

Article

Using Machine Learning to Predict Wind Flow in Urban Areas

Nir BenMoshe ^{1,*} , Eyal Fattal ¹ , Bernd Leitl ²  and Yehuda Arav ¹ 

¹ Department of Applied Mathematics, Israel Institute for Biological Research,
P.O. Box 19, Ness-Ziona 7410001, Israel

² Department of Mathematics, Meteorological Institute, Informatics and Natural Sciences,
University of Hamburg, Bundesstr. 55, 20146 Hamburg, Germany

* Correspondence: nirb@iibr.gov.il

Abstract: Solving the hydrodynamical equations in urban canopies often requires substantial computational resources. This is especially the case when tackling urban wind comfort issues. In this article, a novel and efficient technique for predicting wind velocity is discussed. Reynolds-averaged Navier–Stokes (RANS) simulations of the Michaelstadt wind tunnel experiment and the Tel Aviv center are used to supervise a machine learning function. Using the machine learning function it is possible to observe wind flow patterns in the form of eddies and spirals emerging from street canyons. The flow patterns observed in urban canopies tend to be predominantly localized, as the machine learning algorithms utilized for flow prediction are based on local morphological features.

Keywords: wind; urban; machine learning; CFD; openFOAM

1. Introduction

Calculation of the urban wind field is known to be a computationally demanding task even in the case of a city over a flat terrain. This is mainly due to the need to resolve the obstructed flow affected by the individual buildings; an air parcel can flow around tall buildings on either side or above them [1]. When considering the mean temporal flow within the urban canopy, there are a few typical flow patterns that occur in the urban canopy: (a) street canyon—when the flow passes above buildings and encounters a perpendicular street, it creates an eddy or eddies within the street [2]; (b) channeling—when the flow is parallel to the streets, it can accelerate [3]; (c) junctions—the flow will continue in multiple directions, branching off into streets that are perpendicular to the original flow.

Modeling the flow patterns at all spatial scales using numerical simulation (DNS) requires a very high spatial resolution. This in turn leads to very small time steps and a high computational effort, making it impractical for modeling urban areas. To simplify the model and use larger grid cells, several options are available, such as Large-Eddy Simulation (LES) models, which parameterize the small-scale flow [4], or Reynolds-Averaged Navier–Stokes (RANS) models [5–8] that provide only the temporal mean values of the flow. RANS equations are used to assess wind comfort levels in urban areas [9,10] and urban wind energy potential [11,12]. In this work, a RANS model will also be used, which is computationally less demanding than LES. However, both LES and RANS models still require a significant amount of computational time, ranging from hours to days when utilizing parallel computing. One of the reasons is that the calculation time for these models is long is because they calculate the wind field for the entire area even if we only need the wind speed at pre-selected locations or areas.

Machine learning (ML) is a discipline that enables the acquisition of knowledge through examples [13,14]. The wind field can be modeled using supervised learning, where the machine is trained to identify the relationship between features and the target outcome (e.g., wind speed at specific locations) and then apply that understanding to similar situations. This connection can be established through various machine learning



Citation: BenMoshe, N.; Fattal, E.; Leitl, B.; Arav, Y. Using Machine Learning to Predict Wind Flow in Urban Areas. *Atmosphere* **2023**, *14*, 990. <https://doi.org/10.3390/atmos14060990>

Academic Editors: Serena Falasca and Annalisa Di Bernardino

Received: 18 April 2023

Revised: 1 June 2023

Accepted: 2 June 2023

Published: 7 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

methods, such as Artificial Neural Networks, Support Vector Machines, and k-Nearest Neighbors [13].

Previous studies in atmospheric science utilized machine learning to characterize the flow in simple structures such as a duct, a single rectangular body, or a blade [15–18]. However, to the best of our knowledge, the only study that considered the complexities of urban flows utilized it only above roof tops [19].

This work employs K-nearest neighbors (kNN) [20] as the ML algorithm. kNN is a non-parametric supervised learning algorithm that can be used for both classification and regression tasks. It is a lazy learning algorithm, meaning that it does not build a model during the training phase. Instead, it stores all of the training data and uses it to make predictions at the time of inference. To make a prediction, kNN finds the K most similar training instances to the new instance and then uses the labels of those instances to make a prediction for the new instance. The k nearest neighbors are identified using a distance metric, such as Euclidean distance or Manhattan distance. The predicted value for a new data point is then calculated as the average of the values of the k nearest neighbors. The value of K is a hyperparameter that must be chosen by the user. kNN has several advantages over other machine learning algorithms. First, it is very versatile and can be used for a wide variety of tasks. Second, it is very robust to noise and outliers. Third, it can be used for both classification and regression tasks.

The current study aims to address the computational challenges associated with modeling micro-urban flow patterns. Initially, RANS simulations, which are computationally intensive but provide fairly accurate results, will be performed to gain an understanding of flow patterns in urban areas. Subsequently, machine learning techniques will be utilized to identify the key properties that influence micro-urban flow. Due to the long calculation time of coupled, regional models with CFD models [21], using those properties for machine learning prediction, it is possible to shorten the calculation time in order to obtain downscaled information. Regional models use wind measurements as a data assimilation input to improve their accuracy [22]. ML predictions can assist this process by adjusting the measurements to be less local, by taking into account nearby morphological features. Furthermore, they can also be employed as predictors of the initial conditions for other RANS simulations, thereby reducing the time required for the simulation.

In addition, we examine the utilization of machine learning for the calculation of wind flow in a complex urban area. Given that machine learning requires many learning cases and the number of experiments is limited, we use the results of RANS simulations for the learning stage. We use two test cases. The first test case is based on the simulation of the Michaelstadt experiment conducted in a wind tunnel (WT) with over 2000 measurements [23]. In this case, we compare the results of the machine learning algorithm to both the RANS simulation in the entire domain and the measured results in the WT at the measured locations. The second test case characterizes Tel Aviv city center, a congested city with a variety of buildings, streets, and open spaces. In the second test case, we compare the machine learning algorithm to the RANS simulation.

2. Materials and Methods

2.1. RANS Model

The Navier–Stokes equations, which pertain to the dynamics of an incompressible fluid, can be utilized to analyze micro-urban flow patterns, under the presumption of neutral atmospheric stability. The Reynolds-Averaged Navier–Stokes (RANS) [24] model is employed to solve the mean Navier–Stokes equation by separating the flow into a mean temporal flow, represented by \bar{u} , and fluctuations, represented by u' .

$$u = \bar{u} + u' \quad (1)$$

The basic RANS equations conserve momentum and mass.

$$\overline{u_j} \frac{\partial \overline{u_j}}{\partial x_j} + \frac{\partial \overline{u'_j u'_i}}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \overline{p}}{\partial x_j} + \nu \frac{\partial^2 \overline{u_i}}{\partial x_j \partial x_j} \quad (2)$$

$$\sum_i \frac{\partial u_i}{\partial x_i} = 0 \quad (3)$$

In order to denote the direction of flow, the wind velocity is represented by the subscripts i and $j = 1, 2, 3$, where u_1 and u_2 represent the horizontal wind speed in the ambient flow direction and perpendicular to the ambient flow direction, respectively, while u_3 represents the vertical wind speed. The distance in the i direction is represented by x_i , air density is represented by ρ , pressure is represented by p , and kinematic viscosity is represented by ν . The Reynolds stress tensor $\overline{u'_i u'_j}$ is solved using the turbulence model $k-\omega$ SST closure [25].

