.; Tian, G.; Qin, S.; Yang, S.; Geng, D.; Zhan, D.; Yang, J.; Vidal, D.; and Wang, L. L. 2025. Generalization of Urban Wind Environment Using Fourier Neural Operator Across Different Wind Directions and Cities. arXiv:2501.05499.

Costa Rocha, P. A.; Thé, J. V. G.; Oliveira Santos, V.; and Gharabaghi, B. 2025. Deep Learning for Atmospheric Modeling: A Proof of Concept Using a Fourier Neural Operator on WRF Data to Accelerate Transient Wind Forecasting at Multiple Altitudes. *Atmosphere*, 16(4).

Cuomo, S.; di Cola, V. S.; Giampaolo, F.; Rozza, G.; Raissi, M.; and Piccialli, F. 2022. Scientific Machine Learning through Physics-Informed Neural Networks: Where we are and What's next. arXiv:2201.05624.

Federal Aviation Administration. 2008. Wind Shear. FAA Safety Brochure P–8740-40. Accessed 22 May 2025.

Larson, M. G.; and Bengzon, F. 2013. *The Finite Element Method: Theory, Implementation, and Applications*, volume 10 of *Texts in Computational Science and Engineering*. Springer Berlin, Heidelberg, 1 edition. ISBN 978-3-642-33286-9.

Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; and Anandkumar, A. 2021. Fourier Neural Operator for Parametric Partial Differential Equations. arXiv:2010.08895.

Liu, W. K.; Li, S.; and Park, H. 2021. Eighty Years of the Finite Element Method: Birth, Evolution, and Future. arXiv:2107.04960.

Liu, X.; Abà, A.; Capone, P.; Manfriani, L.; and Fu, Y. 2022. Atmospheric Disturbance Modelling for a Piloted Flight Simulation Study of Airplane Safety Envelope over Complex Terrain. *Aerospace*, 9(2).

Luo, T.; Li, Z.; Yuan, Z.; Peng, W.; Liu, T.; Wang, L. L.; and Wang, J. 2024. Fourier neural operator for large eddy simulation of compressible Rayleigh–Taylor turbulence. *Physics of Fluids*, 36(7): 075165.

Maronga, B.; Banzhaf, S.; Burmeister, C.; Esch, T.; Forkel, R.; Fröhlich, D.; Fuka, V.; Gehrke, K. F.; Geletič, J.; Giersch, S.; Gronemeier, T.; Groß, G.; Heldens, W.; Hellsten, A.; Hoffmann, F.; Inagaki, A.; Kadasch, E.; Kanani-Sühring, F.; Ketelsen, K.; Khan, B. A.; Knigge, C.; Knoop, H.; Krč, P.; Kurppa, M.; Maamari, H.; Matzarakis, A.; Mauder, M.; Pallasch, M.; Pavlik, D.; Pfafferott, J.; Resler, J.; Rissmann, S.; Russo, E.; Salim, M.; Schrempf, M.; Schwenkel, J.; Seckmeyer, G.; Schubert, S.; Sühring, M.; von Tils, R.; Vollmer, L.; Ward, S.; Witha, B.; Wurps, H.; Zeidler, J.; and Raasch, S. 2020. Overview of the PALM model system 6.0. *Geoscientific Model Development*, 13(3): 1335–1372.

MITCHELL, A.; and Griffiths, D. 1980. *The Finite Difference Method in Partial Differential Equations*. John Wiley & Sons.

Moukalled, F.; Mangani, L.; and Darwish, M. 2016. *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab*, volume 113 of *Fluid Mechanics and Its Applications*. Springer Cham, 1 edition. ISBN 978-3-319-16873-9.

Peng, W.; Qin, S.; Yang, S.; Wang, J.; Liu, X.; and Wang, L. L. 2024. Fourier neural operator for real-time simulation of 3D dynamic urban microclimate. *Building and Environment*, 248: 111063.

Qin, S.; Zhan, D.; Geng, D.; Peng, W.; Tian, G.; Shi, Y.; Gao, N.; Liu, X.; and Wang, L. L. 2025. Modeling multivariable high-resolution 3D urban microclimate using localized Fourier neural operator. *Building and Environment*, 273: 112668.

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

Rantanen, E.; and Talleur, D. 2005. Incremental Transfer and Cost Effectiveness of Groundbased Flight Trainers in University Aviation Programs. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 49.

Rienecker, H.; Hildebrand, V.; and Pfifer, H. 2023. Energy optimal 3D flight path planning for unmanned aerial vehicle in urban environments. *CEAS Aeronautical Journal*, 14(3): 621–636.

Sabadas, S.; Fothergill, S.; Silva, J. M.; and Boston, N. 2024. A Review of Training Procedures for Simulated Engine Failure after Take-Off Exercises with Twin-Engine Aircraft under 5700 kg. *Aerospace*, 11(7).

Tran, A.; Mathews, A.; Xie, L.; and Ong, C. S. 2023. Factorized Fourier Neural Operators. arXiv:2111.13802.

Um, K.; Brand, R.; Fei, Y.; Holl, P.; and Thuerey, N. 2020. Solver-in-the-Loop: Learning from Differentiable Physics to Interact with Iterative PDE-Solvers. *Advances in Neural Information Processing Systems*.

Wang, Y.; Li, Z.; Yuan, Z.; Peng, W.; Liu, T.; and Wang, J. 2024. Prediction of turbulent channel flow using Fourier neural operator-based machine-learning strategy. *Phys. Rev. Fluids*, 9: 084604.

Watson, N. A.; Fernandez, N.; Owen, I.; and White, M. D. 2025. The Potential of Flight Simulation to Support Pilot Training for Mountain Helicopter Emergency Medical Services. *Air Medical Journal*.

White, M. D.; Padfield, G. D.; Lu, L.; Advani, S.; and Potter, M. 2021. Review of flight simulation fidelity requirements to help reduce 'rotorcraft loss of control in-flight' accident rates. *CEAS Aeronautical Journal*, 12(4): 701–721.

Wolfson, M. M.; Delanoy, R. L.; Forman, B. E.; Hallowell, R. G.; Pawlak, M. L.; and Smith, P. D. 1994. Automated Microburst Wind-Shear Prediction. *Lincoln Laboratory Journal*, 7(2): 399–426.

Würth, T.; Freymuth, N.; Zimmerling, C.; Neumann, G.; and Kärger, L. 2024. Physics-informed MeshGraphNets (PI-MGNs): Neural finite element solvers for non-stationary and nonlinear simulations on arbitrary meshes. *Computer Methods in Applied Mechanics and Engineering*, 429: 117102.

Zhiyin, Y. 2015. Large-eddy simulation: Past, present and the future. *Chinese Journal of Aeronautics*, 28(1): 11–24.

Štrbac, A.; Greiwe, D. H.; Hoffmann, F.; Cormier, M.; and Lutz, T. 2022. Piloted Simulation of the Rotorcraft Wind Turbine Wake Interaction during Hover and Transit Flights. *Energies*, 15(5).