Bayesian denoising of satellite images using co-registered NO_2 images

Erik Franciscus Maria Koene[1], Gerrit Kuhlmann[1], and Dominik Brunner[1]

[1]Empa, Laboratory for Air Pollution / Environmental Technology, Dübendorf, Switzerland

Correspondence: Erik Koene (erik.koene@empa.ch)

Abstract. Accurate emission tracking (e.g., locating and quantifying hot spots) using satellite images requires a good signal-to-noise ratio (SNR) of total column images. Achieving this SNR is challenging for satellite-based trace gas imagers, especially when enhancements are small relative to the background or small relative to retrieval uncertainty. Therefore, some satellites carry additional trace gas imagers with high SNR, such as NO_2, which is co-emitted with the trace gas of interest. While NO_2 is frequently used qualitatively for plume detection or plume fitting, its potential for quantitative noise reduction remains largely untapped. This paper presents two methods to enhance the SNR of total column images using co-registered NO_2 images through minimum mean square error (MMSE) Bayesian denoising, which are a simple form of a Kalman filter or maximum a posteriori estimate. The first "joint MMSE" method relies on the presence of plumes in both the low- and co-registered high-SNR NO_2 images. The second "self-similar MMSE" method utilizes image self-similarity and is based on an existing technique called BM3D. The methods are evaluated using a synthetic dataset (SMARTCARB) of atmospheric CO_2 and NO_2 concentrations, achieving over +40 decibels improvement in peak SNR (i.e., an over $10^{(40/10)}$ increase in SNR). Additionally, the methods are applied to TROPOMI SO_2 and NO_2 data over South Africa and used to compute a divergence image, demonstrating that an estimated ~~30-60~~40-80% noise reduction is possible. By enhancing the SNR of total column images, these techniques improve the detectability of subtle emission signals, which could benefit atmospheric monitoring applications.

1 Introduction

To quantify emissions and support climate policy, satellite-based monitoring system are developed that will detect and quantify emissions plumes from cities and large point sources (hereafter referred to as 'hot spots'). To perform emission quantification for hot spots, a good signal-to-noise ratio (SNR) is essential; first to be able to detect the plumes, and second to be able to quantify the plume enhancements with good accuracy. Achieving this for satellite observations of CO_2 is challenging, as enhancements (~0-5 ppm) are minor compared to background levels (~420 ppm) and retrieval uncertainties are high (Miller et al., 2007). Therefore, CO_2 monitoring satellites like the Global Observing Satellite for Greenhouse gases and Water cycle (GOSAT-GW~~,~~), the Copernicus Anthropogenic Carbon Dioxide Monitoring (CO2M~~and TANGO~~) mission and the Twin Anthropogenic Greenhouse Gas Observers (TANGO) mission will carry an additional NO_2 instrument. NO_2 is useful because it is co-emitted with CO_2 during high-temperature combustion while it can be measured with a much better SNR. NO_2 thus helps delineating and thereby quantifying the low SNR CO_2 plumes using emission quantification methods. So far, approaches