The RANS equations were solved using the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) solver [26] integrated in the OpenFOAM library [27]. The top boundary of the domain was assigned slip boundary conditions, and the wind velocity near the ground and buildings was fixed to be zero using a wall function. The inlet wind profile imposed a Neuman boundary condition using a logarithmic profile, and the outlet boundary condition was set to have zero gradient. The simulation's configuration is based on the "wind around buildings" example of the openFOAM package.

To model the flow in urban areas, a digital map of the area is required, which distinguishes between air-permeable regions (e.g., streets) and air-impermeable regions (e.g., buildings). Such a digital map can be obtained from map servers (such as openstreetmap) or through CAD software (such as freeCAD).

The initial step in constructing the computational domain is defining the grid over the calculation area. An irregular mesh is employed to minimize grid size. Specifically, the grid is densely spaced near the ground and buildings where the flow is complex and coarser in other parts of the domain where the flow is less complex. The irregular mesh is constructed by first defining a regular mesh with a horizontal resolution of 10 m and a vertical resolution of 4 m. In proximity to the buildings and ground, the regular mesh is gradually divided into four smaller cells in each direction to increase the resolution. This results in cells near the buildings being 2.5 m wide and 1 m high.

In those experiments, the inlet flow was characterized by a logarithmic velocity profile, which resulted in an average flow speed of 7 m/s at a height of 100 m.

In the following sections, we will explain how the RANS solution serves as an input for the learning phase of the supervised machine learning engine.

2.2. Machine Learning

There are various machine learning models that can be employed to learn the flow patterns in urban areas. In this work, we tried several methods of computational learning; the methods were tested according to their level of accuracy, and the method found to have the highest accuracy is the kNN method [28]. The kNN method can be based on different values of K ; we found that setting the value to 6 gives the appropriate balance between the calculation time and the desired accuracy. We use the k -Nearest Neighbors (kNN) regression model implementation by scikit-learn [29]. kNN is a supervised learning model that is based on a training set. In this work, it will link the morphological features to RANS results. When using kNN to predict the flow, it compares the morphological properties of the target location to those of similar properties in the training set. The features utilized for learning include the distance of the location from the nearest building in four directions, the location height, and the height of nearby buildings. These features were selected as they can be utilized to compute the width-to-height ratio (w/H) between buildings, a commonly

used metric for describing flow in street canyons [30,31]. The actual wind speed will be normalized according to the wind speed above the buildings.

While the prediction process takes fractions of a second, it is important to note that an additional stage involving the development of a machine learning function, or regression function, is necessary. Although this stage may be time-consuming, it only needs to be performed once during the lifetime of a project. An extensive database of accurate predictions is crucial for gaining a better understanding of the wind field; therefore, we rely on WT-verified RANS simulations to obtain an accurate depiction of the “true” wind field. To construct the learning database, RANS simulations can be utilized to generate millions of records and local properties of the location can be used as predictors.

In our study, we utilized only 20% of the cell data for training purposes after transforming the simulation results from an irregular grid to a regular grid. The remaining 80% of the cells were utilized to evaluate the accuracy of the learning process. The training process takes time as a function of the number of cells, and thus only a limited portion of the cells were selected for the training phase. The cells were not randomly chosen as we aimed to avoid using cells with similar feature values for both training and prediction phases. Instead, when we checked the machine learning function for the same city that used for the learning process, we selected the training cells from the city center and the testing cells from the outskirts. Another test case used the Tel Aviv learning function and tested it on the Michaelstadt experiment.

2.3. Test Cases

We use two datasets as test cases to examine the validity of our method. The first dataset is the Michaelstadt WT experiment, which is part of the COST ES1006 project [23], as it represents a typical European city. The WT experiment was conducted at the “WOTAN” facility at Hamburg University. The WT has a neutrally stratified model boundary layer flow generated by turbulence spire generators at the inlet. The measurements were carried out with 2D fiber-optic Laser-Doppler-Anemometry. The geometric scale is 1:250. The wind flow measurements were used for validation of RANS and LES simulation by other groups [32–35]. The second test case is Tel Aviv center morphology (Figure 1), which represents an urban area with varying building shapes and heights.



Figure 1. Part of the simulation domain of central Tel Aviv.

The Michaelstadt morphology consists of buildings that range from 18 to 24 m in height and are 15 m wide. The streets are non-perpendicular and are 20 m wide. The buildings are not rectangles and vary in shape. All have patios on the interior (as depicted in Figure 2). The figure displays the locations of 2156 measurement points in the center of the town on various levels during the experiment. The measuring points were below and above the building's top. The advantage of this test case is that it allows for direct comparison and verification of the machine learning results with both WT measurements and RANS simulation results. When we compared the wind velocity of the machine learning prediction with the RANS simulation, we were also able to compare points near the walls of the building.

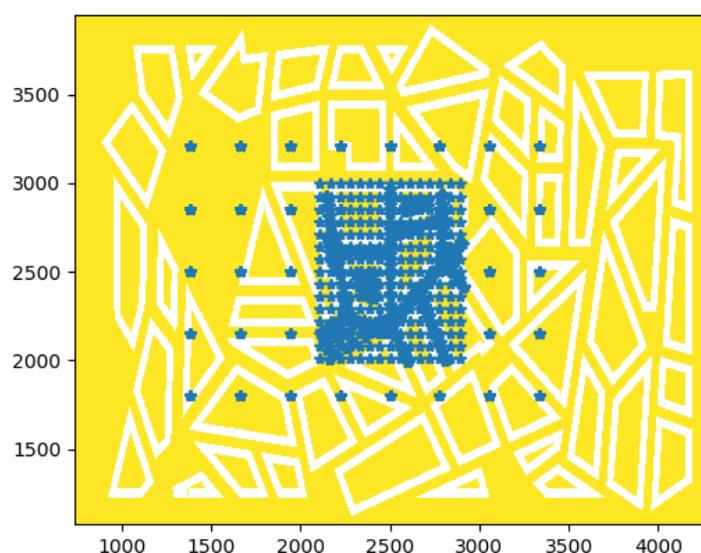


Figure 2. The location of the measuring points at different heights in Michaelstadt wind tunnel experiment.

The Tel Aviv center morphology (Figure 1) aims to test the performance of the kNN regression model in a different urban areas with varying building shapes and heights. The results of this experiment were used to validate the performance of the model (against the WT-verified RANS) and show its generalization capability to other urban areas. The area that was chosen is between the coordinates 32.058 °N 34.757 °E and 32.081 °N 34.794 °E; it has a size of 9 km². The average building height is 12.2 m and the tallest building is 187 m in height. The morphology of the city center contains courtyards and buildings of different sizes, shapes, and directions. The streets are also of different widths, from streets of 1 lane to streets of 8 lanes. The junctions are not always perpendicular.

3. Results

A computational domain consisting of 9.5 million cells was established to model the wind flow in the Michaelstadt experiment. Utilizing a 36-CPU cluster, the solution achieved convergence after 3500 iterations, with a duration of 7.3 h. The convergence criterion was defined as a difference of 10^{-6} between the right-hand side and left-hand side of the Reynolds-Averaged Navier–Stokes equations.

The ML training function was built using an array of sample locations with their morphological properties and their calculated velocity. Using more data gives us better accuracy, as seen in Figure 3. Using more than 80k samples did not change the accuracy by much.

