Segmentation of XCO_2 images with deep learning: application to synthetic plumes from cities and power plants

Joffrey Dumont Le Brazidec[1], Pierre Vanderbecken[1], Alban Farchi[1], Marc Bocquet[1], Jinghui Lian[2,3], Grégoire Broquet[2], Gerrit Kuhlmann[4], Alexandre Danjou[2], and Thomas Lauvaux[2]

[1]CEREA, École des Ponts and EDF R&D, Île-de-France, France
[2]Laboratoire des Sciences du Climat et de l'Environnement, LSCE/IPSL, CEA-CNRS-UVSQ, Université Paris-Saclay, 91198 Gif-sur-Yvette, France
[3]Origins.S.A.S, Suez Group, Île-de-France, France
[4]Swiss Federal Laboratories for Materials Science and Technology (Empa), Dübendorf, Switzerland

Correspondence: Joffrey Dumont Le Brazidec (joffrey.dumont@enpc.fr)

Abstract.

Under the Copernicus programme, an operational CO_2 Monitoring Verification and Support system (CO_2MVS) is being developed and will exploit data from future satellites monitoring the distribution of CO_2 within the atmosphere. Methods for estimating CO_2 emissions from significant local emitters (hotspots, i.e. cities or power plants) can greatly benefit from the availability of such satellite images, displaying atmospheric plumes of CO_2. Indeed, local emissions are strongly correlated to the size, shape and concentrations distribution of the corresponding plume, the visible consequence of the emission. The estimation of emissions from a given source can therefore directly benefit from the detection of its associated plumes in the satellite image.

In this study, we address the problem of plume segmentation, i.e. the problem of finding all pixels in an image that constitute a city or power plant plume. This represents a significant challenge, as the signal from CO_2 plumes induced by emissions from cities or power plants is inherently difficult to detect since it rarely exceeds values of a few ppm and is perturbed by variable regional CO_2 background signals and observation errors. To address this key issue, we investigate the potential of deep learning methods and in particular convolutional neural networks to learn to distinguish plume-specific spatial features from background or instrument features. Specifically, a U-net algorithm, an image-to-image convolutional neural network, with a state-of-the-art encoder, is used to transform an XCO_2 field into an image representing the positions of the targeted plume. Our models are trained on hourly 1 km simulated XCO_2 fields in the regions of Paris, Berlin and several German power plants. Each field represents the plume of the hotspot, the background consisting of the signal of anthropogenic and biogenic CO_2 surface fluxes near or far from the targeted source and the simulated satellite observation errors.

The performance of the deep learning method is thereafter evaluated and compared with a plume segmentation technique based on thresholding in two contexts: the first where the model is trained and tested on data from the same region, and the second where the model is trained and tested in two different regions. In both contexts, our method outperforms the usual segmentation technique based on thresholding and demonstrates its ability to generalise in various cases: city plumes, power plant plumes, and areas with multiple plumes. Although less accurate than in the first context, the ability of the algorithm to

extrapolate on new geographical data is conclusive, paving the way to a promising universal segmentation model, trained on a well-chosen sample of power plants and cities, and able to detect the majority of the plumes from all of them. Finally, the highly accurate results for segmentation suggest a significant potential of convolutional neural networks for estimating local emissions from spaceborne imagery.

1 Introduction

Under the Paris Agreement on Climate Change, progress on emission reduction efforts is monitored on the basis of regular updates of the national greenhouse gas (GHG) inventories (UNFCCC, 2015). To independently assess the progress of countries towards their targets, objective means of tracking anthropogenic CO_2 emissions and their evolution is needed. Top-down estimates based on atmospheric measurements can provide such observation-based evidence. Developed through the European Earth observation programme, Copernicus, the CO_2 emissions Monitoring and Verification Support capacity (CO_2MVS) will provide an operational emissions monitoring system based on such an approach (Janssens-Maenhout et al., 2020). It will operate in particular on a constellation of dedicated CO_2 imaging satellites, the Copernicus CO_2 Monitoring (CO_2M) mission, as part of the Sentinel programme, which will be launched from 2026.

One aim of CO_2MVS is to provide estimates of local emissions from hotspots such as cities or power plants that account for a major fraction of anthropogenic CO_2 releases. For this purpose, local data assimilation can be applied to individual plumes visible in satellite CO_2 images. A plume is defined as an increase in CO_2 concentration above the background level, caused by emissions from a hotspot. To estimate emissions from the plume, it is essential to detect it on satellite images. Thus, the detection of a plume, i.e. the identification of its contour, in a satellite image is a critical step in the evaluation of source emissions.

The detection and identification of pollutant plumes from simulated fields or observations has been the subject of an important amount of research. Lauvaux et al. (2022) exploit satellite images sampled by the TROPOspheric Monitoring Instrument (TROPOMI) to identify very large emitters of CH_4 using a thresholding technique. Finch et al. (2021) successfully trained Neural Networks (NN) on satellite images of NO_2 to detect the presence of plumes. Recent thresholding techniques have proven effective in detecting large CO_2 plumes in satellite images, either using the Orbital Carbon Observatory-2 (OCO-2, Crisp et al., 2017; Reuter et al., 2019), or Observing System Simulation Experiments (OSSEs) (Kuhlmann et al., 2019a).

Nevertheless, the detection and quantification of CO_2 plumes in satellite images remains a challenge with various obstacles. Conventional threshold-based methods rely on the signal-to-noise ratio of a plume. The signal is the CO_2 enhancement inside the plume above the background field and the noise is the variability in measurements due to single sounding precision of the instrument and the interference of other anthropogenic and biospheric fluxes. Kuhlmann et al. (2019a) showed that for the expected single sounding precision of the CO2M CO_2 product (<0.7 ppm, 2 km resolution, swath width $>$ 250 km, MRDv3), the signal-to-noise ratio of many cities and power plants is too small for a reliable detection of CO_2 plumes with threshold-based methods. CO2M will overcome this limitation through an additional nitrogen dioxide (NO_2) instrument on the same platform that, as a proxy to CO_2, significantly improves plume detection capabilities (Kuhlmann et al., 2019a). CO2M will also provide CH_4 observations [1]. However, not all currently planned CO_2 imaging satellites (such as CO2Image (Butz et al., 2022)) will have NO_2 observations available, which puts a limit on the capabilities for CO_2 imaging instrument to detect emission plumes using threshold-based methods.

Although mainly motivated by CO_2MVS, this study focuses on CO_2 images in general. The objective is to cope with the signal-to-noise ratio (SNR) problem in CO_2 plume detection problems, with the help of deep learning methods (Chollet, 2017; Zhang et al., 2022). In particular, we rely on convolutional neural networks (CNNs) to segment plumes more accurately than thresholding techniques by learning and capturing plume-specific spatial patterns. Plumes may indeed have certain spatial properties or shapes that can be exploited by an algorithm capable of extracting and learning these features. The image dataset used to train and test the CNN model is based on fields of column-averaged dry air mole fractions of CO_2 (XCO_2), simulated in the vicinity of the targeted sources (Grand Paris, Île-de-France (IdF), Berlin, and various power plants). Each image is comprised of (at least) a targeted source plume and the other nearby biogenic and anthropogenic fluxes, plus the instrumental noise typical of the sensor onboard CO_2M. Clouds are not included in the CO_2 images for simplicity. They will be addressed in a separate publication.

A large amount of labelled data is a prerequisite for the use of CNNs. In section 2, we present the two synthetic datasets that are used to train and evaluate the performance of the CNNs. The plume segmentation problem is then mathematically defined in section 3.1. The loss function, which defines what the CNN should target, i.e., what a plume is according to the deep learning model, is described in section 3.2. Next, in section 3.3, the architecture and parameterisation of the CNN are introduced and explained. Subsequently, the trained model is applied in two contexts:

- a context of geographical generalisation, where a model trained to recognise plumes on images from the regions of Paris, Berlin, and various power plants is evaluated on new images from the same regions;
- a context of geographical extrapolation, where a model trained to recognise plumes on images from the regions of Paris, and various power plants is evaluated on images from the region of Berlin.

In both cases, the CNN method is compared for reference to the thresholding plume segmentation method described by Kuhlmann et al. (2019a, 2021), which is available as part of a Python package for data-driven emission quantification (ddeq; https://gitlab.com/empa503/remote-sensing/ddeq). Finally, conclusions on the performances of the deep learning models in these situations are provided.

2 Synthetic datasets

2.1 Simulation of the CO_2 fields

Two different atmospheric transport models are used to simulate the CO_2 fields which provide the XCO_2 images. Simulations in the Paris region by WRF-Chem V3.9.1 (Grell et al., 2005) are based on the configuration of Lian et al. (2021) while simulations

[1] https://www.eoportal.org/satellite-missions/co2m

in the Berlin region including neighbouring power plants are taken from the SMARTCARB project (Kuhlmann et al., 2019b; Brunner et al., 2019).

Paris data consist of 3-month meteorological and CO_2 transport simulations on a nesting of three domains with different spatial resolutions (25, 5, and 1 km). Initial and boundary conditions (ICBC) are forced with ERA-5 re-analysis fields (Hersbach et al., 2020) at a resolution of $0.75°$ for the meteorological simulations and CAMS 3-hourly update interval global CO_2 atmospheric inversion products for the CO_2 simulations (Chevallier, 2018). High-resolution inventories, the TNO GHGco v3.0 TNO-MACC II and the VERIFY D2.1 v1.0 (Denier van der Gon et al., 2021), are used to simulate CO_2 concentrations over the entire domain. Finally, biogenic fluxes are computed with the Vegetation Photosynthesis and Respiration Model (VPRM) model (Mahadevan et al., 2008) coupled online with the WRF-Chem V3.9.1 model.

The SMARTCARB simulations were run with the COSMO-GHG model for a domain centred on Berlin and covering several neighbouring power plants (Jänschwalde, Lippendorf, Boxberg and others). The simulations were used to generate synthetic CO2M observations (Kuhlmann et al., 2020b) and to assess different plume detection and inversion methods (Kuhlmann et al., 2019a, 2020a, 2021; Hakkarainen et al., 2022). The model fields consist of hourly data over one year with a spatial resolution of $0.01°$ and sixty vertical layers from 0 to 24 km. MeteoSwiss COSMO-7 analyses are used as the meteorological initial and boundary conditions, while the CO_2 boundary conditions correspond to the fields of the ECMWF (European Centre for Medium-Range Weather Forecasts) free-running global CO_2 simulations with 137 levels (Agustí-Panareda et al., 2014). Biogenic CO_2 fluxes are modelled offline with the VPRM diagnostic biosphere model (Mahadevan et al., 2008). Finally, the TNO-MACC III inventory (Kuenen et al., 2014) is used for modelling anthropogenic emissions in most of the regions. Berlin emissions, however, are modelled with the help of a detailed inventory (Kuhlmann et al., 2019b). The main configuration parameters are summarised in the table 1.

	Paris	Berlin and PP
Transport model	WRF-Chem V3.9.1	COSMO-GHG
Domain	Île-de-France (IdF) + surroundings	$\sim$ 700 km^2 centred around Berlin
Output spatial resolution	Nested: 25, 5, and 1 km in IdF	1.1 km
Output time resolution	1 hour	1 hour
Vertical resolution	43 levels (until 50 hPa)	60 t.f.l (from 0 to 24 km)
Meteorological fields [ICBC]	ERA-5 ECMWF re-analysis fields at $0.75°$	COSMO-7 analyses of MeteoSwiss
CO_2 tracers [ICBC]	Global CO_2 atmospheric inversion products at 3-hourly update intervals from CAMS	Global free-running CO_2 simulations with 137 levels from ECMWF
Anthropogenic emissions	TNO GHGco emission inventory v3.0 (1 h, 1 km)	TNO/MACC III inventory (1 h, 7 km) Berlin: detailed inventory
Biogenic CO_2 fluxes	online VPRM	offline VPRM

Table 1. Main setup parameters of the transport of CO_2 for the Paris simulations and the SMARTCARB simulations.

2.2 Parameterisation of the CO_2 field simulations

CNN segmentation models are trained and tested on images of fixed size: XCO_2 images of 160×160 pixels are extracted from the Paris and SMARTCARB datasets. Therefore, the images used to train and evaluate the CNN are not synthetic CO_2M observations but a simplified dataset. The images are extracted such that the hotspot is located in the centre of the image and the chosen size ensures that most of the hotspot plume is present in the image. The native resolution of $1.1\,\mathrm{km}$ of the SMARTCARB data is maintained during this extraction phase, while Paris data are mapped from the original 200 pixels in longitude and 165 pixels in latitude to 160×160 pixels: the new image concentrations are calculated by cubic spline interpolation (Virtanen et al., 2020) which gives images with a resolution of $1.25 \times 1.03\,\mathrm{km}^2$ in IdF.