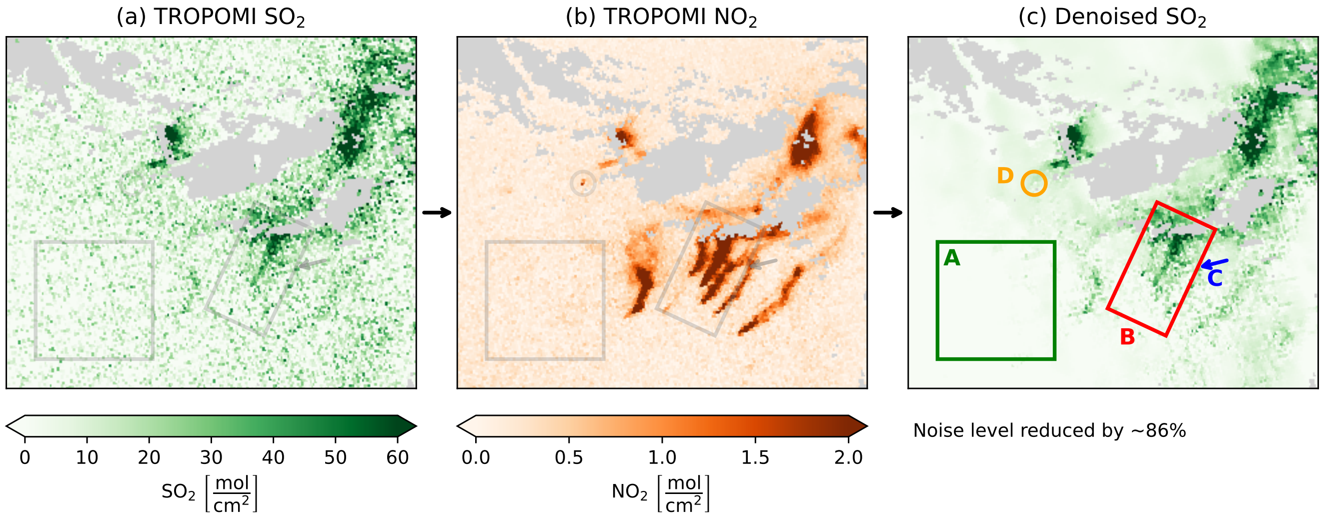

Figure 1. Example of the denoising procedure for a South African region, recorded by TROPOMI on 2021-02-20. Axis labels are omitted to emphasize the clarity of the denoised image rather than geographical location. Data with a quality factor below 0.75 is masked and appears as plain gray. By optimally combining the (a) 'noisy' SO_2 and (b) less 'noisy' NO_2 images, we create a (c) denoised SO_2 image. We highlight some features of the denoised image. Square **A** shows that in areas without signal, noise is effectively removed from the image. Rectangle **B** indicates that plume signals present in the noisy data are retained, resulting in an enhanced signal-to-noise ratio. ~~Rectangle~~ Arrow **C** illustrates an east-west (i.e., right-left) feature with low amplitude but sharp, high contrast edges, identifiable in the original SO_2 image but absent in the NO_2 image, confirming that significant 'positive' and 'negative' signals are preserved during denoising. Circle **D** shows a feature of high amplitude in the NO_2 image which is absent in the SO_2 image, indicating we don't add signal unduly. The noise level estimate is derived from Immerkaer (1996).

in the literature have used the information contained in the NO_2 observations mainly qualitatively. For example, they guided plume detection or constrained a Gaussian curve fitted to plume transects (Reuter et al., 2019; Kuhlmann et al., 2019, 2021). One prominent emission quantification method, the divergence method (Beirle et al., 2019, 2023; Koene et al., 2024), cannot effectively leverage the superior SNR of the co-registered NO_2 data, as it does not depend on plume detection. As the divergence method is highly susceptible to noise in the data due to its derivative operations, Hakkarainen et al. (2022) proposed to apply a mean filter to prepare the noisy CO2M CO_2 images for the divergence method; however, such spatial smoothing risks blurring emission signals at the source.

In this paper, we explore two data-driven methods to enhance the SNR of trace gas images using the co-registered NO_2 images (a process also referred to as 'denoising'). Compared to alternatives like a model-driven denoising approach relying on meteorological, emission, and satellite noise priors, a data-driven method requires fewer assumptions, as the empirical cross-field covariance between co-registered images is learned directly from the data. An example of the proposed methods is given in Figure 1, which illustrates the effectiveness in reducing noise in a SO_2 image recorded by the TROPOspheric Monitoring Instrument (TROPOMI; Veefkind et al., 2012). The two methods are minimum mean square estimators (MMSEs).