Table 1 presents a comparison between the simulation results of the RANS model (Equations (1) to (3)) and the prediction of the ML model using 80k samples to the wind speed obtained from the wind tunnel (WT) in both horizontal directions (u_1 and u_2). The present study used RANS simulation results that were consistent with LES results from the literature [23] and wind turbine (WT) observations. The similarity between the RANS

simulation and the WT observations can be seen in terms of their mean values and standard deviations. The Canonical Correlation Analysis (CCA) [36], which tests the correlation between vectors that can be correlated (e.g., u_1 and u_2), gives a high score. Additionally, the Pearson correlation coefficient (r) between the RANS simulation and the WT observations is high, and the Root Mean Square Deviation (RMSD) between the simulation and the observation is significantly lower than the average and standard deviation. The factor 2, which compares the ratio of simulation points with values between 1/2 and 2 of the measurement and calculated wind speed, is also high [37]. LES simulations obtained Factor 2 values of 0.667 and 0.94 for the coarse and fine resolutions, respectively, for u_1 and 0.404 and 0.471 for u_2 in their regular grid [23]. In contrast, the present study found Factor 2 values of 0.79 and 0.52 for u_1 and u_2 , respectively, in a RANS simulation with an irregular grid.

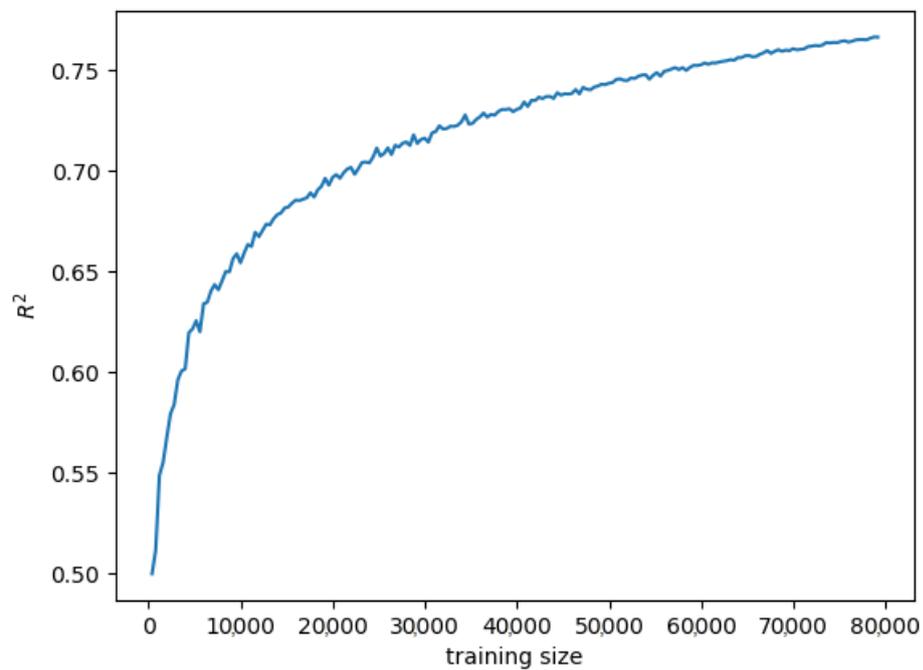
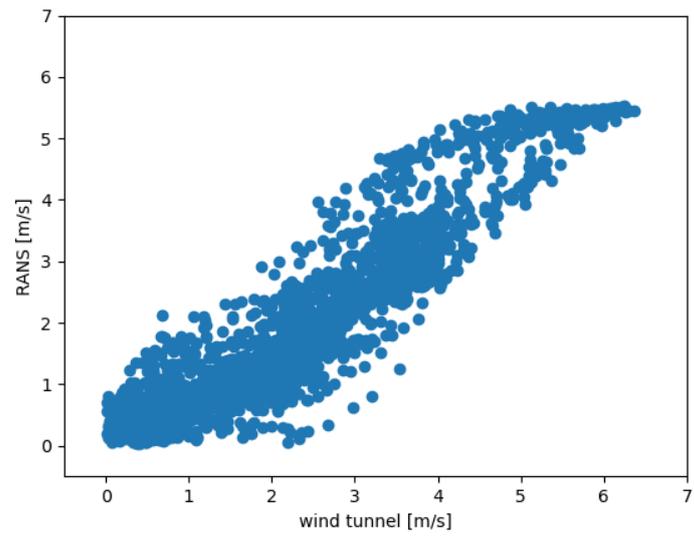


Figure 3. The machine learning function accuracy as a function of the training size for Michaelstadt experiment.

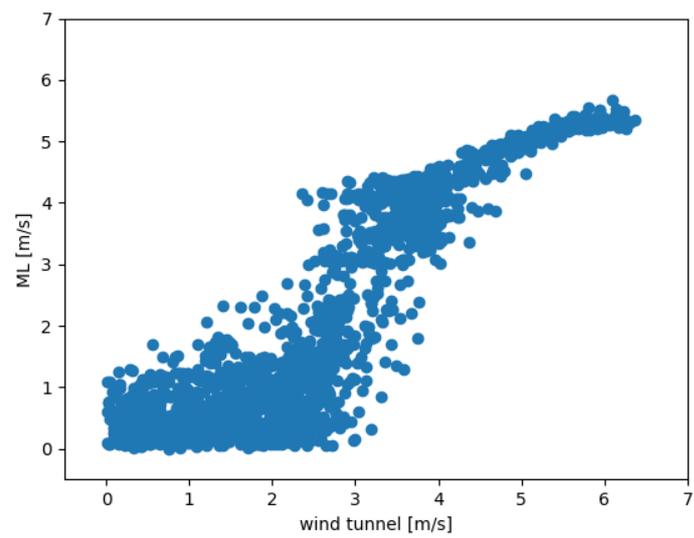
Table 1. Michaelstadt model accuracy.

Index	RANS	ML
CCA	0.96	0.93
r	0.93	0.90
Factor 2	0.84	0.65
RMSD m/s	0.67	0.87
Observation Mean m/s	2.47	2.47
Model Mean m/s	2.11	2.08
Observation STD m/s	1.52	1.52
Model STD m/s	1.52	1.79

Figure 4 depicts the distribution of the different measurement points for the RANS simulation and the ML prediction, respectively. The figures compare the wind speed of the simulation to the WT measurements at each measuring point. It can be seen that both the RANS simulation and the ML prediction exhibit better agreement with the WT measurements for high wind speeds located above the buildings. The RANS simulation has a stronger correlation with the measurements between the buildings as it takes into account the flow upstream. The main flow is also well-captured by the ML prediction, which indicates that it is primarily influenced by local features.

