

VANE-LESS 3D WIND ESTIMATION FOR FIXED-WING UAVS: A PHYSICS-AUGMENTED TEMPORAL RESIDUAL NETWORK APPROACH

Ziyang Zhang[1,2], Shaokun Li[1,2], Xiaoping Xu[1,2] & Zhou Zhou[1,2]

[1]School of Aeronautics, Northwestern Polytechnical University, Xi'an 710071, China
[2]National Key Laboratory of Aircraft Configuration Design, Xi'an 710072, China

Abstract

Accurate wind estimation is essential for fixed-wing unmanned aerial vehicles (UAVs) in low-altitude missions, yet conventional air-data probes increase size, weight and power, while dynamics-based vane-less estimators are sensitive to aerodynamic model errors and inertial noise. This paper presents DeepWindNet, a physics-augmented temporal residual network for instantaneous three-dimensional wind estimation using onboard inertial measurement unit (IMU), Global Navigation Satellite System (GNSS) and control-input signals instead of dedicated air-data hardware. The method uses a rough wind-triangle estimate as a structural anchor and learns the nonlinear residual caused by aerodynamic mismatch, propeller slipstream, sensor noise and unsteady maneuvers. A dilated temporal residual backbone captures multi-scale flight response, and squeeze-and-excitation channel attention recalibrates heterogeneous sensor features according to flight context. We build an active exploration platform using PX4-Gazebo Software-in-the-Loop (SITL) to generate more than 1.2 million synchronized samples over Gaussian thermal updrafts and local power-law gradient winds. Within this controlled SITL setting, DeepWindNet achieves horizontal wind root mean square error (RMSE) below 0.3 m/s and vertical wind RMSE of 0.045 m/s on trajectory-level unseen tests, while suppressing the bias and high-frequency oscillation observed in a diagnostic dynamics-inversion baseline. The paper also identifies the remaining validation requirements for onboard computational footprint, stochastic turbulence and real-flight transfer.

Keywords: Wind estimation, Fixed-wing UAV, Deep learning, Temporal residual network, SITL

1. Introduction

Wind is one of the dominant environmental factors that determines the safety, efficiency and endurance of fixed-wing unmanned aerial vehicles (UAVs). In low-altitude missions such as long-endurance reconnaissance, terrain mapping, disaster monitoring and autonomous soaring, the aircraft is continuously exposed to spatially non-uniform gusts, thermal updrafts and boundary-layer shear. These atmospheric motions influence the aircraft through two distinct mechanisms. On the one hand, unmodelled wind acts as an external disturbance that perturbs attitude, increases tracking error and may drive the vehicle toward an unfavorable flight envelope. On the other hand, vertical updrafts and wind gradients can be exploited as a natural energy source. A fixed-wing UAV that can perceive the local wind vector in real time can reject dangerous disturbances more effectively and can also plan trajectories that harvest atmospheric energy. Therefore, accurate instantaneous three-dimensional (3D) wind estimation is a prerequisite for wind-aware autonomy.

Direct wind measurement is usually performed by five-hole probes, vanes, hot-wire anemometers, ultrasonic sensors or pitot-static systems. Such sensors provide valuable air-data information, but they are difficult to deploy on small fixed-wing platforms. They introduce additional size, weight and power (SWaP) burden, disturb the aerodynamic shape, increase drag and are vulnerable to damage during field landing and transportation. These constraints are especially severe for low-cost UAVs

https://doi.org/10.71945/icas2026_0859

and swarm platforms, where every gram and every watt matters. While avoiding external air-data hardware reduces SWaP burdens, the learned estimator must still remain computationally lightweight for embedded companion computers. A vane-less estimator that uses existing navigation and control signals is therefore attractive only if its computational footprint remains compatible with the aircraft's embedded resources.

Traditional vane-less methods generally reconstruct aerodynamic states through a dynamic model. In principle, the aircraft itself can be treated as a flying probe: inertial acceleration, attitude and thrust can be combined with aerodynamic coefficients to infer angle of attack, sideslip and airspeed, and the wind triangle can then recover the wind vector. However, this approach is fragile in practice. Full-envelope aerodynamic coefficients are expensive to obtain, and coefficients extracted from a simplified simulation description rarely include high-order derivatives, fuselage-wing interaction, propeller slipstream and structural deformation. When the aircraft enters a high-angle-of-attack or high-sideslip condition, the mismatch between the assumed model and the actual aerodynamic force becomes significant. Moreover, algebraic inversion directly propagates accelerometer noise into the solved aerodynamic states, causing high-frequency oscillation that is unsuitable for feedback control.

This paper proposes DeepWindNet, a physics-augmented deep learning framework for vane-less 3D wind estimation. The central idea is to use simple physical relations where they are reliable and let a temporal neural network learn the residual part that analytical models cannot express robustly. DeepWindNet uses IMU, GNSS and control inputs as the observable state, computes a rough wind estimate from kinematic relations as a structural prior, and applies a Physics-Augmented Temporal Residual Network (PA-TRN) to infer the final wind vector. A squeeze-and-excitation (SE) module reweights sensor channels according to instantaneous flight context, for example emphasizing vertical acceleration in updrafts or yaw-rate related features in lateral shear. Rather than replacing all atmospheric sensing in flight-critical certification, the estimator is intended as a lightweight perception front end for wind-aware guidance, autonomous soaring and reinforcement-learning policies.

The main contributions are threefold. First, a physics-augmented temporal residual architecture is developed, integrating kinematic priors, dynamic difference features and control inputs so that the network learns corrections around a physical anchor rather than performing unconstrained black-box regression. Second, a diagnostic evaluation is carried out using an air-data-aided extended Kalman filter (EKF) as a high-information reference and a conventional dynamics-inversion method as a model-sensitivity baseline. Third, a PX4-Gazebo SITL active exploration platform generates labelled data in Gaussian updraft and gradient-wind environments, enabling sub-meter-per-second estimation accuracy in controlled trajectory-level tests and providing interpretable feature-usage evidence. The claims are intentionally scoped to structured wind fields in this simulation setting. Broadband turbulence filtering, sim-to-real transfer and target-hardware profiling are treated as necessary follow-on work rather than already solved problems.

