Expert Systems With Applications 203 (2022) 117294

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Deep learning methods evaluation to predict air quality based on Computational Fluid Dynamics

Xavier Jurado [a,b,c,*], Nicolas Reiminger [c], Marouane Benmoussa [c], José Vazquez [b], Cédric Wemmert [a]

[a] ICube Data mining, 300 Bd Sébastien Brant, Illkirch-Graffenstaden, France
[b] ICube Fluid mechanics, 2 Rue Boussingault, Strasbourg, France
[c] Air&D, 32 rue Whimpheling, Strasbourg, France

ARTICLE INFO

Keywords:
Deep learning
Convolutional neural network
Computational Fluid Dynamics
Air quality

ABSTRACT

Air quality is a major health issue for densified cities nowadays. To evaluate and act upon it, modeling alongside sensors has proved to be a powerful tool. Among the different available models, Computational Fluid Dynamics (CFD) has proved to be formidable to evaluate airborne pollutant dispersion locally in urban areas since it is able to consider buildings and others complexes phenomenon at the scale of the meter. Nevertheless, this method has a major drawback, it is computationally expensive and cannot be applied in real time or over large areas. To overcome this issue, several state-of-the-art deep learning methods to treat spatial information have been trained based on CFD results to predict airborne pollutant dispersion. Among these models, multiResUnet architecture was proved to be the best on overall over seven metrics. It managed to have two out of three air quality metrics within satisfactory range for a good air quality model. These results are obtained in a mere matter of tens of seconds against several hours for CFD.

1. Introduction

Atmospheric pollution represents millions of deaths each year and is one of the major health issues according to the World Health Organisation (WHO) since 91% of people lives in areas exceeding the WHO threshold standards (WHO, 2018). Indeed, airborne pollution can cause several mortal diseases (Calderón-Garcidueñas et al., 2020; Yang et al., 2020). It is also detrimental to the environment, causing acid rains (Zhang et al., 2017) or impacting agricultural yield (Wang et al., 2020). To tackle this issue, regulation has been implemented in Europe through the European Directive in 2008 (EU, 2008). The regulation is based on annual average as well as hourly concentrations that should not be exceeded. To ensure that these standards are respected and to protect health of residents, several tools exist to assess the pollution in an area. These tools can span continents (Blocken et al., 2015) to urban neighborhoods. For local pollution at the scale of the neighborhood, one can either use sensors (Jurado et al., 2020), but they are expensive and only provide very local information, or numerical models based on physical phenomena (Reiminger et al., 2020). A popular approach for local pollution assessment is to simulate its dispersion with Computational Fluid Dynamics (CFD), but this requires a lot of computing resources (Jurado et al., 2021). It is therefore adapted to compute mean annual average but is not ideal for large areas or use in real time. On the other hand, to cover large areas in real time, some models like plume exist. Unfortunately, they are based on hypothesis that make them unsuited for urban areas where the air pollution is the most stringent (Kumar et al., 2015). The recent advances in machine learning and deep learning may provide the answers to these limitations. Indeed, this field have much progressed over the recent years especially thanks to the improvement and democratization of highly threaded parallel computing processors (Boyer & Baz, 2013). Recently, it has proved to outperform previous state-of-the-art methods in various fields such as speech recognition, visual object recognition, object detection and many other domains such as drug discovery or genomics (LeCun et al., 2015). These new methods have not gone unnoticed in the domain of physics and numerical simulation. Their use are still nascent in these domains. For example, deep learning models were trained to perform numerical simulation faster than traditional methods as in Guo et al. (2016), Junfeng Chen (2019) and Prieler et al. (2018). Deep learning has also been used in the domain of air quality to estimate the pollution based on pictures (Chen et al., 2019), to predict the pike of pollution harmful to health (Morabito & Versaci, 2003), concentration

* Correspondence to: Air&D, 32 rue Whimpheling, F-67000 Strasbourg, France.
E-mail addresses: xjurado@air-d.fr (X. Jurado), nreiminger@air-d.fr (N. Reiminger), benmoussa.marouanee@gmail.com (M. Benmoussa), jose.vazquez@engees.eu (J. Vazquez), wemmert@unistra.fr (C. Wemmert).

https://doi.org/10.1016/j.eswa.2022.117294
Received 3 August 2021; Received in revised form 20 January 2022; Accepted 22 April 2022
Available online 10 May 2022

values from sensors data (Du et al., 2021), to forecast air quality indicators (Kurt & Oktay, 2010; Yan et al., 2021), to extract the main features explaining the pollution variation (Qi et al., 2018).

To build a fast and accurate system able to predict air pollution in real time based on wind, traffic and buildings geometry, we use a convolutional network (CNN), that has proven to be able to treat spatial information successfully, to learn pollutant dispersion from CFD. This will overcome the issue of speed related to standard CFD computation while proposing a model that is more appropriate to urban areas. In this paper, 6 CNN models (namely UNet, SegNet, linkNet, MultiResUnet, PSPNet and FCN) are trained and tested, based on 5000 CFD examples. The aim of the paper is to verify the capability of such models to determine pollutant dispersion rapidly and accurately, and which of these well known CNN architecture performs better to solve this problem.