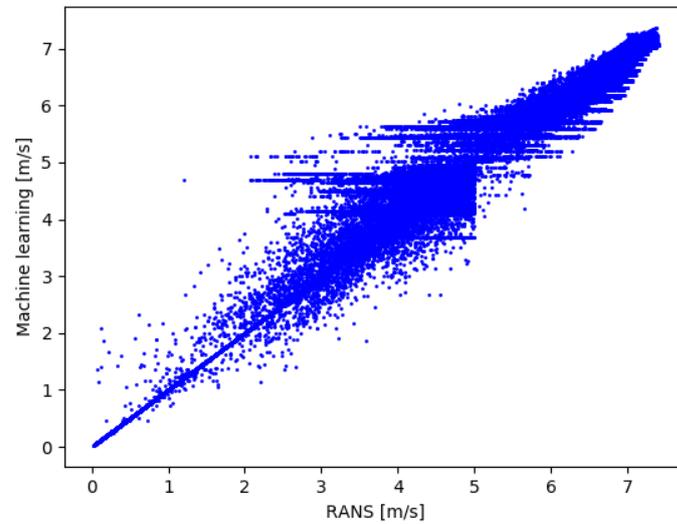


(a)

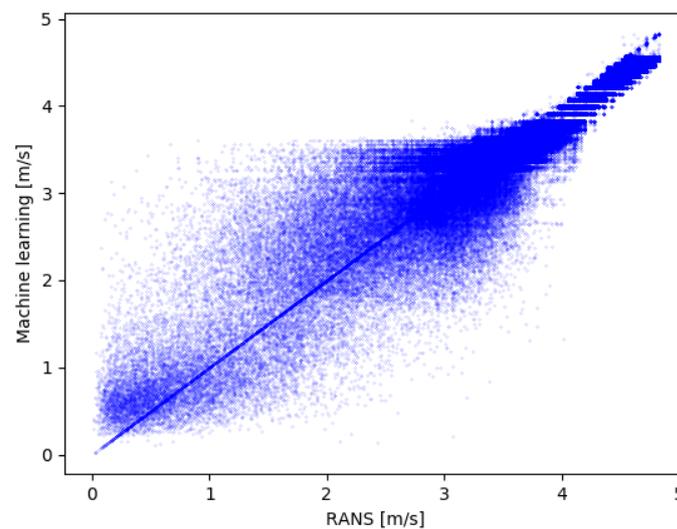


(b)

Figure 4. Comparison between the model wind speed and the wind tunnel observation wind speed. (a) RANS simulation results (b) machine learning results.



(a)



(b)

Figure 5. Comparison between the wind speed of RANS simulation and the machine learning prediction for (a) Michealstadt experiment and (b) Tel Aviv center.

When considering the comparison of the simulation results to the wind tunnel, it should be noticed that the wind tunnel measurements were carried out in the center of the streets. The measurements of the wind tunnel are spot-on, while the results of the simulations represent cells with a width of about one and a half meters; this width can affect the data taken by tens of percent.

The correlation between the RANS velocity results and the ML prediction can be seen in Figure 5 where the upper plot depicts the Michaelstadt experiment and the lower plot displays the Tel Aviv center experiment.

The comparison between the machine learning prediction results and the RANS simulation results is good and even better between the buildings compared to the comparison of the machine learning prediction to the wind tunnel measurements, because in the comparison of the ML prediction to the simulation results, more comparison points are taken into account, including points that are closer to the walls of the building compared to the wind

tunnel measurement points. In the Michelstadt experiment, there is an overestimation of the wind speed in the machine learning prediction; the points of the overestimation are on the outskirts of the city near the open areas.

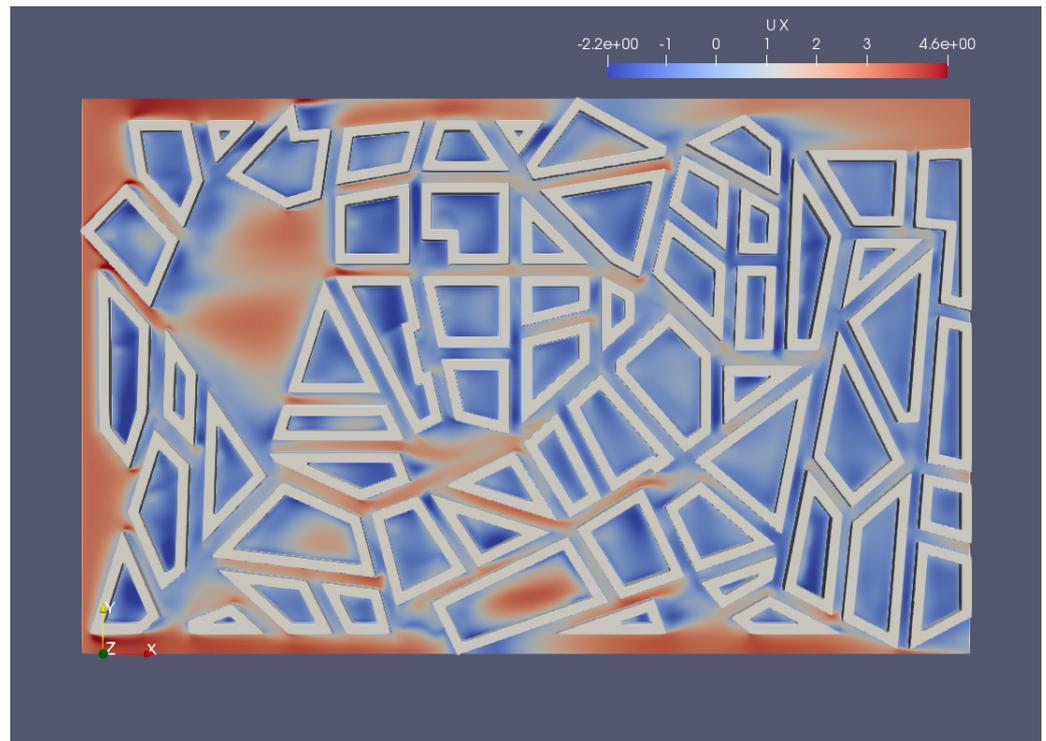
Comparing the wind field at 2 m above the ground shows good qualitative agreement (Figure 6). Both RANS simulation and the ML prediction have low wind speed between the buildings and high wind speed above the canopy. In large courtyards, the wind is higher as well, as seen in the upper left side of the city and within most of the streets that are parallel to the ambient wind direction.

An additional method for verifying the accuracy of the model is to assess its ability to reproduce known flows in micro-urban environments. Figure 7 illustrates a typical street canyon flow in which the flow above the buildings creates an eddy as it encounters them [2]. The upper image (a) shows the result of the RANS simulation, and the lower image (b) depicts the result of the machine learning functions. The length of the arrows is arbitrary. The comparison between the graphs shows us that both the RANS simulation and the machine learning results give high speeds above the roofs of the houses and low speeds between the buildings. In the three eddies shown in the figures, you can see that the wind strength is weaker in the center of the vortex and higher as you move away from the center of the vortex. The arrows in the left vortex and the middle vortex are similar between the RANS simulation and the machine learning prediction. The right vortex has a little less agreement between the simulation and the prediction but it still gives key features of the flow and maintains the flow shape.