The first is based on the joint information in CO_2 and NO_2 pixels. The second is based on image self-similarity. Put simply, the MMSEs we present are operations that extract a denoised signal from two (or more) noisy inputs in a Bayesian optimal way, much like a Kalman filter. We define the estimators in the theory section, and suggest to chain them in series to provide the best results. Within the results section of the paper, we verify the method by applying it to synthetic CO2M data to denoise synthetic CO_2 images. We then show a ‘real data’ example using combined TROPOMI SO_2 and NO_2 data and use the denoised SO_2 data to denoise the corresponding divergence map.

2 Methods

The two denoising methods presented in the following will be referred to as the “joint MMSE” approach and the “self-similar MMSE” approaches. The former is a novel innovation, whereas the latter is a pre-existing method from the field of computer vision, which we adapt for denoising co-registered images.

2.1 Joint MMSE (jMMSE)

In this section, we explore a method that makes use of the joint information in two co-registered signals at the pixel level. The theory may alternatively be derived from a Bayesian inference point of view, as shown in ~~Appendix B~~Supplementary material section A.

2.1.1 Observation model

Satellite data of two co-registered *pixels* of, say, CO_2 and NO_2 follow a general model like

$$\begin{bmatrix} \tilde{CO_2} \\ \tilde{NO_2} \end{bmatrix} = \begin{bmatrix} CO_2 \\ NO_2 \end{bmatrix} + \begin{bmatrix} n_{CO_2} \\ n_{NO_2} \end{bmatrix}, \tag{1}$$

where the tildes indicate *noisy* observations; CO_2 and NO_2 denote the noise-free but unknown true values, and n_{CO_2} and n_{NO_2} indicate the noise on the measurements. We can rewrite this model into a coupled observational model by making it a function of the noise-free CO_2 data,

$$\begin{bmatrix} \tilde{CO_2} \\ \tilde{NO_2} \end{bmatrix} = \begin{bmatrix} 1 \\ d(x,y) \end{bmatrix} CO_2 + \begin{bmatrix} n_{CO_2} \\ n_{NO_2} \end{bmatrix} = \tilde{\boldsymbol{M}} = \boldsymbol{H}c + \boldsymbol{n} \tag{2}$$

where $\tilde{\boldsymbol{M}}$ contains the two noisy observed pixels, $c = CO_2$ is the noise-free column, $\boldsymbol{H} = [1 \quad d(x,y)]^T$ is the observation operator with a spatially varying function $d(x,y)$ that transforms the CO_2 pixel into an equivalent NO_2 observation, and $\boldsymbol{n}$ contains the two noise components.

2.1.2 The maximum a posteriori solution

Our aim is to estimate c (the unknown noise-free CO_2 column) from $\tilde{\boldsymbol{M}}$ (the noisy observations). This can be written as a *maximum a posteriori* problem with a Gaussian distributed prior with mean $\mathbb{E}[c]$, noise mean $\mathbb{E}[\boldsymbol{n}] = \boldsymbol{0}$ and independent errors

$\mathbb{E}[c\boldsymbol{n}] = \mathbb{E}[c]\mathbb{E}[\boldsymbol{n}] = \mathbf{0}$, yielding a minimum mean square error (MMSE) optimal estimate of the underlying CO_2 field, which we will denote by $\hat{c}$,

$$\hat{c} = \arg\min_c E\left[\sigma_c^{-2}(c - \mathbb{E}[c])^2 + (\boldsymbol{H}c - \tilde{\boldsymbol{M}})^T \mathbf{C}_{nn}^{-1}(\boldsymbol{H}c - \tilde{\boldsymbol{M}})\right], \tag{3}$$

where $\sigma_c^2 = \mathbb{E}[c^2] - \mathbb{E}[c]^2$ is the variance of the expected prior, and $\mathbf{C}_{nn} = \mathbb{E}[\boldsymbol{n}\boldsymbol{n}^T]$ is the noise covariance matrix. See Appendix A for details how such quantities may be computed in practice. The solution to this problem is well-known (e.g., Fichtner, 2021, eq. 6.8),

$$\hat{c} = \frac{\boldsymbol{H}^T \mathbf{C}_{nn}^{-1} \tilde{\boldsymbol{M}} + \sigma_c^{-2}\mathbb{E}[c]}{\boldsymbol{H}^T \mathbf{C}_{nn}^{-1} \boldsymbol{H} + \sigma_c^{-2}}. \tag{4}$$