A wide variety of fields and plumes are needed to train an efficient plume segmentation model. The dataset diversity and size is achieved through:

– seasonal variability (January, March, and August for the Paris data to cover summer and winter, and a whole year for SMARTCARB);

– geographical variability (Paris, and various locations in Germany);

– emission range variability across different locations and times. In Berlin, the average emissions based on the inventory is 16.8 Mt/yr with a standard deviation (std) of 7.2 Mt/yr. In Jänschwalde, the emissions average is 33.3 Mt/yr with a std of 7.7 Mt/yr, while in Boxberg, the average is 19.0 Mt/yr with a std of 4.4 Mt/yr. The Grand Paris emissions average is 20.7 Mt/yr with a std of 9.5 Mt/yr;

– plume type variability: single power plant plumes (a single major anthropogenic plume on the image) with Lippendorf, multiple plumes (several major anthropogenic plumes on the image) with Jänschwalde or Boxberg, cities (Grand Paris, Berlin) or cities with an extended suburb (the Île-de-France including Paris region, IdF). The Paris data is split into two parts to assess the ability of the CNN to retrieve plumes from the Paris conurbation alone (Grand Paris) or from the entire Paris region (IdF).

To fully account for the detectability factors affecting the signal-to-noise ratio, the satellite instrumental noise must be taken into account. In this study, a Gaussian random noise, without spatial correlation, of $0.7\,\mathrm{ppm}$, typical of CO_2M (Meijer, 2020), is used and added to the simulated XCO_2 fields. Considering these various factors, the generation of a XCO_2 image can be summarised in three steps: simulation of the hotspot anthropogenic plume, addition of the simulated background (biogenic and other anthropogenic fluxes), and addition of the instrument noise. This is illustrated on Fig. 1. We provide the CNN model with full noisy images (right panels of Fig. 1) and we design it to return the plume masks of the hotspot plumes (left panels of Fig. 1).

Data augmentation techniques (Chollet, 2017) are applied to the training data. The training images are randomly shifted, zoomed, sheared, flipped and rotated variants of the original images. Specifically, each image used for training the CNN has been subject to:

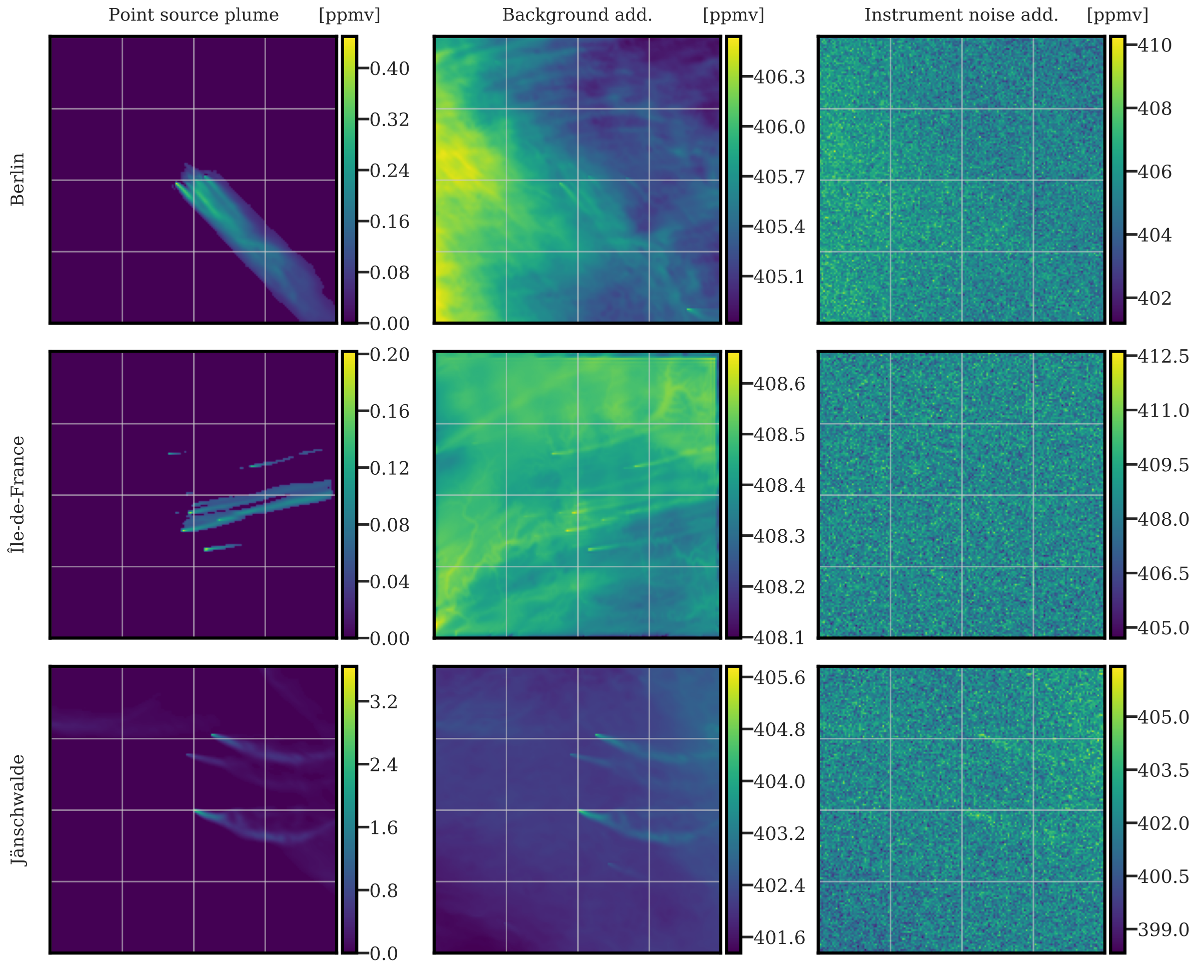

Figure 1. Examples for the construction of three simulated XCO_2 satellite images. Each row shows the generation of a sample XCO_2 image; three hotspots are considered at random times and days: Berlin (top), Paris (middle), and Jänschwalde (bottom) which is located near other power plants, which explains the presence of multiple plumes. The left column displays the anthropogenic hotspot plumes with the concentrations in ppmv indicated on the colour bars. In the middle column is shown the addition of background (biogenic and anthropogenic fluxes). Finally, on the the right column is revealed the full simulated image used as input to the CNN model with the addition of satellite instrument noise.

- a random horizontal and vertical shift of 0 to 20% (the border values are then used to fill the missing values of the new image, as shown in Fig. 2);
- a random zoom of 0 to 20%;
- a potential horizontal or vertical flip with a probability of 0.5;

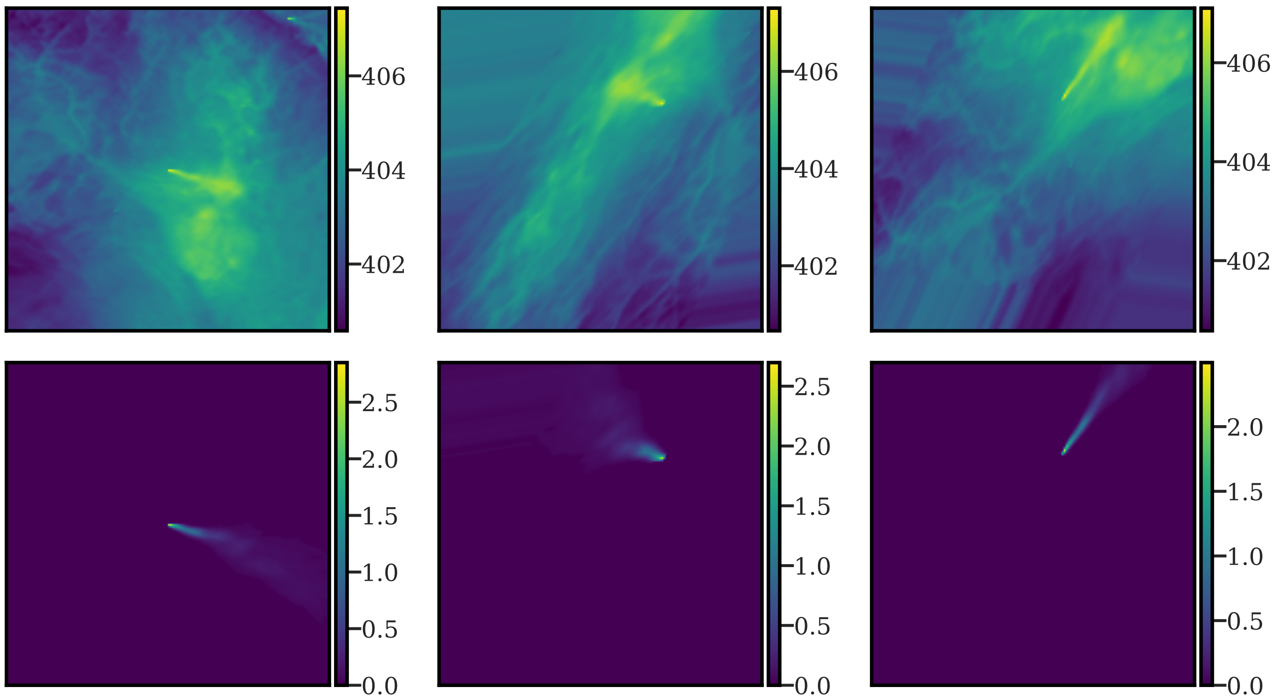

Figure 2. Examples of augmented XCO_2 fields (top row, without instrumental noise) and corresponding plumes (bottom row). The left column corresponds to the original XCO_2 field and plume. The middle and right columns correspond to the same XCO_2 fields, after shearing, flipping, rotating and translating operations. These are typical examples of what is used as input to the CNN model.

– a random rotation of 0 to $180°$;

– a random shear, i.e., a distortion along an axis (while the other axis is fixed), of 0 to $45°$.

Data augmentation is meant to (i) raise the performance of the CNN model: data augmentation artificially and substantially increase the number of training data, thus reducing the risk of overfitting, (ii) raise the representativeness of our plume database through the enforcement of geometrical invariance.

Figure 2 shows two examples of data augmented fields and associated plumes. The selection of the data augmentation techniques used and their characteristics was based on experimentation. Extrapolation or distortion of plumes due to data augmentation can lead to non-physical plumes. Yet, we empirically found that the use of such plumes improves the ability of the CNN model to segment real plumes.

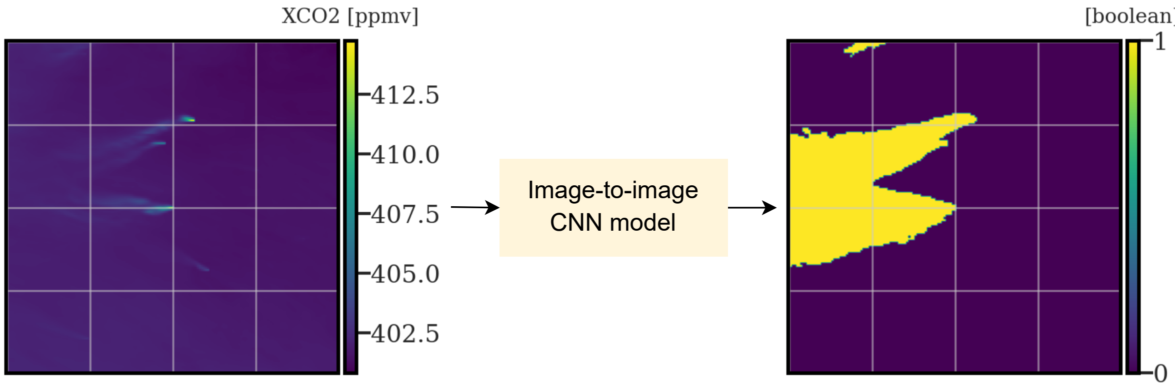

Figure 3. Illustrating the principle of plume segmentation. An image-to-image model (here, a CNN) is used to extract from a XCO_2 field the anthropogenic plumes from one or several hotspots.

3 Segmentation: methodology

3.1 Problem description

The plume segmentation problem can be defined, for a given image, as the detection of all pixels composing the plume. This problem can be seen as an image-to-image problem, where the goal is to translate the original image into a Boolean map where pixels are assigned to True (part of a hotspot plume) or False (not part of a hotspot plume), as shown in Fig. 3. Many algorithms can be designed to perform such a translation. However, in this study, we dispose of a labelled dataset as both the input XCO_2 field and the corresponding targeted plume are available. In this context of supervised learning, for image processing, CNNs are particularly effective (Chollet, 2017; Zhang et al., 2022).

These algorithms are based on learning specific patterns of increasing complexity using smaller and simpler patterns (the filters). The larger and more complex patterns are specific to the learned targets (here, the plumes from the targeted sources). The filters are optimised to allow the learning of these complex target-specific features. This optimisation is done automatically, unlike most algorithms where the filters would have to be chosen manually (feature engineering).

