AI for Air Quality: Leveraging Data Fusion for Deep Downscaling of Atmospheric Pollutants

Giannis Ashiotis[1], Evgenios Tsigkanos[1], Theodoros Christoudias[1,2,*] and Mihalis A. Nicolaou[1,*]

[1]*Computation-based Science and Technology Research Center, The Cyprus Institute, Nicosia, Cyprus*

[2]*Climate and Atmosphere Research Center, The Cyprus Institute, Nicosia, Cyprus*

Abstract

Air pollution is detrimental to the environment and contributes to the global burden of disease. Accurate prediction of surface concentrations of atmospheric pollutants, including fine particulate matter ($PM_{2.5}$) and gaseous NO_2, is key to mitigating these harmful effects. However, most forecasts produced by state-of-the-art models exhibit biases and limitations due to low-resolution (typically $\sim 40\,km$) and intense computational requirements. While deep learning-based approaches promise cheap inference, and can inherently scale and assimilate large amounts of data, ground truth measurements are obtained by station networks that are costly and sparsely distributed spatially - making the training of data-hungry deep learning methods tedious. In this context, we propose a novel deep learning framework that can accurately downscale the resolution of surface concentrations down to 1×1 km by leveraging additional data sources of static and time-varying nature (such as elevation and land use type). To do so, we carefully design a tailored data alignment and sampling process that balances the distribution across land use type, enhancing model generalization. To handle ground truth sparsity, we propose a tailored loss function that leads the network to model correlations across data sources close to ground station, while encouraging it to approximate model-based outputs elsewhere. With a set of qualitative and quantitative experiments, we demonstrate that the proposed method outperforms all compared modeling- and learning-based approaches, in *both* the aerosol and gaseous phase, with a significant increase in accuracy (up to 19% for NO_2 and 13% for $PM_{2.5}$) - while generalizing the improvement across the various land use types.

Keywords
Air Quality, Earth Observation, Deep Learning

1. Introduction

Air pollution is detrimental to human health and its contribution to the global burden of disease is now well known [1]. Regulated components of atmospheric pollution, including particulate matter $PM_{2.5}$ and gaseous NO_2, are known to cause morbidity and premature mortality in human populations [2]. Accurate prediction of the local surface concentrations of atmospheric pollutants is key to mitigating these harmful effects on human health and the environment.

While air pollution models play a pivotal role in this direction, it is well accepted that machine learning approaches can provide further insights, identifying underlying patterns, trends and relationships between parameters and data sources [3] - such as remote sensing atmospheric composition retrievals from satellite platforms, augmented by in situ ground station measurements [4].

Improvements in prediction accuracy require advances in assimilation, encompassing both observing systems as well as models [3]. Methods to assimilate data in order to improve air quality predictions based on machine learning, or autoregressive models have been developed, targeting a more realistic representation of air quality in inhabited areas [5, 6]. Air quality sensor network data were assimilated in numerical simulation using neural networks, but *limited* to a single urban area, *without* incorporating auxiliary sources of information [7, 8]. At the same time, while model-based approaches can provide regional to global coverage, they are computationally expensive and limited in terms of resolution. Furthermore, ground-based stations have very high acquisition, installation and operation costs. This limits the density of stations in order to monitor pollution accurately over large areas of interest.

Motivated by the above challenges, in this paper we present a deep learning framework that leverages publicly available data sources in order to improve downscaling accuracy of atmospheric pollutants. We leverage properties of convolutional networks, that learn hierarchical, non-linear representations of high-dimensional multimodal data, along with additional publicly available data sources - including in situ measurements, elevation maps, and aerosol optical depth columns. To address the issue of ground station *sparsity*, we design a custom loss function that heavily penalizes ground station errors, while encouraging the network to fit model-based predictions in case ground stations are not available. Furthermore, to avoid modelling spurious correlations between land use and measurements, we design a novel *tile sampling* approach that balances the land use type tiles per batch - while keeping the standard deviation over class membership as stable as possible.

The proposed framework improves the predictive performance of CAMS by alleviating biases, downscaling to 1×1 km spatial resolution. Our experiments demonstrate the intrinsic value of additional data sources such as the CORINE Land Cover (CLC) and Digital Elevation Models (DEM) in improving accuracy and successfully increasing product resolution. Additionally, in contrast to related work we show that the proposed method is generalizable to multiple pollutant species in *both* the aerosol and gaseous phases. This enables detailed studies of the impacts of pollution to drive science and policy making - in contrast to the relatively coarse resolution of model-based approaches. The contributions of this work are summarized below:

- We propose a novel deep learning framework for downscaling atmospheric pollutants in both the aerosol and gaseous phase, that can readily leverage additional data sources of different resolution and coordinate reference systems.
- We design a custom loss function that leads to state-of-the-art results using only a handful of ground truth station measurements. Significance weighting is introduced for model-based forecasts and ground-station based measurements, while network is encouraged to learn correlations between supplementary data types and ground station measurement distribution.
- We design a novel tile sampling algorithm that creates a diverse and balanced dataset per

batch. This avoids modelling spurious correlations between land use and measurements, and enhances model generalization.

- By a set of quantitative and qualitative experiments, we showcase the generality and accuracy of the proposed method against both model-based and deep-learning based approaches. Also, we provide insights regarding the data sources utilized for each pollutant that can aid future research.