2. Related Work

2.1 Model-based wind estimation

Wind estimation for aerial vehicles commonly relies on the wind triangle. When airspeed, angle of attack and sideslip are measured, the air-relative velocity can be rotated into the inertial frame and subtracted from GNSS ground velocity; this relation is often embedded in Kalman or complementary filters [1]. Such filters are accurate with reliable air-data sensors because the observation directly constrains the wind vector. The limitation is not the filter principle itself, but the measurement requirement: small UAVs often cannot carry accurate and well-calibrated probes throughout field operation. Vane-less estimators instead use force and moment balance to infer aerodynamic angles from inertial response. The approach can be accurate in ideal simulation, but it requires a high-fidelity aerodynamic database and precise thrust modelling. In practical UAV applications, static coefficients may be obtained from simplified configuration files, while dynamic derivatives, propeller-induced flow and local separation remain unknown. The inverse problem then becomes ill-conditioned: coefficient errors become steady bias, and sensor noise is transferred directly to the recovered aerodynamic angles.

2.2 Learning-based time-series estimation

Learning-based estimators can approximate nonlinear mappings from high-dimensional sensor histories. Early multilayer perceptrons are compact but cannot represent temporal aerodynamic lag explicitly. Recurrent networks can consume historical states, but their serial structure increases latency and makes embedded implementation less predictable. Temporal convolutional and residual networks are attractive because they process sequence windows in parallel and use dilation to enlarge the receptive field without a proportional increase in parameter count [2].

Recent work also shows that wind-aware autonomy is increasingly coupled with learning-based guidance and control. Deep reinforcement learning has been demonstrated for fixed-wing attitude control with field experiments [3], integrated updraft mapping and exploitation [4, 5], and orographic soaring control of fixed-wing UAVs [6]. These studies strengthen the need for a compact wind-perception front end: guidance policies can exploit atmospheric energy only when the local wind state is available with sufficient temporal resolution.

However, a purely data-driven model may learn correlations that do not extrapolate well beyond the simulated wind cases used for training. For wind estimation, the wind triangle and rigid-body dynamics provide useful partial knowledge even when they are not accurate enough to act as final solvers. DeepWindNet therefore follows a physics-guided strategy: the wind triangle supplies a rough but meaningful anchor, dynamic difference features expose the mismatch between expected and observed motion, and the network learns corrections associated with unmodelled dynamics, uncertain parameters and noisy measurements.

3. Problem Formulation and Wind Models

3.1 Coordinate frames and wind triangle

Three coordinate frames are used in this study. The inertial frame is defined as a North-East-Down (NED) frame, denoted by $\mathscr{F}_n$, with its origin fixed at the take-off location. The body frame $\mathscr{F}_b$ is attached to the aircraft center of gravity, with the x_b axis pointing forward, the y_b axis pointing to the right wing and the z_b axis pointing downward. The wind frame $\mathscr{F}_w$ is aligned with the relative air velocity and parameterized by the angle of attack α and sideslip angle β. The body attitude with respect to the NED frame is represented by the rotation matrix R_{nb}.

The fundamental kinematic relation is

$$\mathbf{v}_g^n = \mathbf{v}_a^n + \mathbf{w}^n, \tag{1}$$

where $\mathbf{v}_g^n = [v_N, v_E, v_D]^T$ is the GNSS ground velocity, $\mathbf{v}_a^n$ is the air-relative velocity expressed in the NED frame, and $\mathbf{w}^n = [w_N, w_E, w_D]^T$ is the wind vector to be estimated. If the true airspeed V_a, angle of attack and sideslip are known, the body-frame air-relative velocity can be written as

$$\mathbf{v}_a^b = V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix}, \quad \mathbf{v}_a^n = R_{nb}\mathbf{v}_a^b. \tag{2}$$

In a vane-less configuration, α, β and V_a are not measured. The goal is to construct an estimator

$$\hat{\mathbf{w}}_k^n = f_\theta\left(\mathbf{x}_{k-L+1:k}, \mathbf{u}_{k-L+1:k}\right), \tag{3}$$

where $\mathbf{x}$ denotes the onboard navigation measurements and derived features, $\mathbf{u}$ denotes actuator commands, L is the temporal window length and f_θ is the learned estimator. The estimator must run online and must be robust to stochastic sensor noise, local model mismatch and non-uniform wind fields.

This formulation intentionally avoids solving for α, β and V_a as explicit intermediate outputs. In small UAV operation, these variables are difficult to validate and can introduce additional error propagation. Instead, DeepWindNet treats the observable flight response over a short time history as the sufficient input for wind inference. The learning target is the NED wind vector itself, while physical quantities such as rough air-relative motion are used only as auxiliary priors and feature channels.

3.2 Gaussian thermal updraft

To evaluate vertical-wind estimation in autonomous-soaring scenarios, a Gaussian thermal model is used. This choice follows the simplified analytical tradition used in autonomous-soaring research, where Gaussian or smoothed bell-shaped thermal profiles provide controllable macroscopic updraft structures for algorithm development [7]. Allen's updraft model and the Konovalov flight-data discussion reported therein also emphasize that real thermals vary in size and shape, so the present model should be interpreted as a structured training and evaluation field rather than a universal atmospheric model. Compared with a constant cylindrical updraft, the Gaussian form introduces a smooth radial decay and a height-dependent profile. Let r be the horizontal distance from the thermal center and $\zeta = z/z_i$ be the normalized altitude, where z_i is the convective boundary-layer height. The characteristic thermal radius is modelled as

$$R(\zeta) = R_0 \left(1 + a_1 \zeta + a_2 \zeta^2\right), \tag{4}$$

and the vertical wind component is

$$w_D(r,z) = -W_c \, g(\zeta) \exp\left[-\left(\frac{r}{R(\zeta)}\right)^2\right], \tag{5}$$

where W_c is the core updraft strength and $g(\zeta)$ describes the vertical intensity distribution. The negative sign follows the NED convention, in which an upward wind corresponds to a negative Down component. This model provides smooth but spatially localized vertical excitation, forcing the estimator to distinguish true updrafts from aircraft maneuvers.

