SHAPING THE WIND: NESTED POTENTIALS FOR KINEMATICALLY ADMISSIBLE URBAN WIND PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting transient urban winds is fundamental to understanding urban microclimates and designing climate-resilient cities. Building-resolving large-eddy simulation produces detailed incompressible urban wind fields at substantial computational cost for each layout. Neural surrogates offer a faster alternative by learning to predict the evolution of velocity fields. However, minimizing velocity prediction error does not guarantee local mass conservation and wall impermeability, which together define kinematic admissibility. This limitation stems from an unconstrained output representation: geometry conditioning guides predictions but does not restrict them to admissible velocity fields. Correcting boundary violations in these outputs changes the flux balance in adjacent fluid cells and may consequently compromise local mass conservation. To address the challenge, we propose **Sculpt**, a nested potential framework that builds the coupled, geometry-dependent constraints directly into its parameterization. This nested parameterization generates divergence-free velocity updates through the discrete curl of a volume vector potential on the native three-dimensional staggered grid. A shared scalar potential constrains the vector potential's boundary values so that the same operator also enforces impermeability, without a per-step pressure projection. Because backpropagation through this curl attenuates large-scale gradient signals, we parameterize the volume potential at multiple resolutions to better capture large-scale flow structures. We introduce **UrbanWindFlow**, an LES dataset spanning urban morphologies and inflow conditions, to evaluate accuracy and kinematic admissibility together. Across all three evaluation settings, **Sculpt** consistently outperforms both the reduced-order and the neural baselines, reducing mean squared error by 11.7–39.3% while maintaining constraint residuals near numerical precision. The code and data are available at `https://anonymous.4open.science/r/SCULPT-Anonymous`.

1 INTRODUCTION

Predicting transient urban winds is essential for understanding urban microclimates and informing climate-resilient city design. Buildings redirect the incompressible flow into wakes, recirculation zones, and street-level jets whose evolution depends on both layout and atmospheric conditions (Britter & Hanna, 2003; Blocken et al., 2012). Building-resolving large-eddy simulation (LES) captures these three-dimensional dynamics, but repeated solves across layouts and inflow conditions are computationally demanding (Okaze et al., 2026). Neural surrogates offer a faster route by learning to advance velocity fields from their recent history and building geometry (Brunton et al., 2020; Gupta & Brandstetter, 2022; Lippe et al., 2023). Their practical value depends on predicting fields that remain compatible with the surrounding geometry.

Geometry conditioning alone does not ensure this compatibility. Local mass conservation requires balanced fluxes in each incompressible fluid cell, while wall impermeability requires zero normal flow through solid surfaces. Here, *kinematic admissibility* (Drucker et al., 1952; Batchelor, 2000) refers to the joint satisfaction of these two conditions. They constrain shared velocity degrees of freedom: changing a wall-face velocity also changes the flux balance in its adjacent cell. Fig. 1 illustrates this coupling: a U-Net prediction closely resembles the reference field but permits flow through buildings; masking the solid region suppresses penetration while amplifying divergence near building edges. The challenge is to ensure kinematic admissibility across different building geometries.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

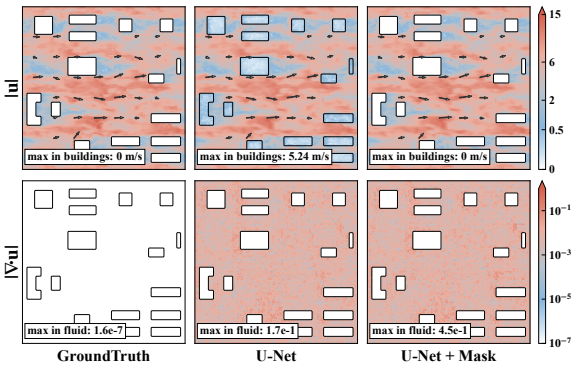

Figure 1: Wind speed (top) and divergence magnitude (bottom) for the ground truth and U-Net predictions before and after enforcing wall impermeability (with direct building mask). Boundary correction eliminates flow through buildings but amplifies divergence near boundaries, compromising local mass conservation.

Existing approaches offer different ways to incorporate these constraints. Soft penalties encourage their satisfaction during training (Raissi et al., 2019; Karniadakis et al., 2021), while a geometry-aware pressure projection enforces both through an additional solve at each prediction step. Structured output representations impose conservation directly, including through vector potentials and divergence-free neural fields (Beucler et al., 2021; Wandel et al., 2021; Richter-Powell et al., 2022). Geometry-dependent compatible spaces have also been developed for steady PDE problems (Shaffer et al., 2026). For an explicit urban-flow surrogate, the central construction problem is to encode solid boundaries in the potential while retaining a divergence-free update on the native three-dimensional grid. A curl guarantees zero divergence, but impermeability depends on the potential's boundary values.

To address this representational challenge, we propose **Sculpt**, a nested potential framework that enforces kinematic admissibility through a geometry-dependent parameterization of velocity updates. For a layout g, discrete conservation and wall conditions define a linear operator $\mathcal{C}(g)$ and an admissible space $\mathcal{V}(g) = \ker \mathcal{C}(g)$. **Sculpt** maps unconstrained potential variables into this space through nested volume and boundary potentials on the native three-dimensional staggered grid. The discrete curl of the vector potential yields divergence-free increments through cancellation of edge contributions within each cell. To enforce impermeability through the same operator, the vector potential's boundary values are assembled from differences of a shared scalar potential, together with compatible global circulation modes. This makes circulation around every boundary face vanish, ensuring zero normal flow. A volumetric module and a surface graph network predict these potential variables, which are assembled before the curl readout. Starting from an admissible state, the resulting updates preserve both constraints throughout a rollout on the fixed layout without a per-step pressure projection. This representation also shapes learning: backpropagation of the velocity loss through the adjoint of the curl reweights gradients across spatial scales and attenuates large-scale signals. We therefore parameterize the volume potential at multiple resolutions, giving coarse flow structures dedicated prediction branches while retaining the same admissible readout.

To evaluate prediction accuracy and kinematic admissibility together, we introduce **UrbanWindFlow**, a dataset of 85 building-resolving LES sequences spanning urban morphologies and inflow conditions. Its three evaluation settings separate unseen layout instances, unseen morphology classes, and held-out inflow combinations. Across these settings, **Sculpt** reduces 16-step mean squared error by 11.7–39.3% relative to nine baselines while maintaining constraint residuals near numerical precision. The nested potentials yield lower error than penalty and projection variants, while multiresolution potentials improve accuracy under three shifts.

Our contributions are threefold:

- **A framework for learning kinematic admissibility.** We introduce a nested volume-boundary potential construction that jointly enforces discrete mass conservation and wall impermeability during autoregressive urban wind-flow prediction.
- **Learning effects of physical structure.** We characterize the scale dependence of curl-induced gradients and develop multiresolution potentials to support the learning of large-scale dynamics.
- **Joint evaluation of accuracy, admissibility, and generalization.** We provide a public LES dataset and evaluation protocol that assess prediction accuracy, kinematic admissibility, and transfer across urban morphologies and inflow conditions.

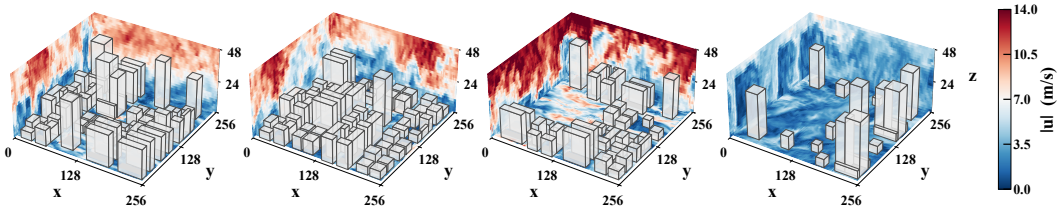

Figure 2: Visualization of the layouts and the instantaneous wind speed $|\mathbf{u}|$ in four **UrbanWindFlow** cases, which differ in both morphology and inflow condition. Axes are in meters.

2 PRELIMINARIES

In building-resolving urban wind modeling, we consider the wind in a city neighborhood as an incompressible fluid, discretized on a uniform staggered grid that is periodic in the two horizontal directions and bounded below by the ground and above by a lid. The building layout is a per-cell occupancy mask that marks each cell as fluid or solid. We denote that mask by g and $\Omega(g)$ for the set of cells it marks as fluid. The instantaneous velocity $\mathbf{u}$ is stored on fluid cell faces of the Arakawa C-grid (Harlow et al., 1965). The layout therefore determines the fluid cells and the boundary faces on which the velocity field must satisfy the kinematic conditions defined below.

2.1 KINEMATICALLY ADMISSIBLE URBAN WIND FIELD

Definition 2.1 (Kinematic Admissibility)**.** A velocity field $\mathbf{u}$ is kinematically admissible (Drucker et al., 1952; Batchelor, 2000) for the layout g when it jointly satisfies the two conditions over the whole geometry. The first is local mass conservation in every fluid cell. The second is wall impermeability, meaning that no flow can cross an impermeable surface:

$$(\nabla_h \cdot \mathbf{u})_c = 0 \quad \forall\, c \in \Omega(g), \qquad\qquad u_f = 0 \quad \forall\, f \in \mathcal{F}_\partial(g), \tag{1}$$

where c ranges over the fluid cells $\Omega(g)$, $(\nabla_h \cdot \mathbf{u})_c$ is the finite-volume divergence of that cell, f ranges over $\mathcal{F}_\partial(g)$, the set of impermeable faces, namely the building surfaces, the ground and the lid, and u_f is the velocity component normal to f. Only that normal component is constrained.

Because both conditions in Eq. (1) are linear in $\mathbf{u}$, they can be written jointly as $\mathcal{C}(g)\mathbf{u} = \mathbf{0}$, where $\mathcal{C}(g)$ is a sparse linear operator read off from the mask. Their solutions constitute the admissible space $\mathcal{V}(g) = \ker \mathcal{C}(g) \subset \mathbb{R}^N$, where $\ker \mathcal{C}(g)$ denotes the kernel of $\mathcal{C}(g)$ and N is the total number of faces of the fluid cells. This formulation highlights three properties of the admissible fields. **(1)** $\mathcal{V}(g)$ is fixed by the geometry: a different layout corresponds to a different subspace rather than a different value of a conditioning variable. **(2)** For a given layout g, every true wind field $\mathbf{u}$ should belong to $\mathcal{V}(g)$ because it satisfies both kinematic conditions. Hence, for any prediction $\hat{\mathbf{u}}$ of that field, any component outside $\mathcal{V}(g)$ cannot match the true field and can only add to the prediction error (Beucler et al., 2021; Zanetta et al., 2023). Mathematically, the orthogonal decomposition gives $\|\hat{\mathbf{u}} - \mathbf{u}\|^2 = \|\Pi_g \hat{\mathbf{u}} - \mathbf{u}\|^2 + \|(\mathbf{I} - \Pi_g)\hat{\mathbf{u}}\|^2$, where Π_g denotes the orthogonal projector onto $\mathcal{V}(g)$. **(3)** The two kinematic conditions are coupled through shared velocity degrees of freedom. A boundary face is also part of an adjacent fluid cell, so its velocity is involved in both wall impermeability and the cell's mass balance. Changing that velocity to satisfy one condition can affect the other. These suggest that admissibility should be built into the output itself, rather than imposed afterward.

Urban Wind Field Dataset. To systematically study the kinematic admissibility in Definition 2.1 across different urban geometries and inflow conditions, we construct **UrbanWindFlow**, a dataset of three-dimensional instantaneous wind fields that resolve buildings on the grid and span a range of layouts and inflow conditions, using the PALM large-eddy simulation model (Maronga et al., 2020). It contains 85 building-resolving wind-field sequences that vary in both morphology and inflow condition. Each sequence consists of 100 consecutive snapshots (frames) at $\Delta t = 1$ s after the flow has become statistically stationary. Fig. 2 shows representative layouts and wind fields. Every snapshot satisfies the two kinematic conditions within numerical precision.