2. Related Work

Historically, statistical methods [9] as well as artificial neural networks (ANN) have been used to study atmospheric effects [10], from air quality and pollution to weather patterns [11]. One such statistical method utilizing both ground station as well as satellite data is [12], using regression techniques. Spatial interpolation methods [13] like spatial averaging, nearest neighbor, inverse distance weighting and kriging have also been prevalent. The limitation with these methods however, is that they usually predict the mean (or weighted average) and miss tail events such as low or high pollutant concentrations [14]. Recently, deep learning models have been utilized for increasing the spatio-temporal resolution of EO data, e.g. [15]. Due to the time series component and the temporal relationship that air quality data possess, an appropriate network suited to this kind of data is often employed: the long short-term memory (LSTM) model which has been shown to capture well this kind of spatial-temporal correlation features [16]. In addition to an LSTM in this case, a convolutional neural network (CNN) can be utilized in unison. The CNN is used to capture spatial correlations and the combination of these architectures shows good accuracy results. Another example of DL methods used in this context is a Super Resolution Deep Residual Network (SRDRN) that work on daily precipitation and temperature downscaling [17] or a Deep Belief Network (DBN) for soil moisture downscaling [18]. Within methods such as these we can also find proof that data fusion enhances results, such as LSTMs with multimodal data [19], or an attention-based, deep convolutional neural network (AU-Net) with multimodal data [20] that downscale projected precipitation. As can be seen, the majority of related work in terms of *atmospheric pollutants* deals mostly with *one* species, without leveraging the vast amounts of additional data sources available - and is applied only in a limited spectrum of land use types.

3. Methodology

In this section, we describe the proposed methodology. It comprises of *data preprocessing* and harmonization of all publicly available data sources utilized (Section 3.1, Table 1), a *sampling* module that aims to balance the representation of land cover classes in the dataset, as well as the final deep learning *architecture*, trained on a *custom loss function* (Sections 3.3 and 3.4) that heavily penalizes ground stations measurements in comparison to CAMS. By leveraging properties of deep networks and convolutional kernels, this enables leveraging correlations with globally available side-information to mitigate the scarsity of ground stations. An overview of the proposed method is presented in Fig. 1.
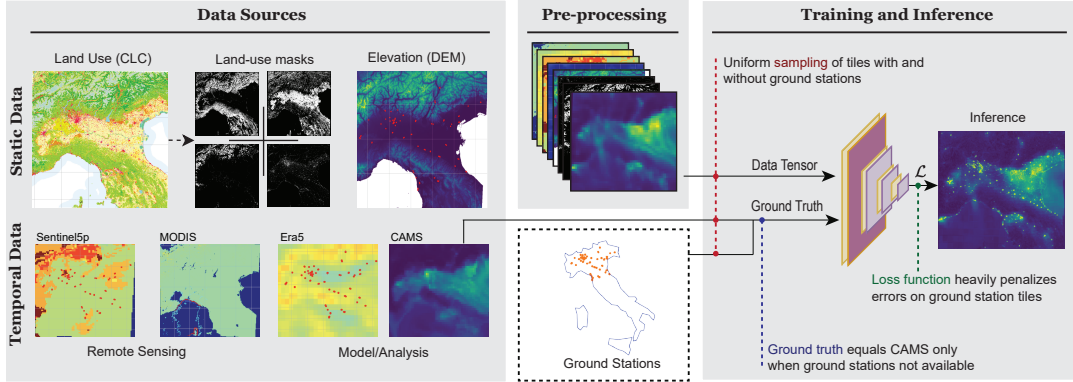
Figure 1: An overview of our method. From left to right; Static and temporal data sources are initially presented. Subsequently, all data sources are preprocessed and harmonized (coordinate system match, resolution match, normalization). The resulting data tensor is fed into the model, along with the ground truth that results from augmenting CAMS with ground station measurements. Finally, our model is trained by uniformly sampling tiles with and without ground stations, using a custom loss function that heavily penalizes errors in terms of ground station tiles.

3.1. Preprocessing

The data used to train the model originate from a 500×500 km region over the north of Italy and consist of several views, each coming with its own coordinate system and resolution. This necessitates preprocessing and harmonization of the data in order to leverage and fuse multiple data sources of heterogeneous nature. We re-project (using Nearest Neighbours) all input data to the coordinate system and resolution of CORINE Land Cover (CLC), which is the highest resolution product along with DEM. It is our hypothesis that land use would factor greatly in the inference of the requested metrics ($PM_{2.5}$ and in particular NO_2 concentrations) [21]. The products are aligned at the intersection of their bounding boxes, normalized to [0,1] and then stacked in a composite raster. As not all of the 44 different land cover classes are equally represented in our area of interest, with some classes not appearing at all, we group semantically similar classes together, and subsequently extract 15 binary masks, one for each composite CLC class. Details about this categorisation can be found in Appendix E.

3.2. Tile Sampling and Extraction

We adopt a training process that utilizes 64×64 input tiles extracted from the re-projected composite raster as input, with each of the tiles corresponding to one pixel in the output. We extend the tiles beyond the actual area that would correspond to one pixel in the output, as our working hypothesis is that pollutant concentrations in any spatial location would be influenced by conditions in the surrounding area.