2. Material and methods

2.1. Physical numerical model

To learn pollutant dispersion in open urban areas, deep learning architectures need examples to be trained on. To simulate wind flow and underlying pollutant dispersion, a popular technique is to use CFD as in Blocken (2014), Murakami (1998) and Reiminger et al. (2020). To perform simulations, Openfoam 5.0 was used. OpenFoam[1] is an open source software dedicated to numerical simulations, ranging from financial to radiation to fluids mechanics. The hypothesis for the simulation were the following:

- Reynolds Averaged Navier Stokes (RANS) approach was used;
- unsteady simulations were performed;
- the turbulence model for the RANS model is k-epsilon renormalization group (RNG) proposed by Yakhot et al. (1992);
- a transport equation for the pollutant dispersion;
- upper and lateral boundaries are symmetry conditions;
- the outlet is a freestream condition;
- buildings have no slip conditions;
- the atmosphere is considered neutral, therefore using a logarithmic inlet profile and turbulence for k and epsilon parameter calculated as proposed in (Richards & Norris, 2011):

$$U = \frac{u_*}{\kappa_{k-\epsilon}} ln \frac{z_0 + z}{z_0} \tag{1}$$

$$\epsilon = \frac{u_*^2}{\sqrt{C_\mu}} \tag{2}$$

$$k = \frac{u_*^3}{\kappa_{k-\epsilon} z} \tag{3}$$

where, U is the inlet speed [m s^{-1}], ϵ is the turbulent dissipation rate [kg m^{-1} s^{-4}], k is the turbulent kinetic energy [kg m^{-1} s^{-3}], u_* is the shear velocity [m/s], $\kappa_{k-\epsilon}$ is the von Kármán constant, z_0 is the roughness length [m] and z is the altitude [m].

Guidelines provided by Franke et al. (2007) were respected when constructing the domain and the meshes of every simulation. For each simulation, the top of the domain is situated at a minimum distance of $5 \times H$ from highest building and the lateral, inlet and outlet boundaries at a minimum distance of $5 \times H$ from the closest building, with H the height of the tallest building in the domain. A mesh sensitivity analysis was made and a mesh with 0.5 m for the cell closest to the building were found to be enough to be insensitive. An example of a neighborhood of the meshing is shown on Fig. 1.

[1] https://www.openfoam.org/.

For more details on the model, equations and validation, please refer to Reiminger et al. (2020) where the same approach has been described and properly validated.

The approach, model and meshes described above have been found to be able to reach an error which is less than 10% compared to experimental measures as show in Reiminger et al. (2020) and a similar approach has been proven to have an overall error of about 30% compared to a real *in situ* situation in urban areas (Rivas et al., 2019). The numerical results will be considered as the ground truth for the deep learning algorithms.

For the sake of simplicity the wind will always come down from the y axis. Around 5000 examples of couples of building layouts and pollutant sources have been computed to be used for the deep learning training and validation.

2.2. Deep learning architectures

Deep learning architectures have shown to be very effective to tackle spatial information in various domain such as predicting urban traffic (Pan et al., 2020; Tedjopurnomo et al., 2020) or to forecast crop yield (Gavahi et al., 2021). Furthermore, convolutional architectures have shown to be particularly effective to deal with spatial information. Indeed, for semantic segmentation, CNNs have proven to be able to overcome issues that were not achievable before in a lot of different fields. For example, it has been used in the medical field to identify certain cell types as in Das and Meher (2021), for small object detection (Liu et al., 2021), predict ozone concentration 24 h in advance as in Sayeed et al. (2020) or remote sensing images analysis (Wagner et al., 2020).

The strength of CNNs to treat spatial information has also started to be used to predict physical phenomena as in Guo et al. (2016) and Junfeng Chen (2019). To simulate physical phenomena, such as fluid mechanics, it is common to define a set of fundamental equations describing the phenomena and then, if needed, to implement a numerical code that will solve them step by step, until reaching convergence (or pseudo convergence) or during the transient wanted time. These steps generally require vast computing time resources.

Deep learning has already been used in fluid mechanics, especially to determine the velocity vector field in Guo et al. (2016) and Junfeng Chen (2019).

Other recent strategies have been used that impose physical constrain such as the continuity equation to the network under the name of "physical informed neural network" (Raissi et al., 2019). This type of network has shown to be able to reproduce CFD results on classical simple case. The Deepmind team has also proposed very promising results with another physical approach using graphical network (Sanchez-Gonzalez et al., 2020). Nevertheless, these two interesting methods are still nascent approaches, are applied to integrally solve CFD and for the moment, can only be used to mimic classical simple case. These approaches are very interesting but are still limited in the real flow they can solve. Here, we have the ambition to go further and study the ability of such architectures to build a model able to determine pollution dispersion given buildings' geometry, wind and traffic information. The previous methods are still not mature enough to be able to solve air pollution dispersion over wide areas in a real urban setting with complex and heterogeneous shapes of building, because of underlying complex flow. Thus, to solve this we propose to use state-of-the-art models that can handle complex spatial problem. We tested different CNN's architectures designed for another spatial issue, image segmentation tasks. These architectures were applied to pollutant dispersion and their results compared.

The first architectures used are encoder–decoder, with, chronologically, Unet (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), linkNet (Chaurasia & Culurciello, 2017) and multiResUnet (Ibtehaz & Rahman, 2020). They follow the same principle of encoding the information to get the context and then decoding it to get the precise

Fig. 1. Example of the meshing on a building layout used to create the examples.