2.2 PROBLEM FORMULATION

Task Definition. Considering the temporal nature of wind fields, we formulate a history-based prediction task that incorporates the building layout and inflow condition under which the wind field

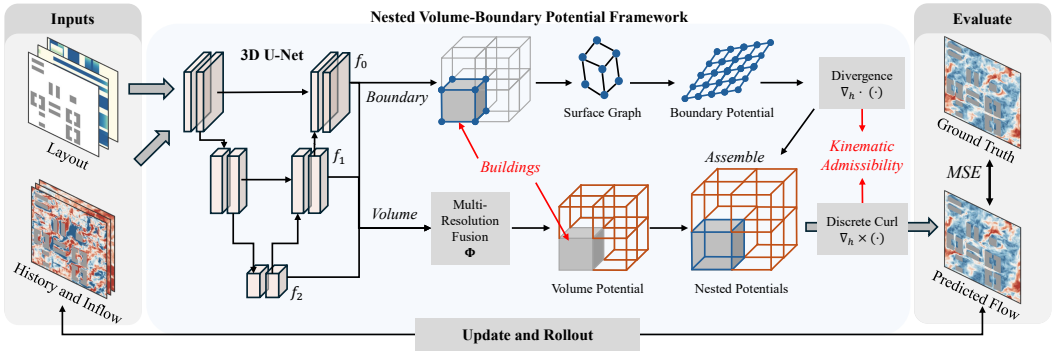

Figure 3: Architecture overview. Given the layout, flow history, and inflow condition, **Sculpt** consists of two parts: a volume potential module that captures multi-scale circulation on the volume grid, and a boundary potential module that models geometry-dependent boundary interactions on the surface mesh. Their outputs are assembled into a nested potential, whose discrete curl yields a kinematically admissible velocity increment for autoregressive rollout.

evolves, with the latter including wind direction, free-stream speed, and atmospheric stability. Given a set of sequences $\mathcal{D}$, the n most recent snapshots, the layout g and the inflow condition a, a predictor f_ω with parameters ω produces the next frame as $\hat{\mathbf{u}}^{t+1} = f_\omega(\mathbf{u}^{t-n+1}, \ldots, \mathbf{u}^t;\, g,\, a)$, and a k-step prediction $\hat{\mathbf{u}}^{t+k}$ is obtained autoregressively. The objective function is as follows:

$$\omega^* = \arg\min_{\omega}\ \mathbb{E}_{(g,a)\sim\mathcal{D}}\ \mathbb{E}_{t,k}\Big[\ell\big(\hat{\mathbf{u}}^{t+k},\, \mathbf{u}^{t+k}\big)\Big], \quad \text{s.t. } \hat{\mathbf{u}}^{t+k} \in \mathcal{V}(g), \tag{2}$$

where $\ell(\cdot)$ is a loss function, $\mathbf{u}^{t+k}$ is the ground-truth frame at horizon k, and t ranges over the frames with $t \geq n$. Eq. (2) requires the admissibility constraint to hold for any ω, layout, inflow condition, and at every rollout horizon. Beyond admissibility, the predictor should generalize across variations in urban geometry and inflow conditions, thus motivating three evaluation settings: unseen layout instances, unseen morphology classes, and held-out inflow combinations.

Making Admissibility a Property of the Architecture. Existing approaches typically impose admissibility on an unconstrained prediction, for example through a soft penalty $\lambda|\mathcal{C}(g)\hat{\mathbf{u}}|^2$ (Raissi et al., 2019; Karniadakis et al., 2021), leaving the residual to the optimizer, or through pressure projection (Li et al., 2026b), which maps each prediction to $\mathcal{V}(g)$ but requires a geometry-dependent elliptic solve at every step. A structured output representation (Ciaremalla & Vanzan, 2024) instead parameterizes predictions directly within the admissible space through a map:

$$\Psi_g :\ \boldsymbol{\theta} \longmapsto \hat{\mathbf{u}}, \qquad \boldsymbol{\theta} \in \mathbb{R}^P \text{ unconstrained}, \tag{3}$$

where $\boldsymbol{\theta}$ contains P unconstrained variables. If $\mathrm{range}(\Psi_g) \subseteq \mathcal{V}(g)$, every output of Ψ_g is kinematically admissible regardless of $\boldsymbol{\theta}$. A direct construction of Ψ_g is to use a basis of $\ker \mathcal{C}(g)$. However, such a basis must be factorized separately for each layout and has no natural correspondence across different geometries, making it difficult to share a single predictor across layouts. We instead construct Ψ_g as a nested map using two discrete identities of the staggered grid: a discrete curl has no divergence, and a discrete gradient has no curl. Specifically, we define the update $\Delta\mathbf{u}$ as the curl of a volume vector potential, which conserves mass in every cell regardless of the mask. The mask is therefore needed only for the second condition. Impermeability is enforced within the same operator by assembling the potential on the edges bounding $\mathcal{F}_\partial(g)$ from a shared scalar potential, with these boundary edges providing the layout information for Ψ_g. Since $\mathcal{V}(g)$ is a linear subspace, adding an admissible update to an admissible frame preserves admissibility by the autoregressive rollout.

3 Methodology

As shown in Fig. 3, **Sculpt** emits two potential fields and reads the velocity update from them. A volumetric module produces the volume vector potential on the grid edges, a surface graph network produces the boundary scalar potential on the surface vertices. Then a parameter-free assembly operator combines them and applies the discrete curl. The following sections detail their formulations.

3.1 NESTED VOLUME-BOUNDARY POTENTIAL FRAMEWORK

Volume Vector Potential. Rather than directly predicting the velocity on faces, we follow prior divergence-free parameterizations (Richter-Powell et al., 2022; Wandel et al., 2021) and predict a vector potential on grid edges. Let h be the grid spacing, $\mathcal{E}$ be the set of grid edges, and $A_e \in \mathbb{R}$ be the value carried by edge $e \in \mathcal{E}$, the collection of which forms the volume vector potential $\mathbf{A}$. Following the discrete Stokes theorem (Hyman & Shashkov, 1997; Desbrun et al., 2005), the velocity update on a face f is recovered as the circulation of $\mathbf{A}$ around its boundary via

$$\Delta u_f \;=\; (\nabla_h \times \mathbf{A})_f \;=\; \frac{1}{h} \sum_{e \in \partial f} \sigma_{f,e}\, A_e, \tag{4}$$

where ∂f denotes the four edges bounding f with a fixed orientation and $\sigma_{f,e} \in \{+1, -1\}$ is the incidence sign of e in that oriented boundary. Local mass conservation is guaranteed automatically, as formalized in the following proposition. The proof is provided in Appendix A.1.

Proposition 3.1. *For every $\mathbf{A} \in \mathbb{R}^{|\mathcal{E}|}$, the field defined by Eq. (4) satisfies $(\nabla_h \cdot \Delta\mathbf{u})_c = 0$ in every cell c and in exact arithmetic.*

Shared Scalar Potential on the Boundary. The curl construction above guarantees local mass conservation for any $\mathbf{A}$, but wall impermeability further constrains its values on the boundary edges. Let $\mathcal{E}_\partial(g) \subset \mathcal{E}$ denote the set of boundary edges, including those incident to at least one solid cell and those on the ground and lid. These edges span a two-dimensional complex $\mathcal{B}$ with vertex set $\mathcal{N}$, whose elements are denoted by ν. Every face in $\mathcal{F}_\partial(g)$ is bounded by four edges in $\mathcal{E}_\partial(g)$, so impermeability depends only on $\mathbf{A}|_{\mathcal{E}_\partial}$. In contrast to methods that impose wall impermeability directly on the predicted velocity (Sukumar & Srivastava, 2022; Higaki et al.), the condition is enforced one degree down the discrete de Rham complex, as formalized in the following proposition.

Proposition 3.2. *Let $\{\chi^{(1)}, \ldots, \chi^{(\beta_1)}\}$ be a harmonic basis of a component of $\mathcal{B}$ with first Betti number β_1, and let the boundary values of $\mathbf{A}$ be taken in the form*

$$A_e \;=\; \frac{G_{e^+} - G_{e^-}}{h} \;+\; \sum_{m=1}^{\beta_1} \gamma_m\, \chi_e^{(m)}, \qquad e \in \mathcal{E}_\partial(g), \tag{5}$$

for an arbitrary $G : \mathcal{N} \to \mathbb{R}$ and arbitrary coefficients γ_m. Then $\Delta u_f = 0$ for every $f \in \mathcal{F}_\partial(g)$.

Here, $G : \mathcal{N} \to \mathbb{R}$ is the shared scalar potential on the boundary vertices, e^+ and e^- are the two endpoints of edge e, and γ_m are the coefficients of the global circulation modes. Proposition 3.2 turns impermeability into a recipe for the boundary values of $\mathbf{A}$: take the three Cartesian components on $\mathcal{E}_\partial(g)$ from the differences of one and the same scalar field, so that the circulation around every boundary face telescopes to zero. Predicting G is then a problem posed on $\mathcal{B}$ rather than on the grid.

Assembly Operator. The two modules provide the interior values of $\mathbf{A}$, the boundary potential G, and the coefficients γ_m. The assembly operator retains the predicted values on the interior edges and replaces those on $\mathcal{E}_\partial(g)$ with Eq. (5). The discrete curl in Eq. (4) is then applied to the assembled potential to obtain the velocity update $\Delta\mathbf{u}$. The layout enters Ψ_g only through the boundary-edge set $\mathcal{E}_\partial(g)$, while the interior edges remain unconstrained. We minimize the mean squared error between the updated field and the ground truth over the fluid cell, with no penalty on either condition of Eq. (1) because both hold for every value of the assembled vector potential. For the stability of autoregressive process, we introduce the pushforward scheme (Brandstetter et al., 2022), whose perturbed inputs are predictions of the same map and are therefore admissible as well. The next two subsections describe in detail how the volume potential and the boundary potential are predicted.

3.2 MULTIRESOLUTION VOLUME POTENTIAL

Inputs. The volumetric module is conditioned on the n most recent frames, the building layout and a single inflow scalar, which together form the channels of one input tensor. Each frame supplies its three velocity components. The layout supplies the occupancy mask and four horizontal distance fields, one per horizontal direction, each giving at every fluid cell the distance to the nearest solid cell along that direction, with periodic wrap-around in x and y. The inflow supplies the surface heat flux broadcast over the grid, which carries the stratification; the direction and the speed are already visible in the frames. A three-dimensional U-Net encodes this tensor into feature maps at several resolutions.

Multiresolution Branches. Given a U-Net with S resolution levels, we denote $\mathbf{f}_s$ for the features it computes at level s, which carry C_s channels on the grid coarsened by 2^s. Typically, the finest layer $\mathbf{f}_0$ serves as the output, and the deeper feature maps serve only as intermediates. We take a candidate potential from each level instead, so that every level reaches the output:

$$\mathbf{a}_s \;=\; \mathcal{C}_s^{\text{out}}\big(\mathcal{U}_s(\mathcal{C}_s^{\text{in}}(\mathbf{f}_s))\big), s = 0, \dots, S-1, \tag{6}$$

where $\mathcal{C}_s^{\text{in}}$ is a convolution that mixes $\mathbf{f}_s$ at level s and reduces its C_s channels to a common width $c_{\mathcal{A}}$, $\mathcal{U}_s$ prolongates the result to the grid, and $\mathcal{C}_s^{\text{out}}$ is a convolution taking those $c_{\mathcal{A}}$ channels to the three Cartesian components. Both $\mathcal{C}_0^{\text{in}}$ and $\mathcal{U}_0$ are the identity. The arrangement is that of a feature pyramid (Lin et al., 2017). A learned matrix $\Phi \in \mathbb{R}^{3\times 3S}$ applied at every grid point combines the S candidates. We denote $[\,\cdot\,;\,\cdot\,]$ as concatenation, the volume vector potential $\mathbf{A}$ can be obtained by

$$\mathbf{A} \;=\; \Phi\big([\,\mathbf{a}_0\,;\,\dots\,;\,\mathbf{a}_{S-1}\,]\big). \tag{7}$$

Why Multiresolution? **Sculpt** predicts vector potential rather than the update directly. Gradients travel back to the potential through the adjoint of the curl, and that adjoint is not neutral: it multiplies each Fourier component of the gradient by a gain proportional to its wavenumber. The long wavelengths arrive attenuated and the short ones amplified, compared with the gradient the same loss would hand to a directly predicted velocity. Across the wavelengths the grid resolves, the two ends of that gain differ by a factor of order the domain size divided by the grid spacing. We provide the proof in Appendix A.2. The effect is that different wavelengths learn at very different speeds. Producing the potential from the output of the U-Net alone, at the full resolution, every weight of the volumetric module is trained through this one reweighted gradient and no weight is dedicated to the long wavelengths. Those wavelengths carry the large-scale structure of the flow, and they are the part of the error that improves most slowly. Nothing prevents the parameterization from representing them; what suffers is the accuracy reached within a fixed training budget. Fig. 4 shows how the gradient that $\Delta\mathbf{u}$ and two type of $\mathbf{A}$ receive during backpropagation distributes over the wavelength bands.

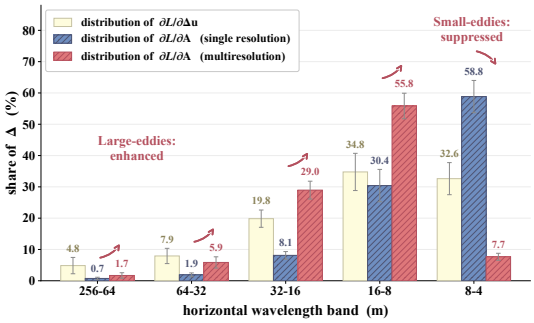

Figure 4: Share of the gradient in each horizontal wavelength band: $\Delta\mathbf{u}$ (left), single-resolution $\mathbf{A}$ (middle), and multiresolution $\mathbf{A}$ (right). The multiresolution read-out raises the share carried by the Large-eddies and lowers the share carried by the small ones, undoing part of the shift that backpropagation through the curl introduces.