To train the model, we extract an equal number of tiles with and without ground stations. However, since the land use type classes where ground stations appear are not balanced across the dataset, we design a novel sampling process that aims to *flatten* the joint distribution of land

Table 1

Data Sources and their utilization in the proposed framework

Data Source	Description
EMEP	Ground station daily measurements of NO_2 and $PM_{2.5}$ surface-level concentrations in Area of Interest (AoI). *Used to augment ground truth during training.*
CAMS	Copernicus Atmospheric Monitoring Service hourly forecasts of anthropogenic and natural pollutants. Single-channel raster for each pollutant, res: 10 km. *NO_2 and $PM_{2.5}$ surface concentrations used as input to the model and as ground truth for tiles not containing a station.*
CLC	CORINE Land Cover characterisation of land use type, grouped according to Appendix E: Table 4. Single-channel raster with discrete values, res: 100 m. *Used as reference for model input resolution.*
Sentinel 5	Sentinel-5P Satellite retrievals of the daily NO_2 tropospheric column densities. Two-channel raster with measurements and QA score, res: 7 km. *Used in the training and inferences of the NO_2 predicting model.*
DEM	Copernicus' Digital Elevation Model (EU-DEM 1.1) provides a high resolution elevation map of the AoI. Single-channel raster, res: 25 m.
ERA5	The ECMWF ERA5 reanalysis provides hourly estimates of meteorological variables. Single-channel rasters, one for each wind direction, res: 30 km. *Used horizontal wind vector components (u, v) as input source.*
MODIS	Aerosol Optical Depth (AOD) acquired by the Moderate Resolution Imaging Spectroradiometer on board the Aqua and Terra satellite mission. Two 4-to-6-channel rasters, one with the measurements and one with the QA score, res: 1 km. *Used as input source.*

use classes in the sampled tiles. This is achieved by iteratively choosing the tile, whose addition to the already sampled set minimizes the standard deviation of the CLC class histogram. More details are available in Appendix B.

3.3. Network Architecture and Implementation Details

As mentioned in Section 3.1, our dataset consists of high-res data, made up by CLC and DEM, and low-res data that for the most part have only one value per input of 10×10 (or even 64×64) tile. In order to reduce the dimensionality of the high-res data, we use a deep CNN to bring everything to a uni-dimensional latent space, out of which one single value can be extracted as our prediction with the use of a fully connected layer.

The model used in our experiments is based on the EfficientNet architecture family, utilizing a pre-trained EfficientNet B0 encoder[1], followed by a linear layer. The first layer of the model is dynamically resized to match the number of products selected at each experiment, while the last layer was resized to one (regression). Both layers were randomly initialized.

---

[1]https://github.com/rwightman/pytorch-image-models

3.4. Loss Function

The model is trained using root mean-squared error (RMSE) as loss function. The error is calculated as the difference between the generated prediction for each tile (*out*) and the ground station value (gs_i). For the tiles that do not contain a ground station, the prediction is instead compared to CAMS ($CAMS_i$).

$$\mathscr{L} = \sqrt{\frac{1}{N_{tiles}} \sum_i^{N_{tiles}} \lambda_{gs} p_i (out - gs_i)^2 + \lambda_{CAMS}(1 - p_i)(out - CAMS_i)^2}$$

$$\text{where} \quad p_i = \begin{cases} 1, & \text{if tile } i \text{ has ground station} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The hyperparameters λ_{gs} and λ_{CAMS} control the relative importance of *ground station* and *CAMS* measurements. To reflect the sparsity of ground stations, as well as the high precision of their measurements, we highly penalize errors on tiles with ground stations. Experimentally, we have determined that $\lambda_{gs} = 10$ and $\lambda_{CAMS} = 1$ yielded the best results. Intuitively, the proposed loss function forces the model to make predictions approximating the CAMS product when dealing with areas dominated by CLC classes of which ground station measurements are not available during training. By heavily penalizing errors on ground stations, we encourage the network to learn correlations between supplementary data types and the accurate distribution of ground station measurements. This enables generalization to neighbouring regions or regions with similar representations in the embedding space.

4. Experiments and Results

In this section, we present extensive experiments to evaluate the efficacy of the proposed data fusion framework for downscaling. This includes (i) comparisons to traditionally used modelling-based approaches (CAMS), as well as bilinear interpolation, (ii) comparisons with deep learning architectures without any data fusion, and (iii) various combinations of static and temporal data sources to enhance accuracy (Table 1), and highlight the value of discovering spatial correlations both within and between data sources.

4.1. Experimental Setting and Description

The input data sources are harmonized as described in Section 3.1. Subsequently, following the tile extraction process described in Section 3.2, two datasets are created; one for training using tiles extracted from days 0-61 (70% of all data) and one for testing using tiles extracted from days 62-87 (30% of all data). Furthermore, because of the availability of each type of data, the datasets are trimmed to create intersection sets with no missing data. We use retrievals with >70% Sentinel 5 quality assurance (QA) score.

As can be seen from Table 1 and Fig. 1, a set of static and time-varying data sources are employed for downscaling the concentrations of air pollutants. Concretely, the static data

includes the 15 stacked CLC masks and DEM. We also stack two more layers depth-wise for ERA5 (one for each wind direction), and one layer for CAMS - all taken at the time of station measurement (denoted as T_0). Time-varying layers for ERA5 and CAMS are added - corresponding to snapshots *before* and *after* the ground station measurement - namely, snapshots from -8h:8h of each day in two-hour increments (T_{multi}). The above static and temporal data define the **full stack** experiments, including static data, multiple temporal snapshots for CAMS and ERA, as well as Sentinel and MODIS data.

Besides experiments using only CAMS and bilinear interpolation, we also provide experiments using *different* subsets of the data sources. In more detail, in the **no fusion** scenario, we utilize the EfficientNET using only CAMS at T_0 - i.e. ground station measurement time. We then extend the no fusion scenario to include CLC and DEM, which are static data, leading to the **static + CAMS@T_0** experiment. The final experiment includes adding further snapshots of CAMS at several times of the day (**static + CAMS@T_{multi}**). For NO_2 downscaling, we also provide experiments that utilize the full stack excluding Sentinel 5. In all experiments, models were trained for a total of 60 epochs.