3.3 Power-law gradient wind

To represent local shear and boundary-layer effects, a power-law gradient-wind model with spatial masking is also used. At height z, the reference horizontal wind speed is

$$V_h(z) = V_{\text{ref}} \left(\frac{\max(z - z_0, \varepsilon)}{z_{\text{ref}}}\right)^{\gamma}, \tag{6}$$

where V_{ref} is the wind speed at reference height z_{ref}, γ is the terrain roughness exponent, and z_0 avoids non-physical values near the ground. The local wind region is defined by a polygonal mask $m(\mathbf{p})$ with a soft boundary transition:

$$\mathbf{w}_h^n(\mathbf{p}, z) = m(\mathbf{p}) \, V_h(z) \begin{bmatrix} \cos\psi_w \\ \sin\psi_w \\ 0 \end{bmatrix}. \tag{7}$$

The third entry is zero because this term represents the horizontal NE-plane wind contribution; any vertical component is supplied by the thermal model when present. The soft mask prevents discontinuities at the boundary of a wind zone, making the simulation more physically plausible and avoiding artificial step inputs that would bias the learning process.

The two analytical wind models are used as controlled excitation fields, not as a complete representation of the atmospheric boundary layer. Real low-altitude turbulence may contain intermittent high-frequency fluctuations described by Dryden or von Karman spectra, local wake interaction and terrain-induced separation. The present dataset therefore evaluates whether the estimator can learn structured vertical and horizontal wind cues from flight response; it does not by itself prove robustness to broadband stochastic turbulence. This distinction is important for interpreting the results and for defining the next validation stage.

4. Diagnostic Baselines

4.1 Sensor-fusion EKF reference

A sensor-fusion EKF is implemented as a high-information reference rather than as an equal competitor. It assumes that air-data measurements, including true airspeed, angle of attack and sideslip, are available. The EKF state is the wind vector in the NED frame,

$$\mathbf{x}_{w,k} = [w_N, w_E, w_D]_k^T, \tag{8}$$

which follows a first-order Gauss-Markov process. The observation is obtained by rearranging Eq. (1):

$$\mathbf{z}_{w,k} = \mathbf{v}^n_{g,k} - R_{nb,k}\mathbf{v}^b_{a,k} = \mathbf{w}^n_k + \mathbf{n}_k. \tag{9}$$

This vector observation updates all three wind components simultaneously and avoids the observability weakness of scalar projections.

To keep the reference realistic, the observation covariance is adaptively inflated when the aerodynamic state approaches regions where probe measurements or linear assumptions are less reliable. For example, when $|\alpha|$ exceeds a prescribed linearity limit $\alpha_{\lim}$, the angle-of-attack measurement variance is increased as

$$\sigma^2_{\alpha,k} = \sigma^2_{\alpha,0}\left[1 + \lambda_\alpha \max\left(0, \frac{|\alpha_k| - \alpha_{\lim}}{\alpha_{\lim}}\right)^2\right]. \tag{10}$$

An analogous rule is applied to sideslip. This air-data-aided reference provides a comparison case under the assumption of dedicated air-data sensing. A lower-cost vane-less method is not expected to dominate this reference under all conditions; instead, the reference indicates the accuracy scale that becomes available when additional air-data hardware is installed.

4.2 Vane-less dynamics inversion

The traditional vane-less baseline uses Newton-Euler dynamics to recover aerodynamic angles from inertial response. After compensating gravity and thrust, the measured specific force is related to the aerodynamic force by

$$\mathbf{F}^b_a = m\mathbf{f}^b_{\text{imu}} - \mathbf{F}^b_T - R_{bn}m\mathbf{g}^n. \tag{11}$$

The aerodynamic force is then represented by dynamic pressure and coefficient tables,

$$\mathbf{F}^b_a = \frac{1}{2}\rho V_a^2 S\mathbf{C}_F(\alpha, \beta, \delta_e, \delta_a, \delta_r). \tag{12}$$

Because Eq. (12) is nonlinear and implicit, a Newton-Raphson iteration is used to solve for α and β. The solved aerodynamic angles are substituted into Eq. (2) and Eq. (1) to estimate wind.

In an ideal Simulink environment where the estimator uses the same aerodynamic model as the simulated aircraft, the inversion accurately recovers the aerodynamic angles. In the PX4-Gazebo environment, however, the baseline deteriorates severely. Figure 1 shows the typical result. The estimated angle of attack and sideslip preserve some phase relation with the true values, confirming that the inertial response contains physical information, but large steady bias and high-frequency oscillation appear. In the tested sequence, the sideslip RMSE reaches 4.073 deg, which is too large for reliable closed-loop wind-aware control. This result should be read as a failure-mode diagnostic for direct algebraic inversion, not as an exhaustive comparison against all possible vane-less estimators. More sophisticated alternatives, such as model-based filters with learned aerodynamic residuals or Gaussian-process wind-state back ends, remain important fair baselines for future work.