The CNNs decomposes as a training step (which includes validation) and a test step. In the training step, the selected CNN model, described in section 3.3, is trained with XCO_2 field/Boolean map pairs. The Boolean map is composed of pixels equal to 1 if the pixel has a positive XCO_2 concentration corresponding to the simulated anthropogenic plume, or 0 if the pixel does not. For a given XCO_2 field, the CNN model knows the target Boolean plume and learns to output a probability map that best matches it (supervised learning). The shapes of the input and output are equal and each pixel in the output represents the probability that the pixel in the input belongs to the anthropogenic plume. In the testing step, the CNN model is applied to new input images, none of them seen during the learning phase, to assess its ability to generalise to new data.

3.2 Loss function

The loss function is a measure of the discrepancy between the truth (the Boolean map representing the real plume) and the prediction (a probability map). Many loss functions can be used, each of them defining what the CNN model should learn from the data, what the priorities are and which differences can be overlooked (Jadon, 2020). The definition of a plume, according to the CNN model, is embedded in the characterisation of the loss function. A classical loss function used for segmentation problems is the binary cross entropy (~~bce~~BCE) between a scalar prediction p and a target y, defined as

$$\mathrm{BCE}(p,y) \triangleq -(y\ln p + (1-y)\ln(1-p)), \tag{1}$$

where $0 < p < 1$ and y is a Boolean value. In our case, the total loss, the discrepancy between a predicted probability map $\mathbf{P} = (p_{i,j})$ and a targeted Boolean map $\mathbf{Y} = (y_{i,j})$, is written:

$$\mathcal{L}(\mathbf{P},\mathbf{Y}) = -\sum_{i,j=1}^{160} (y_{i,j}\ln p_{i,j} + (1-y_{i,j})\ln(1-p_{i,j})), \tag{2}$$

which is the sum of the pixel-wise ~~bces~~ BCE between and $p_{i,j}$ and $y_{i,j}$.

This definition uses the plume Boolean map as target (truth, or label), and gives an equal weight to pixels with a high plume concentration and to pixels with a low plume concentration, which is questionable. Two Boolean plumes are shown in Fig. 4: the middle row images represent the transformation of the top row plumes into Boolean targets: images of 0 and 1 depending on whether the plume concentration of the pixel is greater than the threshold $\tau = 0.05\,$ppmv. These Boolean targets are visually far from representative of the plumes: the bulk of the signal, the mass of CO2, is contained in a much narrower area. In practice, this choice hinders convergence and deeply degrades the performance of the CNN, since many pixels with low plume concentration are difficult to detect. A threshold could be used to generate more representative Boolean targets, but due to the diversity of plume types, no universal threshold exists.

To overcome this problem, the pixel loss is weighted by a function proportional to the plume concentration of the pixel. The weight function, depending on the plume concentration in the pixel, is linear and is defined by

$$w(c) = \begin{cases} 1 & \text{if } c \leq y_{\min}, \\ w_{\min} + \dfrac{w_{\max}-w_{\min}}{y_{\max}-y_{\min}}(c - y_{\min}) & \text{if } y_{\min} < c < y_{\max}, \\ w_{\max} & \text{if } c \geq y_{\max}, \end{cases} \tag{3}$$

where $c \geq 0$ is the plume concentration of the pixel, $y_{\min} = 0.05\,$ppm. Furthermore, we choose to set $y_{\max}$ as the 99$^{\text{th}}$ percentile of the plume concentrations instead of the maximum to avoid outliers. With this weighting, the loss on a prediction field $p_{i,j}$ becomes

$$\mathcal{L}(\mathbf{P},\mathbf{Y}) = -\sum_{i,j}(w(c_{i,j})y_{i,j}\ln p_{i,j} + (1-y_{i,j})\ln(1-p_{i,j})), \tag{4}$$

where $c_{i,j}$ is the true plume concentration at pixel (i,j) and $y_{i,j}$ is a Boolean indicating whether a pixel is part of the plume or not. After preliminary sensitivity experiments (not illustrated here), $w_{\min}$ and $w_{\max}$ are set as 0.01 and 4. With these values, the model:

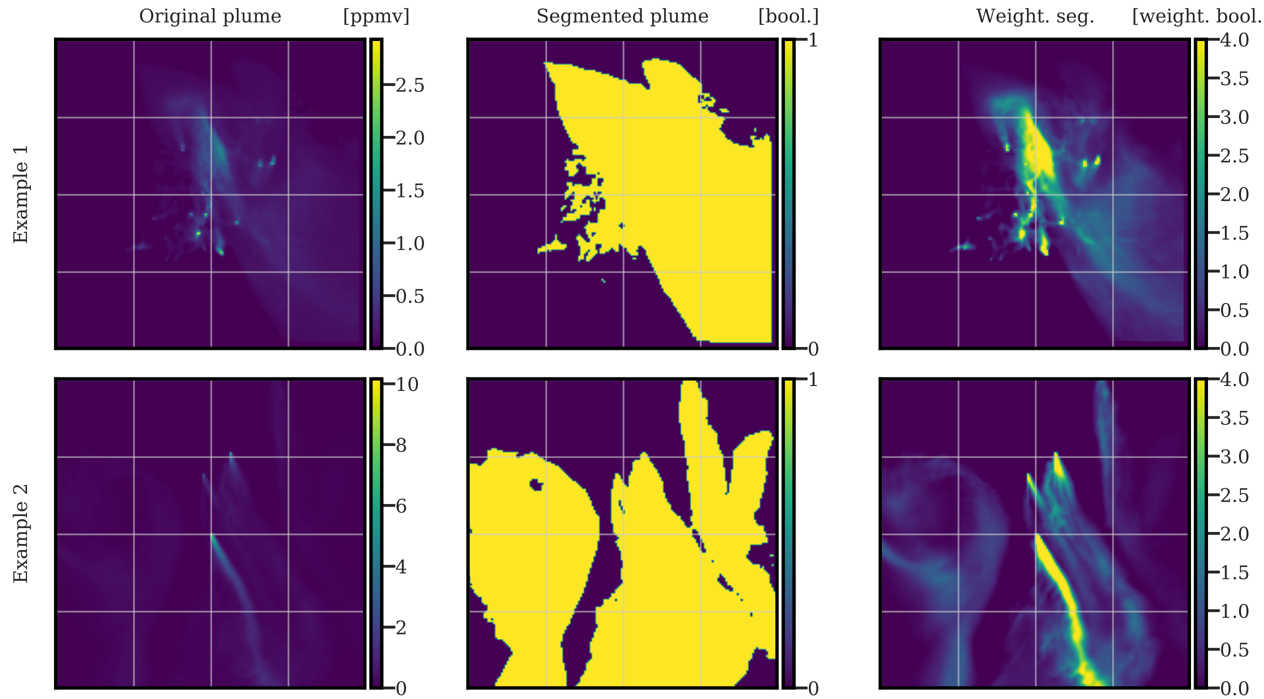

Figure 4. Examples of XCO_2 plumes (left), corresponding Boolean maps representing plume positions (middle) and weighted Boolean maps representing the plume positions (right). The weighting is calculated according to equation (3).

– is heavily penalised if it makes an error on a pixel with a high plume concentration;

– is penalised very little (even insignificantly) if it makes an error on a pixel associated with a low plume concentration;

– is moderately penalised if it makes an error on a non-plume pixel.

The result of this weighting can be observed on the right column of Fig. 4: each pixel is still a Boolean but weighted depending on the plume concentration of the pixel.

The new loss function is differentiable which is a necessary condition for the application of the gradient descent back-propagation algorithm. Moreover, the weighting is carried out independently for each field/plume pair, and not uniformly for the whole dataset. This latter choice would have penalised low-emission hotspots and favoured high-emission ones. This loss function is referred to as weighted ~~bce~~ BCE (weighted binary cross entropy) or WBCE in the following.

In practice, during the training phase, the plumes undergo a two-step transformation process. Firstly, they are transformed using the weight function described in Eq. 3. Subsequently, they undergo further transformation using the data augmentation techniques specified in section 2.2. The resulting transformed plumes are subjected to the loss function defined in Eq. 4 during training.

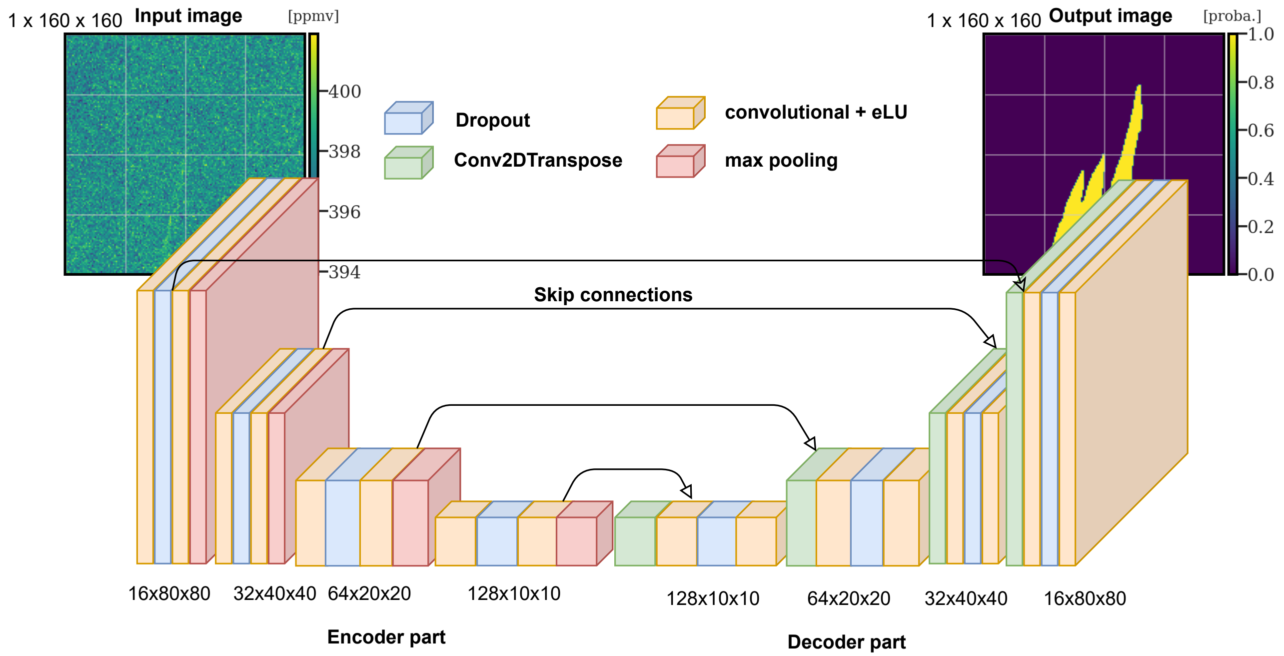

Figure 5. The XCO_2 field/plume pairs are fed into a U-Net that learns to distinguish the spatial features of the plume from the background.

3.3 U-net model

The deep learning model chosen to address this image-to-image problem follows the U-net architecture, a CNN encoder-decoder originally developed for biomedical image segmentation (Ronneberger et al., 2015), but later successively applied in many domains. This architecture is composed of (i) a downsampling or encoder phase where the resolution of the input image decreases and the number of feature channels increases, and (ii) of an upsampling or decoder phase where the resolution is increased to its original shape while the number of feature channels decreases symmetrically to the downsampling phase. The encoder captures and learns aggregated information locally, progressing until it captures close to the entire image. The encoder works in the same way as, for example, a classification model and can be built with any conventional CNN classifier. The decoder use the encoded information to build the output. The particularity of the U-net architecture is the use of skip connections where encoded layers are directly carried to the decoder part. In other words, the decoder part collects the high-resolution features from the encoder part through concatenation, to prevent any information loss.

Many encoder and decoder architectures can be used: an example is illustrated in Fig. 5. Such a U-net is build on top of convolutional layers that locally aggregate the information. Furthermore, the encoder uses maxpooling layers which decrease the resolution of the image, while the decoder resorts to upsampling layers. Finally, dropout layers are used to reduce overfitting. However, in this paper, we use a generalised architecture, not shown for the sake of readability since more than 270 layers and 5 million parameters are used. The encoder used is the EfficientNetB0 CNN architecture (Tan and Le, 2020) which is built with specific convolution layers (based on depth-wise convolutions) and a squeeze-and-excitation optimisation. Several encoders have been considered and tested: ResNet (He et al., 2015), DenseNet (Huang et al., 2018) and self-made alternatives. The decoder phase is a repetition of convolution and upsampling layers.

A dropout rate of 0.2 is used in the encoder part. The activation layers in the encoder part are swish functions (Ramachandran et al., 2017), whereas relu functions are chosen for the decoder part. The normal kernels are chosen to initialise convolutional layers to avoid vanishing or exploding gradients during the first epochs. To get a probability map, the final output is activated by a sigmoid function. We use an initial learning rate of 10^{-3} with Adam optimiser and a *reduce on plateau* strategy after considering different configurations. The batch size is set to 32 samples, and the number of epochs is set to 500 which ensures the convergence of learning. The final model weights are the best performing weights on the validation dataset.

3.4 Training, validation, and test datasets

The complete dataset is divided into training, validation and test subsets. Since a plume at a certain time strongly resembles the plume of the next hour, the validation and test sets consist of subsets of plumes on two consecutive days. For a given month, the test dataset always consists of the plumes of the 4^{th}, 5^{th}, 15^{th}, 16^{th} days of the month. The training, validation and test datasets are used to train the model, to tune its hyperpameter and to test the optimal model, respectively.