The solution in eq. (4) is the *maximum a posteriori solution*, also known as the *generalized least squares solution* or the *Bayesian linear estimator*, which is mathematically also identical to a single prediction step in a Kalman filter framework without recursive time updates (e.g., Fichtner, 2021, eqs. 6.13–6.14).

2.1.3 Solution using only the available data

The solution in eq. (4) is elegant but impractical, as it requires one to know $\boldsymbol{H}$ (i.e. the true NO_2:CO_2 ratio $d(x, y)$ for every pixel). However, in the following, we show that the solution ~~may~~ can be rewritten into a form that depends merely on the data itself. For this, we take a closer look at the data covariance matrix, which can be estimated from the data itself (i.e., the sample covariance matrix):

$$\mathbf{C}_{dd} = \mathbb{E}[\tilde{\boldsymbol{M}}\tilde{\boldsymbol{M}}^T] - \mathbb{E}[\tilde{\boldsymbol{M}}]\mathbb{E}[\tilde{\boldsymbol{M}}]^T = \begin{bmatrix} \mathrm{cov}(\tilde{CO_2}, \tilde{CO_2}) & \mathrm{cov}(\tilde{CO_2}, \tilde{NO_2}) \\ \mathrm{cov}(\tilde{CO_2}, \tilde{NO_2}) & \mathrm{cov}(\tilde{NO_2}, \tilde{NO_2}) \end{bmatrix}. \tag{5}$$

Given the model of eq. (2), we can also write it as (making judicious use of $\mathbb{E}[c\boldsymbol{n}] = \mathbf{0}$),

$$\mathbf{C}_{dd} = \mathbb{E}\left[(\boldsymbol{H}c + \boldsymbol{n})(\boldsymbol{H}c + \boldsymbol{n})^T\right] - \mathbb{E}\left[\boldsymbol{H}c + \boldsymbol{n}\right]\mathbb{E}\left[\boldsymbol{H}c + \boldsymbol{n}\right]^T, \tag{6}$$

$$= \mathbb{E}[\boldsymbol{n}\boldsymbol{n}^T] - E\left[\boldsymbol{n}\right] E\left[\boldsymbol{n}^T\right] + \boldsymbol{H}\boldsymbol{H}^T\left(\mathbb{E}[c^2] - \mathbb{E}[c]^2\right), \tag{7}$$

$$= \mathbf{C}_{nn} + \boldsymbol{H}\boldsymbol{H}^T \sigma_c^2. \tag{8}$$

As detailed in ~~Appendix D1~~Supplementary section C1, we can derive a matrix inversion identity from the Sherman–Morrison formula, $\mathbf{A}(\mathbf{A} + \boldsymbol{B}C\boldsymbol{D}^T)^{-1} = \mathbf{I} - \boldsymbol{B}\boldsymbol{D}^T\mathbf{A}^{-1}/(\boldsymbol{D}^T\mathbf{A}^{-1}\boldsymbol{B} + C^{-1})$, which yields the following relation,

$$\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1} = \mathbf{I} - \frac{\boldsymbol{H}\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}}{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H} + \sigma_c^{-2}}, \tag{9}$$

for which we note that the right-hand side closely resembles eq. (4). By rearranging terms, pre-multiplying with a vector $\boldsymbol{w}^T = [1 \quad 0]$ that satisfies $\boldsymbol{w}^T\boldsymbol{H} = 1$, post-multiplying the result with ~~$(\tilde{\boldsymbol{M}} - \mathbb{E}[\boldsymbol{M}])$~~ $(\tilde{\boldsymbol{M}} - \mathbb{E}[\tilde{\boldsymbol{M}}])$ and adding the expected prior column $\mathbb{E}[c]$, we obtain