4.2. Results and Discussion

Our experiments involve testing against known ground station measurements of pollutant concentrations. Results are presented in Table 2, covering all experiment scenarios discussed in Section 4.1. Overall, it is evident that even using a deep network with no additional data (**no fusion**) slightly improves the results over CAMS and bilinear interpolation. Overall, one can easily see that the proposed framework, utilizing the full stack of data, leads to significant improvements in accuracy. Namely, for NO_2, RMSE is reduced by 15% and MAE by 19% compared to CAMS and its bilinear interpolation to the target resolution. Similarly, we observe a reduction of 11% for RMSE and 13% for MAE for $PM_{2.5}$. Furthermore, the reduction in σ_{AE} allows for more confidence in our predictions - and leads to more stable models which in turn enhance model *generalization*. Finally, our predictions remove the inherent bias in CAMS, in particular for the case of $PM_{2.5}$, where we achieve a reduction of 37% in the MBE.

Experiments utilizing subsets of data sources (c.f., Section 4.1) lead to conclusions that slightly deviate depending on the product to be predicted. In particular, it is clear from Table 2 that data fusion is *necessary* to provide a significant leap in downscaling accuracy. Unexpectedly at a first glance, we can also see that the addition of multiple snapshots of CAMS alone slightly reduce accuracy in comparison to using only one CAMS snapshot at T_0. Adding the corresponding snapshots of the ERA5 product alleviates this, and in the case of $PM_{2.5}$ we obtain the best performing model. This seems to, at least for $PM_{2.5}$, validate our assumptions about possible interactions between the various snapshots of CAMS and ERA5 that result in better predictive capabilities. This does not seem to be the case for NO_2 - something which could be attributed to the pollutant's short atmospheric lifetime (a few hours) - which necessitates higher temporal resolution to capture long term dependencies between pollutant concentration (from CAMS) and wind conditions (ERA5). Regardless, adding the full stack leads to less standard deviation and hence more stable - and generalizable models. The best combination of data sources can always be determined via cross-validation.

It is also important to highlight that the proposed method is successful in terms of generalizing

Table 2

Results comparing the proposed framework and variants (denoted with †) to CAMS and bilinear interpolation for downscaling: root mean square error (RMSE), mean absolute error (MAE), standard deviation of the absolute errors σ_{AE}, mean bias error (MBE). Static refers to CLC and DEM products and T_{multi} refers to -8h:+8h snapshots taken in two-hour increments around ground station measurement.

Species	Experiment	RMSE	MAE	σ_{AE}	MBE
NO_2	CAMS	33.9	30.0	15.7	29.9
	bilinear	33.6	29.9	15.5	29.8
	no fusion	32.0	27.6	16.2	26.9
	†static + CAMS@T_0	**28.1**	**23.5**	15.3	**22.5**
	†static + CAMS@T_{multi}	29.4	25.0	15.5	23.8
	†full stack \ {SEN5}	28.5	24.2	15.3	23.2
	†full stack	28.4	24.1	**15.0**	23.4
$PM_{2.5}$	CAMS	26.3	20.3	16.7	20.1
	bilinear	26.2	20.3	16.6	20.1
	no fusion	26.2	20.1	16.7	18.5
	†static + CAMS@T_0	24.0	18.2	15.6	13.6
	†static + CAMS@T_{multi}	24.5	18.5	16.0	14.7
	†full stack	**23.3**	**17.7**	**15.1**	**12.6**

performance improvement across land use types. Concretely, our method achieves an average improvement of 12.06% ($\sigma = 6.24$) for NO_2 and 13.45% ($\sigma = 9.41$) for $PM_{2.5}$. Results per land use type are shown in Appendix D Table 3 - our method improves results for each and every land use type in all but one case - where results are similar. Fig. 2a shows the underestimation by CAMS of pollutant concentrations in urban and industrial areas (land cover classes 5, 8, 9), while our model predicts values much closer to the measurements at the ground stations.