The numbers of data for training, validation and test differ for each test case. In the last case (extrapolation to Berlin), there are about 23,000 images in the training dataset, 4,000 in the validation dataset and 7,000 in the test dataset. It is worth mentioning that data augmentation techniques enable us to use a significantly greater number of training images in practice.

The input XCO_2 fields are standardised using the mean and variance over all pixels of all images of the training data set. All the results of the following section 4 were obtained on the test dataset, unobserved until the final evaluation. Furthermore, the results are obtained on a non augmented test dataset since WBCE metrics of the model on an augmented or non-augmented test datasets are similar and we are primarily interested in segmenting non-augmented images.

4 Applications

To evaluate the performances of the CNN plume segmentation, two alternative segmentations to compare to (hereafter called references) are described in section 4.1. Then, the U-net algorithm is trained and tested in two configurations.

In section 4.2, the first configuration, we investigate the ability of the U-net to generalise to new data from the same region. The U-net is trained and tested on pairs of XCO_2 and plume images in Grand Paris, IdF, Berlin, Lippendorf and in plume clusters centred at Jänschwalde or Boxberg. Several training setups are considered: the CNN is trained either on all available data or only on data from one location.

In section 4.3, the second configuration, we investigate the ability of the U-net to extrapolate on unseen data from another area. The U-net is trained on $\{$Grand Paris, IdF, Jänschwalde, Lippendorf, Boxberg$\}$ images and tested on Berlin images.

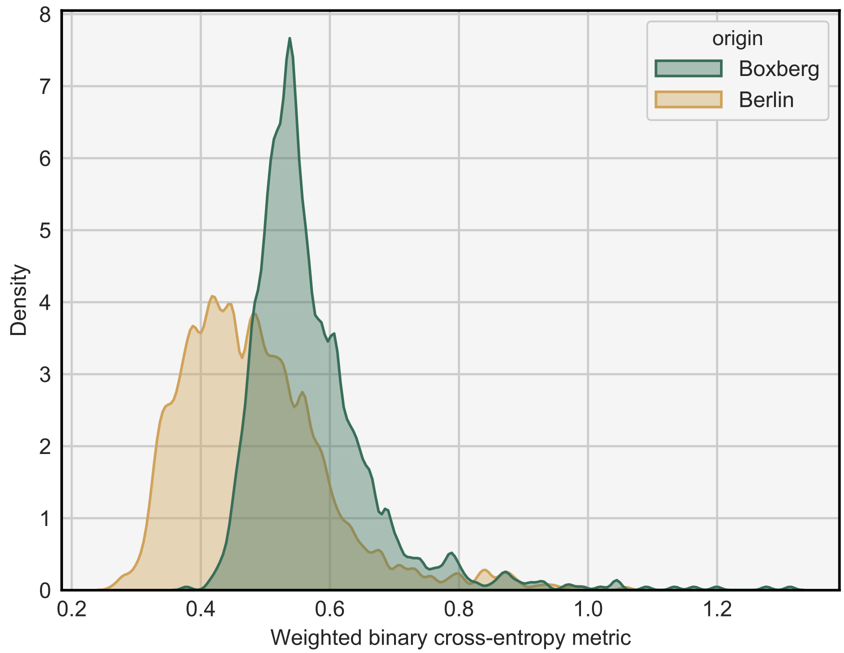

Figure 6. Histograms of the ~~weighted bce~~ WBCE scores over all images in Berlin and Boxberg obtained with the neutral reference.

4.1 Alternative segmentations to compare to

4.1.1 Neutral reference

Two references are considered to assess the quality of the CNN segmentation through their WBCE scores. First, we use a constant probability map as a first reference, which is in practice equivalent to a prediction of non-segmentation of the plume. Since the WBCE metric only deals with probabilities (rather than Boolean values) this constant value is a probability and must be chosen. For each hotspot dataset, this probability is found as the one minimising the WBCE over all images of that hotspot with a differential evolution algorithm (Virtanen et al., 2020; Storn and Price, 1997). In practice, for each hotspot, the calculated probability is close to 0.15-0.2. This first segmentation reference output is called *neutral reference* in the following. Figure 6 shows the histograms of the WBCE computed on the plume cluster centred at Boxberg and the Berlin plume with respect to the neutral reference. Large variations can be observed: the neutral WBCE score for the Berlin images varies between 0.25 and 1. This means that the score associated with a segmentation is very dependent on the image considered: for one image, a score of 0.25 corresponds to a good segmentation and for another, such a score is equal to the neutral score and therefore equivalent to the absence of plume detection. Therefore, in the following, to make the segmentation scores more consistent over the samples, the WBCE of an image is systematically divided by its WBCE obtained with the neutral reference segmentation.

This new metric is called NWBCE (normalised weighted binary cross entropy metric) and a score of 1 means that the resulting segmentation is no better than "no detection".

4.1.2 A segmentation technique based on thresholding: ddeq

The second reference to be compared with our segmentation method is the detection algorithm implemented in the Python package for data-driven emission quantification (ddeq[2]). This algorithm can be described as a thresholding method: it first detects signal enhancements that are significant in relation to instrument noise and background variability, and then identifies plumes as coherent structures (Kuhlmann et al., 2019a, 2021). Since the algorithm returns a Boolean map, the identified non-plume and plume pixels (0 and 1) are mapped to two values, defined independently for each hotspot. These two values are chosen so as to minimise the WBCE over all images from the hotspot. Figure 7 shows four applications of the ddeq algorithm to the CO_2 images (two plume clusters centred in Boxberg, two in Berlin). The first plume cluster (centred in Boxberg) image (1st row) obtains a much better NWBCE (0.79) than the second example (0.97) because the plume signal to background ratio is much higher. The same is true for the Berlin plume segmentations. Furthermore, due to the low signal to noise ratio, no plume is detected on the fourth example and a constant probability map is returned (which gives a score close to the neutral and not 1, because the mapped probability is different). The thresholding method allows the segmentation of plumes, or portions of plumes, associated with signals above the background. But if no visible signal above the background is detected, the plume is not identified.

4.2 Generalisation on new data from the same region

4.2.1 Choice of the training dataset

In this section, we investigate the performance of the U-net when trained and tested on data from the same region. We consider two ways to train the model: to train to segment plumes in images of a given location, the U-net can either exploit only the XCO_2 field/plume pairs from that specific location or the XCO_2 field/plume pairs from all available locations. The two approaches yield different results, as summarised in table 2.

In the case of Berlin and IdF, the two training setups give approximately the same results for the mean and median. In the case of the Grand Paris and, to a lesser extent, Lippendorf, using additional training data improves the quality of the results. For the Paris fields this might come from a lack of data – only three months. By contrast, in the Jänschwalde and Boxberg cases, using additional data degrades the results. This is most likely due to the fact that these two areas are characterised by multiple plumes (on the same image). In the following sections, we present the predictions for the best training configuration: in the case of Berlin, Grand Paris, IdF, and Lippendorf, the U-net trained on all available data is used and in the case of Jänschwalde and Boxberg, the U-net trained with a restricted dataset.

[2] https://gitlab.com/empa503/remote-sensing/ddeq)

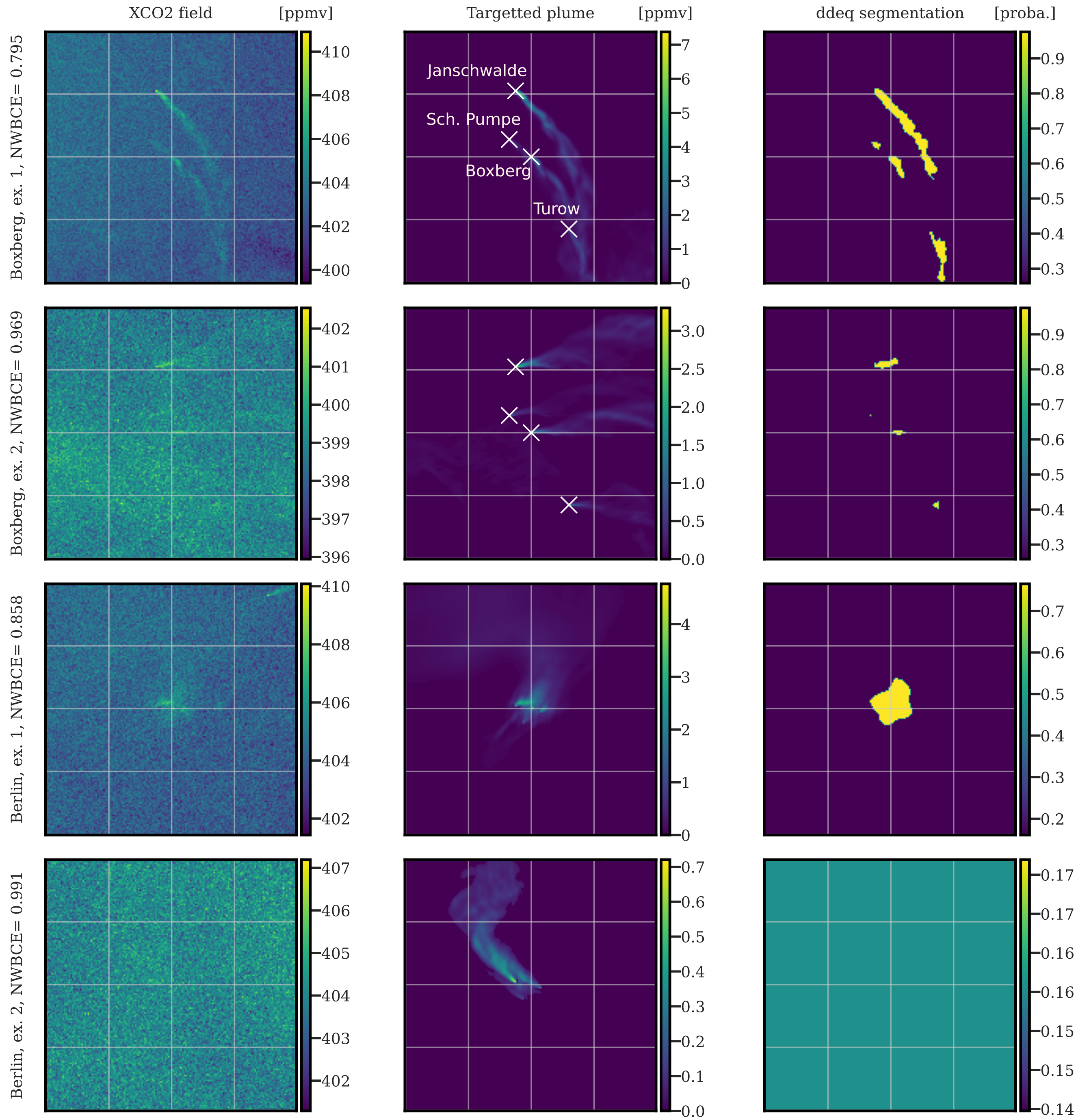

Figure 7. Four examples of ddeq plume segmentation algorithm applications on simulated satellite images centred at Boxberg (first two rows) or Berlin (last two rows). The first column corresponds to XCO_2 simulated satellite images in ppmv, the second column to the targeted plumes, and the third column to the predictions of the ddeq segmentation algorithm mapped to probability images. All times given on the left of the figures are in UTC.

Test / Train location	Same region		All data	
	Mean	Median	Mean	Median
Berlin	0.45	0.35	0.46	0.36
IdF	0.66	0.54	0.60	0.57
Grand Paris	0.68	0.53	0.54	0.48
Lippendorf	0.66	0.58	0.61	0.52
Jänschwalde	0.39	0.31	0.58	0.55
Boxberg	0.40	0.33	0.55	0.47

Table 2. Normalised weighted-bce (NWBCE) mean-median over the XCO_2 field/plume pairs (i.e. overall model performance) of a certain region in two situations: the U-net is trained on fields from the same region, or on all available data. The lower the score the better.

4.2.2 Score histograms

Figure 8 presents the kernel density estimates of the NWBCE scores of the U-net and ddeq segmentation methods according to the origin of the XCO_2 field/plume pair. As a general rule, the lower the score, the better the segmentation. As shown in the following examples, scores between 0 and 0.5 correspond usually to very good to good segmentation, and scores between 0.5 and 0.8 to 0.9 to non-perfect but usable segmentation. A score of 1 is neutral (neither worse nor better than predicting no plume) and a score above 1 corresponds to a worse segmentation than the neutral , i.e. a segmentation of the wrong part of the image. A number of applications with scores are presented in the following.

On all hotspots, our deep learning model consistently outperforms the ddeq segmentation method on the NWBCE metric. For example, the average NWBCE over all Berlin images is 1.0 for the neutral (by definition), 0.95 for the ddeq method, and 0.44 for the CNN segmentation. The average NWBCE is over all Janschwälde images 0.90 for the ddeq method and 0.40 for the CNN method. Note, however, that the CNN is optimised on the NWBCE metric whereas the ddeq segmentation method is not. The choice of a metric is to some extent arbitrary and the difference between the two methods would change if another metric, and/or another definition of the plume, were chosen.