3.3 Surface Graph Network for the Boundary Potential

With these volumetric features, **Sculpt** next explores the shared scalar potential G on the boundary. The boundary potential lives on $\mathcal{B}$, where a three-dimensional convolution is a poor inductive bias: two façades across a narrow street are neighbours in Euclidean distance, while the path between them on $\mathcal{B}$ descends one façade, crosses the ground and ascends the other. We therefore predict G with a network whose spatial operator is an aggregation along $\mathcal{E}_\partial(g)$.

Learning Representations on the Boundary. We regard $\mathcal{B}$ as a graph with node set $\mathcal{N}$ and edge set $\mathcal{E}_\partial(g)$. Its nodes lie on a regular lattice, so each has at most six neighbors, one per axis direction $\delta \in \{1, \dots, 6\}$, and an indicator $o_{\nu,\delta} \in \{0, 1\}$ records whether that neighbour exists. Every node carries a static descriptor $\mathbf{d}_\nu \in \mathbb{R}^{d_{\text{des}}}$ that collects the six indicators, the height of the node normalized by the domain height, and a slip fraction $\eta_\nu \in [0, 1]$ that is zero on a no-slip surface and one on a free-slip one, obtained from the aerodynamic roughness length. Using the full-resolution output $\mathbf{f}_0$ of the volumetric module, we can obtain a representation of each cell. Then we introduce a learnable aggregation $\mathbf{p}_\nu \in \mathbb{R}^8$ to map cells to central nodes within $\mathcal{B}$. Let $\mathbf{X}_\nu \in \mathbb{R}^{8\times C_0}$ stack the eight rows $\mathbf{f}_0[c_j(\nu)]$ of the finest feature map of Section 3.2, taken at the eight cells incident to ν in slots whose order encodes direction, and let $\mathbf{1}_\nu \in \{0, 1\}^8$ mark the slots inside the domain. The node feature $\mathbf{F}_\nu \in \mathbb{R}^{C_0}$ can be obtained by:

$$\mathbf{p}_\nu \;=\; \mathrm{softmax}\big(\mathrm{MLP}(\mathbf{X}_\nu, \mathbf{d}_\nu) + \log \mathbf{1}_\nu\big), \qquad \mathbf{F}_\nu \;=\; \mathbf{X}_\nu^\top \mathbf{p}_\nu. \tag{8}$$

The node features are then refined on $\mathcal{B}$. A graph module of K layers starts from $\mathbf{h}_\nu^{(0)} = \mathbf{W}_{\text{in}}[\,\mathbf{F}_\nu\,;\,\mathbf{d}_\nu\,]$, and layer τ updates every node by:

$$\mathbf{h}_\nu^{(\tau)} \;=\; \mathbf{h}_\nu^{(\tau-1)} \;+\; \mathrm{SiLU}\Big(\mathbf{U}^{(\tau)}\big[\,\mathbf{h}_\nu^{(\tau-1)}\;;\;\sum_{\delta=1}^{6} o_{\nu,\delta}\,\mathbf{W}_\delta^{(\tau)}\,\mathbf{h}_{\mathrm{nbr}(\nu,\delta)}^{(\tau-1)}\,\big]\Big), \tag{9}$$

where $\mathrm{nbr}(\nu,\delta)$ is the neighbour of ν in direction δ. $\mathbf{W}_{\text{in}} \in \mathbb{R}^{c_{\mathcal{B}}\times(C_0+d_{\text{des}})}$, $\mathbf{W}_\delta^{(\tau)} \in \mathbb{R}^{c_{\mathcal{B}}\times c_{\mathcal{B}}}$ and $\mathbf{U}^{(\tau)} \in \mathbb{R}^{c_{\mathcal{B}}\times 2c_{\mathcal{B}}}$ are the learnable weights. Each direction carries its own $\mathbf{W}_\delta^{(\tau)}$, since a step along a façade and a step around a corner are distinct operations on $\mathcal{B}$. We denote $\mathbf{H} \in \mathbb{R}^{|\mathcal{N}|\times c_{\mathcal{B}}}$ for the final states $\mathbf{h}_\nu^{(K)}$ stacked over the nodes.

Multiresolution Readout. The reweighting of Section 3.2 reaches G through a curl followed by a gradient, so its long wavelengths sit one power of the gain further from the loss than those of $\mathbf{A}$, as Remark A.3 makes precise. We therefore read G out at several levels, as we do for $\mathbf{A}$:

$$g_s \;=\; \mathrm{pool}_s\big(\mathbf{H}\mathbf{w}_s\big), \qquad s = 0,\dots,S-1, \tag{10}$$

where $\mathbf{w}_s \in \mathbb{R}^{c_{\mathcal{B}}}$ reads one value from each node, and pool_s averages it within the corresponding level-s block followed by parameter-free neighbor averaging along $\mathcal{E}_\partial(g)$. The S readouts are combined as in Eq. (7). Since adding a component-wise constant to G leaves the velocity update unchanged, we subtract the component-wise mean from the predicted boundary potential.

Harmonic Completion. We specialize the harmonic part of Proposition 3.2 to our urban domain. The number of harmonic coefficients is determined by the topology of each component of $\mathcal{B}$. Here, $\mathcal{B}$ consists of the ground, the lid, together with the building surfaces. The lid is periodic in both horizontal directions and is therefore topologically a torus with first Betti number $\beta_1 = 2$, giving two harmonic coefficients, γ_1 and γ_2, associated with the two periodic directions. These modes are necessary because the gradient of G has zero circulation around every closed loop and cannot represent these global circulations. In our setting, they control the mean horizontal wind. Summing Eq. (4) along a vertical column cancels the vertical differences in pairs, while the horizontal differences vanish under periodic averaging, giving

$$\langle \Delta u\rangle_{\text{dom}} = -\frac{\langle A^y\rangle_{\text{lid}}}{L_z}, \qquad \langle \Delta v\rangle_{\text{dom}} = +\frac{\langle A^x\rangle_{\text{lid}}}{L_z}, \tag{11}$$

where $\langle\cdot\rangle_{\text{dom}}$ and $\langle\cdot\rangle_{\text{lid}}$ average over the domain and over the top layer of edges, and L_z is the domain height. Setting $\gamma_1 = \gamma_2 = 0$ would force both means to vanish and freeze the mean wind at the value of the first frame. We hence take them from the lid means of the horizontal components the volumetric module emits, and set those of the ground component to zero. Substituting G and the harmonic coefficients into Eq. (5) gives the boundary values $\mathbf{A}|_{\mathcal{E}_\partial(g)}$. These values replace those produced by Eq. (7) on $\mathcal{E}_\partial(g)$, while the interior values of $\mathbf{A}$ are retained. The assembly operator of Section 3.1 then applies the discrete curl once to the merged potential. By Propositions 3.1 and 3.2, the resulting update satisfies local mass conservation and wall impermeability for every layout and every value of $\boldsymbol{\theta}$. Finally, we can obtain the kinematically admissible wind velocity based on historical frames.

4 Experiments

In this section we evaluate **Sculpt** on **UrbanWindFlow** by autoregressive wind-field prediction tasks. We report the mean squared error (MSE) of the predicted velocity over the fluid cells at several rollout horizons, and report two physical residuals to measure the kinematic admissibility: the maximum divergence $\max|\nabla\cdot\mathbf{u}|$, which measures whether mass is conserved, and the maximum normal velocity at solid surfaces $\max|\mathbf{u}\cdot\mathbf{n}|$, which measures whether air is allowed to pass through boundaries. The two physical residuals are worst cases, taken over all test sequences and all four horizons. Details and supplementary experiments are provided in Appendix B, C, and D.

4.1 Experiment Setup

Evaluation Settings. We construct three evaluation settings on **UrbanWindFlow**. **Unseen layouts** uses 1,608 / 192 samples for training / validation under one inflow, while the 108 samples on the 6

Table 1: Performance on the **unseen layouts** setting of **UrbanWindFlow**. MSE is reported at four rollout horizons. Highlighted are the top first, second results. '<' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Methods	MSE↓ ($\times 10^{-3}$)				worst over all k	
	k=1	k=4	k=8	k=16	max $\lvert\nabla\cdot\mathbf{u}\rvert$ ↓	max $\lvert\mathbf{u}\cdot\mathbf{n}\rvert$ ↓
DMD	55.27±17.53	55.40±17.66	63.10±19.82	65.58±20.46	$< 10^{-6}$	0
Time-Mean	29.84±9.59	46.59±14.63	52.06±16.13	53.98±16.61	$< 10^{-6}$	0
POD	54.75±16.93	54.69±16.92	54.87±16.95	55.36±17.21	$< 10^{-5}$	1.70
Dil-ResNet	0.85±0.43	8.36±3.16	29.37±8.94	69.03±15.56	0.46	1.10
TFNO	21.10±7.11	35.49±11.25	53.42±14.94	91.05±17.71	1.14	1.84
AFNO	10.44±3.71	28.27±9.52	42.05±13.41	60.85±18.87	2.02	2.92
AViT	7.69±2.63	17.60±6.30	32.47±11.01	58.01±17.46	1.52	1.64
P3D	1.00±0.49	5.85±2.69	16.79±7.48	43.44±16.47	0.44	0.87
U-Net	0.61±0.30	3.30±1.60	9.06±4.21	25.94±10.32	0.16	1.06
Sculpt	**0.22±0.13**	**1.07±0.55**	**3.89±2.02**	**15.91±7.52**	$< 10^{-6}$	$< 10^{-6}$

Table 2: Performance on the **unseen morphologies** setting of **UrbanWindFlow**. MSE is reported at four rollout horizons. Highlighted are the top first, second results. '<' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Methods	MSE↓ ($\times 10^{-3}$)				worst over all k	
	k=1	k=4	k=8	k=16	max $\lvert\nabla\cdot\mathbf{u}\rvert$ ↓	max $\lvert\mathbf{u}\cdot\mathbf{n}\rvert$ ↓
DMD	68.62±17.04	68.61±17.11	79.65±17.42	79.47±15.27	$< 10^{-6}$	0
Time-Mean	36.77±9.47	58.37±13.67	66.23±13.28	65.53±11.70	$< 10^{-6}$	0
POD	80.82±21.94	80.97±22.11	81.72±22.41	83.71±22.99	$< 10^{-5}$	1.86
Dil-ResNet	1.30±0.71	12.24±4.93	39.94±11.54	86.07±17.04	0.47	1.15
TFNO	27.31±8.10	46.72±12.84	66.60±15.25	102.29±16.90	1.22	2.03
AFNO	13.89±4.84	37.66±11.05	57.53±14.49	86.32±21.09	3.28	4.62
AViT	10.58±3.46	27.09±8.50	51.49±13.86	86.27±18.79	2.09	1.53
P3D	1.58±0.84	9.25±4.42	26.45±10.55	63.19±16.99	1.37	0.99
U-Net	0.94±0.52	5.57±2.96	15.52±7.56	41.33±15.35	0.20	1.05
Sculpt	**0.32±0.21**	**1.85±1.03**	**6.61±3.34**	**25.10±9.88**	$< 10^{-6}$	$< 10^{-6}$

Table 3: Statistics of three settings on **UrbanWindFlow**.

Setting	#layouts	building density	#inflows	#snapshots	#samples
Unseen layouts	30	0.17–0.38	1	3,000	1,908
Unseen morphologies	34	0.11–0.38	1	3,400	1,980
Unseen inflows	5	0.11–0.38	9	4,500	2,520

held-out layouts (same morphology) are used for testing. **Unseen morphologies** uses 1,608 / 192 samples for training / validation under one inflow, while the 180 samples on the 10 layouts (unseen morphology) are used for testing. **Unseen inflows** uses 2,010 / 240 samples for training / validation under 9 inflow conditions over wind direction, free-stream speed and stability, while the 270 samples on the 3 held-out conditions are used for testing. Every test sequence is rolled out from the same 18 start frames, and each MSE entry reports the mean and standard deviation over these rollouts. The statistics of every setting are given in Table 3.

Baselines. We compare our method with nine wind-field predicting methods, which can be divided into three groups:

- Reduced-order methods include the temporal mean of the input window (Time-Mean), dynamic mode decomposition (DMD) (Schmid, 2010; Tu, 2013), and proper orthogonal decomposition (POD) (Berkooz et al., 1993). They predict from linear combinations of observed snapshots.
- Convolutional surrogates include the dilated residual network (Dil-ResNet) (Stachenfeld et al., 2021), the pseudo-3D residual network (P3D) (Holzschuh et al., 2025), and a three-level 3D U-Net with ConvNeXt blocks (U-Net) (Ohana et al., 2024).
- Spectral and attention operators include the tensorized Fourier neural operator (TFNO) (Li et al., 2020; Kossaifi et al., 2023), the adaptive Fourier neural operator (AFNO) (Guibas et al., 2021), and the axial vision transformer (AViT) (Ohana et al., 2024).