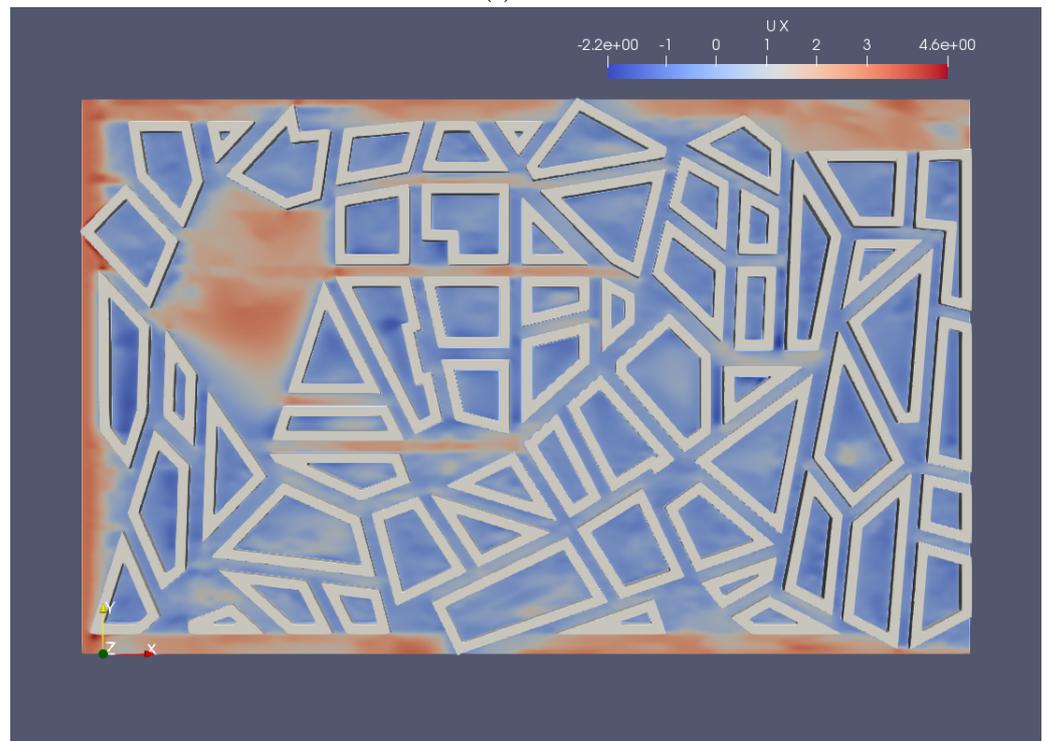
The wind direction above the buildings is the ambient flow direction. Below the rooftop, the wind direction is determined by the urban morphology. When an air parcel goes in the direction of a wall, it will change its direction according to the angle between the air parcel and the wall (e.g., it will create vortices, as seen in Figure 7). Thus, the wind direction is determined mainly by a few local features. We can see in the upper plot of Figure 8 that the horizontal wind direction predicted by the ML function has very good agreement with the RANS simulation results. The main deviation is near the ambient flow direction (0 degrees) and it is because of the slowing down of the wind speed in the U direction due to the vertical wind (W).

In Figures 4–8, the machine learning function was built using one urban neighborhood. This in turn was used for the calculation of the other neighborhood. A better validation would be a study on one city and using the machine learning function on another city. In our opinion, it can be accomplished when there is a wide enough variety for learning samples, as in our case. Figure 9 shows the correlation of the prediction of the wind speed in Tel Aviv using the Michaelstadt machine learning function. The graph shows a reasonable match between the simulation and the prediction of the machine learning. An underestimation can be seen above the rooftops, so more features related to this should be added, such as the distance from the beginning of the house.

Furthermore, additional machine learning functions can be employed to predict other important quantities such as the turbulence kinetic energy (TKE) or the pressure field. Figure 10 illustrates the correlation between the predicted and observed TKE values, demonstrating the effectiveness of the machine learning approach in capturing the turbulent characteristics of the flow.

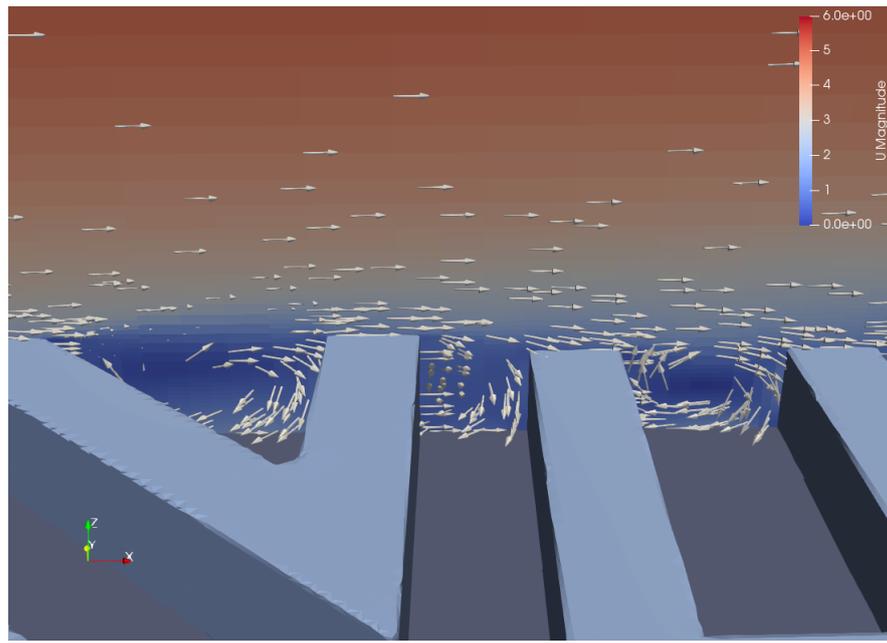


(a)

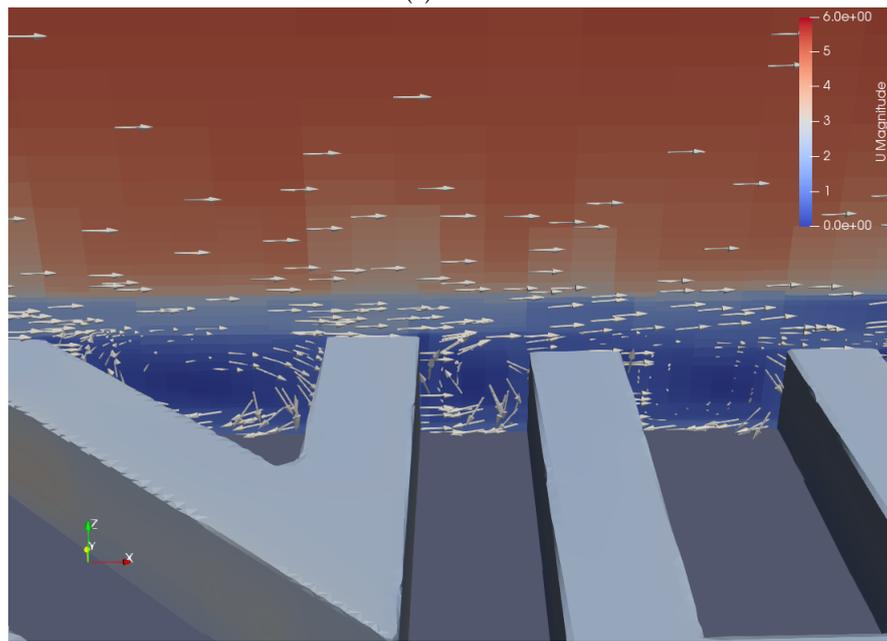


(b)

Figure 6. Comparison between the wind field at 2 m above the ground of (a) RANS simulation and (b) the machine learning prediction for Michealstadt experiment.