The best segmentation scores are obtained on Jänschwalde and Boxberg, which is consistent with the fact that these images contain several plumes of high intensity. The histogram of the Lippendorf NWBCE metric shows overall very good results but with a large variance, and a significant part of the scores above 1. The distribution of the Berlin fields has a wider tail than that of the power plant fields: this can be explained by the shape of the city's plumes, which are generally more complex and therefore more difficult to segment than the straight power plant plumes. The poorer results over Grand Paris and IdF on average are due to the smaller amount of available images and the low SNR of Grand Paris plumes. The small plumes specific

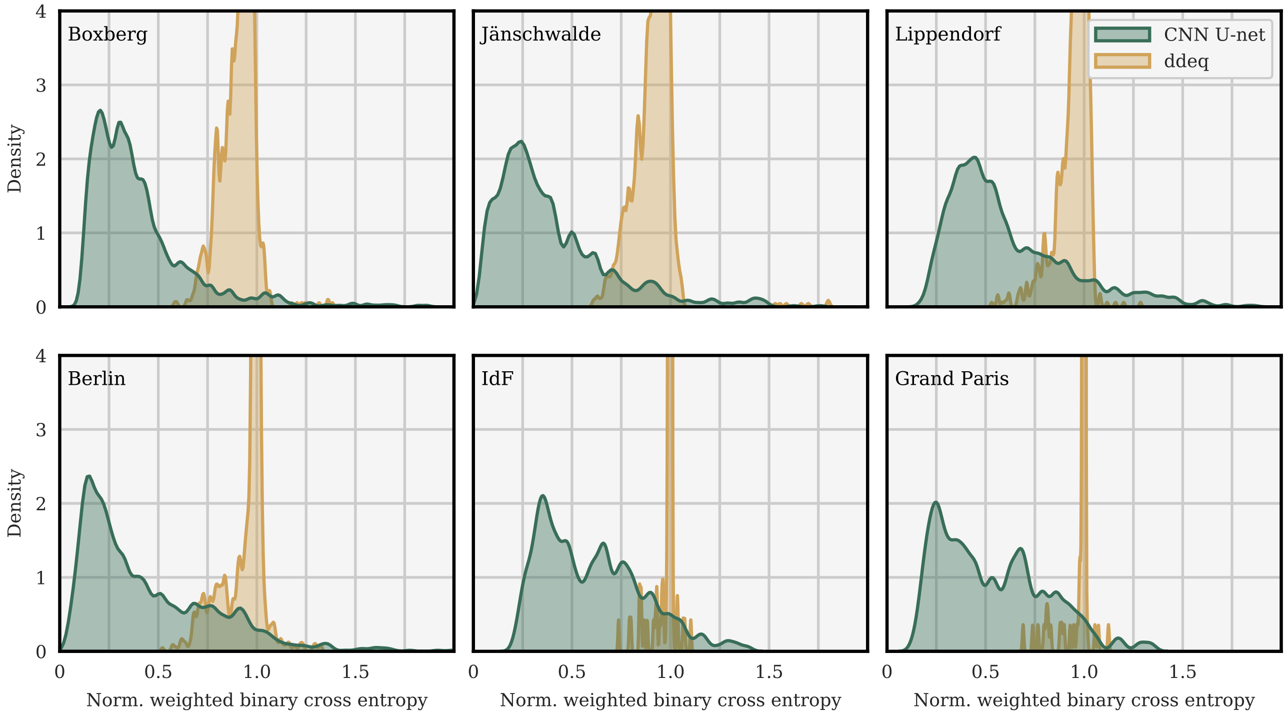

Figure 8. Kernel densities of the histograms of the NWBCE scores of the U-net and ddeq segmentation techniques over all images of the various geographical domains.

to IdF (outside Grand Paris) are almost never recovered as shown in the IdF histogram, similar to the Grand Paris histogram but slightly shifted to the right.

4.2.3 Berlin region predictions

In Fig. 9, we present four typical Berlin plume segmentations with the U-net algorithm. The XCO_2 images (left column) are fed into the CNN which yields the segmentation probability maps (right column) of the XCO_2 plumes. For the classical binary cross-entropy metric, a pixel with a probability equal to 0.5 means a "no-information" on the class of the pixel (plume or non-plume). We assume that this can be extended to the WBCE metric: all pixels with a probability greater than 0.5 can be considered as part of a plume, while pixels below 0.5 can be considered as pixels that are not part of a plume. Consequently, a divergent (at 0.5) colour map is used to represent the CNN model predictions. The middle images are the weighted Boolean maps transformations (according to equation (3)) of the actual plumes for comparison. The four images in order are illustrative of the four quartiles in terms of performances, respectively (according to their NWBCE score).

The first and second rows show a very accurate segmentation: the model predicts the correct direction, shape and thickness of the plume. The third plume is rather well recovered with some inaccuracies: in particular, the tail of the plume is reconstructed

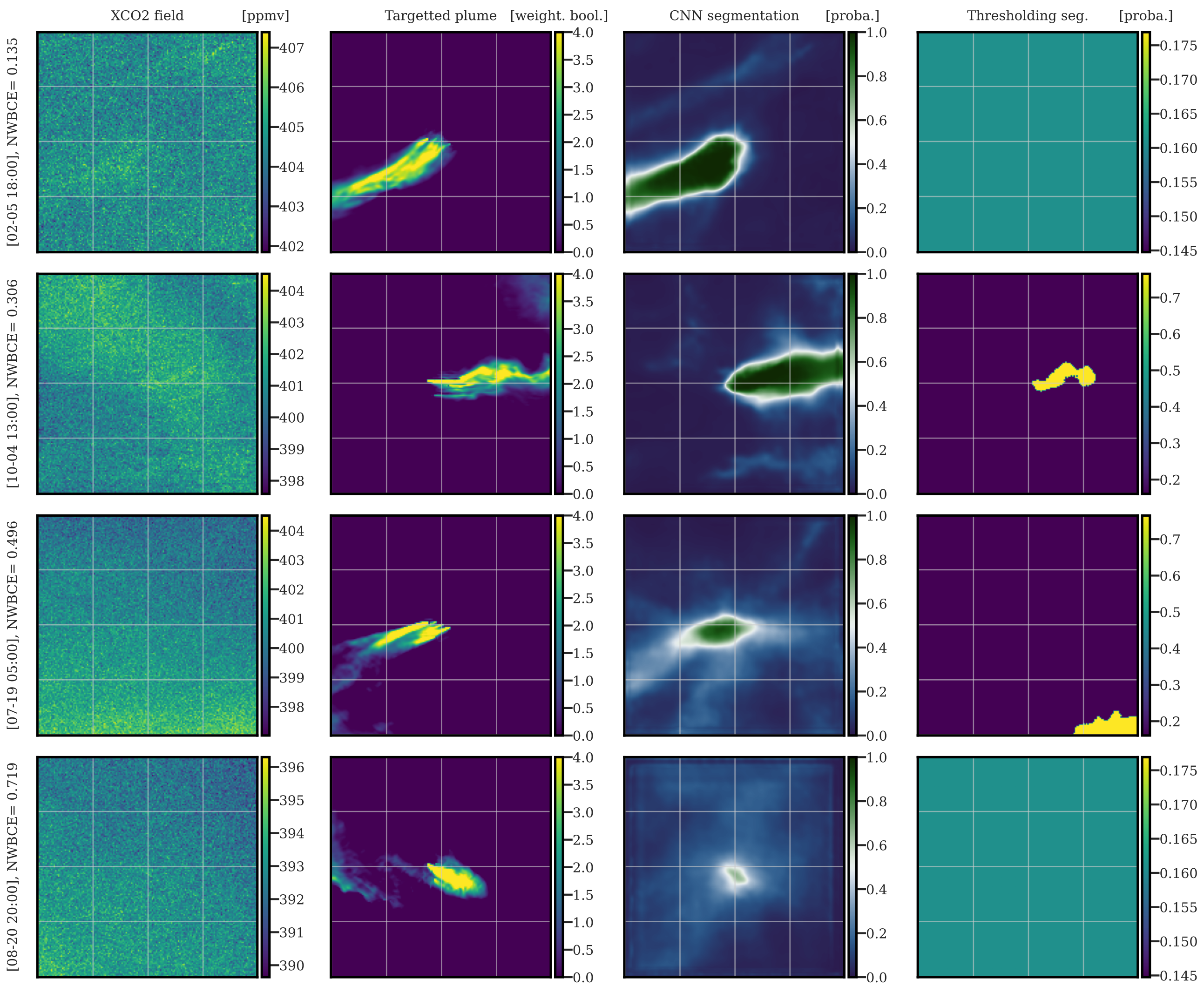

Figure 9. Examples of the application of U-net on images in the Berlin region. The first, second and third columns correspond to XCO_2 images of Berlin, weighted Boolean plumes and CNN predictions as probability maps, respectively. The fourth column shows the application of the detection algorithm implemented in the Python package ddeq. The first, second, third and fourth rows are representative of the first, second, third and fourth quartiles of NWBCE scores, respectively. All times given on the left of the figures are in UTC.

with less accuracy, which was expected since the concentrations on the tail reach very low values. Moreover, the core of the plume is segmented with less confidence: the probabilities of the plume pixels are close to 0.8. In general, prediction confidence is positively correlated with the NWBCE score. Similarly, uncertainty, represented by the number of pixels close to 0.5, and the NWBCE score are inversely correlated. To a certain extent, this is true for all hotspots and is a measure of model uncertainty. It can also be used in evaluations without access to the truth to quantify how certain the predictions are. Confirming this correlation, the fourth row shows a very uncertain prediction that still correctly finds the direction and core of the plume. In all images, the position of the plume origin is always accurate. This is not trivial because in the training set horizontal and vertical shifts are used, which means that the plume origin is not known in advance by the model. We note that the plume is often masked by background variability and instrument noise, which does not prevent its detection by the CNN.

4.2.4 Plume cluster centred in Boxberg predictions

In Fig. 10 are shown the segmentation of four images centred at Boxberg power plant. Sources are from bottom to top of the images: Turow, Boxberg, Schwarze Pumpe and Jänschwalde. The four images in order are representative of the four quartiles of their NWBCE score, respectively. All first three segmentations are very accurate: the origins, thicknesses and directions of the plumes are accurately reconstructed. Some failures are the mixing of the two plumes in the centre of the first image, the no detection of the Schwarze Pumpe plume in the second image, or the wrong evaluation of the direction of the Turow plume in the third image. The fourth segmentation high NWBCE is mainly due to the addition by the model with a high probability of a ghost plume on the left of the image: a clear enhancement on the XCO_2 field at the same location explains the U-net error. The absence of power plants or major cities in the area raises questions on the origin of this enhancement.

4.2.5 Lippendorf predictions

In Fig. 11, we show the segmentation of four images centred at Lippendorf power plant. The four images from top to bottom are illustrative of the four quartiles of their NWBCE score. First two XCO_2 images are well segmented: the origins, thicknesses and directions of the plumes are retrieved by the CNN model. The third row shows a strange behaviour of the plume which is not well anticipated by the model: the plume returns on itself. The fourth line shows a very poor recovery with a score of less than 1: the Lippendorf plume is in fact well segmented but the residuals of other plumes are not, resulting in a significant error. This problem is at the origin of a large part of the errors on Lippendorf images and could be solved by using a better loss/metric that would take into account the position of the source. A complementary study on the overall performance of the model can be found in the supplementary material.