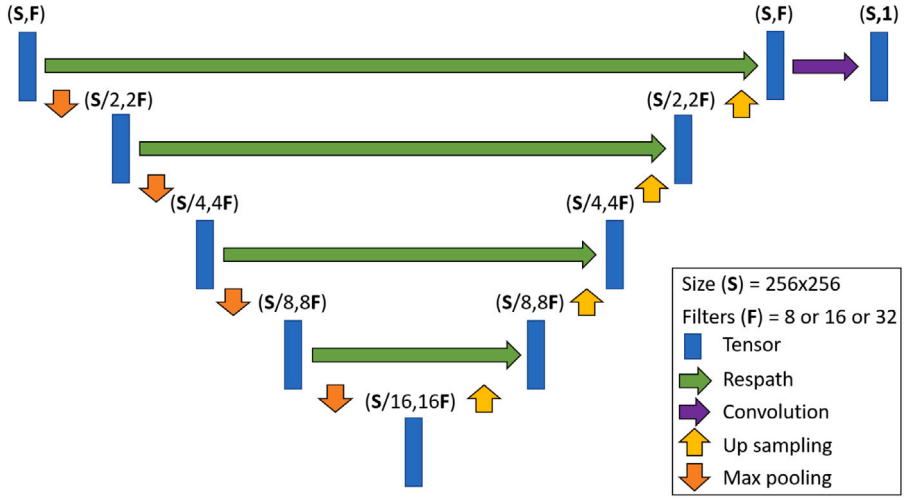


Fig. 2. Diagram of the architecture of the multiResUnet.

location of the wanted features. However they have small variants on the way they handle spatial information through the layers. A multi scale representation method with PSPNet (Zhao et al., 2017) will also be used. And finally, a classical full convolutional network (FCN) (Long et al., 2015).

The models can have different number of free parameters depending on the number of layers and filters at each layer. To test different numbers of trainable parameters, the architectures will be tested with several filters per level. Each of this architectures have a level in which the number of filters is minimal as it can seen on 2 noted "F". This min filter will be used to describe the variation of free parameters in the models.

2.3. Input and output data for the deep learning models

The computation from the physical model are turned into 2D maps of 150×150 m^2 at a height of 1.5 m. Two maps will be used as input, the first map representing the height of the buildings and the second map the distance from the pollutant source. The last map, will be the normalized pollutant dispersion field. An example of the images used the architectures are shown below Fig. 3:

In this study, 4919 examples were produced, divided with 3687 for training, 410 for validation and 822 divided into 28 subsets for testing according to the methodology provided by Fawaz et al. (2019). All the architectures are trained with the same dataset and tested on the same dataset. The training was performed for 25 epochs with a batch size of 6. The optimizer used is Adam. A callback patience of 5 epochs was used on the validation data loss.

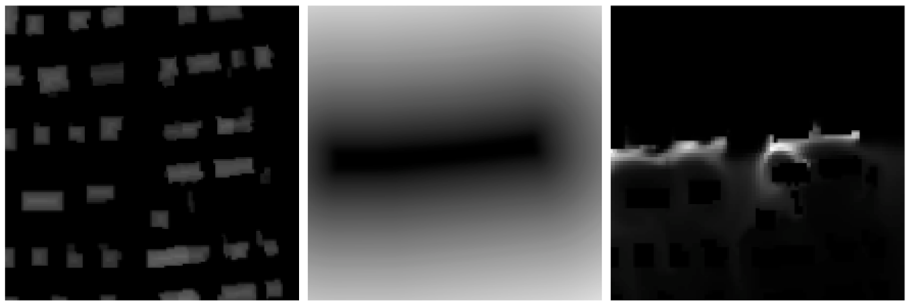
2.4. Deep learning loss

For every model, three losses are tested. Two well known losses, binary crossentropy (bce) and mean squared error (mse) as defined in Eqs. (4) and (5).

$$bce = \frac{1}{N}\sum_{i=1}^{N} y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i), \tag{4}$$

$$mse = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{5}$$

A custom loss, called $J_{3D}loss$, was also tested (see Eq. (6)). It is based on the Jaccard index, originally called community coefficient, that aims at comparing the intersection with the union of two binary



(a) Buildings (b) Pollution source (c) CFD result

Fig. 3. Images given as input to the network (a) the height, shape and position of each building in the area, (b) the distance from the pollution source, and (c) the corresponding CFD simulation pollutant dispersion field, considered as the right output for the CNN.

Table 1
Summary of the different variants of each model tested in this study.

Models	Min filters	Losses
FCN	1 - 2 - 4 - 8	J_{3D} - bce - mse
PSPNet	8 - 16	J_{3D} - bce - mse
linkNet	8 - 16 - 32	J_{3D} - bce - mse
SegNet	8 - 16 - 32	J_{3D} - bce - mse
multiResUnet	8 - 16 - 32	J_{3D} - bce - mse
Unet	8 - 16 - 32	J_{3D} - bce - mse

set. This index is often used in segmentation to compare the predicted binary mask to a ground truth segmentation mask. But here, the pollutant concentration is a continuous value, so areas cannot be compared as in segmentation. However, the continuous value can be considered as a third dimension and so the intersection over the union is not computed between two surfaces but two volumes. The loss is computed between two pairs of images as following:

$$J_{3D}\ loss = 1 - \frac{V_{pred} \bigcap V_{true}}{V_{pred} \bigcup V_{true}} \simeq 1 - \frac{1}{N}\sum_{i=1}^{N} \frac{min(y_i, \hat{y}_i)}{max(y_i, \hat{y}_i)} \tag{6}$$

where V_{pred} and V_{true} are the respective volume of the two images with the pixel value as the third dimension respectively for the predicted and ground truth image, N is the number of pixels, yi_i^{true} is the value of the ith pixel of the true image and yi_i^{true} is the value of the ith pixel in the predicted image.