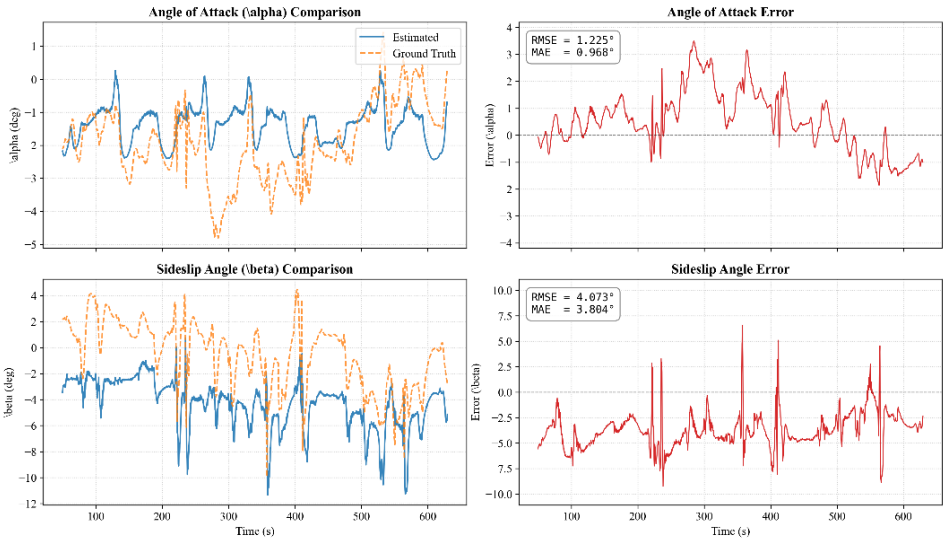


Figure 1 – Dynamics-inversion result in the PX4-Gazebo environment. Model mismatch creates steady bias, while accelerometer noise is amplified by the algebraic inverse solver.

5. DeepWindNet Framework

5.1 Hybrid input representation

The input representation is designed to separate aircraft control action from wind-induced motion while preserving physical interpretability. The raw measurement group contains IMU acceleration, angular rates, attitude, GNSS velocity and related navigation states. A four-dimensional control group contains the normalized actuator commands. These control inputs are important because a roll or pitch response can be caused either by pilot/autopilot command or by wind disturbance. Without actuator information, the network may incorrectly attribute commanded maneuvers to wind.

The second group is a rough physics prior. A simplified kinematic reconstruction is computed from available navigation states to obtain an approximate wind vector $\mathbf{w}_{\text{rough}}$. In this work the prior is formed with a trim-angle, zero-sideslip approximation,

$$\mathbf{w}^n_{\text{rough},k} = \mathbf{v}^n_{g,k} - R_{nb,k}\hat{V}_{a,k}\begin{bmatrix}\cos\alpha_0\\0\\\sin\alpha_0\end{bmatrix}, \tag{13}$$

where α_0 is the trimmed angle of attack and $\hat{V}_{a,k}$ is a nominal airspeed proxy obtained from the flight condition. This prior is deliberately imperfect under high sideslip or model mismatch; it acts as an anchor rather than a final estimator. The network learns a residual correction:

$$\hat{\mathbf{w}}^n_k = \mathbf{w}^n_{\text{rough},k} + \Delta\mathbf{w}^n_{\theta,k}. \tag{14}$$

Residual learning narrows the search space because the model mainly explains the part of the problem that the rough physics cannot solve. The zero-sideslip term is a practical approximation for nominal fixed-wing operation: during routine cruise, loitering and thermalling, the flight controller usually attempts to maintain coordinated turns, so the sideslip angle is driven toward a small value. Under this coordinated-flight assumption, Eq. (13) provides a low-cost rough anchor over a large part of the intended envelope. The prior is nevertheless a potential failure source under strong gusts, large maneuvers or flight outside the trimmed aerodynamic envelope, where both α_0 and $\hat{V}_{a,k}$ may be biased. DeepWindNet therefore also receives raw IMU, GNSS and control histories as well as dynamic difference features, so the residual branch is not forced to trust the prior alone. Consequently, the kinematic prior establishes a robust baseline, allowing the temporal residual network to selectively compensate for unmodelled aerodynamic nonlinearities. The present paper evaluates this idea in the trained SITL envelope; robust deployment should add prior-perturbation training, prior-dropout ablations or confidence gating to prevent a badly biased physical anchor from misleading the network.

The third group consists of temporal difference features. Wind often appears as a discrepancy between expected and observed state rates. Therefore, differences of ground speed, attitude and rough airspeed indicators are explicitly provided. These dynamic features help the estimator capture transient gusts and the rapid boundaries of wind zones. All input channels are normalized using training-set statistics so that low-amplitude vertical-wind cues are not numerically dominated by larger horizontal velocity or attitude channels.

5.2 Temporal residual backbone

The PA-TRN backbone is a one-dimensional temporal convolutional residual network. Given a window of length L, stacked dilated convolutional blocks extract features at multiple temporal scales:

$$\mathbf{h}_{i+1} = \mathbf{h}_i + \Phi_i(\mathbf{h}_i; d_i), \tag{15}$$

where Φ_i denotes a convolution-normalization-activation module and d_i is the dilation factor. Increasing dilation expands the receptive field without increasing the number of parameters proportionally. As a result, the network can respond to short gust edges while still using longer flight context to suppress random sensor noise. The residual connection stabilizes training gradients and facilitates deeper temporal feature extraction, which is crucial for efficient execution on embedded companion computers.

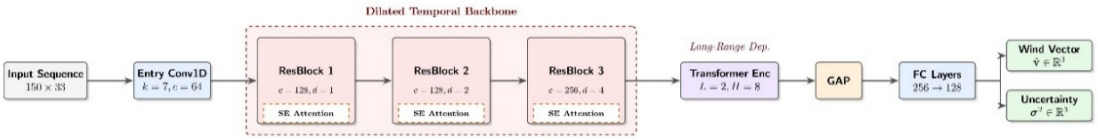


Figure 2 – DeepWindNet architecture. Raw IMU/GNSS states, actuator commands, physical rough wind priors and dynamic difference features are fused by a temporal residual backbone with SE attention.