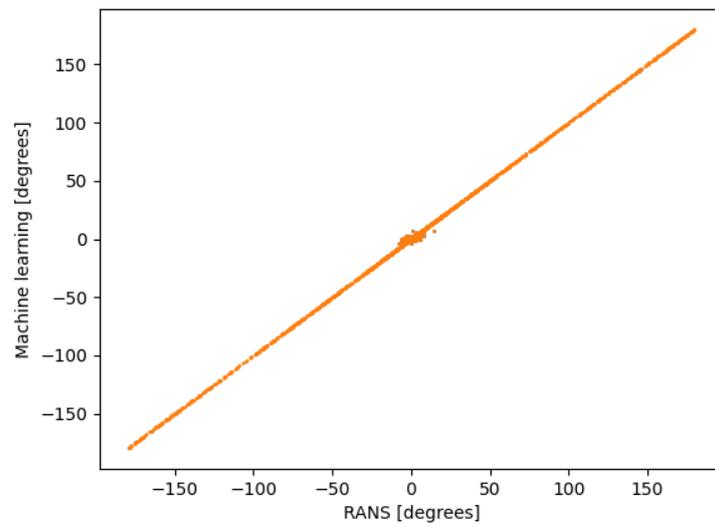


(a)

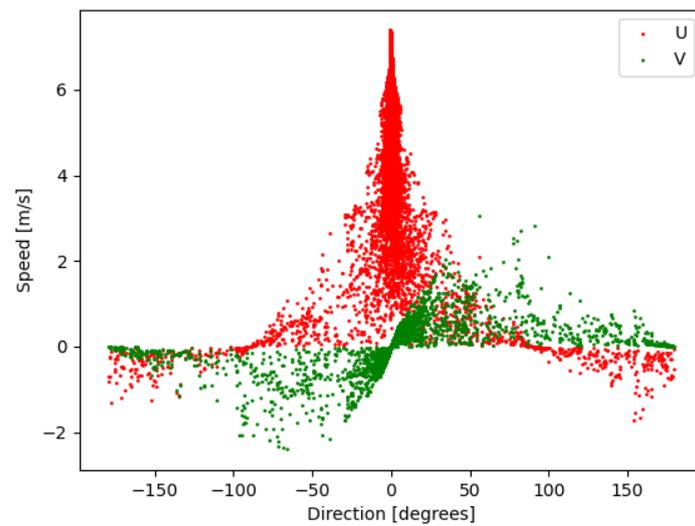


(b)

Figure 7. Flow pattern simulation of an eddy that is formed in a street canyon using (a) RANS equations (b) machine learning.



(a)



(b)

Figure 8. The ML prediction of the horizontal wind direction vs. the wind direction of RANS simulation in the upper plot (a). The U (red dots) and V (green dots) components of the wind direction from the RANS simulation (b).

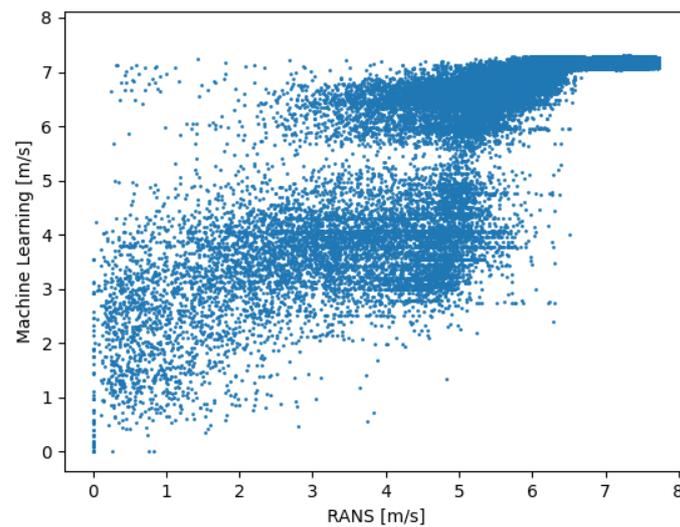


Figure 9. Comparison between the ML prediction of Tel Aviv (using Michaelstadt ML function) to the Tel Aviv RANS simulation results.

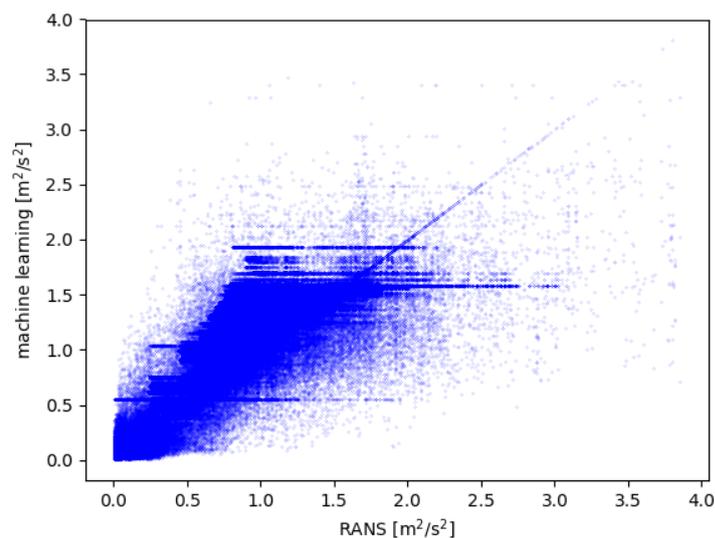


Figure 10. Comparison between the TKE of RANS simulation and the machine learning prediction for the Michealstadt experiment.

The resolution of the model determines the scope of the phenomena that the model can describe. The present study employs an irregular grid with a high resolution near the buildings and the ground, allowing for a more detailed representation of the complex flow between the buildings. To reduce computational resources, a coarser resolution is used in areas of less turbulence, high above the ground. To account for the varying resolutions, the total number of cells in the simulation domain is used for comparison, rather than the resolution. As seen in Table 2, a comparison of Michaelstadt simulation results with wind tunnel measurements shows similar accuracy (RMSD of 0.494 and 0.472 for low and high cell count, respectively), despite a significant difference in simulation duration (0.1 and 7.3 h for low and high cell count, respectively). In this case, the similarity in measurement accuracy was due to the fact that the measurement points were situated in the center of the streets. However, there was a noticeable difference in flow on the sides of the street near the buildings. The transition between the maximum speed in the center of the streets that are parallel to the direction of flow to the zero speed near the walls of the buildings is more gradual in the simulation with the larger number of cells. Furthermore, in the streets perpendicular to the direction of the flow, the resulting vortices are less accurate

in the simulation with fewer cells. The required number of cells in the simulation depends on the specific research question. For the use of machine learning functions, a high number of cells in the simulation is necessary for the learning process to capture the wind speed near the buildings and predict the wind velocity in these areas accurately.

Table 2. The model accuracy for different numbers of cells.

	Coarse	Medium	Fine
number of cells	0.5 M	2.5 M	9.5 M
simulation duration [h]	0.1	0.9	7.3
RMSD (vs. Observation)	0.494	0.483	0.472

Three sets of features were tested. In the first set, we choose fewer features than in the second and third sets. We calculated the weight of each feature for each set. Table 3 contains the full weight list. The weight of some features is neglected because it is already allocated to other features. For example, the distance from the building in the right to the u_2 direction in set II is a superposition of the distance from the building on the left and the street width. We can determine if the added features are significant by adding more features and checking their weight and the change in the prediction accuracy.

Table 3 reveals that the primary factors influencing wind flow are the separation from the ground level, the angle between the street and the ambient wind direction, and the proximity to neighboring buildings. These factors hold considerable significance when examining wind flow that is perpendicular to the street, as they determine vortex properties. This relationship is closely related to the width-to-height (W/H) ratio, which is well-established for urban street canyons [30,31]. The W factor denotes the sum of the distance from the neighboring buildings. Additionally, two other crucial factors are the distances from the adjacent buildings on either side, which carry significant importance when investigating wind flow parallel to the street.

Table 3. The flow property weights.