4.2 Experimental Results

Main Results. Tables 1 and 2 compare all ten methods on the unseen layout and the unseen morphology settings, and Table 7 in Appendix D.1 reports the unseen inflow setting. Against the strongest baseline, **Sculpt** lowers the error at k=16 by 29% averaging on the three settings, and by 61.9% at k=1, so the relative margin is widest on the short horizons. Time-Mean and DMD build their prediction from historical snapshots and therefore inherit exact physical residuals, but they carry no dynamics and perform poorly throughout. POD draws its basis from the training sequences whose impermeable boundaries lie elsewhere. U-Net is the most accurate among learnable baselines, yet its worst face lets air through at 1.06, 1.05 and 1.30 times the free-stream speed, more than five orders of magnitude above **Sculpt**. AFNO diverges outright under inflow shift, to an MSE of 1.6×10^{3} with a divergence of 5.9×10^{2}. The **unseen inflows** setting is the hardest for every learned model, to the point that POD overtakes all of them at k=16, while **Sculpt** still leads by 11.7%. In summary, **Sculpt** consistently outperforms both reduced-order methods and neural surrogates across all settings, demonstrating the nested potential framework's effectiveness in accuracy and kinematic admissibility.

Table 4: Ablation on the **unseen layouts** setting of **UrbanWindFlow**. MSE is reported at two rollout horizons. Highlighted are the top first, second results. '$<$' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Variant	MSE↓		$\max\|\nabla\cdot\mathbf{u}\|\downarrow$		$\max\|\mathbf{u}\cdot\mathbf{n}\|\downarrow$	
	k=1	k=16	k=1	k=16	k=1	k=16
Sculpt	**0.22±0.13**	**15.91±7.52**	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$
w/o null-space Ψ_g	0.43±0.24	31.03±12.05	0.20	1.24	1.58	2.00
w/ soft penalty	25.00±4.58	112.95±10.40	0.14	0.36	0.76	1.02
w/ post-projection	0.38±0.22	25.16±9.81	$< 10^{-6}$	$< 10^{-6}$	0	0
w/o boundary G	0.27±0.18	19.94±8.29	$< 10^{-6}$	$< 10^{-6}$	0.71	0.80
w/o increment $\Delta\mathbf{u}$	13.63±4.61	2867.98±265.31	$< 10^{-6}$	$< 10^{-5}$	$< 10^{-6}$	$< 10^{-5}$
w/o multi-scale $\mathbf{A}$	0.23±0.15	16.20±6.87	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$
w/o learned fusion	0.26±0.18	18.46±7.98	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$

Table 5: Ablation on the **unseen morphologies** setting of **UrbanWindFlow**. MSE is reported at two rollout horizons. Highlighted are the top first, second results. '$<$' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Variant	MSE↓		$\max\|\nabla\cdot\mathbf{u}\|\downarrow$		$\max\|\mathbf{u}\cdot\mathbf{n}\|\downarrow$	
	k=1	k=16	k=1	k=16	k=1	k=16
Sculpt	**0.32±0.21**	**25.10±9.88**	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$
w/o null-space Ψ_g	0.69±0.41	45.84±13.89	0.23	1.03	1.69	2.20
w/ soft penalty	34.05±2.83	150.08±15.74	0.14	0.34	1.13	1.45
w/ post-projection	0.59±0.38	38.01±12.52	$< 10^{-6}$	$< 10^{-6}$	0	0
w/o boundary G	0.39±0.28	29.95±11.58	$< 10^{-6}$	$< 10^{-6}$	0.95	0.95
w/o increment $\Delta\mathbf{u}$	22.23±5.58	2891.92±366.57	$< 10^{-6}$	$< 10^{-5}$	$< 10^{-6}$	$< 10^{-5}$
w/o multi-scale $\mathbf{A}$	0.33±0.23	27.30±10.89	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$
w/o learned fusion	0.37±0.27	29.03±11.52	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-7}$	$< 10^{-6}$

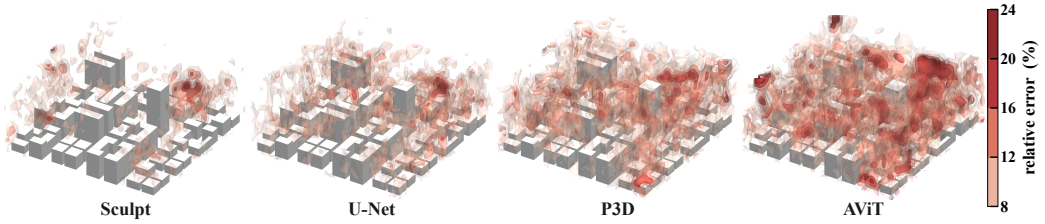

Figure 5: Isosurfaces of the prediction error at k=16 on an unseen inflow case, for **Sculpt** and the three learned baselines. Buildings are grey.

Ablation Study. To evaluate the efficacy of **Sculpt**'s components, we conduct an ablation study comparing seven variant configurations, with comprehensive details provided in Appendix C.2 and quantitative results presented in Tables 4, Table 5, and Table 8 of Appendix D.2. The variant **w/o null-space** Ψ_g raises the error and brings violations of constraints. Moving the constraints into the loss **w/ soft penalty** is worse on both counts. Repairing the field afterwards **w/ post-projection** does restore both constraints exactly, yet it costs a linear solve at every step and trails **Sculpt**. Within the parameterization, the shared boundary potential is what makes the walls exact: removing it (**w/o boundary** G) leaves the divergence at the same numerical floor but lets air through the walls at up to 1.72 times the free-stream speed, at a cost of 19.3–39.7% in 16-step error. Predicting the updates rather than the absolute field is what keeps the rollout stable. Removing any one leads to performance declines, underscoring the critical contributions of these components.

Visualizations. Figure 5 draws the 16-step rollout error as isosurfaces, which demonstrate that **Sculpt** keeps the error confined to a far smaller volume. Meanwhile, we can find where methods fail. Every one of them is least accurate in the roof-level shear layer: measured against the local wind speed, the mean error of **Sculpt** falls from 9.3% in the two cells adjacent to a wall to 3.3% beyond 16 m from any wall, and the same monotone decay holds for U-Net, P3D and AViT. The region where all four exceed 12% of the free stream sits at a mean height of 21.5 m, above the mean roof of 12.9 m and below the tallest building at 36 m, which is where the flow separating from the roofs meets the faster air above them. One explanation is that shear layer is also the least steady one, which would make it difficult for any surrogate. How to address the problem is left to future work.

5 Conclusion

This paper studies how a neural surrogate for transient urban wind can keep its predictions compatible with the geometry. Local mass conservation and wall impermeability constrain shared velocity field, so enforcing one of them disturbs the other. We formulate their joint satisfaction as kinematic admissibility and propose **Sculpt**, which reaches it by a nested volume-boundary potential framework: a volumetric module learns vector potential for fluid cell to get divergence-free, while a surface graph network to learn the scalar potential on the boundary to satisfy the wall impermeability. The two levels assemble into one potential field from which the velocity update is read out, and the guarantee holds inherently. To measure accuracy and admissibility together we release **UrbanWindFlow**, a building-resolving large-eddy simulation dataset with three evaluation settings. Across them **Sculpt** lowers MSE by 11.7–39.3% against nine baselines while holding both residuals below 10^{-6}.

AI USE STATEMENT

In this work, the conceptual framework and core ideas presented in this paper represent the authors' original contributions. We used generative AI tools solely for language-related assistance, including manuscript translation and linguistic polishing. We have not used generative AI tools for developing the methodology, designing algorithms, conducting experiments, analyzing results, or generating scientific claims, and other generative AI-related tasks are not applicable to this work. We have reviewed all AI-assisted text and ensured that it accurately reflects the authors' original ideas and contributions. The authors take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI tools.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide the anonymous code link of the source code and detailed implementation of **Sculpt**. The model configurations, hyper-parameters, training details, and computational resources are provided in Appendix C.1. All datasets used in our experiments are publicly available in our link, with their statistics summarized in Section 4.

REFERENCES

Douglas Arnold, Richard Falk, and Ragnar Winther. Finite element exterior calculus: from hodge theory to numerical stability. *Bulletin of the American mathematical society*, 47(2):281–354, 2010.

George Keith Batchelor. *An introduction to fluid dynamics*. Cambridge university press, 2000.

Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

Tom Beucler, Michael Pritchard, Stephan Rasp, Jordan Ott, Pierre Baldi, and Pierre Gentine. Enforcing analytic constraints in neural networks emulating physical systems. *Physical review letters*, 126(9):098302, 2021.

Bert Blocken, WD Janssen, and Twan van Hooff. Cfd simulation for pedestrian wind comfort and wind safety in urban areas: General decision framework and case study for the eindhoven university campus. *Environmental Modelling & Software*, 30:15–34, 2012.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural pde solvers. *arXiv preprint arXiv:2202.03376*, 2022.

Rex Edward Britter and Steven R Hanna. Flow and dispersion in urban areas. *Annual review of fluid mechanics*, 35(1):469–496, 2003.

Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual review of fluid mechanics*, 52(1):477–508, 2020.

Cheng Chen, Geng Tian, Shaoxiang Qin, Senwen Yang, Dingyang Geng, Dongxue Zhan, Jinqiu Yang, David Vidal, and Liangzhu Wang. Urban wind field prediction using fourier neural operators across different wind directions and cities. In *Building Simulation*, volume 19, pp. 271–285. Springer, 2026.

Gabriele Ciaremalla and Tommaso Vanzan. Variable reduction as a nonlinear preconditioning approach for optimization problems. *arXiv preprint arXiv:2409.01032*, 2024.

Armand de Villeroché, Rem-Sophia Mouradi, Vincent Le Guen, Sibo Cheng, Marc Bocquet, Alban Farchi, Patrick Armand, and Patrick Massin. Anchored-branched steady-state wind flow transformer (ab-swift): a metamodel for 3d atmospheric flow in urban environments. *arXiv preprint arXiv:2603.25635*, 2026.

Mathieu Desbrun, Anil N Hirani, Melvin Leok, and Jerrold E Marsden. Discrete exterior calculus. *arXiv preprint math/0508341*, 2005.

Daniel C Drucker, William Prager, and Harvey J Greenberg. Extended limit design theorems for continuous media. *Quarterly of applied mathematics*, 9(4):381–389, 1952.

Hang Du, Yi-Qing Ni, Zheng-Wei Chen, Yuan-Jiang Zeng, and Xiu-Yu Chen. St-swinunet: Real-time image-to-image reconstruction and multi-step forecasting of multi-height urban wind fields. *Building and Environment*, pp. 114593, 2026.

John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Adaptive fourier neural operators: Efficient token mixers for transformers. *arXiv preprint arXiv:2111.13587*, 2021.

Jayesh K Gupta and Johannes Brandstetter. Towards multi-spatiotemporal-scale generalized pde modeling. *arXiv preprint arXiv:2209.15616*, 2022.

Francis H Harlow, J Eddie Welch, et al. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of fluids*, 8(12):2182, 1965.

Takefumi Higaki, Yuki Tanabe, Hirotada Hashimoto, and Takahito Iida. Enhancing graph neural network-based surrogate models for lagrangian fluid simulations: Accurate pressure prediction, flexible boundary condition handling, and time-step convergence. *Flexible Boundary Condition Handling, and Time-Step Convergence*.

Henrik Hoeiness, Kristoffer Gjerde, Luca Oggiano, Knut Erik Teigen Giljarhus, and Massimiliano Ruocco. Positional encoding augmented gan for the assessment of wind flow for pedestrian comfort in urban areas. *arXiv preprint arXiv:2112.08447*, 2021.

Benjamin Holzschuh, Georg Kohl, Florian Redinger, and Nils Thuerey. P3d: Scalable neural surrogates for high-resolution 3d physics simulations with global context. *arXiv preprint arXiv:2509.10186*, 2025.

Yunfei Huang and David S Greenberg. Geometric and physical constraints synergistically enhance neural pde surrogates. *arXiv preprint arXiv:2506.05513*, 2025.

James M Hyman and Mikhail Shashkov. Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *Computers & Mathematics with Applications*, 33(4):81–104, 1997.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

Patrick Kastner and Timur Dogan. A gan-based surrogate model for instantaneous urban wind flow prediction. *Building and Environment*, 242:110384, 2023.

Dmitrii Kochkov, Jamie A Smith, Ayya Alieva, Qing Wang, Michael P Brenner, and Stephan Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118, 2021.

Georg Kohl, Li-Wei Chen, and Nils Thuerey. Benchmarking autoregressive conditional diffusion models for turbulent flow simulation. *Neural Networks*, pp. 108641, 2026.