5.3 Physics-aware channel attention

Fixed-wing aerodynamic response is anisotropic. A vertical updraft is strongly reflected in vertical acceleration and energy-rate changes, while lateral shear can be more visible in yaw rate, sideslip-related motion and cross-track acceleration. Standard convolution treats all input channels with fixed kernels and does not explicitly adapt to this variation. DeepWindNet embeds an SE block in each residual stage. For a latent feature tensor $\mathbf{H} \in \mathbb{R}^{C \times L}$, temporal squeeze produces a channel descriptor

$$s_c = \frac{1}{L} \sum_{t=1}^{L} H_{c,t}. \tag{16}$$

Two lightweight fully connected layers then generate channel weights

$$\mathbf{a} = \sigma(W_2\, \delta(W_1 \mathbf{s})), \tag{17}$$

where δ is a nonlinear activation and σ is the sigmoid function. The recalibrated feature is $\tilde{H}_{c,t} = a_c H_{c,t}$. This mechanism follows the squeeze-and-excitation principle [8] and allows the network to dynamically emphasize channels that are physically relevant to the current disturbance mode.

5.4 Uncertainty-weighted composite loss

The three wind components have different scales and learning difficulty. Horizontal wind often contains larger step-like changes, while the vertical component is smaller but mission-critical for energy-aware flight. A fixed loss weight can cause one axis to dominate training. DeepWindNet therefore uses a multi-task uncertainty formulation with learnable log variances:

$$\mathcal{L}_{\text{data}} = \sum_{j \in \{N,E,D\}} \left[\exp(-s_j)\, \text{SmoothL1}(\hat{w}_j - w_j) + s_j \right]. \tag{18}$$

A temporal smoothness term is added to reduce non-physical high-frequency oscillation:

$$\mathscr{L}_{\text{total}} = \mathscr{L}_{\text{data}} + \lambda_g \left\| \nabla_t \hat{\mathbf{w}} - \nabla_t \mathbf{w} \right\|_1 . \tag{19}$$

The model is trained using AdamW [9] with cosine-annealing warm restarts. The Smooth L1 component improves robustness near gust boundaries. The gradient term in Eq. (19) penalizes mismatch between predicted and labelled temporal gradients, rather than penalizing large gradients by itself. Therefore, it can preserve sharp changes when such changes exist in the labels. However, because the present labels come from smooth analytical wind fields, this loss still needs stress testing under Dryden/von Karman turbulence before any claim is made about high-frequency gust reconstruction.

5.5 Implementation characteristics

The network is intended for real-time onboard use, but this paper does not claim that the current prototype has already satisfied every embedded-computing constraint. One-dimensional convolutions avoid the recurrent dependence of long short-term memory (LSTM)-like architectures, so all samples in the input window can be processed in parallel. The SE blocks operate on channel descriptors rather than on the full time sequence, which limits their incremental cost. These design choices are compatible with embedded deployment, but final SWaP assessment requires direct measurement of inference latency, memory footprint and power consumption on the target companion computer. In deployment, the estimator would be called at the navigation update rate: the newest IMU/GNSS/control sample is appended to the rolling window, normalized, passed through PA-TRN, and converted into the NED wind estimate for guidance or monitoring.

The output is intentionally kept in the inertial NED frame. This choice matches the representation used by navigation and path-planning modules and avoids repeatedly transforming a noisy body-frame wind estimate through the attitude solution. It also simplifies the downstream use of vertical wind for autonomous soaring, because the Down component directly indicates whether the surrounding air mass contains usable upward motion.

6. SITL Dataset Construction

6.1 PX4-Gazebo active exploration platform

Labelled wind data are difficult to obtain from real flight because the true 3D wind field is rarely known with sufficient spatial and temporal resolution. This work therefore builds a physics-based SITL platform integrating the PX4 autopilot stack [10] with the Gazebo physics engine [11]. PX4 provides realistic control-loop delay and actuator commands, while Gazebo provides rigid-body dynamics, sensor noise and wind-field interaction. A custom wind interface injects analytical wind fields into the simulation loop and records synchronized state, control and ground-truth wind labels.

The simulated airspace from 50 m to 300 m is divided into altitude layers. In each episode, Gaussian updrafts and power-law gradient zones are generated with randomized location, intensity, direction and size, providing diverse vertical and horizontal excitation without requiring expensive outdoor wind instrumentation. The randomization is hierarchical: the global layer fixes the altitude band, each local wind zone samples its geometry and strength, and each trajectory segment samples an entry direction and traversal pattern. This prevents the dataset from being dominated by a small number of repeated wind signatures. Nevertheless, training and testing remain within one simulator stack and one aircraft configuration, so the split should be interpreted as intra-domain trajectory and parameter generalization, rather than full sim-to-real transfer.