2.5. Evaluation of the results

2.5.1. Popular metrics in the air quality field

To evaluate the predictions made by the deep learning architectures, several metrics will be used. Indeed, each metric measures different aspects of the model and helps to see strength and weaknesses better than reducing the analysis on one single one. In the air quality field, the study of Chang and Hanna (2004) provides several metrics to be used to evaluate and conclude on the quality of a model. Six metrics are provided, but some are equivalent and evaluate the same aspect of the prediction. Thus, only four of the metrics are kept for this study. Fractional Bias (FB) measures if the prediction mean is globally the same as the ground truth mean value. Normalized Mean Squared Error ($NMSE$) measures if there are extreme differences between the prediction and the ground truth. The fraction of predictions within a factor of two of observations ($FAC2$) measures that on overall, the predictions are within an accepting error margin. And finally, R index, that compares the correlation between the two datasets (ground truth and predictions). FB and $NMSE$ are to be minimized at 0, $FAC2$ and R are to be maximized at 1.

$$FB = \frac{(\overline{C_{ref}} - \overline{C_{pred}})}{0.5(\overline{C_{pred}} + \overline{C_{ref}})}, \tag{7}$$

$$NMSE = \frac{\overline{(C_{ref} - C_{pred})^2}}{C_{pred} C_{ref}}, \tag{8}$$

$$FAC2 = \text{fraction of data that satisfy } 0.5 < \frac{C_{pred}}{C_{ref}} < 2, \tag{9}$$

$$R = \frac{\overline{(C_{ref} - \overline{C_{ref}})(C_{pred} - \overline{C_{pred}})}}{\sigma_{C_{pred}} \sigma_{C_{ref}}}, \tag{10}$$

with C_{pred} the predicted concentration field and C_{ref} the reference concentration field (ground truth).

In Chang and Hanna (2004), the authors propose ranges of values for the metrics presented above to assess if an air quality model is satisfying. They also underline that for spatial models, these values are harder to reach. The suggested values are:

- FAC2 > 0.5,
- NSME < 1.5,
- |FB| < 0.3.

2.5.2. Metrics related to images

Three more metrics will be used to compare the predictions and ground truth. The relative mean absolute error (MAE_{rel}), J_{3D} that is also used as a loss and described previously, and the Structural Similarity Index ($SSIM$) designed to measure the visual quality between a compressed image and the original one. MAE_{rel} is to be minimized to 0. $SSIM$ and J_{3D} are to be maximized to 1.

$$MAE_{rel} = \frac{|C_{ref} - C_{pred}|}{\overline{C_{pred}}} \tag{11}$$

$$J_{3D} \simeq \frac{min(C_{ref}, C_{pred})}{max(C_{ref}, C_{pred})} \tag{12}$$

with C_{pred} the model prediction concentration and C_{ref} the reference concentration (ground truth).

$$SSIM(A, B) = \frac{(2\mu_A\mu_B + c_1)(2\sigma_{AB} + c2)}{(\mu_A^2 + \mu_B^2 + c1)(\sigma_A^2 + \sigma_B^2 + c2)} \tag{13}$$

$$c_1 = (k_1 L)^2 \quad c_2 = (k_2 L)^2 \tag{14}$$

where μ_A and μ_B are the respective average of A and B, σ_A^2 and σ_B^2 are the respective variances of A and B, σ_{AB} is the covariance of A and B, L is the dynamic range of the pixel values and k_1 and k_2 are two constants respectively 0.01 and 0.03 (by default).

3. Results

To compare the architectures, the methodology provided in Fawaz et al. (2019) will be used. This methodology allows to compare different models by ranking them on their performance on a metric over several datasets. This ranking can then be used to make a critical difference diagrams. To compare the models, the test dataset composed of 822 examples divided into 28 subdatasets will be used. A subdataset corresponds to an emission source (road) with a building outlet.

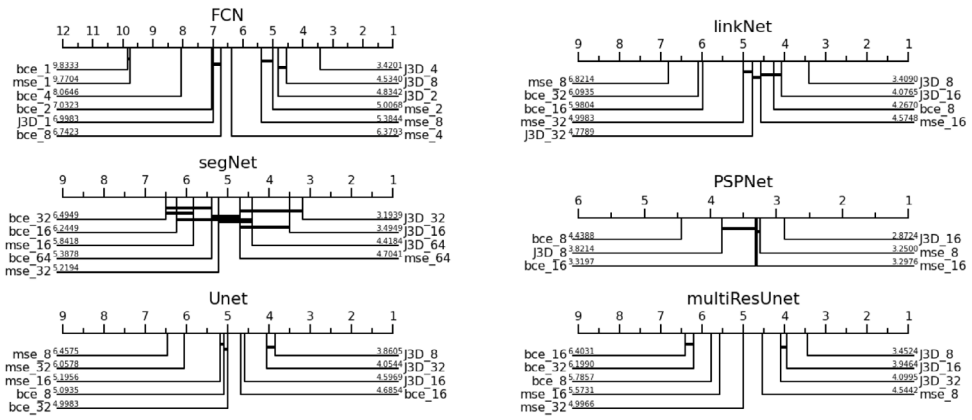


Fig. 4. Ranking of the different variants for each model using all the metrics.

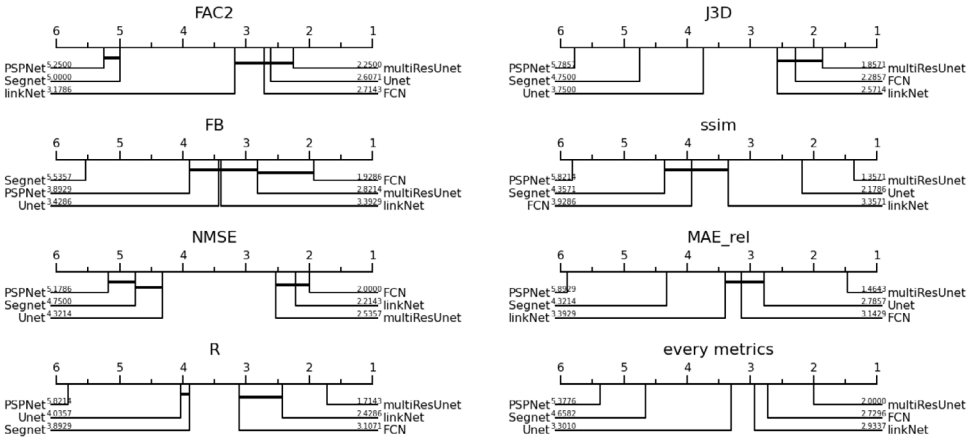


Fig. 5. Ranking of each best variant for each model according to each metric.