Feature	Set I	Set II	Set III
Distance from ground	1.90	2.01	2.66
mean height (including buildings)	-	0	0
distance from next building in u_1 direction	1.047	0.873	1.61
distance from previous building in u_1 direction	1.961	1.946	1.77
street width in u_1 direction	-	0.187	0
street width in u_2 direction	-	0.709	0
distance from building in left in u_2 direction	0.589	0.259	1.51
distance from building in right in u_2 direction	0.413	0	0.98
height of the previous building in u_1 direction	0.321	0.097	-0.01
height of the following building in u_1 direction	0.176	0.053	0.01
street angle	-	-	-2.54
ratio between the distance from the previous building and the previous building height	-	-	-0.02
ratio between the distance from the following building and the following building height	-	-	0.05

In the previous section, we discussed the utilization of the RANS equations to describe the flow behavior. Despite the simplicity of these equations and the agreement they provide between the model and the wind tunnel observations (as evidenced in Table 1), solving these equations requires a substantial number of iterations until convergence is reached. An effective approach to reducing the number of required iterations is to use good initial conditions. The machine learning engine we employed in this section allows us to predict the velocity field within the domain and use it as an initial condition for solving the RANS equations. Figure 11 demonstrates the convergence of the standard RANS simulation

and the machine learning-based RANS simulation by comparing the root mean square deviation of the velocity field between the simulation's iteration before convergence and the last iteration. It can be seen that simulations based on machine learning-generated initial conditions converge faster than those without. In cases where full convergence of the simulation is not necessary, the machine learning option can provide reasonable results.

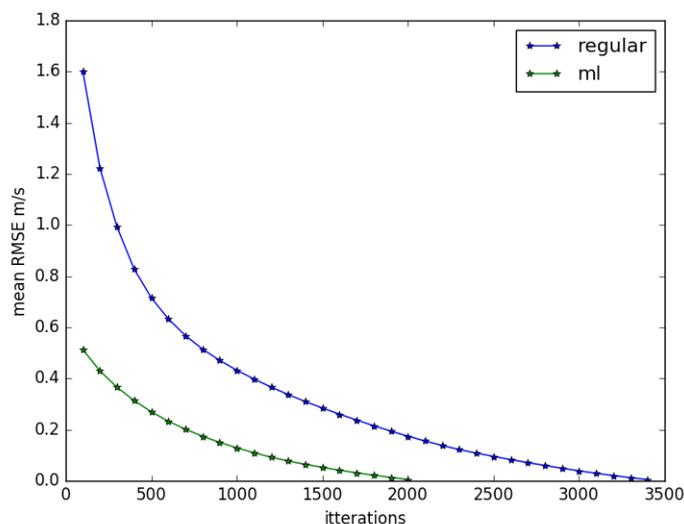


Figure 11. Convergence of machine learning-based simulation (green) and “regular” RANS simulation (blue).

4. Discussion and Conclusions

In this study, a solution to the flow patterns in urban areas through the use of machine learning functions that are based on the Reynolds-Averaged Navier–Stokes solution was presented. The proposed method shows good agreement with wind tunnel observations above and below the tops of building. Predicting the wind field using machine learning was able to reproduce complex phenomena such as the creation of vortices between buildings. We tested the set of equations using both uniform initial conditions and machine learning-based initial conditions for two case studies: the Michaelstadt wind tunnel experiment and the Tel Aviv center. The latter approach resulted in a reduced convergence rate by 50%. The machine learning engine utilized only local features, such as the surrounding building characteristics and cell height, indicating that the flow patterns are primarily influenced by the local structures. To describe the local flow, we selected a few crucial features, but further investigation may be necessary to identify additional important features, which could be explored through the use of machine learning. Other factors, such as distance from previous buildings that are higher than the closest buildings, may also be considered. The machine learning engine can serve as a downscaling tool, where a numerical weather prediction model (e.g., WRF) with a coarse resolution can be complemented by fine-resolution data obtained from the machine learning prediction tool. Different wind speeds and directions in different parts of a large city can add complexity to the wind. We show in this paper that the wind patterns between the buildings are mostly local. In order to predict the wind field, one has to calculate the wind field using a larger-scale model and then downscale the results using ML. Although the learning process was carried out using one wind direction, the variety in street directions and house sizes allows the use of this learning function for different ambient wind directions.

The utilization of machine learning is a less complex alternative to conducting a Reynolds-averaged Navier–Stokes (RANS) simulation. The reason is that, for the latter, comprehensive data on all the structures in the vicinity are necessary, including their geometries, heights, orientations, and relative distances from one another. Conversely, after establishing a machine learning algorithm for a comparable urban setting, the only

pertinent variables needed for subsequent predictions are the distance of the target location from its neighboring buildings, the altitude of the target point, and the average wind velocity across the urban area.

This paper describes the use of computational learning for urban flow in flat terrains; although most cities in the world are built in flat areas, attention should also be given to cities built over complex topography. In future work, such a flow will be described. When computing urban flow in a city over complex topography, additional features such as the height difference of the ground in different directions should be added. Another scenario that will be examined in future works is the effect of temperature field on flow. In this case, the RANS equations will be augmented with the temperature field to be used for the learning process, and the temperature gradient with height will also be added to the list of features.

Author Contributions: Conceptualization, N.B., E.F. and Y.A.; methodology, N.B., E.F. and Y.A.; software, N.B. and Y.A.; validation, N.B., E.F. and Y.A.; formal analysis, N.B.; investigation, N.B., E.F. and Y.A.; resources, N.B., E.F., B.L. and Y.A.; data curation, N.B., B.L. and Y.A.; writing—original draft preparation, N.B.; writing—review and editing, N.B., E.F., B.L. and Y.A.; visualization, N.B.; supervision, N.B., E.F. and Y.A.; project administration, E.F.; funding acquisition, E.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank David Avissar for fruitful discussions.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

kNN	k Nearest Neighbors
ML	Machine learning
RANS	Reynolds-Averaged Navier–Stokes
WT	Wind Tunnel

References

1. Hanna, S.R.; Brown, M.J.; Camelli, F.E.; Chan, S.T.; Coirier, W.J.; Hansen, O.R.; Huber, A.H.; Kim, S.; Reynolds, R.M. Detailed simulations of atmospheric flow and dispersion in downtown Manhattan: An application of five computational fluid dynamics models. *Bull. Am. Meteorol. Soc.* **2006**, *87*, 1713–1726. [[CrossRef](#)]
2. Gromke, C.; Buccolieri, R.; Di Sabatino, S.; Ruck, B. Dispersion study in a street canyon with tree planting by means of wind tunnel and numerical investigations—Evaluation of CFD data with experimental data. *Atmos. Environ.* **2008**, *42*, 8640–8650. [[CrossRef](#)]
3. Spirn, A. Better Air Quality at Street Level: Strategies for Urban Design. In *Public Streets for Public Use*; Van Nostrand Reinhold: New York, NY, USA, 1987.
4. Cheng, W.C.; Yang, Y. Scaling of Flows Over Realistic Urban Geometries: A Large-Eddy Simulation Study. *Bound. Layer Meteorol.* **2022**, *186*, 125–144. [[CrossRef](#)]
5. Blocken, B.; Persoon, J. Pedestrian wind comfort around a large football stadium in an urban environment: CFD simulation, validation and application of the new Dutch wind nuisance standard. *J. Wind. Eng. Ind. Aerodyn.* **2009**, *97*, 255–270. [[CrossRef](#)]
6. Moonen, P.; Defraeye, T.; Dorer, V.; Blocken, B.; Carmeliet, J. Urban Physics: Effect of the micro-climate on comfort, health and energy demand. *Front. Archit. Res.* **2012**, *1*, 197–228. [[CrossRef](#)]
7. Neophytou, M.; Gowardhan, A.; Brown, M. An inter-comparison of three urban wind models using Oklahoma City Joint Urban 2003 wind field measurements. *J. Wind. Eng. Ind. Aerodyn.* **2011**, *99*, 357–368. [[CrossRef](#)]
8. Tominaga, Y.; Stathopoulos, T. Numerical simulation of dispersion around an isolated cubic building: Model evaluation of RANS and LES. *Build. Environ.* **2010**, *45*, 2231–2239. [[CrossRef](#)]
9. Blocken, B.; Stathopoulos, T.; Van Beeck, J. Pedestrian-level wind conditions around buildings: Review of wind-tunnel and CFD techniques and their accuracy for wind comfort assessment. *Build. Environ.* **2016**, *100*, 50–81. [[CrossRef](#)]
10. Nimarshana, P.; Attalage, R.; Perera, K.K.C. Quantification of the impact of RANS turbulence models on airflow distribution in horizontal planes of a generic building under cross-ventilation for prediction of indoor thermal comfort. *J. Build. Eng.* **2022**, *52*, 104409. [[CrossRef](#)]