Jean Kossaifi, Nikola Kovachki, Kamyar Azizzadenesheli, and Anima Anandkumar. Multi-grid tensorized fourier neural operator for high-resolution pdes. *arXiv preprint arXiv:2310.00120*, 2023.

Hojin Lee, Andreas Lintermann, Sangseung Lee, and Mario Rüttgers. Urbanflow-3k: A dataset of 3,000 lattice-boltzmann simulations of random building layouts. *arXiv preprint arXiv:2603.16554*, 2026.

Tianyi Li, Michele Buzzicotti, Fabio Bonaccorso, and Luca Biferale. Physics-constrained diffusion model for synthesis of 3d turbulent data. *arXiv preprint arXiv:2603.12834*, 2026a.

Xigui Li, Hongwei Zhang, Ruoxi Jiang, Deshu Chen, Chensen Lin, Limei Han, Yuan Qi, Xin Guo, and Yuan Cheng. Project and generate: Divergence-free neural operators for incompressible flows. *arXiv preprint arXiv:2603.24500*, 2026b.

Yi Li, Eric Perlman, Minping Wan, Yunke Yang, Charles Meneveau, Randal Burns, Shiyi Chen, Alexander Szalay, and Gregory Eyink. A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, (9):N31, 2008.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 936–944. Ieee, 2017.

Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard Turner, and Johannes Brandstetter. Pde-refiner: Achieving accurate long rollouts with neural pde solvers. *Advances in Neural Information Processing Systems*, 36:67398–67433, 2023.

Björn Maronga, Sabine Banzhaf, Cornelia Burmeister, Thomas Esch, Renate Forkel, Dominik Fröhlich, Vladimir Fuka, Katrin Frieda Gehrke, Jan Geletič, Sebastian Giersch, et al. Overview of the palm model system 6.0. *Geoscientific Model Development*, 13(3):1335–1372, 2020.

Michael McCabe, Peter Harrington, Shashank Subramanian, and Jed Brown. Towards stability of autoregressive neural operators. *arXiv preprint arXiv:2306.10619*, 2023.

Rajat Mittal and Gianluca Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37(1): 239–261, 2005.

Arvind T Mohan, Nicholas Lubbers, Misha Chertkov, and Daniel Livescu. Embedding hard physical constraints in neural network coarse-graining of three-dimensional turbulence. *Physical Review Fluids*, 8(1):014604, 2023.

Andreĭ Sergeevich Monin and Aleksandr Mikhaılovich Obukhov. Basic laws of turbulent mixing in the surface layer of the atmosphere. *Contrib. Geophys. Inst. Acad. Sci. USSR*, 151(163):e187, 1954.

Negin Nazarian, Jiachen Lu, Mathew J Lipson, Melissa A Hart, Sijie Liu, E Scott Krayenhoff, Lewis Blunn, and Alberto Martilli. Urbantales: A large-eddy simulation dataset for urban canopy layer turbulence and parameterization. *Bulletin of the American Meteorological Society*, 106(12): E2461–E2478, 2025.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J Agocs, Miguel Beneitez, Marsha Berger, Blakesley Burkhart, Stuart B Dalziel, Drummond B Fielding, et al. The well: a large-scale collection of diverse physics simulations for machine learning. *Advances in Neural Information Processing Systems*, 37:44989–45037, 2024.

Tsubasa Okaze, Hideki Kikumoto, Naoki Ikegaya, Keisuke Nakao, Hiroki Ono, Keigo Nakajima, Masashi Imano, Takamasa Hasama, Yuichi Tabata, Takeshi Kishida, et al. Aij guidelines on the applications of large-eddy simulation to pedestrian wind environment: Recommendations and validation benchmarks. *Journal of Wind Engineering and Industrial Aerodynamics*, 269:106321, 2026.

Vivek Oommen, Siavash Khodakarami, Aniruddha Bora, Zhicheng Wang, and George Em Karniadakis. Learning turbulent flows with generative models: Super-resolution, forecasting, and sparse flow reconstruction. *arXiv preprint arXiv:2509.08752*, 2025.

Wenhui Peng, Shaoxiang Qin, Senwen Yang, Jianchun Wang, Xue Liu, and Liangzhu Leon Wang. Fourier neural operator for real-time simulation of 3d dynamic urban microclimate. *Building and Environment*, 248:111063, 2024.

Shaoxiang Qin, Theodore Potsis, Dongxue Zhan, Xue Liu, Ted Stahopoulos, and Liangzhu Leon Wang. Flume-fno: data-efficient and scalable prediction of 3d wind and temperature fields in unseen urban morphologies. *arXiv preprint arXiv:2503.19708*, 2025a.

Shaoxiang Qin, Dongxue Zhan, Dingyang Geng, Wenhui Peng, Geng Tian, Yurong Shi, Naiping Gao, Xue Liu, and Liangzhu Leon Wang. Modeling multivariable high-resolution 3d urban microclimate using localized fourier neural operator. *Building and Environment*, 273:112668, 2025b.

Ali Rabeh, Ethan Herron, Aditya Balu, Soumik Sarkar, Chinmay Hegde, Adarsh Krishnamurthy, and Baskar Ganapathysubramanian. Geometry matters: benchmarking scientific ml approaches for flow prediction around complex geometries. *arXiv preprint arXiv:2501.01453*, 2024.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pp. 5301–5310. PMLR, 2019.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. 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, 2019.

Jack Richter-Powell, Yaron Lipman, and Ricky TQ Chen. Neural conservation laws: A divergence-free perspective. *Advances in Neural Information Processing Systems*, 35:38075–38088, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.

Benjamin D Shaffer, Shawn Koohy, Brooks Kinch, M Ani Hsieh, and Nathaniel Trask. Structure-preserving learning improves geometry generalization in neural pdes. *arXiv preprint arXiv:2602.02788*, 2026.

Reda Snaiki, Jiachen Lu, Shaopeng Li, and Negin Nazarian. A hierarchical deep learning model for predicting pedestrian-level urban winds. *Building and Environment*, pp. 114354, 2026.

Kimberly Stachenfeld, Drummond B Fielding, Dmitrii Kochkov, Miles Cranmer, Tobias Pfaff, Jonathan Godwin, Can Cui, Shirley Ho, Peter Battaglia, and Alvaro Sanchez-Gonzalez. Learned coarse models for efficient turbulence simulation. *arXiv preprint arXiv:2112.15275*, 2021.

Natarajan Sukumar and Ankit Srivastava. Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 389:114333, 2022.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Daniel MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. Pdebench: An extensive benchmark for scientific machine learning. *Advances in neural information processing systems*, 35:1596–1611, 2022.

Jonathan H Tu. *Dynamic mode decomposition: Theory and applications*. Princeton University, 2013.

Nils Wandel, Michael Weinmann, and Reinhard Klein. Teaching the incompressible navier–stokes equations to fast neural surrogate models in three dimensions. *Physics of Fluids*, 33(4), 2021.

Houzhi Wang, Wei Ma, Jianlei Niu, and Ruoyu You. Evaluating a deep learning-based surrogate model for predicting wind distribution in urban microclimate design. *Building and Environment*, 269:112426, 2025a.

Houzhi Wang, Wei Ma, Jianlei Niu, and Ruoyu You. A geometry-conditioned transformer solver for predicting three-dimensional wind distribution in urban microclimate design. *Urban Climate*, 64: 102634, 2025b.

Shizheng Wen, Arsh Kumbhat, Levi Lingsch, Sepehr Mousavi, Yizhou Zhao, Praveen Chandrashekar, and Siddhartha Mishra. Geometry aware operator transformer as an efficient and accurate neural surrogate for pdes on arbitrary domains. *Advances in Neural Information Processing Systems*, 38: 155423–155501, 2026.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. *arXiv preprint arXiv:2402.02366*, 2024.

Yihan Wu, Zihan Cao, Yubin Lei, Yuxuan Han, Man Yuan, Lihui Wang, and Xingang Zhou. Multi-constraint physics-informed generative adversarial networks (pigan) enable small-data learning for urban wind field prediction. *Building and Environment*, pp. 114202, 2026.

Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *arXiv preprint arXiv:1901.06523*, 2019.

Francesco Zanetta, Daniele Nerini, Tom Beucler, and Mark A Liniger. Physics-constrained deep learning postprocessing of temperature and humidity. *Artificial Intelligence for the Earth Systems*, 2(4):e220089, 2023.

Shixiong Zhou, Huanxia Wei, Chao Xia, Yingying Xing, Changmin Jiang, Hai Yang, and Shuai Jia. U3dwind: A low altitude wind field dataset and benchmark for urban air mobility. *arXiv preprint arXiv:2607.04495*, 2026.

This appendix contains details about proofs, dataset, experimental implementation, supplementary experiments, related works, limitations, and future works.

A PROOF

A.1 PROOFS OF PROPOSITIONS 3.1 AND 3.2

Proof of Proposition 3.1. Fix a cell c and sum Eq. (4) over its six oriented faces. Every edge of c is shared by exactly two of those faces and enters them with opposite incidence signs $\sigma_{f,e}$, so all edge contributions cancel in pairs and $(\nabla_h \cdot \Delta \mathbf{u})_c = 0$. The argument uses only the incidence structure of the grid, hence holds for every $\mathbf{A}$ and every mask g. □

Proof of Proposition 3.2. Every $f \in \mathcal{F}_\partial(g)$ has all four of its edges in $\mathcal{E}_\partial(g)$, so Eq. (4) evaluates Δu_f from boundary values alone. The first term of Eq. (5) is a discrete gradient, whose circulation around the oriented boundary of f telescopes to zero. The second term is a closed 1-cochain on $\mathcal{B}$, so its circulation around the 2-cell f vanishes by definition of closedness. Summing the two gives $\Delta u_f = 0$. □

A.2 SPECTRAL GAIN OF THE CURL READOUT

This appendix proves the statement made in Section 3.2, that eliminating the constraints through a potential attenuates the gradient signal that reaches the potential at large scales and amplifies it at small ones.

Setting and Notation. Throughout this appendix we work on the uniform grid of spacing h with periodic boundary conditions in all three directions, which isolates the effect of the readout from that of the boundary treatment of Section 3.1. Write $\mathcal{R}$ for the linear map that assembles the update from the volume vector potential, so that $\Delta \mathbf{u} = \mathcal{R}\mathbf{A}$ is Eq. (4) written for all faces at once, and write $\mathcal{R}^\top$ for its adjoint. For any loss $\mathcal{L}$ the chain rule gives

$$\frac{\partial \mathcal{L}}{\partial \mathbf{A}} \;=\; \mathcal{R}^\top \mathbf{r}, \qquad \mathbf{r} \;=\; \frac{\partial \mathcal{L}}{\partial \Delta \mathbf{u}}, \tag{12}$$

so the quantity to be characterized is the gain of $\mathcal{R}^\top$ as a function of spatial scale. Let $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3)$ denote an admissible wavevector of the periodic grid, with $\xi_\alpha \in (-\pi/h, \pi/h]$ for $\alpha \in \{1, 2, 3\}$, and write $\mathbf{A}_{\boldsymbol{\xi}}$ and $\mathbf{r}_{\boldsymbol{\xi}}$ for the discrete Fourier coefficients of $\mathbf{A}$ and of $\mathbf{r}$ at $\boldsymbol{\xi}$. On a uniform periodic grid, $\mathcal{R}$ is translation-invariant and hence is block-diagonalized by the discrete Fourier transform, with one 3×3 block $\mathcal{R}_{\boldsymbol{\xi}}$ per wavevector. Write D_α for the one-sided difference along axis α and define

$$q_\alpha(\boldsymbol{\xi}) \;=\; \frac{e^{\mathrm{i}\xi_\alpha h} - 1}{h}, \qquad |q_\alpha(\boldsymbol{\xi})| \;=\; \frac{2}{h}\Big|\sin\frac{\xi_\alpha h}{2}\Big|, \tag{13}$$

so that q_α is the symbol of D_α. We collect the three symbols into the vector $\mathbf{q}(\boldsymbol{\xi})$.

Proposition A.1. *The block $\mathcal{R}_{\boldsymbol{\xi}}$ is the cross-product matrix of $\mathbf{q}(\boldsymbol{\xi})$, and for every $\boldsymbol{\xi} \neq \mathbf{0}$,*

$$\mathcal{R}_{\boldsymbol{\xi}}^* \mathcal{R}_{\boldsymbol{\xi}} \;=\; |\mathbf{q}(\boldsymbol{\xi})|^2 \left(\mathbf{I} - \Pi_{\boldsymbol{\xi}}\right), \qquad \Pi_{\boldsymbol{\xi}} \;=\; \frac{\mathbf{q}(\boldsymbol{\xi})\,\mathbf{q}(\boldsymbol{\xi})^*}{|\mathbf{q}(\boldsymbol{\xi})|^2}, \tag{14}$$

where $\Pi_{\boldsymbol{\xi}}$ is the orthogonal projector onto the span of $\mathbf{q}(\boldsymbol{\xi})$, which is the range of the discrete gradient at that wavevector. Consequently the singular values of $\mathcal{R}_{\boldsymbol{\xi}}$ are $|\mathbf{q}(\boldsymbol{\xi})|$ twice and 0 once, and the same holds for $\mathcal{R}_{\boldsymbol{\xi}}^*$.