Table 2
Evaluation of the results of the multiResUnet on each metric.

Metric	FAC2	NMSE	FB	R	MAE rel	J3D	ssim
Mean value	0.8	3.7	0.3	0.8	0.7	0.5	0.8
Expected value	≈>0.5	≈<1.5	≈<0.3	1	0	1	1

3.1. Loss functions and filters

Three loss functions were tested along several number of filters for each 6 model as summurized in Table 1. The difference between prediction and ground truth was evaluated according to the 7 metrics presented above. Nevertheless, as this would produce $7 \times 6 = 49$ diagrams, to sum up the result, the 7 metrics of each variant were concatenated together for each model to determine the best performing variant for each model. Thus, the 6 models diagrams are presented on the critical difference diagrams in Fig. 4. Notations on the diagram for the model are "loss"_"min filters", for example a model that uses binary crossentropy and 4 min filters will be noted "bce_4".

As it can be seen on Fig. 4, the J_{3D} loss always comes first for every model.

3.2. Architectures

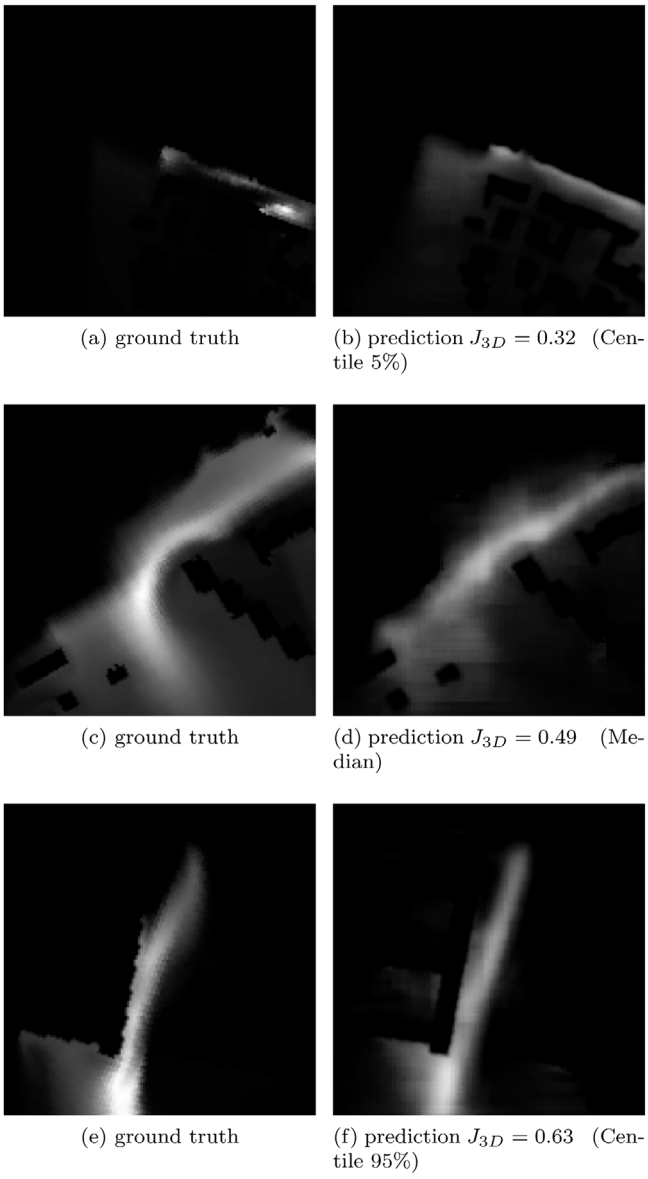
Using the best variant of each model as determined in the previous subsection, the same approach of the critical difference diagram will be used to determine which model performs best. The results for all metrics with the best variant of each model is presented on Fig. 5

The architecture that predicted the best the pollutant dispersion on overall is the multiResUnet which is first 5/7 times and always at least in the first statistically indistinguishable group. When all metrics are considered together, multiResUnet becomes first. The results on the metrics of the best model which is the multiResUnet using 8 min filters and J_{3D} are given in Table 2. It can be seen that the multiResUnet using the J_{3D} loss managed to perform within the standard performance of a good model for 2 out of 3 metrics widely used in air quality.

Examples of the multiResUnet predictions against the CFD model for the centile 5%, the median and the centile 95% of J_{3D} are shown on Fig. 6.

3.3. Deep learning speed up performance

Deep learning methods have been applied in the air quality field in recent years. Deep learning needs data to learn from, this come generally either from sensors or numerical physics models. For sensors authors have used deep learning model to predict air quality from monitoring site data using parameters influencing pollution level such as meteorological data or emission sources (Li et al., 2016; Sun et al., 2020). In their review (Liao et al., 2020) showed these several aspects covering deep learning models based on sensors data but they also underlines another aspect, the speed up of existing physics modeling approach on Chemical Transport Models (CTM). The approach of the current article is in line with this last point, the acceleration of calculations from physics models. Indeed, currently only fast computing model with simplified equations such as Gaussian can be used to perform computation on large scale or real time. These models are nevertheless not suited for urban areas as they do not take buildings or terrain



(a) ground truth (b) prediction $J_{3D} = 0.32$ (Centile 5%)

(c) ground truth (d) prediction $J_{3D} = 0.49$ (Median)

(e) ground truth (f) prediction $J_{3D} = 0.63$ (Centile 95%)

Fig. 6. Examples of predictions from the multiResUnet.

in a satisfying manner compared to more complex method such as CFD (Bady, 2017; Kumar et al., 2015).