6.2 Feature-oriented trajectories

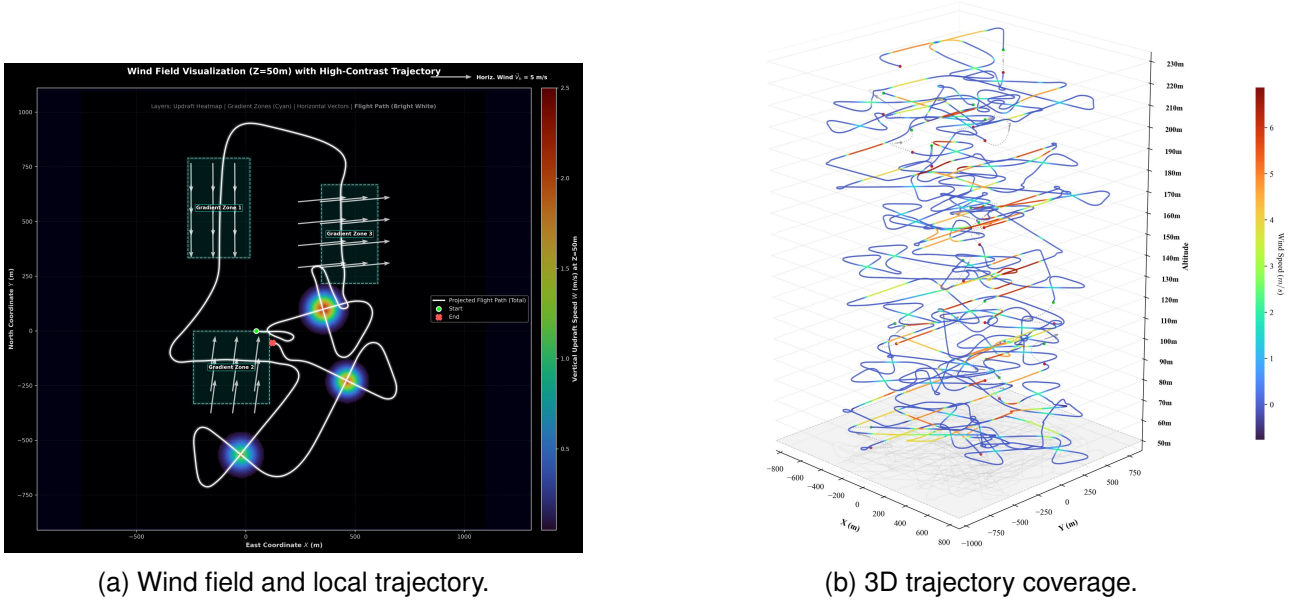
Random wandering is inefficient because much of the collected data may correspond to calm or weakly informative regions. The platform therefore uses active exploration. Cross-scan trajectories pass through thermal centers to excite radial and vertical gradients, while rectangular scans traverse gradient-wind boundaries and interiors. A rule-based scheduler orders sub-tasks by entry and exit distance, reducing uninformative transit time.

Figure 3 illustrates the generated environment and trajectory coverage. The resulting dataset contains more than 1.2 million synchronized tuples of IMU, GNSS, control input, derived features and

ground-truth wind. The dense altitude and spatial coverage require the model to estimate wind from flight response rather than memorize a two-dimensional map.

Table 1 – SITL dataset design used for DeepWindNet training and testing.

Item	Configuration
Simulation stack	PX4 autopilot integrated with Gazebo physics
Altitude range	50–300 m layered airspace
Wind structures	Gaussian thermal updrafts and power-law gradient zones
Exploration patterns	Cross scans for updrafts; rectangular scans for shear zones
Recorded signals	IMU, GNSS, attitude, actuator commands and ground-truth wind
Dataset scale	More than 1.2 million synchronized samples



(a) Wind field and local trajectory.

(b) 3D trajectory coverage.

Figure 3 – SITL dataset construction. Active exploration generates informative crossings of updrafts and gradient-wind zones over multiple altitude layers.

7. Experiments and Results

7.1 Evaluation setup

The model is trained on randomized SITL episodes and evaluated on unseen trajectories and wind-field configurations. The primary metrics are root mean square error (RMSE) and mean absolute error (MAE) for each NED wind component. The reported comparisons are diagnostic rather than a complete leaderboard. The EKF reference uses air-data measurements and therefore represents the performance achievable with dedicated sensing; the dynamics-inversion baseline uses only the vehicle dynamics model and onboard inertial response and is used to expose the sensitivity of direct algebraic inversion. A fully fair comparison among vane-less estimators should additionally include filters with learned residual aerodynamics, nonparametric wind-state regression and compact sequence models trained on the same input set.

The unseen test sequences include step-like horizontal wind changes, repeated thermal crossings and mixed conditions in which updrafts and gradient zones appear in the same flight. This split is important because random train-test sampling at the sample level would overestimate performance: adjacent points along one trajectory are highly correlated. The evaluation therefore emphasizes trajectory-level generalization, where both the wind-zone placement and the vehicle path differ from those used during optimization.

Table 2 reports representative results. The sensor-fusion EKF is highly accurate because it receives direct air-data information. DeepWindNet, despite using no vane or air-data probe, achieves horizon-

tal RMSE below 0.3 m/s and vertical RMSE of 0.045 m/s in the same SITL domain. The dynamics-inversion result is not treated as a numerical competitor in the table because its intermediate angle estimates suffer from bias and oscillation, making the downstream wind RMSE dependent on solver failure details.

Table 2 – Representative SITL evaluation and diagnostic baseline outcome.

Method	North RMSE (m/s)	East RMSE (m/s)	Down RMSE (m/s)
Sensor-fusion EKF reference	0.118	0.071	0.059
Diagnostic dynamics inversion	unstable angle reconstruction; wind RMSE not meaningful		
DeepWindNet	0.216	0.282	0.045

7.2 DeepWindNet time-domain performance

Figure 4 shows the time-domain prediction of DeepWindNet. The network tracks step-like changes in horizontal wind and narrow vertical updraft pulses with little delay. In the North and East components, the prediction converges within approximately 0.5 s after gust transitions and maintains low residual error during steady intervals. In the Down component, the network captures the short negative peaks associated with updrafts and achieves the smallest RMSE. This is important for autonomous soaring, where vertical wind determines whether the aircraft should circle, exit or search for a stronger thermal. The right column of Fig. 4 also demonstrates the benefit of temporal convolution and the Smooth L1 based composite loss within the tested analytical wind fields. Error spikes appear near sharp transitions, but they decay rapidly and do not evolve into persistent oscillation. This contrasts with dynamics inversion, where noise is amplified continuously by the algebraic inverse solver. The learned temporal filter therefore behaves as a denoising estimator in the current SITL setting. Whether the same smoothing behavior preserves broadband turbulent gusts must be verified with stochastic wind spectra before being used as a flight-critical claim.