Proof. On the staggered grid the three components of Eq. (4) read $\Delta u = D_2 A^3 - D_3 A^2$, $\Delta v = D_3 A^1 - D_1 A^3$ and $\Delta w = D_1 A^2 - D_2 A^1$, where A^α collects the entries of $\mathbf{A}$ on the edges aligned with axis α. Applying the discrete Fourier transform turns D_α into multiplication by $q_\alpha(\boldsymbol{\xi})$ of Eq. (13), so $\mathcal{R}_{\boldsymbol{\xi}} \mathbf{A}_{\boldsymbol{\xi}} = \mathbf{q}(\boldsymbol{\xi}) \times \mathbf{A}_{\boldsymbol{\xi}}$, which is the first claim. Cross-product matrices are antisymmetric, so $\mathcal{R}_{\boldsymbol{\xi}}^* = -[\bar{\mathbf{q}}]_\times$, and the bilinear identity $[\mathbf{a}]_\times [\mathbf{b}]_\times = \mathbf{b}\mathbf{a}^\top - (\mathbf{a}^\top \mathbf{b})\mathbf{I}$ gives

$$\mathcal{R}_{\boldsymbol{\xi}}^* \mathcal{R}_{\boldsymbol{\xi}} \;=\; -[\bar{\mathbf{q}}]_\times [\mathbf{q}]_\times \;=\; (\bar{\mathbf{q}}^\top \mathbf{q})\,\mathbf{I} - \mathbf{q}\,\bar{\mathbf{q}}^\top \;=\; |\mathbf{q}|^2 \mathbf{I} - \mathbf{q}\mathbf{q}^*, \tag{15}$$

which is Eq. (14). The matrix $\mathbf{q}\mathbf{q}^*/|\mathbf{q}|^2$ is Hermitian, idempotent and of rank one, hence the orthogonal projector onto $\mathrm{span}(\mathbf{q})$, and $\mathbf{q}$ is exactly the Fourier block of the discrete gradient, whose entries are the same one-sided differences. Eq. (14) therefore has eigenvalues $|\mathbf{q}|^2$ with multiplicity two and 0 with multiplicity one, which are the squared singular values of $\mathcal{R}_{\boldsymbol{\xi}}$. At $\boldsymbol{\xi} = \mathbf{0}$ one has $\mathbf{q} = \mathbf{0}$ and $\mathcal{R}_{\mathbf{0}} = 0$, so a spatially constant potential produces no update. □

The projector in Eq. (14) is the Fourier statement of the identity that a discrete gradient has no curl, which Section 3.1 uses at the boundary: the one direction that $\mathcal{R}_{\boldsymbol{\xi}}$ annihilates is the gradient direction.

Corollary A.2. *Let $\boldsymbol{\xi} \neq \mathbf{0}$ and let $\mathbf{r}_{\boldsymbol{\xi}}^{\parallel}$ be the component of $\mathbf{r}_{\boldsymbol{\xi}}$ in the range of $\mathcal{R}_{\boldsymbol{\xi}}$. Then the gradient that reaches the potential satisfies*

$$\left\|\mathcal{R}_{\boldsymbol{\xi}}^{*}\,\mathbf{r}_{\boldsymbol{\xi}}\right\| \;=\; |\mathbf{q}(\boldsymbol{\xi})|\,\left\|\mathbf{r}_{\boldsymbol{\xi}}^{\parallel}\right\| \;\leq\; |\mathbf{q}(\boldsymbol{\xi})|\,\left\|\mathbf{r}_{\boldsymbol{\xi}}\right\|, \tag{16}$$

with

$$|\mathbf{q}(\boldsymbol{\xi})|^2 = \frac{4}{h^2}\sum_{\alpha=1}^{3}\sin^2\frac{\xi_\alpha h}{2}, \qquad |\mathbf{q}(\boldsymbol{\xi})| = |\boldsymbol{\xi}| + O(|\boldsymbol{\xi}|^3 h^2)\ \text{ as } |\boldsymbol{\xi}|h \to 0, \qquad \max_{\boldsymbol{\xi}} |\mathbf{q}(\boldsymbol{\xi})| = \frac{2\sqrt{3}}{h}. \tag{17}$$

The gain $|\mathbf{q}(\boldsymbol{\xi})|$ is nondecreasing in each $|\xi_\alpha|$ on $[0, \pi/h]$.

Proof. Eq. (16) is Proposition A.1 applied to $\|\mathcal{R}_{\boldsymbol{\xi}}^{*}\mathbf{r}_{\boldsymbol{\xi}}\|^2 = \mathbf{r}_{\boldsymbol{\xi}}^{*}\mathcal{R}_{\boldsymbol{\xi}}\mathcal{R}_{\boldsymbol{\xi}}^{*}\mathbf{r}_{\boldsymbol{\xi}}$, together with $\mathcal{R}_{\boldsymbol{\xi}}\mathcal{R}_{\boldsymbol{\xi}}^{*} = |\mathbf{q}|^2(\mathbf{I} - \Pi_{\boldsymbol{\xi}}^{\perp})$, where $\Pi_{\boldsymbol{\xi}}^{\perp}$ projects onto $\ker\mathcal{R}_{\boldsymbol{\xi}}^{*}$, which is the orthogonal complement of the range of $\mathcal{R}_{\boldsymbol{\xi}}$. The remaining statements follow from Eq. (13) and from $\sin x = x + O(x^3)$, with the maximum attained when every $|\xi_\alpha|$ equals π/h. □

The factor by which the gradient is rescaled on its way to the potential is $|\mathbf{q}(\boldsymbol{\xi})|$, which vanishes linearly as the wavelength grows and is largest at the grid scale, so a mode of wavelength $2\pi/|\boldsymbol{\xi}|$ receives a gradient smaller by a factor of order $|\boldsymbol{\xi}|h$ than a mode at the grid scale receives, and the gain at the largest resolved wavelength falls below the gain at the Nyquist wavenumber by a factor of order the ratio of the domain size to the grid spacing. Any elimination of a differential constraint through a potential contributes the singular values of the eliminating operator in the same way, so the attenuation is a property of the change of variables. The component of the face-side gradient outside the range of $\mathcal{R}_{\boldsymbol{\xi}}$ never reaches the potential, which is the gradient-side counterpart of the fact that the construction emits only fields inside $\mathcal{V}(g)$.

Remark A.3. On a flat portion of the boundary the same computation applies to the composition through which the shared scalar potential acts, namely a gradient followed by a curl. Each factor contributes one power of $|\mathbf{q}(\boldsymbol{\xi})|$, so the gain from the loss to G carries $|\mathbf{q}(\boldsymbol{\xi})|^2$ and the attenuation at large scales is squared. This scaling motivates the multi-resolution readout of G in Section 3.3. The boundary complex is not a periodic lattice, so the argument is exact where the boundary is flat and serves as a guide elsewhere.

B Data Details

B.1 Dataset Generation

Every layout is a building height field on the horizontal grid, extruded from the ground so that each building is a filled column, and is fixed by a morphology type and a random seed. The type sets the composition of the sixteen blocks that tile the domain, from archetypes differing in footprint, spacing and storey count: courtyard blocks, rows of elongated buildings, isolated towers, towers on a low podium, low-rise fabric and open ground. It thereby fixes the plan-area density and the height distribution. The seed draws the street-grid offsets, the archetype of each block, the storey count of each building and the placement of each footprint. Streets run straight through the domain at widths of 24, 16 and 6 m on a block rhythm of 128 m, which divides the domain width and so respects the horizontal periodicity. Storeys are 3 m high and every height is quantized to h, so that no building edge cuts a cell.

All sequences come from the PALM large-eddy simulation model (Maronga et al., 2020) on a $24\times128\times128$ grid of uniform spacing $h = 2$ m, a domain of $48\times256\times256$ m that is doubly periodic

horizontally, bounded below by rough ground and above by a free-slip lid. The solver blocks the cells inside a building and writes velocities on the native Arakawa-C staggered grid. Subgrid stresses use the 1.5-order closure with the mixing length limited near surfaces, advection uses the Piacsek–Williams scheme, and time stepping is third-order Runge–Kutta with the perturbation pressure solved at every substep. That projection uses the multigrid solver with masking disabled, an unbounded cycle count, a residual limit of 10^{-13} and four Gauss–Seidel sweeps per level. The roughness length is $z_0 = 0.1\,\mathrm{m}$ on the ground and on all building surfaces, which carry a stress-type condition from Monin-Obukhov similarity theory (Monin & Obukhov, 1954). The normal velocity vanishes on the ground, the buildings and the lid, and the Coriolis parameter is zero.

Each run starts from constant profiles at a surface potential temperature of $300\,\mathrm{K}$ with small random disturbances and integrates for $1{,}000\,\mathrm{s}$, after which 100 consecutive snapshots at intervals of $1\,\mathrm{s}$ form one sequence. The inflow enters through the two surface geostrophic wind components, which carry the free-stream speed and the angle to the fixed street grid, so that direction varies independently of geometry. Stratification enters through the surface heat flux and the temperature lapse rate: $0.1\,\mathrm{K\,m\,s^{-1}}$ with no lapse for the convective condition, zero with no lapse for the neutral one, and $-0.02\,\mathrm{K\,m\,s^{-1}}$ with $0.01\,\mathrm{K\,m^{-1}}$ for the stable one. The nine inflow conditions combine three wind directions ($15°$, $30°$, $45°$), three free-stream speeds (3, 6, $9\,\mathrm{m\,s^{-1}}$), and stability (convective, neutral, stable). The three conditions held out for testing are $(15°, 9\,\mathrm{m\,s^{-1}}, \mathrm{convective})$, $(30°, 3\,\mathrm{m\,s^{-1}}, \mathrm{neutral})$ and $(45°, 6\,\mathrm{m\,s^{-1}}, \mathrm{stable})$. **UrbanWindFlow** holds 85 such sequences. A PALM run uses 16 MPI processes. During the whole experiments, we use an Intel(R) Xeon(R) Gold 5318Y CPU and one NVIDIA A800 80GB PCIe GPU with 80GB memory to train and inference.

Table 6: Related flow-field benchmark that wind surrogates could be evaluated on. **Solver** names the method that generated the fields. **3D** marks volumetric fields. **Obstacles** marks solid bodies that sit inside the domain and are resolved by the grid. **Transient** marks instantaneous slices. **Geometry** marks the geometric shift.

Dataset	Solver	3D	Obstacles	Transient	Geometry
PDEBench (Takamoto et al., 2022)	Canonical PDE solvers	✓	×	✓	×
The Well (Ohana et al., 2024)	Multi-physics solvers	✓	×	✓	×
PDEArena (Gupta & Brandstetter, 2022)	Canonical PDE solvers	×	×	✓	×
JHTDB (Li et al., 2008)	DNS	✓	×	✓	×
Gen4Turbulence (Oommen et al., 2025)	DNS	✓	×	✓	×
Geometry Matters (Rabeh et al., 2024)	Steady 2D solver	×	✓	×	✓
UrbanFlow-3K (Lee et al., 2026)	LBM, 2D	×	✓	×	✓
UrbanTALES (Nazarian et al., 2025)	LES (PALM)	✓	✓	×	✓
U3DWind (Zhou et al., 2026)	LBM-LES	✓	✓	×	✓
UrbanWindFlow (ours)	**LES (PALM)**	✓	✓	✓	✓

B.2 Comparison with Existing Benchmarks

Measuring Eq. (1) requires data in which the solid boundaries are present and resolved, and measuring it under time stepping requires instantaneous snapshots. Table 6 summarizes the public benchmarks against the two requirements. To our knowledge no publicly released dataset provides volumetric, instantaneous velocity fields over resolved urban geometry. **UrbanWindFlow** is built to fill this gap. We use the tensor layout, resolution class and snapshot count of Gen4Turbulence (Oommen et al., 2025) as a reference, and extend it with the two elements that our setting requires, a per-layout solid-occupancy mask and a sample axis spanning procedurally generated building layouts.

C Experimental Details

C.1 Implementation Details.

During the evaluation, all velocities are normalized by the free-stream speed, and the divergence is computed from these normalized velocities. The wall residual is taken over every face that is not bounded by fluid on both sides. Within every training sequence the frames are split as [3,70) for training and [76,84) for validation, with a six-frame gap so that no validation frame is temporally

adjacent to a training frame. Training and evaluation are separated over sequences: a test sequence is absent from training at every frame, so the two may occupy the same time range. Every test sequence contributes the same 18 start frames $t_0 = 66, \ldots, 83$, each rolled out to $t_0 + 16 \leq 99$.