The main issue with CFD is the computing resources and time that are necessary to obtain a reliable result. This makes CFD model unsuited for real time pollutant dispersion assessment since the time it requires to perform a simulation is too long. It cannot be performed live but a bank of CFD results could be computed in advances. Nevertheless, this option is too costly to use on several neighborhoods. Moreover, if the neighborhoods have buildings that are destroyed or new buildings built, the simulations would need to be computed again even if the flow field of the previous state of the neighborhood could be used to speed up the computation of the new configuration. To illustrate this issue, to obtain a pollutant dispersion map on an area of interest of 200×200 m^2 with the validated meshing used by Reiminger et al. (2020) and the rules of Franke et al. (2007) it requires around 4 h of computation on 48 CPU. Meanwhile, the trained deep neural network needs around 22 s to perform 450 areas of 150×150 m^2 on a GPU GTX 1080Ti. It is hard to compare the two approaches right away in term of computing time because the CFD resolves in 3D while the deep learning performs in 2.5D (it solves for a given height but takes into account building altitudes in the Z axis). Moreover, the GPU and CPU are two different components. Also, the step that requires the most computing time is the flow field computation: on wide areas, several pollutant dispersion sources could be computed at the same time using the same flow field. Knowing these, however, to determine the pollution dispersion at pedestrian level in an urban area with one road, the deep learning model performs 10 000 magnitude of order faster than the traditional CFD method. If the flow field would compute for 10 sources of pollution at the same time the speed up would be of a magnitude of 1000. Even if several multiResUnet models were made to predict different altitudes with a resolution close to the CFD (around 1 m) then the speed up would be of an order of magnitude of a 100.

Deep Learning methods could therefore represent the future of assessing pollution dispersion from traffic in crowded cities since it is promising to reproduce CFD results known for their accuracy in a fraction of the time the CFD would need. With its performance, the deep learning architecture associated with real time data feedback on traffic and meteorological data could be used to assess pollutant dispersion in real time on a city scale.

3.4. Limitations of the current approach

The current approach in its settings has some limitations some improvements could be provided in later work. Regarding the ground truth results from the CFD, they could be improved by using more complex resolution schemes such as Reynold Stress Model (RSM) or Large Eddy Simulations (LES) models. However, these methods take longer to converge and are less stable. Hence, it would be harder to create many examples as needed by deep learning models. Regarding the limitations, they are mostly due to the provided examples that followed the hypothesis given in Section 2.1:

- Atmospheric conditions: examples provided to the multiResUnet were computed only in neutral atmospheric conditions. Examples with stable and unstable atmospheric conditions would broaden the application of the deep learning model.
- The ground is considered as flat: this could be easily solved by adding either an 3D model of the terrain or adding it directly to the height of the buildings.
- The building foundation and road start at $z = 0$: this would need more examples and maybe to add a 3D model of the height of the road or use a 3D multiResUnet variant.

Regarding the deep learning approach in itself, some improvements and changes could be made on the multiResUnet model. For instance, we could apply hypertuning to improve the performance, or adapt the architecture to consider wider areas. Air pollutant dispersion is also a 3 dimensional problem, thus, the results may be improved by using a 3D variants of multiResUnet. Furthermore, it would also provide the pollution dispersion at different heights. Finally, physical constrains could be added to the model as it was done in Raissi et al. (2019), but this would require to solve the velocity and pressure field which are more complex phenomena.

4. Conclusion

Several architectures that have proved their efficiency in other fields have been applied to pollutant dispersion modeling. For each of these architectures, several variants with different amount of minimal filters were trained using three different losses. For each model, the variants were compared against several metrics and it was found that J_{3D} loss gave the best results for every model to predict airborne pollutant dispersion. The architectures were then compared one against the others and it was found that multiResUnet had the overall best results. Using metrics wildly accepted in the air quality field, 2 out of the 3 metrics are in the accepted range for a good air quality model when compared

to the ground truth. The architecture was able to obtain these results in tens of seconds compared to the classical CFD computation that requires several hours. These results are promising to enable real time pollutant dispersion in urban cities with CFD accuracy. Indeed, this Deep Learning model could be the milestone of an intelligent system to assess air pollution from traffic in real time.

CRediT authorship contribution statement

Xavier Jurado: Conceptualization, Formal analysis, Methodology, Software, Writing – original draft, Writing – review & editing, Project administration. **Nicolas Reiminger:** Methodology, Resources, Writing – review & editing. **Marouane Benmoussa:** Software, Validation. **José Vazquez:** Methodology, Resources, Writing – review & editing. **Cédric Wemmert:** Supervision, Conceptualization, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank Nvidia for providing a free Titan V GPU through their academic grant. The authors would like to thank the *Association Nationale de la Recherche et de la Technologie* (ANRt), France for supporting the Ph.D.

References

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(12), 2481–2495. http://dx.doi.org/10.1109/tpami.2016.2644615.

Bady, M. (2017). Evaluation of Gaussian plume model against CFD simulations through the estimation of CO and NO concentrations in an urban area. *American Journal of Environmental Sciences*, *13*(2), 93–102. http://dx.doi.org/10.3844/ajessp.2017.93.102.

Blocken, B. (2014). 50 Years of computational wind engineering: Past, present and future. *Journal of Wind Engineering and Industrial Aerodynamics*, *129*, 69–102. http://dx.doi.org/10.1016/j.jweia.2014.03.008.

Blocken, B., van der Hout, A., Dekker, J., & Weiler, O. (2015). CFD simulation of wind flow over natural complex terrain: Case study with validation by field measurements for Ria de Ferrol, Galicia, Spain. *Journal of Wind Engineering and Industrial Aerodynamics*, *147*, 43–57. http://dx.doi.org/10.1016/j.jweia.2015.09.007.