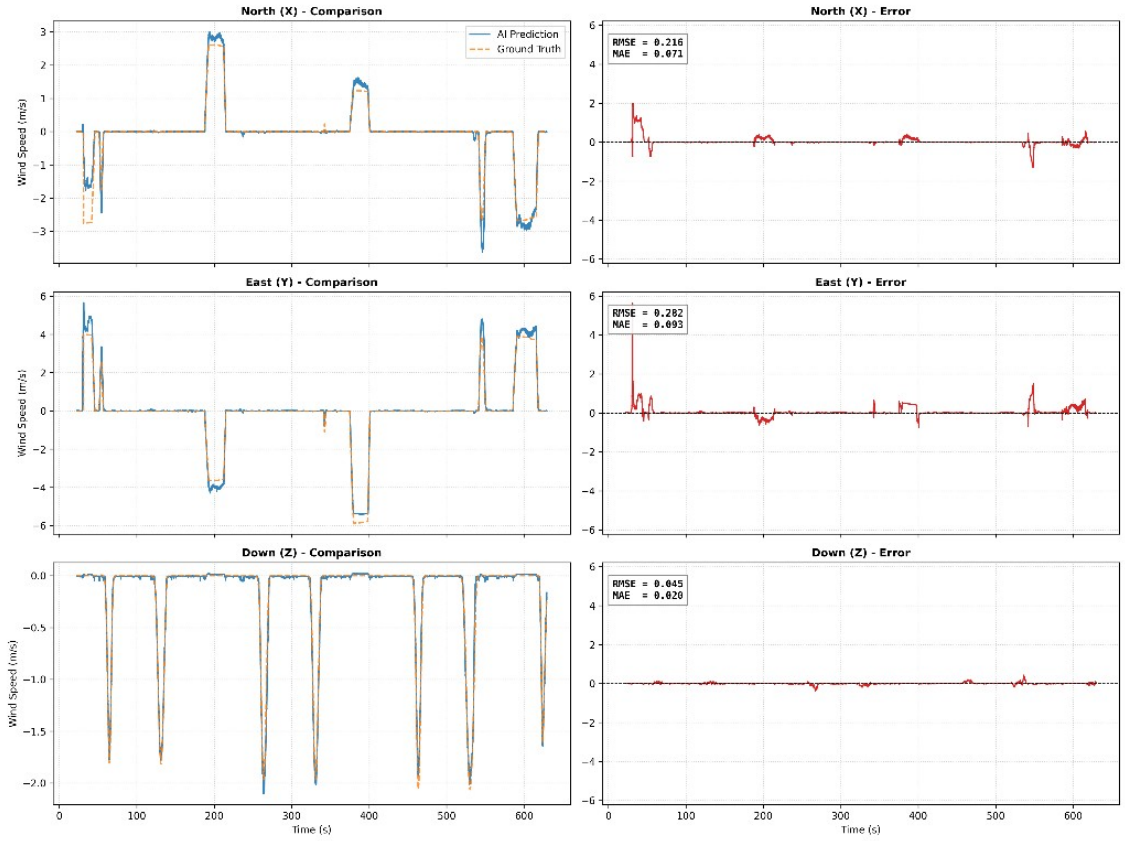


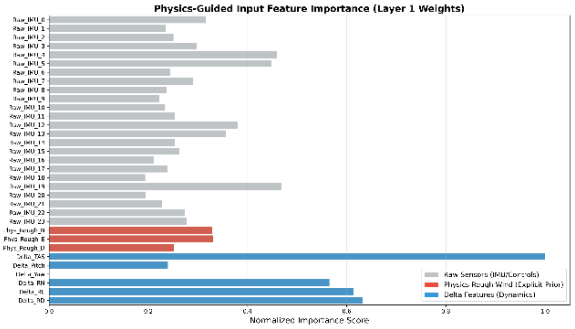
Figure 4 – DeepWindNet estimation result on an unseen sequence. The left column compares predicted and ground-truth wind components; the right column shows residual errors.

7.3 Feature-usage inspection and physical consistency

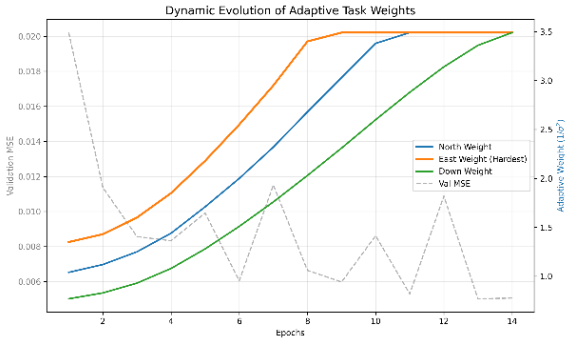
As a heuristic feature-usage inspection, the first-layer convolutional weights are normalized and grouped by feature type. Figure 5(a) shows that dynamic difference features have the largest importance scores, especially changes related to airspeed and inertial velocity. This agrees with the wind-triangle intuition: wind is expressed through the discrepancy between ground-relative and air-relative motion. The rough physics prior also receives a clear weight, indicating that it acts as an anchor rather than an unused auxiliary input.

The adaptive loss weights in Fig. 5(b) reveal a curriculum-like training process. During early epochs, the model keeps task confidences low while validation error drops rapidly. As prediction improves, the learned confidences increase and saturate differently across axes, confirming that the uncertainty-weighted objective balances the difficulty of horizontal and vertical estimation.