4.2.6 Paris predictions

In Fig. 12, we present four typical IdF plume segmentations, with or without the IdF specific plumes, with the U-net algorithm. The four images in order are representative of the four quartiles of their NWBCE score, respectively. The first three images show segmentations that reconstruct the direction and origin of the plume. However, the thickness is less and less well defined

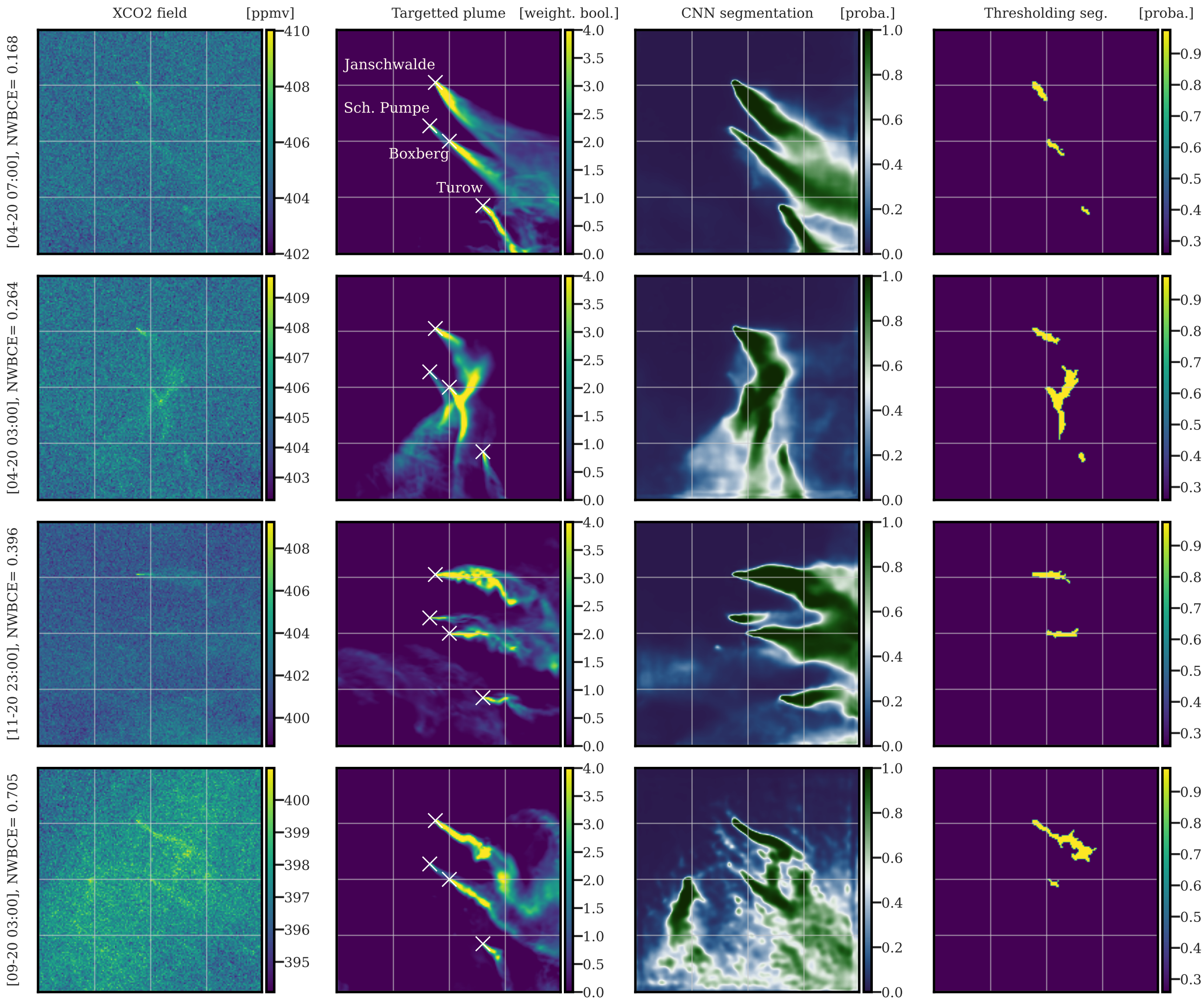

Figure 10. Examples of the application of U-net on images centred at Boxberg. Sources are from bottom to top in each image: Turow, Boxberg, Schwarze Pumpe and Jänschwalde. The first, second and third columns correspond to XCO_2 images, weighted Boolean plumes and CNN predictions as probability maps, respectively. The fourth column shows the application of the detection algorithm implemented in the Python package ddeq. The first, second, third and fourth rows are representative of the first, second, third and fourth quartiles of NWBCE scores, respectively. All times given on the left of the figures are in UTC.

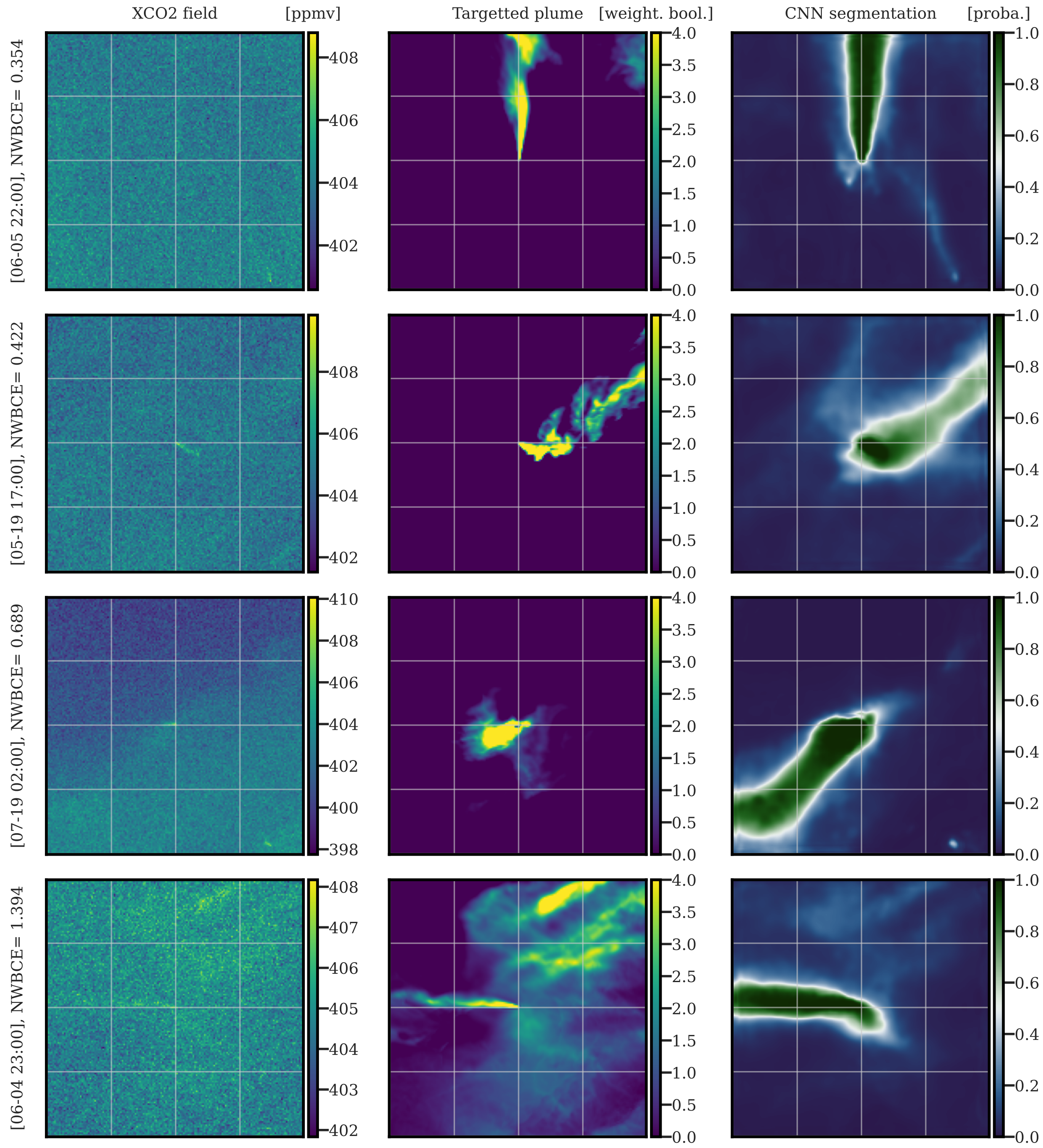

Figure 11. Examples of the application of U-net on images centred at Lippendorf. The first, second and third columns correspond to XCO_2 images, weighted Boolean plumes and CNN predictions as probability maps, respectively. The first, second, third and fourth rows are in the first, second, third and fourth quartiles of NWBCE scores, respectively. All times given on the left of the figures are in UTC.

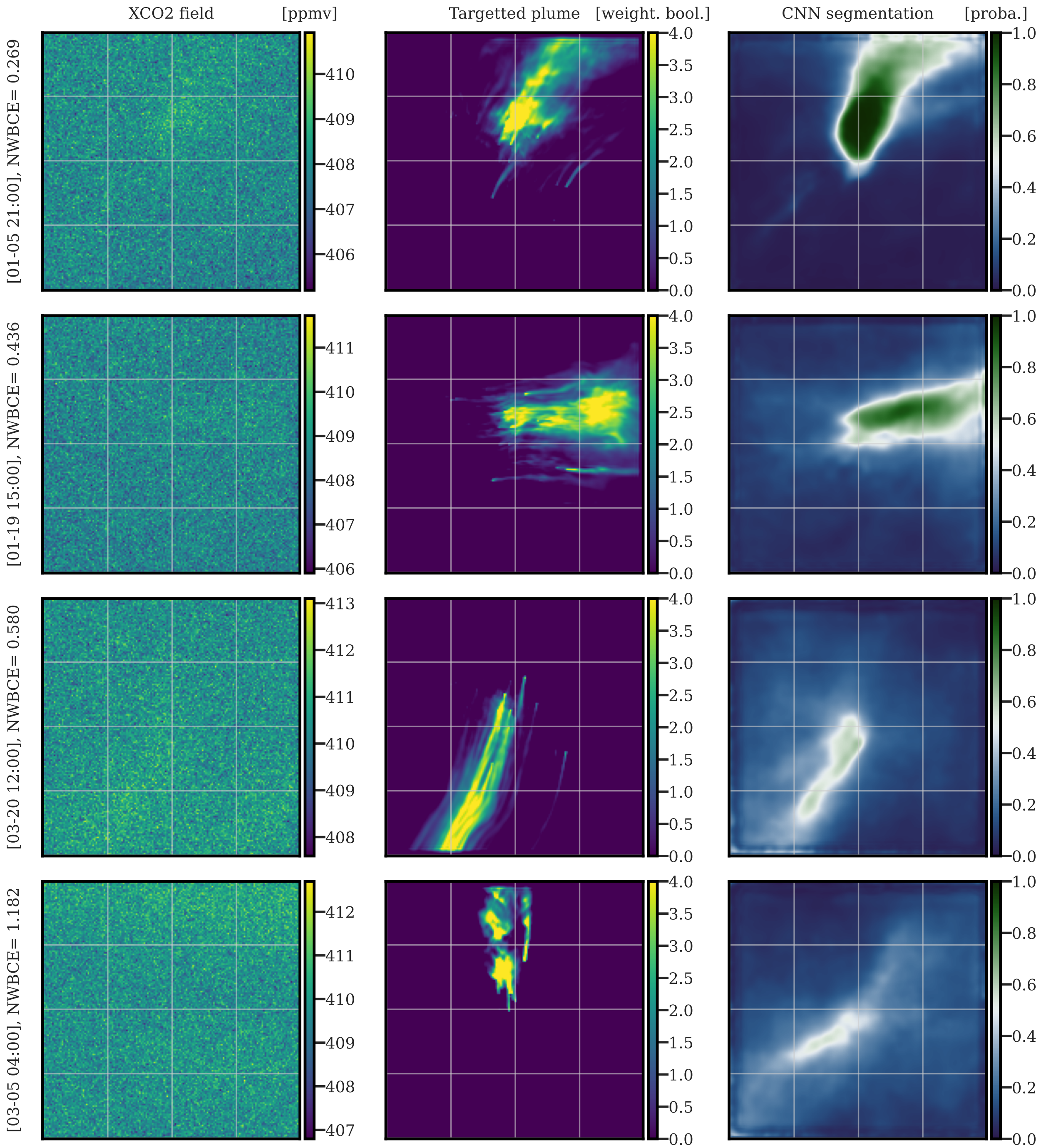

Figure 12. Examples of the application of the U-net on images of IdF. The first, second and third columns correspond to XCO_2 images of IdF, weighted Boolean plumes and CNN predictions as probability maps, respectively. The first, second, third and fourth rows are representative of the first, second, third and fourth quartiles of NWBCE scores, respectively. All times given on the left of the figures are in UTC.

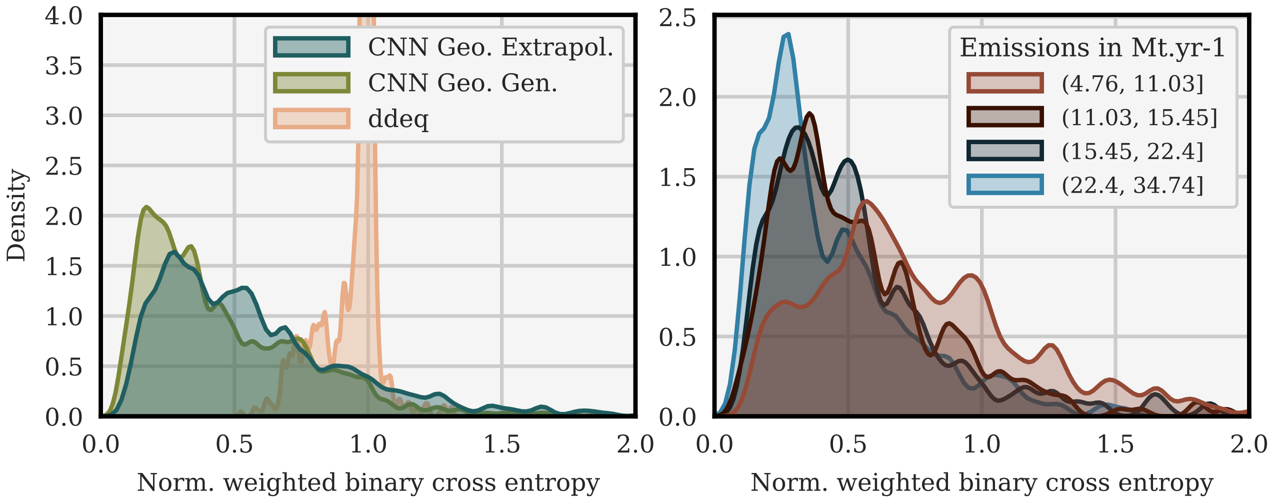

Figure 13. Histograms of the NWBCE image scores over all test images of Berlin. Plumes/segmentations are classified depending on the method used (left). In the geographical extrapolation case, the CNN segmentation outputs are classified in four equivalent clusters according to the corresponding emissions (right).

as the NWBCE increases. The other potential plumes outside Paris are systematically missed by the model because of their too low concentration. Moreover, as the NWBCE increases, the pixel-plume predictions are closer and closer to 0.5 showing the hesitations of the model. Finally, the plume in the last image is completely missed and the model makes no predictions above 0.55, expressing its inability to find the plume.