5. Conclusion

We presented a novel deep learning framework that is able to model correlations between and within multiple data sources in order to improve downscaling accuracy - leading to a wide spectrum of downstream applications that carry significant impact to public health and policy. A detailed sampling and normalization process is employed, feeding into a deep network that trained with a tailored loss function to address the sparse availability of ground truth station measurements. Our model outperforms all compared methods in downscaling NO_2 and $PM_{2.5}$ concentrations to 1 km resolution with improved accuracy. We also provide rigorous experiments that crystallize the contribution of additional data sources in this problem - which, as demonstrated, can significantly increase accuracy and stability of models. The improved performance in both pollutant concentrations evinces of the generality of the proposed method, while relying on the properties of depth-wise convolutional kernels avoids introducing costly and data-hungry temporal models (such as recurrent neural networks). This reduces both the training-time, as well as the need for site- and time- specific data sources [22].
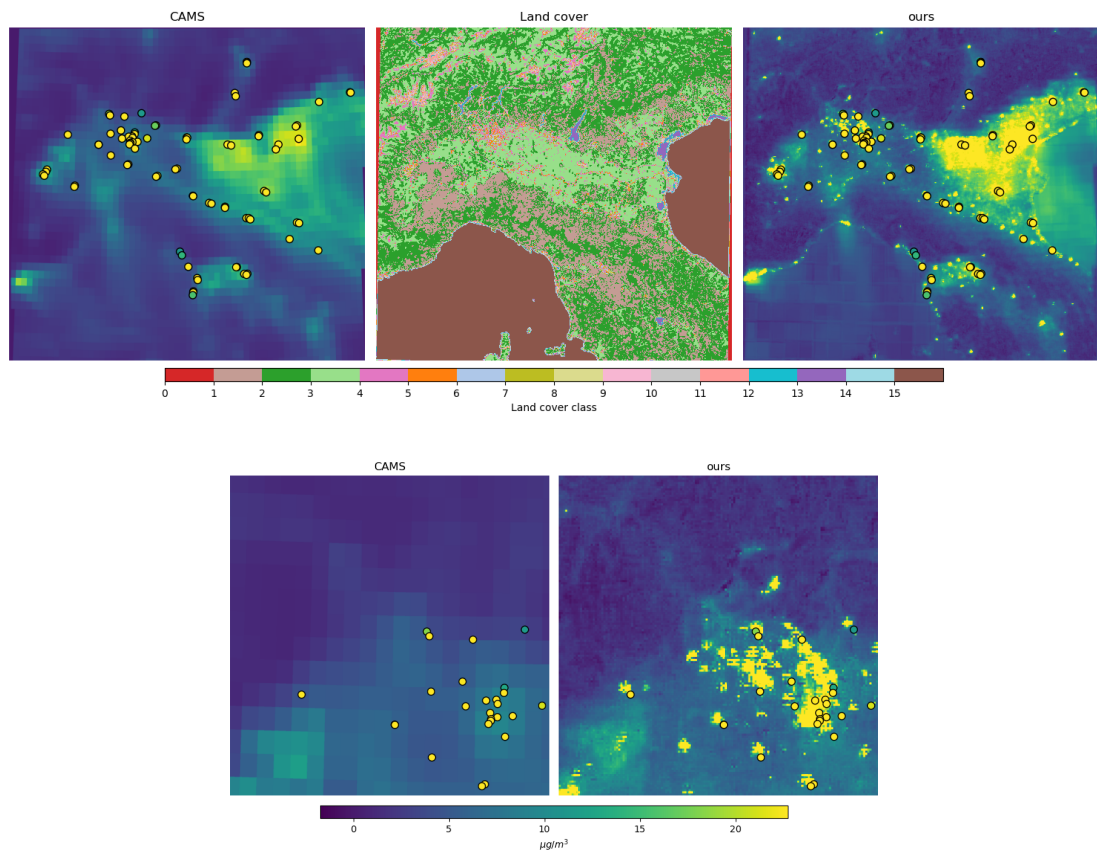
Figure 2: (top) Prediction of the concentration of NO_2 from our model compared to CAMS, and land use cases. Ground-truth station measurements are overlaid in circles using the same colormap legend as the predictions. (bottom) Area finer detail (zoom for clarity).

Acknowledgments

References

[1] World Health Organization, WHO global air quality guidelines: particulate matter (PM2. 5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide, World Health Organization, 2021.

[2] A. J. Cohen, H. Ross Anderson, B. Ostro, K. D. Pandey, M. Krzyzanowski, N. Künzli, K. Gutschmidt, A. Pope, I. Romieu, J. M. Samet, K. Smith, The global burden of disease due to outdoor air pollution, J. Toxicol. Environ. Health Part A. 68 (2005) 1301–1307.

[3] R. S. Sokhi, N. Moussiopoulos, A. Baklanov, J. Bartzis, I. Coll, S. Finardi, R. Friedrich, C. Geels, T. Grönholm, T. Halenka, M. Ketzel, A. Maragkidou, V. Matthias, J. Moldanova, L. Ntziachristos, K. Schäfer, P. Suppan, G. Tsegas, G. Carmichael, V. Franco, S. Hanna, J.-P. Jalkanen, G. J. M. Velders, J. Kukkonen, Advances in air quality research – current and emerging challenges, Atmospheric Chemistry and Physics 22 (2022) 4615–4703.

[4] A. van Donkelaar, R. V. Martin, M. Brauer, B. L. Boys, Use of satellite observations for long-term exposure assessment of global concentrations of fine particulate matter, Environ. Health Perspect. 123 (2015) 135–143.

[5] I. Kalate Ahani, M. Salari, A. Shadman, Statistical models for multi-step-ahead forecasting of fine particulate matter in urban areas, Atmospheric Pollution Research 10 (2019) 689–700.

[6] A. C. Just, K. B. Arfer, J. Rush, M. Dorman, A. Shtein, A. Lyapustin, I. Kloog, Advancing methodologies for applying machine learning and evaluating spatiotemporal models of fine particulate matter (pm2.5) using satellite data over large regions, Atmospheric Environment 239 (2020) 117649.

[7] B. Heinold, M. Weger, O. Knoth, R. Schrödner, T. Müller, L. Tõnisson, High-resolution air-quality modeling in urban areas—a case study for the city of leipzig, in: C. Mensink, V. Matthias (Eds.), Air Pollution Modeling and its Application XXVII, Springer Berlin Heidelberg, Berlin, Heidelberg, 2021, pp. 277–282.

[8] M. Otalora, L. Soulhac, C. V. Nguyen, C. Derognat, Challenges in the assimilation of mobile sensors data for urban air quality – analysis of a paris study, in: Proceedings of Abstracts 12th International Conference on Air Quality Science and Application, 2020, p. 117.

[9] D. S. Wilks, Copyright, in: Statistical Methods in the Atmospheric Sciences (Fourth Edition), fourth edition ed., Elsevier, 2019, p. iv.

[10] M. Gardner, S. Dorling, Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences, Atmospheric Environment 32 (1998) 2627–2636.

[11] A. Chattopadhyay, P. Hassanzadeh, S. Pasha, Predicting clustered weather patterns: A test case for applications of convolutional neural networks to spatio-temporal climate data, Scientific Reports 10 (2020) 1317.

[12] P. Gupta, S. A. Christopher, Particulate matter air quality assessment using integrated surface, satellite, and meteorological products: 2. a neural network approach, Journal of Geophysical Research: Atmospheres 114 (2009).

[13] D. W. Wong, L. Yuan, S. A. Perlin, Comparison of spatial interpolation methods for the estimation of air quality data, Journal of Exposure Science & Environmental Epidemiology 14 (2004) 404–415.