11. Anup, K.; Whale, J.; Urmee, T. Urban wind conditions and small wind turbines in the built environment: A review. *Renew. Energy* **2019**, *131*, 268–283.
12. Toja-Silva, F.; Kono, T.; Peralta, C.; Lopez-Garcia, O.; Chen, J. A review of computational fluid dynamics (CFD) simulations of the wind flow around buildings for urban wind energy exploitation. *J. Wind. Eng. Ind. Aerodyn.* **2018**, *180*, 66–87. [[CrossRef](#)]
13. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. [[CrossRef](#)]
14. Mahesh, B. Machine learning algorithms—A review. *Int. J. Sci. Res.* **2020**, *9*, 381–386.
15. Zhao, Y.; Akolekar, H.D.; Weatheritt, J.; Michelassi, V.; Sandberg, R.D. RANS turbulence model development using CFD-driven machine learning. *J. Comput. Phys.* **2020**, *411*, 109413. [[CrossRef](#)]
16. Wu, J.L.; Xiao, H.; Paterson, E. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Phys. Rev. Fluids* **2018**, *7*, 1–42. [[CrossRef](#)]
17. Maulik, R.; Sharma, H.; Patel, S.; Lusch, B.; Jennings, E. Accelerating RANS Turbulence Modeling Using Potential Flow and Machine Learning. *arXiv* **2019**, arXiv:1910.10878.
18. Vuppala, R.K.; Kara, K. A non-intrusive reduced order model using deep learning for realistic wind data generation for small unmanned aerial systems in urban spaces. *AIP Adv.* **2022**, *12*, 085020. [[CrossRef](#)]
19. Mortezaadeh, M.; Zou, J.; Hosseini, M.; Yang, S.; Wang, L. Estimating Urban Wind Speeds and Wind Power Potentials Based on Machine Learning with City Fast Fluid Dynamics Training Data. *Atmosphere* **2022**, *13*, 214. [[CrossRef](#)]
20. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **1992**, *46*, 175–185. [[CrossRef](#)]
21. Chrit, M.; Majdi, M. Improving Wind Speed Forecasting for Urban Air Mobility Using Coupled Simulations. *Adv. Meteorol.* **2022**, *2022*, 2629432. [[CrossRef](#)]
22. Chrit, M. Reconstructing urban wind flows for urban air mobility using reduced-order data assimilation. *Theor. Appl. Mech. Lett.* **2023**, *13*, 100451. [[CrossRef](#)]
23. Baumann-Stanzer, K.; Andronopoulos, S.; Armand, P.; Berbekar, E.; Efthimiou, G.; Fuka, V.; Gariazzo, C.; Gasparac, G.; Harms, F.; Hellsten, A.; et al. COST ES1006 Model Evaluation. 2015. Available online: https://www.researchgate.net/publication/275952924_COST_ES1006_Model_evaluation_case_studies_Approach_and_results (accessed on 17 January 2023).
24. Reynolds, O. On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion. *Philos. Trans. R. Soc. Lond.* **1895**, *186*, 123–164.
25. Menter, F.R. *Improved Two-Equation k - Ω Turbulence Models for Aerodynamic Flows—NASA Technical Memorandum TM-103975*; NASA Technical Memorandum NASA; NASA Ames Research Center: Moffett Field, CA, USA, 1992; pp. 1–31.
26. Caretto, L.S.; Gosman, A.D.; Patankar, S.V.; Spalding, D.B. Two Calculation Procedures for Steady, Three-Dimensional Flows with Recirculation. In *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*; Springer: Cham, Switzerland, 2004; pp. 60–68. [[CrossRef](#)]
27. Churchfield, M.J.; Moriarty, P.J.; Vijayakumar, G.; Basseur, J.G. Wind Energy-Related Atmospheric Boundary Layer Large-Eddy Simulation Using OpenFOAM. In *Proceedings of the 19th Symposium on Boundary Layers and Turbulence*, Keystone, CO, USA, 2–6 August 2010; pp. 1–26.
28. Soomro, A.A.; Mokhtar, A.A.; Salilew, W.M.; Abdul Karim, Z.A.; Abbasi, A.; Lashari, N.; Jameel, S.M. Machine Learning Approach to Predict the Performance of a Stratified Thermal Energy Storage Tank at a District Cooling Plant Using Sensor Data. *Sensors* **2022**, *22*, 7687. [[CrossRef](#)]
29. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
30. Oke, T.R. Street design and urban canopy layer climate. *Energy Build.* **1988**, *11*, 103–113. [[CrossRef](#)]
31. Vardoulakis, S.; Fisher, B.E.; Pericleous, K.; Gonzalez-Flesca, N. Modelling air quality in street canyons: A review. *Atmos. Environ.* **2003**, *37*, 155–182. [[CrossRef](#)]
32. Rakai, A.; Franke, J. Validation of two RANS solvers with flow data of the flat roof Michelstadt case. *Urban Clim.* **2014**, *10*, 758–768. [[CrossRef](#)]
33. Efthimiou, G.C.; Bartzis, J.G.; Berbekar, E.; Hertwig, D.; Harms, F.; Leitl, B. Modelling short-term maximum individual exposure from airborne hazardous releases in urban environments. Part II: Validation of a deterministic model with wind tunnel experimental data. *Toxics* **2015**, *3*, 259–267. [[CrossRef](#)]
34. Efthimiou, G.C.; Kovalets, I.V.; Argyropoulos, C.D.; Venetsanos, A.; Andronopoulos, S.; Kakosimos, K.E. Evaluation of an inverse modelling methodology for the prediction of a stationary point pollutant source in complex urban environments. *Build. Environ.* **2018**, *143*, 107–119. [[CrossRef](#)]
35. Castelli, S.T.; Armand, P.; Tinarelli, G.; Duchenne, C.; Nibart, M. Validation of a Lagrangian particle dispersion model with wind tunnel and field experiments in urban environment. *Atmos. Environ.* **2018**, *193*, 273–289. [[CrossRef](#)]
36. Knapp, T.R. Canonical correlation analysis: A general parametric significance-testing system. *Psychol. Bull.* **1978**, *85*, 410–416. [[CrossRef](#)]
37. Chang, J.C.; Hanna, S.R. Air quality model performance evaluation. *Meteorol. Atmos. Phys.* **2004**, *87*, 167–196. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.