4.3 Extrapolation on unseen data from another region

In this section, we investigate the performance of the U-net when trained and tested on data from different regions. Such a task is more difficult than generalising on plumes from the same region, where the training and test sets have more similarities from the local meteorological and pollution climatology. To study the potential for extrapolation, the U-net model is trained on the Paris, Jänschwalde, Boxberg and Lippendorf fields and tested on the Berlin fields. Berlin is chosen because cities are a particularly complicated case as their signal is lower and because, in this way, we can rely on a large set of images to validate and test the CNN model.

Figure 13 shows the histograms of the NWBCE scores for all Berlin test images depending on the method used (left), and, with the CNN method in the case of geographical extrapolation, of all NWBCE scores for several ranges of the Berlin emission rate at the time of the image (right). The mean NWBCE score of all prediction-truth pairs is 0.57 and the median is 0.49 – higher in both cases than when the model is trained on Berlin images only (see Section 4.2), but still very satisfying: the model extrapolates well and outperforms the ddeq segmentations according to the NWBCE metric. In addition, the main divergence between the generalisation and extrapolation histograms is a shift to the right of the part of the histogram between the scores of 0 and 0.75. It is explained later in this section that segmentations with an NWBCE below 0.75 are generally good enough for inversion. In other words, the switch from generalisation to extrapolation mainly degrades highly accurate segmentations to "only" accurate segmentations.

Plumes that we assess are the consequence of a variety of emission levels. For example, Berlin emissions range from approximately 4 to 35 Mt/yr. On the right histogram, it can be observed that the results, quite naturally, deteriorate in the case of low-emission plumes: for high-emission plumes, the density peaks at 0.25 whereas it peaks at 0.5 in the case of low-emission plumes. The variance of the low-emission plumes NWBCE metric density is also significantly higher.

In Fig. 14, we present four typical Berlin plume segmentations with the U-net algorithm. The four images from top to bottom are illustrative of the four quartiles, respectively (according to their NWBCE score). The first three images show segmentations that recover the direction and origin of the plume. The thickness of the plume is also well reconstructed in the first two examples but a part of the plume is missed in the third example, which gives the largest fraction of the error; this miss is probably due to a gradient in the background field. The second and third examples also show a significant number of pixels at values around 0.5, expressing the uncertainty of the model. In these examples, the plume is masked by background variability and instrument noise, yet is detected by the CNN. On the fourth example, the deep learning model fails to detect the plume and yet diagnoses higher uncertainty. Further analysis shows that most of the last quartile results are difficult to use for inversion because they partially or completely miss the plume, or express too much uncertainty.

5 Conclusions

The future availability of satellite images of CO_2 columns, such as the Copernicus CO_2 Monitoring (CO_2M) mission, opens up new possibilities for the assessment of local CO_2 emissions. Emissions can be assessed from CO_2 plumes of hot spots in the satellite images ~~(Nassar et al., 2017)~~(Nassar et al., 2017, 2022). This data-driven assessment needs to detect plumes from satellite images, which is difficult for thresholding method due to the low signal-to-noise ratio of the plume. Deep learning and convolutional neural network (CNN) techniques could provide more accurate plume detection, because of their ability to learn and capture plume-specific spatial patterns, which do not necessarily depend on a significant concentration enhancement.

In this paper, we evaluate the ability of CNNs to accurately detect the mask of a plume in an XCO_2 satellite image using simulated CO_2 fields. Each synthetic XCO_2 image is the sum of the anthropogenic plume of a major hotspot (a city or a power plant), a background from other biogenic and anthropogenic fluxes, and a random Gaussian noise to simulate the satellite instrumental errors.

Our plume detection model is based on a CNN encoder-decoder, the U-net algorithm with an EfficientNetB0 backbone. It is an image-to-image model, which transforms the full XCO_2 field into a map showing the positions of the anthropogenic emission plumes. For training, we develop a novel loss function that penalises more the errors made on pixels associated with high plume concentrations and thus yields a more accurate definition of a plume than a simple threshold value. This CNN is trained and tested in two contexts. First, the capacity of the model to generalise on unseen data from the same region is

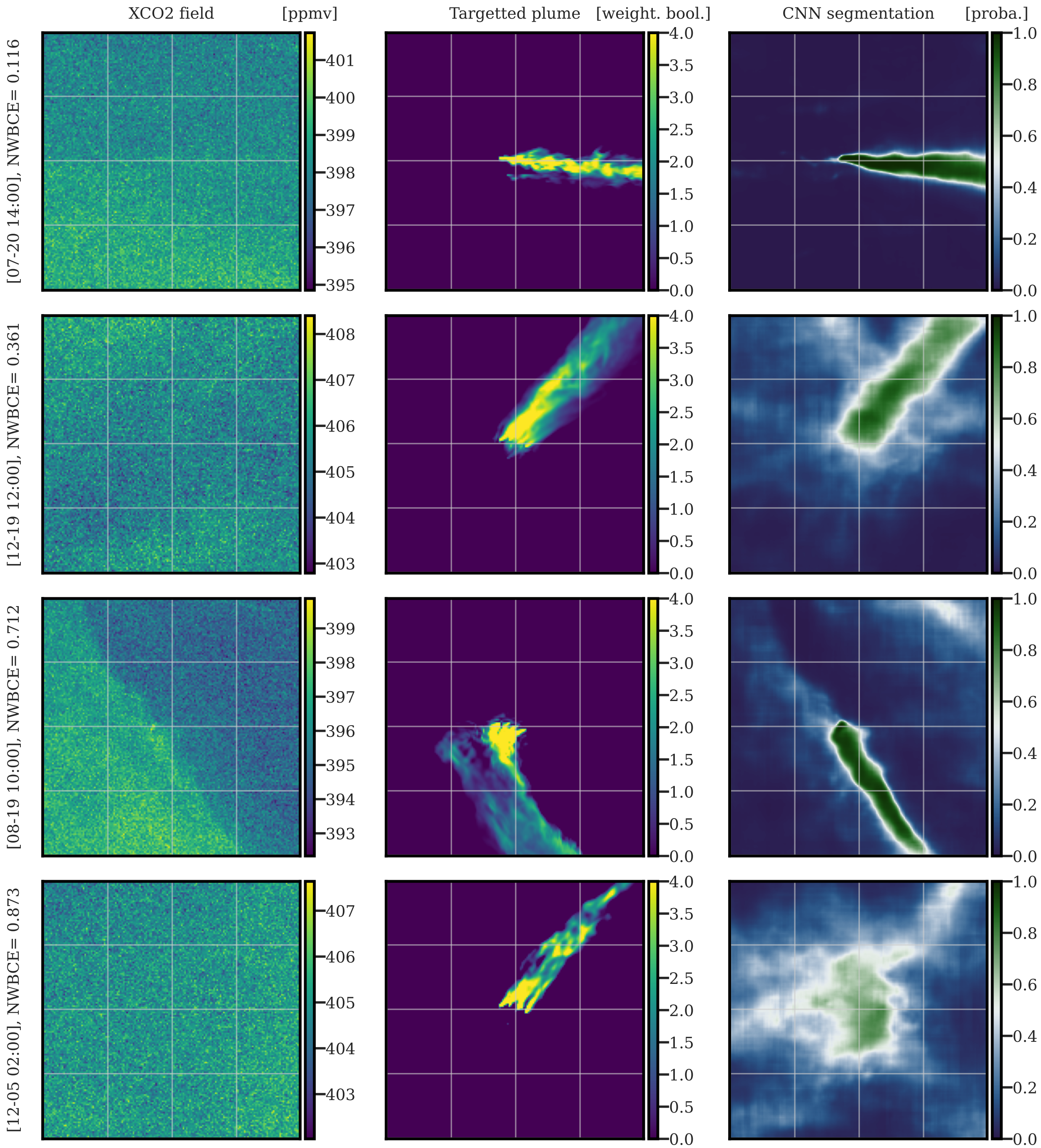

Figure 14. Examples of the application of the U-net on images of Berlin. The first, second and third columns correspond to XCO_2 images of Berlin, weighted Boolean plumes and CNN predictions as probability maps, respectively. The first, second, third and fourth rows are representative of the first, second, third and fourth quartiles, respectively. All times given on the left of the figures are in UTC.

evaluated. The U-net shows very good performance: most plumes are precisely segmented, the origin, thickness and shape of the plume being often accurately retrieved. Second, we evaluate the ability of the model to extrapolate to unobserved data from another region. Specifically, the model is trained with simulated fields of Paris and power plants and tested with fields in the Berlin area. The segmentations are slightly less accurate than in the first context, but are nevertheless very satisfactory: about half of the Berlin plumes are accurately segmented, with plume shape, thickness, direction and origin recovered, and 75% of the segmentations are accurate enough to be used for inversion.

The observed good performance of the U-net architecture is due to the ability of the convolution layers to capture detailed spatial patterns corresponding to plumes even when the concentrations of these plumes are partially covered by high satellite noise or background variability. It allows the model to outperform segmentations by the thresholding technique according to the concentration-weighted metric, whether the model trained on some data is tested on data from the same region or not. The U-net is effective over a wide variety of plumes (cities, power plants, diverse regions, several levels of hotspot emissions). Its training time is less than one day, while once the model trained, the evaluation of a new image is less than a second. However, although the model performs better when trained and tested on data from the same region, it would be too expensive to generate simulations on all the cities and power plants whose plumes we wish to segment. Therefore, we believe that the goal is the development of a "universal" CNN, trained only on a limited sample of cities and power plants and highly efficient on all of them. The model in this paper, trained on Paris and power plants data and tested on Berlin, already shows accurate and very satisfactory segmentations of the Berlin plumes, but the results need to be confirmed on multiple cases.

It is very likely that many other techniques could be applied to improve these segmentations, which could be based on:

- more advanced and powerful NN architectures such as transformers or on CNN networks with more parameters;
- improving the distribution of the data, by increasing the amount of images used, or by using more carefully chosen augmentation techniques.

For all these reasons, CNN methods appear to be very suitable for CO_2 plume segmentation problems on satellite data. However, the model was evaluated on simulated data which does not take into account all the problems of plume detectability presented by real satellite images in particular clouds and patterns of systematic errors due to surface reflectance and aerosol dependency of the retrievals. Consequently, the method needs to be extended and validated on full OSSEs: fields with clouds and satellite swaths taken into account, and afterwards on real satellite data.

Finally, as the CO_2 emission rate is proportional to the mass of the corresponding plume, accurate plume segmentation should lead to an accurate emission estimate. The reliability and accuracy of the CNN model segmentations suggest that a well-trained CNN fed by these segmented plumes could be a very efficient hotspot estimator.

Data availability. The datasets used in this paper are available on a compliant repository on on 10.5281/zenodo.4048228 for the SMART-CARB data (Berlin, and German power plants). The code is available on Github at https://github.com/cerea-daml/co2-images-seg.

Author contributions. Joffrey Dumont Le Brazidec: Conceptualisation, Methodology, Software, Investigation, Formal analysis, Visualisation, Resources, Project administration, Writing - Original Draft; Pierre Vanderbecken: Investigation, Formal analysis, Writing - Review; Alban Farchi: Conceptualisation, Methodology, Project administration, Writing - Review; Marc Bocquet: Conceptualisation, Methodology, Project administration, Funding acquisition, Writing - Review; Jinghui Lian: Resources, Writing - Review; Grégoire Broquet: Resources, Writing - Review; Gerrit Kuhlmann: Resources, Writing - Review; Alexandre Danjou: Resources, Writing - Review; Thomas Lauvaux: Resources

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. This project has been funded by the European Union's Horizon 2020 research and innovation programme under grant agreement N° 958927 (Prototype system for a Copernicus CO_2 service). CEREA is a member of Institut Pierre Simon Laplace (IPSL). The authors would like to thank Hugo Denier van der Gon for the TNO inventory and Élise Potier for her implication in the constitution of the Paris dataset.