Boyer, V., & Baz, D. E. (2013). Recent advances on GPU computing in operations research. In *2013 IEEE international symposium on parallel & distributed processing, workshops and phd forum*. IEEE, http://dx.doi.org/10.1109/ipdpsw.2013.45.

Calderón-Garcidueñas, L., Herrera-Soto, A., Jury, N., Maher, B. A., González-Maciel, A., Reynoso-Robles, R., Ruiz-Rudolph, P., van Zundert, B., & Varela-Nallar, L. (2020). Reduced repressive epigenetic marks, increased DNA damage and Alzheimer's disease hallmarks in the brain of humans and mice exposed to particulate urban air pollution. *Environmental Research*, *183*, Article 109226. http://dx.doi.org/10.1016/j.envres.2020.109226.

Chang, J. C., & Hanna, S. R. (2004). Air quality model performance evaluation. *Meteorology and Atmospheric Physics*, *87*(1–3), http://dx.doi.org/10.1007/s00703-003-0070-7.

Chaurasia, A., & Culurciello, E. (2017). LinkNet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE visual communications and image processing (VCIP)*. IEEE, http://dx.doi.org/10.1109/vcip.2017.8305148.

Chen, L., Ding, Y., Lyu, D., Liu, X., & Long, H. (2019). Deep multi-task learning based urban air quality index modelling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *3*(1), 1–17. http://dx.doi.org/10.1145/3314389.

Das, P. K., & Meher, S. (2021). An efficient deep convolutional neural network based detection and classification of acute lymphoblastic leukemia. *Expert Systems with Applications*, *183*, Article 115311. http://dx.doi.org/10.1016/j.eswa.2021.115311.

Du, S., Li, T., Yang, Y., & Horng, S. (2021). Deep air quality forecasting using hybrid deep learning framework. *IEEE Transactions on Knowledge & Data Engineering*, *33*(06), 2412–2424. http://dx.doi.org/10.1109/TKDE.2019.2954510.

EU (2008). *Directive 2008/50/EC of the European parliament and of the council of 21 may 2008 on ambient air quality and cleaner air for Europe*. European Union.

Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P.-A. (2019). Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, *33*(4), 917–963. http://dx.doi.org/10.1007/s10618-019-00619-1.

Franke, J., Hellsten, A., Schlünzen, H., & Carissimo, B. (2007). Best practice guideline for the CFD simulation of flows in the urban environment. In *COST action 732*.

Gavahi, K., Abbaszadeh, P., & Moradkhani, H. (2021). DeepYield: A combined convolutional neural network with long short-term memory for crop yield forecasting. *Expert Systems with Applications*, *184*, Article 115511. http://dx.doi.org/10.1016/j.eswa.2021.115511.

Guo, X., Li, W., & Iorio, F. (2016). Convolutional neural networks for steady flow approximation. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, http://dx.doi.org/10.1145/2939672.2939738.

Ibtehaz, N., & Rahman, M. S. (2020). MultiResUNet : Rethinking the U-net architecture for multimodal biomedical image segmentation. *Neural Networks*, *121*, 74–87. http://dx.doi.org/10.1016/j.neunet.2019.08.025, arXiv:1902.04049.

Junfeng Chen, E. H. (2019). U-net architectures for fast prediction in fluidmechanics. hal-02401465.

Jurado, X., Reiminger, N., Vazquez, J., & Wemmert, C. (2021). On the minimal wind directions required to assess mean annual air pollution concentration based on CFD results. *Sustainable Cities and Society*, *71*, Article 102920. http://dx.doi.org/10.1016/j.scs.2021.102920.

Jurado, X., Reiminger, N., Vazquez, J., Wemmert, C., Dufresne, M., Blond, N., & Wertel, J. (2020). Assessment of mean annual NO2 concentration based on a partial dataset. *Atmospheric Enviroment*, *221*, Article 117087. http://dx.doi.org/10.1016/j.atmosenv.2019.117087.

Kumar, P., Feiz, A.-A., Ngae, P., Singh, S. K., & Issartel, J.-P. (2015). CFD simulation of short-range plume dispersion from a point release in an urban like environment. *Atmospheric Enviroment*, *122*, 645–656. http://dx.doi.org/10.1016/j.atmosenv.2015.10.027.

Kurt, A., & Oktay, A. B. (2010). Forecasting air pollutant indicator levels with geographic models 3days in advance using neural networks. *Expert Systems with Applications*, *37*(12), 7986–7992. http://dx.doi.org/10.1016/j.eswa.2010.05.093.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. http://dx.doi.org/10.1038/nature14539.

Li, X., Peng, L., Hu, Y., Shao, J., & Chi, T. (2016). Deep learning architecture for air quality predictions. *Environmental Science and Pollution Research*, *23*(22), 22408–22417. http://dx.doi.org/10.1007/s11356-016-7812-9.

Liao, Q., Zhu, M., Wu, L., Pan, X., Tang, X., & Wang, Z. (2020). Deep learning for air quality forecasts: a review. *Current Pollution Reports*, *6*(4), 399–409. http://dx.doi.org/10.1007/s40726-020-00159-z.

Liu, Y., Sun, P., Wergeles, N., & Shang, Y. (2021). A survey and performance evaluation of deep learning methods for small object detection. *Expert Systems with Applications*, *172*, Article 114602. http://dx.doi.org/10.1016/j.eswa.2021.114602.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE, http://dx.doi.org/10.1109/cvpr.2015.7298965.

Morabito, F. C., & Versaci, M. (2003). Fuzzy neural identification and forecasting techniques to process experimental urban air pollution data. *Neural Networks*, *16*(3–4), 493–506. http://dx.doi.org/10.1016/s0893-6080(03)00019-4.