[14] M. C. Hubbard, W. Cobourn, Development of a regression model to forecast ground-level ozone concentration in louisville, ky, Atmospheric Environment 32 (1998) 2637–2647.

[15] K. Stengel, A. Glaws, D. Hettinger, R. N. King, Adversarial super-resolution of climatological wind and solar data, Proceedings of the National Academy of Sciences 117 (2020) 16805–16815.

[16] S. Du, T. Li, Y. Yang, S.-J. Horng, Deep air quality forecasting using hybrid deep learning framework, IEEE Transactions on Knowledge and Data Engineering 33 (2021) 2412–2424.

[17] F. Wang, D. Tian, L. Lowe, L. Kalin, J. Lehrter, Deep learning for daily precipitation and temperature downscaling, Water Resources Research 57 (2021) e2020WR029308.

[18] H. Zhao, J. Li, Q. Yuan, L. Lin, L. Yue, H. Xu, Downscaling of soil moisture products using

deep learning: Comparison and analysis on tibetan plateau, Journal of Hydrology 607 (2022) 127570.

[19] D. Tran Anh, S. P. Van, T. D. Dang, L. P. Hoang, Downscaling rainfall using deep learning long short-term memory and feedforward neural network, International Journal of Climatology 39 (2019) 4170–4188.

[20] A. Y. Sun, G. Tang, Downscaling satellite and reanalysis precipitation products using attention-based deep convolutional neural nets, Frontiers in Water 2 (2020).

[21] G. Xu, L. Jiao, S. Zhao, M. Yuan, X. Li, Y. Han, B. Zhang, T. Dong, Examining the impacts of land use on air quality from a spatio-temporal perspective in wuhan, china, Atmosphere 7 (2016).

[22] J. Kukkonen, L. Partanen, A. Karppinen, J. Ruuskanen, H. Junninen, M. Kolehmainen, H. Niska, S. Dorling, T. Chatterton, R. Foxall, et al., Extensive evaluation of neural network models for the prediction of no2 and pm10 concentrations, compared with a deterministic modelling system and measurements in central helsinki, Atmospheric Environment 37 (2003) 4539–4550.

A. Introduction

In the Appendix, we present additional information regarding the proposed method, along with more experiments and results. In more detail, in Appendix B we present algorithms and detailed descriptions of our sampling process. In Appendix D, additional results are presented - including quantitative results per land use type, as well as visualizations of results and comparisons with other methods. In Appendix C, we include a visualization of the EfficientNet encoder architecture that is utilized. Finally, in Appendix E, we show the land use characterization grouping that is employed in our work.

B. Tile Sampling and Extraction

In this section, we provide some more details on the tile extraction and sampling process used for training introduced in Section 3.2. The goal of this process is to extract CLC class balanced set of X_1 tiles that contain a ground station and another X_1 tiles that don't. In our experiments we set $X_1 = 200$.

Creation of tile set basis. We loop over all the stations for X_2 times and choose a random 10×10 CLC tile that includes the station (we chose $X_2 = 2$). This is done to create a basis onto which we base our sampling. We then calculate the CLC class histogram of the sampled tiles and its standard deviation, as we will need then in the next step. For more details see Algorithm 1.

Sampling of ground station tiles. We then, again, loop over the stations, choosing random tiles around them as before. Out of the new sampled tiles, we select the one that minimizes the standard deviation of the recalculated histogram. We repeat the same process until we have X_1 sampled tiles. To add some randomness, instead of the best performing tile
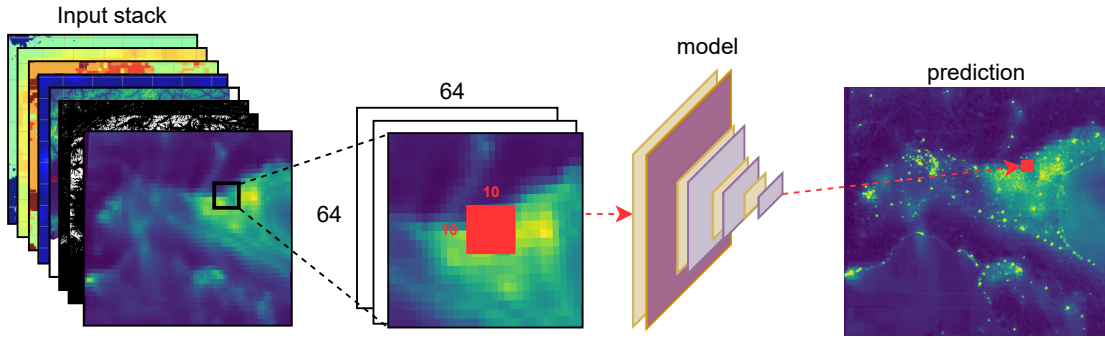
Figure 3: 64×64 tile to pixel correspondence. The 64×64 tile from the input stack is used as context to improve the final prediction, while the 10×10 inner 'generator' tile marked in red covers the extent used to generate 1 pixel in the prediction.

at each step, we can randomly select one out of the X_3 best performing tiles (in our experiments this was set to $X_3 = 5$). For more details see Algorithm 2.

Sampling of non ground station tiles. We proceed with a similar process as in the previous step. This time, instead of using ground stations as the seed for the tile, we choose pixels of the least represented CLC class. More specifically, X_4 pixels (in our experiments $X_4 = 1000$) of the least represented class are randomly selected. Following the same steps as before we randomly choose one of the X_5 best performing tiles in reducing the standard deviation of the CLC class histogram to add to the sampled set we chose $X_5 = 50$). Again, we repeat until we have the required X_1 tiles. For more details see Algorithm 3.

Finally, the tiles for the full stack experiments are extracted following Algorithm 4.