References

Agustí-Panareda, A., Massart, S., Chevallier, F., Boussetta, S., Balsamo, G., Beljaars, A., Ciais, P., Deutscher, N. M., Engelen, R., Jones, L., Kivi, R., Paris, J.-D., Peuch, V.-H., Sherlock, V., Vermeulen, A. T., Wennberg, P. O., and Wunch, D.: Forecasting global atmospheric CO_2, Atmos Chem Phys, 14, 11 959–11 983, https://doi.org/10.5194/acp-14-11959-2014, publisher: Copernicus GmbH, 2014.

Brunner, D., Kuhlmann, G., Marshall, J., Clément, V., Fuhrer, O., Broquet, G., Löscher, A., and Meijer, Y.: Accounting for the vertical distribution of emissions in atmospheric CO_2 simulations, Atmos Chem Phys, 19, 4541–4559, https://doi.org/10.5194/acp-19-4541-2019, publisher: Copernicus GmbH, 2019.

Butz, A., Scheidweiler, L., Baumgartner, A., Feist, D. G., Gottschaldt, K.-D., Jöckel, P., Kern, B., Köhler, C., Krutz, D., Lichtenberg, G., Marshall, J., Paproth, C., Slijkhuis, S., Sebastian, I., Strandgren, J., Wilzewski, J. S., and Roiger, A.: CO2Image: a next generation imaging spectrometer for CO2 point source quantification, Tech. Rep. EGU22-6324, Copernicus Meetings, https://doi.org/10.5194/egusphere-egu22-6324, conference Name: EGU22, 2022.

Chevallier, F.: Validation report for the inverted CO2 fluxes, v18r1 – version 1.0, Copernicus Atmosphere Monitoring Service, p. 20, https://atmosphere.copernicus.eu/sites/default/files/2019-01/CAMS73_2018SC1_D73.1.4.1-2017-v0_201812_v1_final.pdf, 2018.

Chollet, F.: Deep Learning with Python, Manning Publications Co., USA, 1st edn., 2017.

Crisp, D., Pollock, H. R., Rosenberg, R., Chapsky, L., Lee, R. A. M., Oyafuso, F. A., Frankenberg, C., O'Dell, C. W., Bruegge, C. J., Doran, G. B., Eldering, A., Fisher, B. M., Fu, D., Gunson, M. R., Mandrake, L., Osterman, G. B., Schwandner, F. M., Sun, K., Taylor, T. E., Wennberg, P. O., and Wunch, D.: The on-orbit performance of the Orbiting Carbon Observatory-2 (OCO-2) instrument and its radiometrically calibrated products, Atmos. Meas. Tech., 10, 59–81, https://resolver.caltech.edu/CaltechAUTHORS:20180212-091712957, 2017.

Denier van der Gon, H., Dellaert, S., Super, I., Kuenen, J., and Visschedijk, A.: VERIFY : Observation-based system for monitoring and verification of greenhouse gases, https://cordis.europa.eu/project/id/776810/reporting, 2021.

Finch, D., Palmer, P., and Zhang, T.: Automated detection of atmospheric NO_2 plumes from satellite data: a tool to help infer anthropogenic combustion emissions, Atmos. Meas. Tech., pp. 1–21, https://doi.org/10.5194/amt-2021-177, 2021.

Grell, G. A., Peckham, S. E., Schmitz, R., McKeen, S. A., Frost, G., Skamarock, W. C., and Eder, B.: Fully coupled "online" chemistry within the WRF model, Atmos. Environ., 39, 6957–6975, https://doi.org/10.1016/j.atmosenv.2005.04.027, 2005.

Hakkarainen, J., Ialongo, I., Koene, E., Szelag, M., Tamminen, J., Kuhlmann, G., and Brunner, D.: Analyzing Local Carbon Dioxide and Nitrogen Oxide Emissions From Space Using the Divergence Method: An Application to the Synthetic SMARTCARB Dataset, Frontiers in Remote Sensing, 3, https://doi.org/10.3389/frsen.2022.878731, 2022.

He, K., Zhang, X., Ren, S., and Sun, J.: Deep Residual Learning for Image Recognition, https://doi.org/10.48550/arXiv.1512.03385, 2015.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., Chiara, G. D., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., Rosnay, P. d., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.-N.: The ERA5 global reanalysis, Q. J. R. Meteorol. Soc., 146, 1999–2049, https://doi.org/10.1002/qj.3803, 2020.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q.: Densely Connected Convolutional Networks, https://doi.org/10.48550/arXiv.1608.06993, arXiv:1608.06993 [cs], 2018.

Jadon, S.: A survey of loss functions for semantic segmentation, https://doi.org/10.1109/CIBCB48159.2020.9277638, 2020.

Janssens-Maenhout, G., Pinty, B., Dowell, M., Zunker, H., Andersson, E., Balsamo, G., Bézy, J.-L., Brunhes, T., Bösch, H., Bojkov, B., Brunner, D., Buchwitz, M., Crisp, D., Ciais, P., Counet, P., Dee, D., Gon, H. D. v. d., Dolman, H., Drinkwater, M. R., Dubovik, O., Engelen, R., Fehr, T., Fernandez, V., Heimann, M., Holmlund, K., Houweling, S., Husband, R., Juvyns, O., Kentarchos, A., Landgraf, J., Lang, R., Löscher, A., Marshall, J., Meijer, Y., Nakajima, M., Palmer, P. I., Peylin, P., Rayner, P., Scholze, M., Sierk, B., Tamminen, J., and Veefkind, P.: Toward an Operational Anthropogenic CO2 Emissions Monitoring and Verification Support Capacity, Bull. Amer. Meteor., 101, E1439–E1451, https://doi.org/10.1175/BAMS-D-19-0017.1, 2020.

Kuenen, J. J. P., Visschedijk, A. J. H., Jozwicka, M., and Denier van der Gon, H. a. C.: TNO-MACC_II emission inventory; a multi-year (2003–2009) consistent high-resolution European emission inventory for air quality modelling, Atmos Chem Phys, 14, 10 963–10 976, https://doi.org/10.5194/acp-14-10963-2014, 2014.

Kuhlmann, G., Broquet, G., Marshall, J., Clément, V., Löscher, A., Meijer, Y., and Brunner, D.: Detectability of CO_2 emission plumes of cities and power plants with the Copernicus Anthropogenic CO_2 Monitoring (CO2M) mission, Atmos. Meas. Tech., 12, 6695–6719, https://doi.org/10.5194/amt-12-6695-2019, 2019a.

Kuhlmann, G., Clément, V., Marshall, J., Fuhrer, O., Broquet, G., Schnadt-Poberaj, C., Löscher, A., Meijer, Y., and Brunner, D.: SMART-CARB – Use of satellite measurements of auxiliary reactive trace gases for fossil fuel carbon dioxide emission estimation, Tech. rep., Zenodo, https://zenodo.org/record/4034266, 2019b.

Kuhlmann, G., Brunner, D., Broquet, G., and Meijer, Y.: Quantifying CO_2 emissions of a city with the Copernicus Anthropogenic CO_2 Monitoring satellite mission, Atmos. Meas. Tech., 13, 6733–6754, https://doi.org/10.5194/amt-13-6733-2020, 2020a.

Kuhlmann, G., Clément, V., Marshall, J., Fuhrer, O., Broquet, G., Schnadt-Poberaj, C., Löscher, A., Meijer, Y., and Brunner, D.: Synthetic XCO2, CO and NO2 observations for the CO2M and Sentinel-5 satellites, https://zenodo.org/record/4048228/export/hx, 2020b.

Kuhlmann, G., Henne, S., Meijer, Y., and Brunner, D.: Quantifying CO2 Emissions of Power Plants With CO2 and NO2 Imaging Satellites, Front. remote sens., 2, https://www.frontiersin.org/article/10.3389/frsen.2021.689838, 2021.

Lauvaux, T., Giron, C., Mazzolini, M., d'Aspremont, A., Duren, R., Cusworth, D., Shindell, D., and Ciais, P.: Global assessment of oil and gas methane ultra-emitters, Science, 375, 557–561, https://doi.org/10.1126/science.abj4351, 2022.

Lian, J., Bréon, F.-M., Broquet, G., Lauvaux, T., Zheng, B., Ramonet, M., Xueref-Remy, I., Kotthaus, S., Haeffelin, M., and Ciais, P.: Sensitivity to the sources of uncertainties in the modeling of atmospheric CO_2 concentration within and in the vicinity of Paris, Atmos Chem Phys, 21, 10 707–10 726, https://doi.org/10.5194/acp-21-10707-2021, 2021.

Mahadevan, P., Wofsy, S., Matross, D., Xiao, X., Dunn, A., Lin, J., Gerbig, C., Munger, J., Chow, V., and Gottlieb, E.: A Satellite-based Biosphere Parameterization for Net Ecosystem CO2 Exchange: Vegetation Photosynthesis and Respiration Model (VPRM), Glob. Biogeochem. Cycles., 22, https://doi.org/10.1029/2006GB002735, 2008.

Meijer, Y.: Copernicus CO2 Monitoring Mission Requirements Document, Earth and Mission Science Division, 84, https://esamultimedia.esa.int/docs/EarthObservation/CO2M_MRD_v3.0_20201001_Issued.pdf, 2020.

Nassar, R., Hill, T. G., McLinden, C. A., Wunch, D., Jones, D. B. A., and Crisp, D.: Quantifying CO2 Emissions From Individual Power Plants From Space, Geophys. Res. Lett., 44, 10,045–10,053, https://doi.org/10.1002/2017GL074702, 2017.

Nassar, R., Moeini, O., Mastrogiacomo, J.-P., O'Dell, C. W., Nelson, R. R., Kiel, M., Chatterjee, A., Eldering, A., and Crisp, D.: Tracking CO2 emission reductions from space: A case study at Europe's largest fossil fuel power plant, Frontiers in Remote Sensing, 3, https://www.frontiersin.org/articles/10.3389/frsen.2022.1028240, 2022.

Ramachandran, P., Zoph, B., and Le, Q. V.: Searching for Activation Functions, https://doi.org/10.48550/arXiv.1710.05941, 2017.

Reuter, M., Buchwitz, M., Schneising, O., Krautwurst, S., O'Dell, C. W., Richter, A., Bovensmann, H., and Burrows, J. P.: Towards monitoring localized CO_2 emissions from space: co-located regional CO_2 and NO_2 enhancements observed by the OCO-2 and S5P satellites, Atmos Chem Phys, 19, 9371–9383, https://doi.org/10.5194/acp-19-9371-2019, 2019.

Ronneberger, O., Fischer, P., and Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation, arXiv:1505.04597 [cs], http://arxiv.org/abs/1505.04597, 2015.

Storn, R. and Price, K.: Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, J Glob Optim, 11, 341–359, https://doi.org/10.1023/A:1008202821328, 1997.

Tan, M. and Le, Q. V.: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, arXiv:1905.11946 [cs, stat], http://arxiv.org/abs/1905.11946, 2020.

UNFCCC: Paris Agreement, {FCCC/CP/2015/L.9/Rev1}, http://unfccc.int/resource/docs/2015/cop21/eng/l09r01.pdf, 2015.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, I., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, Vijaykumar, A., Bardelli, A. P., Rothberg, A., Hilboll, A., Kloeckner, A., Scopatz, A., Lee, A., Rokem, A., Woods, C. N., Fulton, C., Masson, C., Häggström, C., Fitzgerald, C., Nicholson, D. A., Hagen, D. R., Pasechnik, D. V., Olivetti, E., Martin, E., Wieser, E., Silva, F., Lenders, F., Wilhelm, F., Young, G., Price, G. A., Ingold, G.-L., Allen, G. E., Lee, G. R., Audren, H., Probst, I., Dietrich, J. P., Silterra, J., Webber, J. T., Slavič, J., Nothman, J., Buchner, J., Kulick, J., Schönberger, J. L., de Miranda Cardoso, J. V., Reimer, J., Harrington, J., Rodríguez, J. L. C., Nunez-Iglesias, J., Kuczynski, J., Tritz, K., Thoma, M., Newville, M., Kümmerer, M., Bolingbroke, M., Tartre, M., Pak, M., Smith, N. J., Nowaczyk, N., Shebanov, N., Pavlyk, O., Brodtkorb, P. A., Lee, P., McGibbon, R. T., Feldbauer, R., Lewis, S., Tygier, S., Sievert, S., Vigna, S., Peterson, S., More, S., Pudlik, T., Oshima, T., Pingel, T. J., Robitaille, T. P., Spura, T., Jones, T. R., Cera, T., Leslie, T., Zito, T., Krauss, T., Upadhyay, U., Halchenko, Y. O., and Vázquez-Baeza, Y.: SciPy 1.0: fundamental algorithms for scientific computing in Python, Nature Methods, 17, 261–272, https://doi.org/10.1038/s41592-019-0686-2, 2020.

Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J.: Dive into Deep Learning, https://doi.org/10.48550/arXiv.2106.11342, 2022.