Murakami, S. (1998). Overview of turbulence models applied in CWE–1997. *Journal of Wind Engineering and Industrial Aerodynamics*, *74–76*, 1–24. http://dx.doi.org/10.1016/s0167-6105(98)00004-x.

Pan, Z., Zhang, W., Liang, Y., Zhang, W., Yu, Y., Zhang, J., & Zheng, Y. (2020). Spatio-temporal meta learning for urban traffic prediction. *IEEE Transactions on Knowledge and Data Engineering*, 1. http://dx.doi.org/10.1109/tkde.2020.2995855.

Prieler, R., Mayrhofer, M., Gaber, C., Gerhardter, H., Schluckner, C., Landfahrer, M., Eichhorn-Gruber, M., Schwabegger, G., & Hochenauer, C. (2018). CFD-based optimization of a transient heating process in a natural gas fired furnace using neural networks and genetic algorithms. *Applied Thermal Engineering*, *138*, 217–234. http://dx.doi.org/10.1016/j.applthermaleng.2018.03.042.

Qi, Z., Wang, T., Song, G., Hu, W., Li, X., & Zhang, Z. (2018). Deep air learning: Interpolation, prediction, and feature analysis of fine-grained air quality. *IEEE Transactions on Knowledge and Data Engineering*, *30*(12), 2285–2297. http://dx.doi.org/10.1109/tkde.2018.2823740.

Raissi, M., Perdikaris, P., & Karniadakis, G. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, *378*, 686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045.

Reiminger, N., Vazquez, J., Blond, N., Dufresne, M., & Wertel, J. (2020). CFD evaluation of mean pollutant concentration variations in step-down street canyons. *Journal of Wind Engineering and Industrial Aerodynamics*, *196*, Article 104032. http://dx.doi.org/10.1016/j.jweia.2019.104032.

Reiminger, N., Vazquez, J., Blond, N., Dufresne, M., & Wertel, J. (2020). CFD evaluation of mean pollutant concentration variations in step-down street canyons. *Journal of Wind Engineering and Industrial Aerodynamics*, *196*, Article 104032. http://dx.doi.org/10.1016/j.jweia.2019.104032.

Richards, P., & Norris, S. (2011). Appropriate boundary conditions for computational wind engineering models revisited. *Journal of Wind Engineering and Industrial Aerodynamics*, *99*(4), 257–266.

Rivas, E., Santiago, J. L., Lechón, Y., Martín, F., Ariño, A., Pons, J. J., & Santamaría, J. M. (2019). CFD modelling of air quality in Pamplona city (Spain): Assessment, stations spatial representativeness and health impacts valuation. *Science of the Total Environment*, 19.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 234–241). Springer.

Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., & Battaglia, P. W. (2020). Learning to simulate complex physics with graph networks. arXiv: 2002.09405.

Sayeed, A., Choi, Y., Eslami, E., Lops, Y., Roy, A., & Jung, J. (2020). Using a deep convolutional neural network to predict 2017 ozone concentrations, 24 hours in advance. *Neural Networks*, *121*, 396–408. http://dx.doi.org/10.1016/j.neunet.2019.09.033.

Sun, X., Xu, W., Jiang, H., & Wang, Q. (2020). A deep multitask learning approach for air quality prediction. *Annals of Operations Research*, *303*(1–2), 51–79. http://dx.doi.org/10.1007/s10479-020-03734-1.

Tedjopurnomo, D. A., Bao, Z., Zheng, B., Choudhury, F., & Qin, A. K. (2020). A survey on modern deep neural network for traffic prediction: Trends, methods and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 1. http://dx.doi.org/10.1109/tkde.2020.3001195.

Wagner, F. H., Dalagnol, R., Tarabalka, Y., Segantine, T. Y. F., Thomé, R., & Hirye, M. C. M. (2020). U-net-id, an instance segmentation model for building extraction from satellite images—Case study in the Joanópolis city, Brazil. *Remote Sensing*, *12*(10), http://dx.doi.org/10.3390/rs12101544.

Wang, Z., Wei, W., & Zheng, F. (2020). Effects of industrial air pollution on the technical efficiency of agricultural production: Evidence from China. *Environmental Impact Assessment Review*, *83*, Article 106407. http://dx.doi.org/10.1016/j.eiar.2020.106407.

WHO (2018). *Burden of disease from ambient air pollution for 2016*. World Health Organization 2018.

Yakhot, V., Orszag, S. A., Thangam, S., Gatski, T. B., & Speziale, C. G. (1992). Development of turbulence models for shear flows by a double expansion technique. *Physics of Fluids A (Fluid Dynamics)*, *4*(7), 1510–1520. http://dx.doi.org/10.1063/1.858424.

Yan, R., Liao, J., Yang, J., Sun, W., Nong, M., & Li, F. (2021). Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering. *Expert Systems with Applications*, *169*, Article 114513. http://dx.doi.org/10.1016/j.eswa.2020.114513.

Yang, Z., Hao, J., Huang, S., Yang, W., Zhu, Z., Tian, L., Lu, Y., Xiang, H., & Liu, S. (2020). Acute effects of air pollution on the incidence of hand, foot, and mouth disease in wuhan, China. *Atmospheric Enviroment*, *225*, Article 117358. http://dx.doi.org/10.1016/j.atmosenv.2020.117358.

Zhang, G., Liu, D., He, X., Yu, D., & Pu, M. (2017). Acid rain in jiangsu province, eastern China: Tempo-spatial variations features and analysis. *Atmospheric Pollution Research*, *8*(6), 1031–1043. http://dx.doi.org/10.1016/j.apr.2017.02.001.

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE, http://dx.doi.org/10.1109/cvpr.2017.660.