Algorithm 1 Creation of tile set basis

1: **procedure** GenBasis(CLC, St, X_2)
2:     **for** X_2 times **do**
3:         **for** each station pixel St_i **do**
4:             Choose random 10×10 sample T_i from CLC with $St_i \in T_i$
5:             Add tile T_i to $Tiles$
6:         **end for**
7:     **end for**
8:     **return** $Tiles$
9: **end procedure**

Algorithm 2 Sampling of ground station tiles

1: **procedure** SampleStations($CLC, St, Tiles, X_1, X_3$)
2:     $H_{Tiles} \leftarrow HISTOGRAM(Tiles)$
3:     $Std \leftarrow \sigma(H_{Tiles})$
4:     **while** $Num(Tiles) < X_1$ **do**
5:         **for** each station pixel St_i **do**
6:             Choose random 10×10 tile T_i from CLC with $St_i \in T_i$
7:             $H_{Tiles+T_i} \leftarrow HISTOGRAM(Tiles + T_i)$
8:             $Std_i \leftarrow \sigma(H_{Tiles+T_i})$
9:             $D_i \leftarrow Std_i - Std$
10:         **end for**
11:         Choose randomly T_{sel} from the X_3 T_is with the lowest D_i
12:         Add T_{sel} to $Tiles$
13:         $H_{Tiles} \leftarrow HISTOGRAM(Tiles)$
14:     **end while**
15:     **return** $Tiles$
16: **end procedure**

Algorithm 3 Sampling of tiles without ground stations

1: **procedure** SampleNoStations($CLC, Tiles, X_1, X_4, X_5$)
2:     **while** $Num(Tiles) < 2 \times X_2$ **do**
3:         Find least represented class C_i
4:         Choose randomly X_4 pixels P of class C_i
5:         **for** each pixel P_i of P **do**
6:             Choose random 10×10 tile T_i from CLC with $P_i \in T_i$
7:             $H_{Tiles+T_i} \leftarrow HISTOGRAM(Tiles + T_i)$
8:             $Std_i \leftarrow \sigma(H_{Tiles+T_i})$
9:             $D_i \leftarrow Std_i - Std$
10:         **end for**
11:         Choose randomly T_{sel} from the X_5 T_is with the lowest D_i
12:         Add T_{sel} to $Tiles$
13:         $H_{Tiles} \leftarrow HISTOGRAM(Tiles)$
14:     **end while**
15:     **return** $Tiles$
16: **end procedure**

Algorithm 4 Extract full-stack tiles

1: **procedure** ExtractTɪles(*Tiles*, *days*, *stackedRaster*)
2:     **for** Every day d_i in *days* **do**
3:         Update *stackedRaster* for day d_i
4:         **for** every tile T_i in *Tiles* **do**
5:             Find the 64×64 tile T_i' with T_i in its center
6:             Extract and save T_i' from *stackedRaster*
7:         **end for**
8:     **end for**
9: **end procedure**

C. Network architecture visualization

(a) EfficientNet encoder architecture.
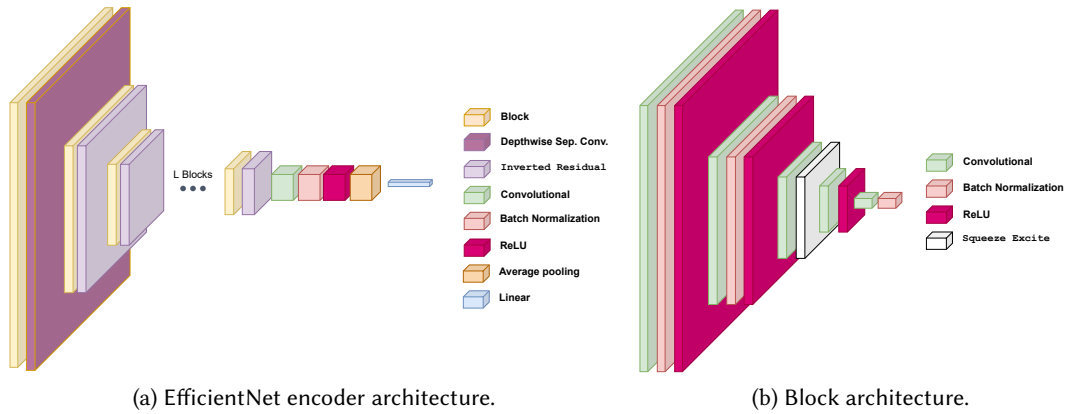

(b) Block architecture.

Figure 4: Model architecture overview. In (a) the block architecture is simplified as an abstraction and further defined in (b).

D. Additional Results

Table 3
Per land cover class performance and improvement.

Class	Stations	CAMS μ/m^3	Ours μ/m^3	Improv. %
1	19	36.6	29.5	19.4
2	10	38.2	31.4	17.8
3	29	29.2	24.6	15.9
5	11	36.2	30.6	15.4
7	1	25.8	23.4	9.4
8	2	45.2	45.9	-1.4
9	9	32.4	28.4	12.4
10	4	28.7	24.0	16.2
13	1	26.3	25.3	3.6
14	2	34.4	30.3	12.0

(a) RMSE per land cover class for NO_2.