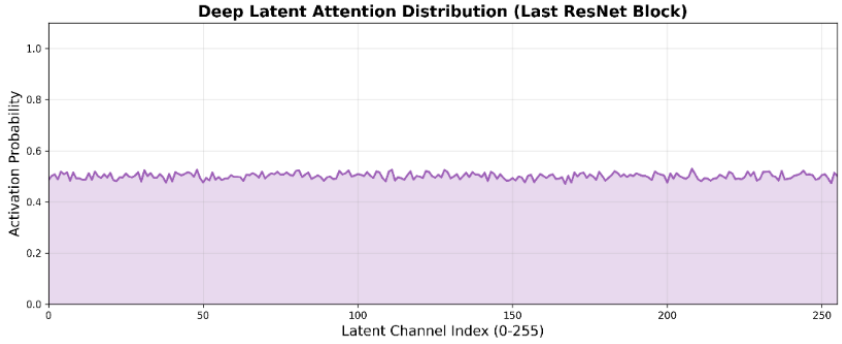
Figure 5(c) further shows that deep SE activations remain distributed across the latent channels rather than collapsing to a few dominant features. This should not be interpreted as a failure to identify one overwhelmingly important channel. In the vane-less setting, the wind estimate is inferred from distributed evidence across IMU, GNSS, attitude and control histories. A distributed and balanced activation pattern across latent channels is consistent with distributed feature usage and supports dense sensor fusion: the network does not degenerate into relying only on a single surrogate variable such as pitch attitude, vertical acceleration or a rough airspeed proxy. Such distributed use of evidence is desirable because it reduces sensitivity to transient noise or bias in any one sensing channel.



(a) Input feature importance.



(b) Adaptive task weights.



(c) Latent SE attention distribution.

Figure 5 – Feature-usage inspection of DeepWindNet. Dynamic features dominate the first layer, learnable uncertainty balances axis difficulty, and SE attention maintains full-channel activity.

7.4 Discussion

The results indicate that the main advantage of DeepWindNet is not only lower average error, but also improved signal usability. Dynamics inversion can preserve a rough trend because the inertial response does contain wind information, yet the output is contaminated by bias and high-frequency noise. Such an estimate is difficult to use in a feedback controller because the controller cannot easily distinguish real gust changes from inversion artifacts. DeepWindNet converts the same inertial

evidence into a smoother and better calibrated wind estimate, which is more suitable for guidance and energy-management logic.

The vertical component is particularly relevant for autonomous soaring. Although the absolute magnitude of w_D is smaller than the horizontal components in many sequences, the estimator must detect short updraft pulses without mistaking aircraft maneuvers for atmospheric energy. The low vertical RMSE suggests that the combined use of temporal context, rough priors and control inputs allows the network to separate motion commanded by the aircraft from motion induced by the surrounding air mass.

The current baseline set should therefore be interpreted according to its purpose. The dynamics-inversion case shows why direct algebraic inverse solutions are brittle under nonlinear Gazebo aerodynamics and sensor noise. The core contribution at this stage is to verify that a "rough physical prior plus temporal residual network" architecture can converge and produce usable wind estimates in a closed SITL loop. Future embedded studies should benchmark it against EKF/GPR fusion and learned-residual filters on Jetson Orin-class computers, using both accuracy and computational cost.

Several limitations remain. The SWaP advantage currently reflects a reduction in external hardware sensors; complete validation requires profiling onboard computational latency, memory footprint and power consumption. The rough prior can be misleading far from trim, so robustness should be tested by perturbing α_0 and $\hat{V}_a$ and by ablating the prior channel. The labels come from smooth analytical fields; Dryden/von Karman turbulence, measured gust records and mixed deterministic-stochastic winds are required to evaluate high-frequency gust preservation. All quantitative tests are same-domain SITL tests under one simulator and one airframe, demonstrating trajectory-level generalization but not full sim-to-real transfer. Stronger vane-less baselines should be included before positioning the method as state of the art.

8. Conclusion and Future Work

This paper presented DeepWindNet, a physics-augmented temporal residual network for vane-less 3D wind estimation on fixed-wing UAVs. This work serves as a physics-based SITL proof of concept for structured wind perception, especially thermal updrafts and gradient winds, rather than as a complete estimator for all low-altitude turbulence. Using IMU, GNSS and control signals instead of dedicated air-data hardware, it combines a rough kinematic prior, temporal difference features, dilated residual convolutions, SE attention and uncertainty-weighted training. On trajectory-level unseen sequences in PX4-Gazebo, DeepWindNet achieved horizontal wind RMSE below 0.3 m/s and vertical wind RMSE of 0.045 m/s, while avoiding the bias and oscillation of a direct dynamics-inversion diagnostic baseline.

Future work will measure latency, memory and power on representative companion computers; add Dryden/von Karman turbulence and measured gust records; collect real-flight data for SITL-to-outdoor adaptation; and compare against stronger vane-less baselines with prior perturbation and removal ablations.

9. Contact Author Email Address

Contact author: Ziyang Zhang.
zzyzzy@mail.nwpu.edu.cn

10. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

References

[1] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(1):35–45, 1960.

[2] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[3] Eivind Eigil Bohn, Erlend M. Coates, Dirk Peter Reinhardt, and Tor Arne Johansen. Data-efficient deep reinforcement learning for attitude control of fixed-wing uavs: Field experiments. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[4] Sven Notter, Christoph Gall, Gerhard Mueller, Aamir Ahmad, and Walter Fichter. Deep reinforcement learning approach for integrated updraft mapping and exploitation. *Journal of Guidance, Control, and Dynamics*, 46:1997–2004, 2023.

[5] Yoav Flato, Roi Harel, Aviv Tamar, Ran Nathan, Tsevi Beatus, et al. Revealing principles of autonomous thermal soaring in windy conditions using vulture-inspired deep reinforcement-learning. *Nature Communications*, 2024.

[6] Tom Suys, Sunyou Hwang, Guido C. H. E. de Croon, and Bart D. W. Remes. Autonomous control for orographic soaring of fixed-wing uavs. *arXiv preprint arXiv:2305.13891*, 2023.

[7] Michael J. Allen. Updraft model for development of autonomous soaring uninhabited air vehicles. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006. AIAA 2006-1510.

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[10] Lorenz Meier, Dominik Honegger, and Marc Pollefeys. PX4: A node-based multithreaded robotics framework for embedded platforms. In *IEEE International Conference on Robotics and Automation*, pages 6235–6240, 2015.

[11] Nathan Koenig and Andrew Howard. Design and use paradigms for Gazebo. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2149–2154, 2004.