All evaluation use $n = 4$ history frames. Every method receives the same input and the same split, and is trained for 120 epochs with Adam at a learning rate of 10^{-3}, no learning-rate schedule. **Sculpt** is trained with a mean squared reconstruction term alone, with no penalty on the constraints, and it is trained with the pushforward scheme of Brandstetter et al. (2022): with probability p the most recent input frame is replaced by the model's own one-step prediction, taken without gradient, and the target is advanced by one step so that it remains the increment relative to that prediction. All convolutions pad circularly along the two horizontal axes, which is what the doubly periodic domain requires, and with zeros along the vertical, where the ground and the lid are physical boundaries. This applies to **Sculpt** and to every convolutional baseline alike. **Sculpt** uses backbone widths $C_0, C_1, C_2 = 32, 64, 128$ over $S = 3$ resolution levels, multiresolution width $c_{\mathcal{A}} = 16$, surface width $c_{\mathcal{B}} = 32$, descriptor width $d_{\mathrm{des}} = 8$ and $K = 4$ message-passing jumps. For every method, including **Sculpt**, we keep the epoch at which the 16-step rollout error on the validation frames is lowest, following the rollout-based selection of PDEArena (Gupta & Brandstetter, 2022).

C.2 Ablation Variants

The first four variants concern where the constraint is imposed.

w/o null-space Ψ_g. The structured output representation of Eq. (3) is removed and the network regresses the three velocity components directly, through the same read-out head that the baselines use.

w/ soft penalty. Starting from W/O NULL-SPACE Ψ_g, the training objective gains a divergence term and a wall term, $\lambda(\|\nabla_h \cdot \mathbf{u}\|^2 + \|u_f\|^2)$ with $\lambda = 1$ on both.

w/ post-projection. Starting from W/O NULL-SPACE Ψ_g, and without any further training, each predicted step is passed through a pressure projection: the normal velocity is cleared on solid faces, a Poisson problem $\nabla_h^2 \phi = \nabla_h \cdot \mathbf{u}$ with homogeneous Neumann conditions on solid boundaries is solved by conjugate gradients in double precision to a tolerance of 10^{-10}, and the gradient of ϕ is subtracted.

w/o boundary G. The discrete curl of Eq. (4) is kept, so Proposition 3.1 still applies, but the substitution of Eq. (5) on boundary edges is skipped: those edges retain the free output of the network instead of being set to a discrete gradient of the boundary potential.

The remaining variants vary design choices that are not required for either guarantee.

w/o increment $\Delta\mathbf{u}$. The network predicts the absolute field $\mathbf{u}^{t+1}$ instead of the increment $\mathbf{u}^{t+1} - \mathbf{u}^t$.

w/o multi-scale A. The set of resolutions in Eq. (6) is reduced from $\mathcal{S} = \{1, 2, 4\}$ to $\mathcal{S} = \{1\}$, so the potential is read out from the finest backbone features alone.

w/o learned fusion. At each boundary node the surface network gathers the eight surrounding cells of the backbone feature map and combines them with weights produced by a small learned scorer. Here that scorer is deleted and the eight slots are averaged uniformly over the valid ones.

D Supplementary experiments

D.1 Performance on **unseen inflows** setting

Table 7 reports the **unseen inflows** setting, whose test sequences hold out three combinations of wind direction, free-stream speed and stability. **Sculpt** is first at all four horizons, ahead of the strongest neural baseline by 55.7% at k=1 and by 33.2% at k=16. This setting separates the methods differently from the other two: every neural baseline degrades far enough by k=16 that POD overtakes all of them, and AFNO leaves the range of the table altogether.

Table 7: Performance on the **unseen inflows** setting of **UrbanWindFlow**. MSE is reported at four rollout horizons. Highlighted are the top first, second results. '<' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Methods	MSE↓ ($\times 10^{-3}$)				worst over all k	
	k=1	k=4	k=8	k=16	$\max\lvert\nabla\cdot\mathbf{u}\rvert\downarrow$	$\max\lvert\mathbf{u}\cdot\mathbf{n}\rvert\downarrow$
DMD	87.65±27.47	87.49±27.13	98.42±29.17	100.31±26.95	$< 10^{-6}$	0
Time-Mean	47.47±15.62	73.29±22.01	80.51±24.51	82.36±23.10	$< 10^{-6}$	0
POD	50.64±15.09	51.21±14.96	52.01±15.16	52.44±15.12	$< 10^{-5}$	0.30
Dil-ResNet	21.56±10.67	56.34±17.34	84.91±21.44	201.57±42.63	2.09	2.26
TFNO	35.86±10.78	82.91±23.58	137.90±50.52	188.84±86.16	0.83	1.60
AFNO	22.19±8.15	51.87±15.38	64.85±17.64	1.6×10^{3}	5.9×10^{2}	2.63
AViT	14.64±4.59	42.54±13.20	70.31±18.66	93.57±22.99	1.40	1.64
P3D	3.02±1.90	19.57±14.73	44.39±24.34	83.77±26.10	0.37	1.70
U-Net	2.01±1.29	10.85±5.60	30.62±12.51	69.36±17.57	0.24	1.30
Sculpt	**0.89±0.67**	**6.35±5.41**	**19.53±15.00**	**46.31±25.52**	$< 10^{-6}$	$< 10^{-6}$

D.2 Ablation on **unseen inflows** setting

Table 8 repeats the ablation of Section 4.2 on the same setting, where **Sculpt** is again first at both horizons. Two components change rank here. The multiresolution potential and its learned fusion carry far more weight under inflow shift, costing 30.7% and 51.6% at k=16 against 1.8% and 16.0% on the unseen layout setting.

Table 8: Ablation on the **unseen inflows** setting of **UrbanWindFlow**. MSE is reported at two rollout horizons. Highlighted are the top first, second results. '<' in two constraint columns means the power-of-ten bound on the measured value, and 0 is exact.

Variant	MSE↓		$\max\lvert\nabla\cdot\mathbf{u}\rvert\downarrow$		$\max\lvert\mathbf{u}\cdot\mathbf{n}\rvert\downarrow$	
	k=1	k=16	k=1	k=16	k=1	k=16
Sculpt	**0.89±0.67**	**46.31±25.52**	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
w/o null-space Ψ_g	1.71±1.52	63.81±32.34	0.28	1.24	1.37	3.32
w/ soft penalty	12.89±3.36	109.27±23.41	0.70	0.84	0.51	0.71
w/ post-projection	1.63±1.49	60.98±33.39	$< 10^{-6}$	$< 10^{-6}$	0	0
w/o boundary G	1.93±2.07	64.68±38.22	$< 10^{-6}$	$< 10^{-6}$	1.34	1.72
w/o increment $\Delta\mathbf{u}$	3.18±2.13	2122.96±178.64	$< 10^{-6}$	$< 10^{-5}$	$< 10^{-6}$	$< 10^{-5}$
w/o multi-scale $\mathbf{A}$	2.56±3.50	60.51±31.26	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
w/o learned fusion	1.05±0.78	70.22±37.12	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$

D.3 Computational complexity

We analyze the computational complexity of **Sculpt** to show its efficiency. Let $N = D \cdot H \cdot W$ denote the number of grid cells, $|\mathcal{N}|$ and $|\mathcal{E}_\partial(g)|$ represent the numbers of nodes and of edges of the boundary complex, C_0 indicate the width of the finest backbone level, and $c_\mathcal{A}$, $c_\mathcal{B}$ and d_{des} the widths of the multiresolution branches, of the surface graph network and of the node descriptor, respectively, with S resolution levels and K rounds of message passing. A $3\times3\times3$ convolution at level s acts on $N/8^s$ cells with $C_s = 2^s C_0$ channels, so the computational complexity of the model during the volumetric stage is given by $O\big(\sum_s (C_s^2 + c_\mathcal{A} C_s)\, N/8^s + SN\big)$. Generally speaking, C_0, $c_\mathcal{A}$ and S are significantly smaller than N, and the geometric sum over the levels is dominated by the finest one. So the time complexity of the first stage of **Sculpt** is $O(C_0^2 N)$.

The computational complexity of the second stage is $O\big(|\mathcal{N}|\, c_\mathcal{B}(C_0 + d_{\text{des}} + K c_\mathcal{B} + S)\big)$, where the first term collects the learned gather of Eq. (8) over the eight cells incident to each node, the second

collects the K rounds of Eq. (9) with one $c_{\mathcal{B}} \times c_{\mathcal{B}}$ matrix per axis direction, and the third collects the read-out of Eq. (10). The assembly operator writes one value on each edge of $\mathcal{E}_\partial(g)$ and the discrete curl of Eq. (4) sums four terms on each face, which together cost $O(|\mathcal{E}_\partial(g)| + N)$ and carry no parameters and no linear system. On **UrbanWindFlow** the boundary complex spans between 9.6% and 12.3% of the cells, so $|\mathcal{N}|$ and $|\mathcal{E}_\partial(g)|$ are $O(N)$ with a small constant. Therefore, the time complexity of **Sculpt** is $O(C_0^2 N + K c_{\mathcal{B}}^2 |\mathcal{N}| + N)$, which reduces to $O(N)$ at fixed widths. Although our model enforces both constraints by construction, it has comparable complexity with the convolutional baselines, whose cost is $O(C^2 N)$ at the same resolution.

D.4 Parameters Analysis

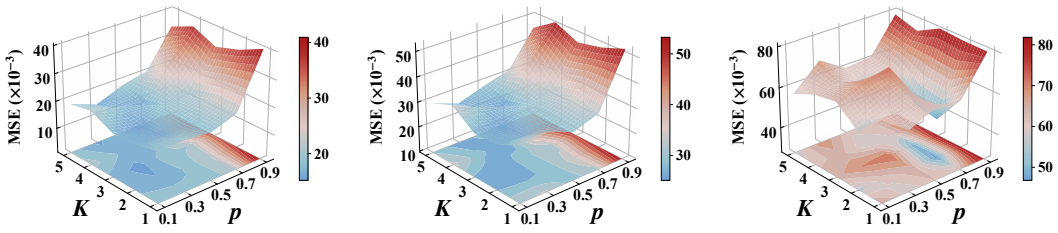

Figure 6: Sensitivity of the 16-step MSE to the number of message-passing rounds K on the boundary complex and to the pushforward probability p.

Sculpt introduces two hyper-parameters: the number of message-passing jump K on the boundary complex and the pushforward probability p. Figure 6 reports the 16-step MSE of different hyper-parameters setting. The shape along p is the same in all three settings: a plateau up to $p = 0.5$, followed by an increase error at $p = 0.9$. Training almost exclusively on the model's own predictions leaves few steps anchored to the ground truth, leading to the instability of training process. The effect of the K value is relatively stable, and the optimal value is around 4. We therefore fix $K = 4$ and $p = 0.5$ for all evaluation settings.

D.5 Visualization

Figures 11, 16 and 21 extend Figure 5 to three further test sequences and to all four reported horizons. Each row of a figure shows one horizon, and the four panels of a row place **Sculpt** beside U-Net, P3D and AViT on the same field with one common colour scale, so that panels are comparable within a row and across rows. Three observations hold on all three sequences. The error volume grows with the horizon for every method, and the shells of the baselines merge into connected sheets between $k = 8$ and $k = 16$ while those of **Sculpt** remain separated. At $k = 1$ the four methods are close, and the ordering that the tables report at $k = 16$ is already visible at $k = 4$. The regions that fill first lie in the layer above the roofs, which is where Section 4.2 locates the largest error of every method.

E Related Works

Neural surrogates for urban wind. Surrogates for neighbourhood-scale wind divide into two families according to what they return. The first predicts a field in a single forward pass. U-Net architectures regress the pedestrian-level field from a layout mask and recover its small scales with a generative refiner (Hoeiness et al., 2021; Kastner & Dogan, 2023; Snaiki et al., 2026; Wu et al., 2026; Wang et al., 2025a; Du et al., 2026). Neural operators transfer across wind directions and across cities (Chen et al., 2026; Peng et al., 2024; Qin et al., 2025a;b), and geometry-aware operators evaluate on arbitrary domains (Wu et al., 2024; Wen et al., 2026; de Villeroché et al., 2026; Wang et al., 2025b). The fields these methods return are steady or time-averaged, and in most cases two-dimensional. The second family consists of general neural PDE surrogates (Gupta & Brandstetter, 2022; Brandstetter et al., 2022; Kochkov et al., 2021), which learn a single-step evolution operator with convolutional, spectral or attention mechanisms (Ronneberger et al., 2015; Stachenfeld et al., 2021; Li et al., 2020; Guibas et al., 2021; Holzschuh et al., 2025) and apply it to their own output



Figure 7: Isosurfaces of the prediction error at k=1 on a case. Buildings are grey.



Figure 8: Isosurfaces of the prediction error at k=4 on a case. Buildings are grey.