Class	Stations	CAMS μ/m^3	Ours μ/m^3	Improv. %
1	8	20.0	17.2	14.4
2	5	20.9	15.3	26.6
3	16	29.5	26.8	9.2
5	7	23.8	21.0	11.5
7	1	27.0	25.6	5.2
8	1	21.8	18.4	15.7
9	4	29.9	24.2	18.9
10	2	27.2	26.3	3.4
13	1	42.1	42.4	-0.7
14	1	20.9	14.5	30.4

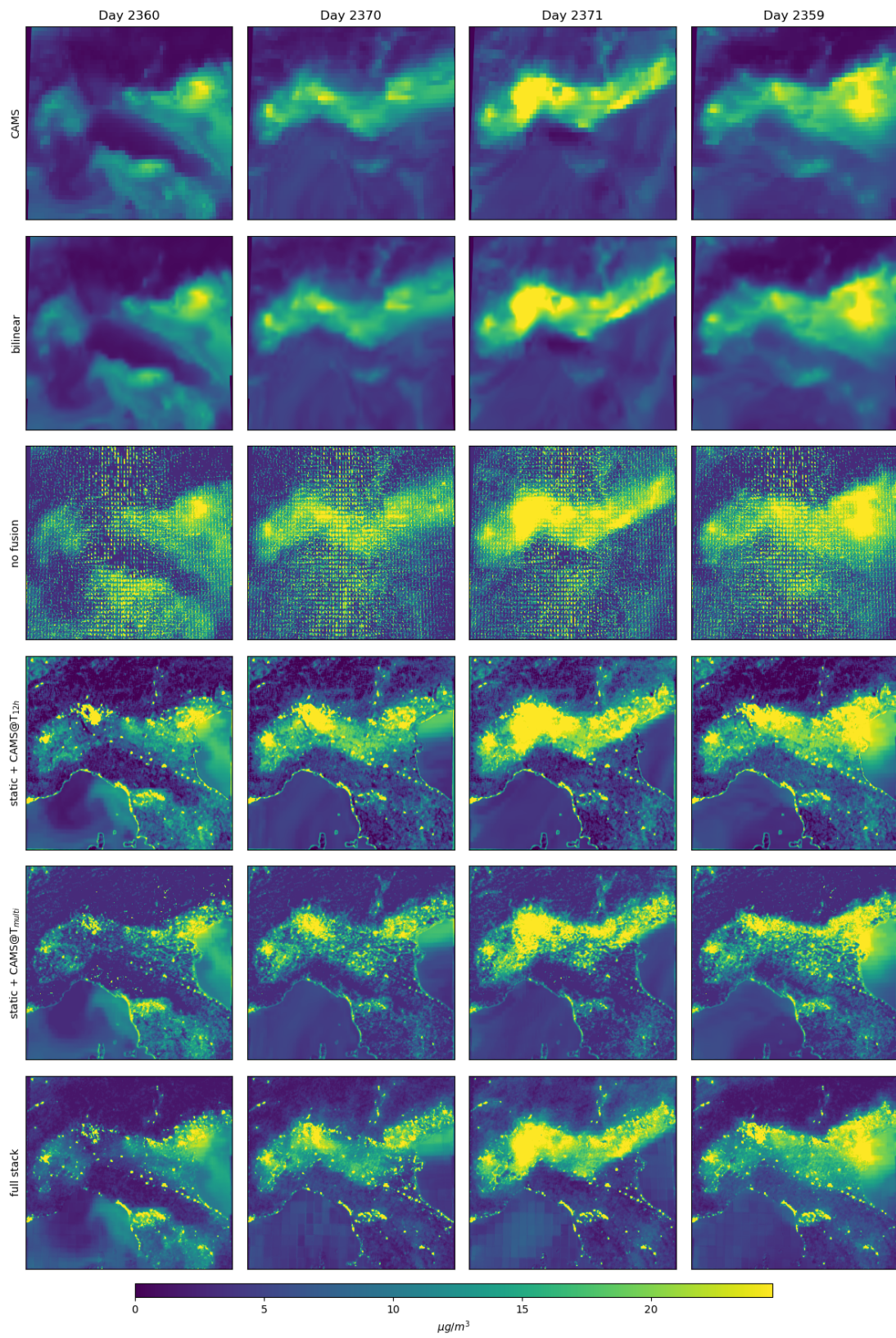(b) RMSE for each of the land cover classes for $PM_{2.5}$.

Figure 5: Additional results and comparisons.

E. Land use type grouping in CLC

Table 4
Land use characterization grouping in CLC.

Characterization	Label	New Category	New Label
Vineyards	15		
Fruit trees and berry plantations	16		
Olive groves	17		
Annual crops associated with permanent crops	19	Agriculture	1
Complex cultivation patterns	20		
Principally agriculture land with significant areas of natural vegetation	21		
Agro-forestry areas	22		
Broad-leaved forest	23		
Coniferous forest	24	Forests	2
Mixed forest	25		
Non-irrigated arable land	12		
Permanently irrigated land	13		
Rice fields	14		
Pastures	18		
Natural grasslands	26	Meadows	3
Moors and heathland	27		
Sclerophyllous vegetation	28		
Transitional woodland-shrub	29		
Beaches dunes sands	30		
Bare rocks	31		
Sparsely vegetated areas	32	Beaches	4
Intertidal flats	39		
Discontinuous urban fabric	2	Discontinuous urban fabric	5
Construction sites	9	Construction sites	6
Mineral extraction sites	7	Mines	7
Dump sites	8		
Continuous urban fabric	1	City	8
Sport and leisure facilities	11		
Industrial or commercial units	3	Industrial or commercial units	9
Road and rail networks and associated land	4		
Port areas	5	Traffic	10
Airports	6		
Burnt areas	33	Burnt areas	11
Glaciers and perpetual snow	34	Glaciers and perpetual snow	12
Inland marshes	35		
Peat bogs	36		
Salt marshes	37		
Salines	38		
Water courses	40	Inland water	13
Water bodies	41		
Coastal lagoons	42		
Estuaries	43		
Green urban areas	10	Green urban areas	14
Sea and ocean	44	Sea and ocean	15