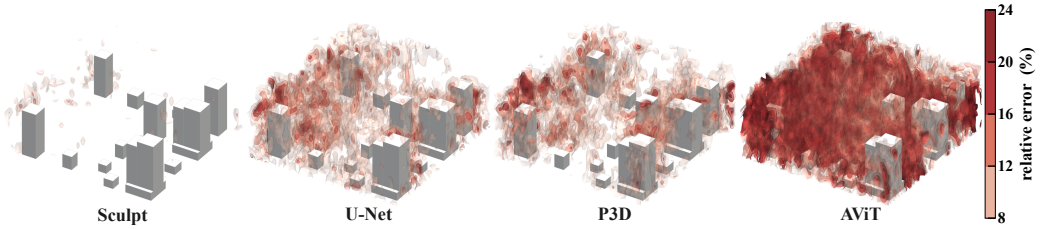

Figure 9: Isosurfaces of the prediction error at k=8 on a case. Buildings are grey.

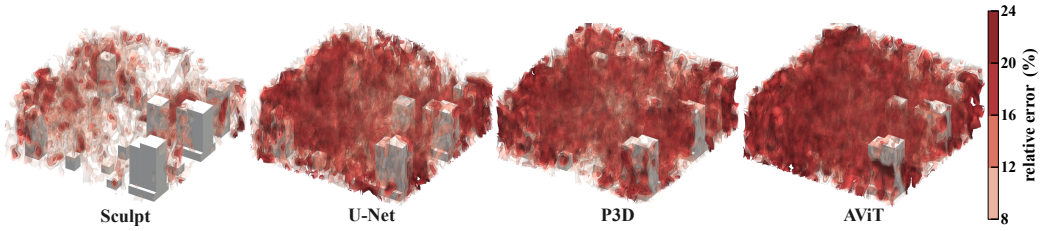

Figure 10: Isosurfaces of the prediction error at k=16 on a case. Buildings are grey.

Figure 11: Prediction error on a test sequence of the **unseen morphologies** setting, at the four reported horizons. Shells share one scale across all panels and all horizons.

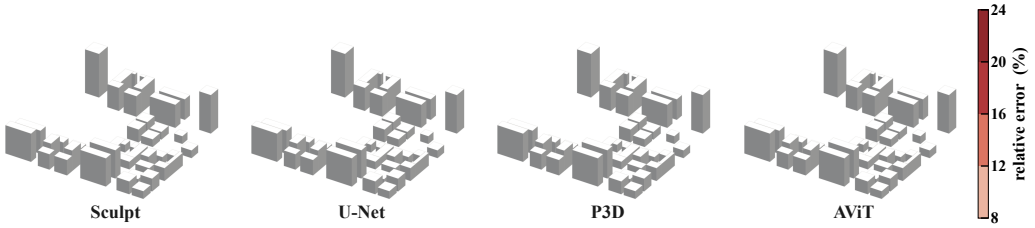

Figure 12: Isosurfaces of the prediction error at k=1 on a case. Buildings are grey.

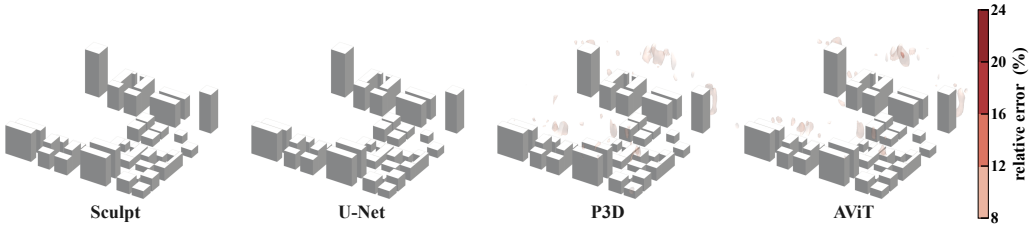

Figure 13: Isosurfaces of the prediction error at k=4 on a case. Buildings are grey.

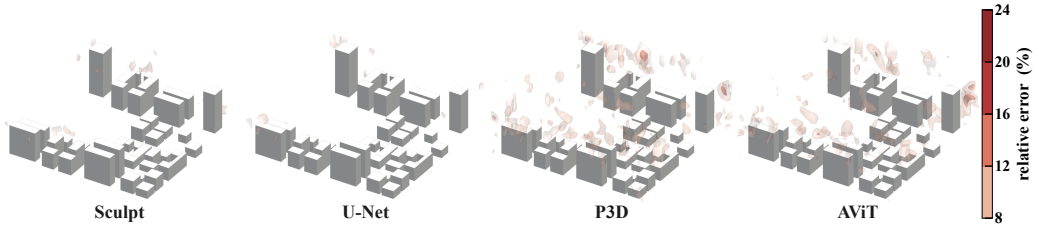

Figure 14: Isosurfaces of the prediction error at k=8 on a case. Buildings are grey.

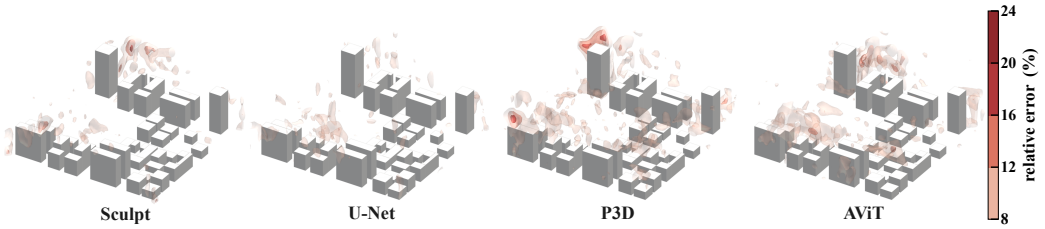

Figure 15: Isosurfaces of the prediction error at k=16 on a case. Buildings are grey.

Figure 16: Prediction error on a test sequence of the **unseen inflows** setting, at the four reported horizons. Shells share one scale across all panels and all horizons; buildings are grey.

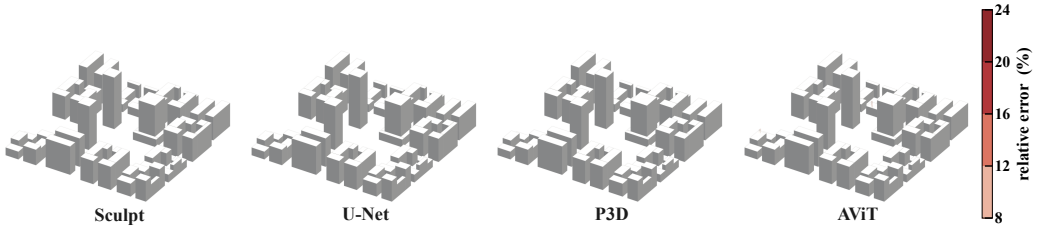

Figure 17: Isosurfaces of the prediction error at k=1 on a case. Buildings are grey.

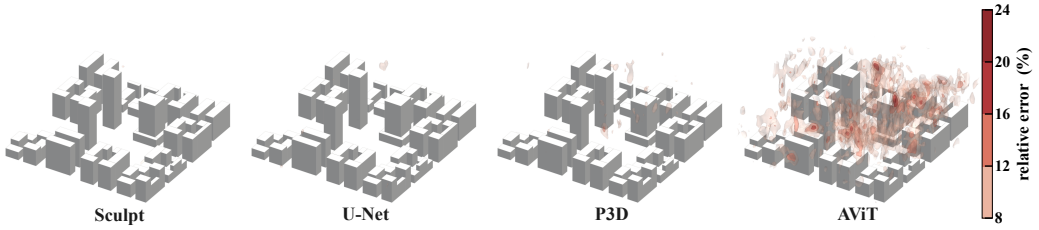

Figure 18: Isosurfaces of the prediction error at k=4 on a case. Buildings are grey.

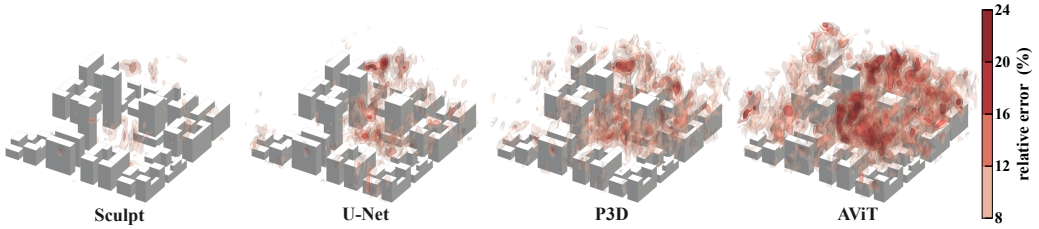

Figure 19: Isosurfaces of the prediction error at k=8 on a case. Buildings are grey.

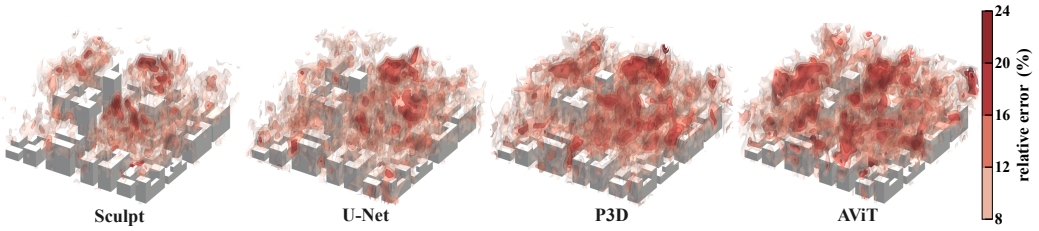

Figure 20: Isosurfaces of the prediction error at k=16 on a case. Buildings are grey.

Figure 21: Prediction error on a test sequence of the **unseen inflows** setting, at the four reported horizons. Shells share one scale across all panels and all horizons; buildings are grey.

over a rollout (Lippe et al., 2023; McCabe et al., 2023; Kohl et al., 2026). These methods return three-dimensional transient fields, and the layout enters them as an input channel that conditions the fitted map. **Sculpt** belongs to the second family, and the layout additionally determines the space in which its output is expressed.

Constraint satisfaction in neural surrogates. Three mechanisms are in use. Penalty methods add the residual of the constraint to the training objective (Raissi et al., 2019; Karniadakis et al., 2021), so that the constraint holds to the accuracy the optimiser reaches. Projection methods correct the prediction after inference, typically through a pressure solve, and restore the constraint exactly at the cost of a linear solve at every step. Structured output representations satisfy the constraint by construction, through divergence-free parameterisations and vector potentials (Beucler et al., 2021; Wandel et al., 2021; Richter-Powell et al., 2022; Mohan et al., 2023; Huang & Greenberg, 2025; Li et al., 2026a) or through geometry-dependent compatible spaces for steady problems (Shaffer et al., 2026). The variants W/ SOFT PENALTY and W/ POST-PROJECTION of Appendix C.2 instantiate the first two mechanisms on the backbone of **Sculpt**. Existing structured representations are formulated on periodic domains, or on domains whose boundary is aligned with the grid. A building surface is immersed in a structured grid (Mittal & Iaccarino, 2005) and moves with the layout, so the two constraints considered here act on a shared set of degrees of freedom.

Spectral properties of learned evolution operators. The tendency of neural networks to fit low frequencies first is documented (Rahaman et al., 2019; Xu et al., 2019; Oommen et al., 2025) and follows from the optimisation dynamics. Appendix A.2 characterises a second mechanism, in which the adjoint of the discrete curl, a fixed operator carrying no parameters, rescales the gradient by the wavenumber before it reaches the backbone. The multiresolution read-out of Eq. (6) responds to this rescaling, with each level supplying a candidate potential.

F Limitations and Future Work

The construction is tied to a regular grid. Both guarantees rest on the incidence structure of the uniform staggered grid on which the data are stored. Local mass conservation follows because every interior edge appears in two faces of a cell with opposite orientation, so the six face circulations of Eq. (4) cancel in pairs. Wall impermeability follows because the four edges of a boundary face carry differences of a single vertex potential, so their circulation telescopes to zero. Carrying the same parameterization to an unstructured mesh, to a cut-cell discretization of the building surfaces, or to a curved manifold requires a discrete de Rham complex on which both relations continue to hold, together with an assembly operator that identifies the boundary edges of that complex. The first relation is available on simplicial complexes through discrete exterior calculus (Arnold et al., 2010; Desbrun et al., 2005); the second depends on how the solid boundary is represented, and constructing it is open.

The roof-level shear layer is difficult for every surrogate considered. Section 4.2 reports that all four learned methods are least accurate in the layer where the flow separating from the roofs meets the faster air above, and that this layer is also the least steady one. A one-step map trained on a squared-error objective approaches the conditional mean of the next frame, so the component of the flow that fluctuates most is the component it reproduces least. This concerns the training objective and applies to surrogates of this class independently of the output representation. Whether a probabilistic or ensemble objective recovers that component, and at what cost in pointwise error, is left to future work.