$$\boldsymbol{w}^T\left(\mathbf{I} - \mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}} - \mathbb{E}[\boldsymbol{M}]\mathbb{E}[\tilde{\boldsymbol{M}}]\right) + \mathbb{E}[c] = \frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}} + \sigma_c^{-2}\mathbb{E}[c]}{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H} + \sigma_c^{-2}}, \tag{10}$$

(the details of this step are given in ~~Appendix D2~~Supplementary section C2).

The ~~noise-free~~ denoised column estimate $\hat{c}$ of the Bayesian optimal solution of eq. (4) may thus be obtained entirely from the data itself, using the left-hand side of eq. (10). It relieves us of the need to know the forward model $\boldsymbol{H}$ that maps the noise-free CO_2 field into NO_2 columns. Simplifying the left-hand side of eq. (10), the details of which are given in ~~Appendix D3~~Supplementary section C 3, we obtain the optimal joint MMSE estimate,

$$\hat{c} = \tilde{CO}_2 - \boldsymbol{w}^T \mathbf{C}_{nn} \mathbf{C}_{dd}^{-1} (\tilde{\boldsymbol{M}} - \mathbb{E}[\boldsymbol{M}]\mathbb{E}[\tilde{\boldsymbol{M}}]). \tag{11}$$

The various covariance matrices and expected values need to be computed using small patches of size $T \times T$ for small values of T (e.g., 5) around a given pixel. See Appendix A for an example implementation in the Python programming language, and ~~Appendix C~~ Supplementary section B for an explicit version of the jMMSE estimate without vector notation.

The ratio $\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}$ in eq. (11) is quite literally the inverse of the SNR. Thus, in regions of a high SNR ($\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1} \approx \mathbf{0}$) we simply keep the measurement as it is, $\hat{c} = \tilde{CO}_2$. In regions without enhanced signals, we have $\mathbf{C}_{nn} = \mathbf{C}_{dd} \Longleftrightarrow \mathbf{C}_{nn}\mathbf{C}_{dd}^{-1} = \mathbf{I}$ and thus take the expected value $\hat{c} = \mathbb{E}[c]$, e.g., the local mean or local median. Conversely, noise will be optimally subtracted in the case of a lower SNR ($\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1} > \mathbf{I}$) based on the correlations between the CO_2 and NO_2 measurements. Hence, the derived expression has all the properties that we would expect from a SNR perspective.

2.2 Self-similar MMSE (BM3D)

An alternative method for denoising is called block matching and 3D filtering (BM3D). This method was introduced by Dabov et al. (2007) in the field of computer vision. It is another MMSE method, but this time it makes use of the self-similarity of patches within single color images (with three channels) to denoise them. We adapt it for denoising joint satellite images using two channels by linearly combining min-max-normalized CO_2 and NO_2 data (i.e., fitting the data range of the two satellite images into the range 0 to 1) into the first channel with a factor 0.5 each, and placing the min-max-normalized NO_2 image into the second channel. After computing the denoised estimates for both channels, we subtract the denoised NO_2 channel (with a factor 0.5) from the first channel to extract the denoised CO_2 signal. Note that the factor 0.5 is one of the two hyperparameters that can be modified when applying BM3D (the other being the prior observation error, σ_c, mentioned in the following).

BM3D is still considered to be a state-of-the-art image denoising algorithm (e.g., Yahya et al., 2020), and computes the following MMSE result for the noise-free CO_2 field (compare with eq. 4):

$$\hat{c}(\boldsymbol{x}) = \mathcal{T}^{-1}\left[\frac{C_{nn}^{-1}(\boldsymbol{f})}{C_{nn}^{-1}(\boldsymbol{f}) + \sigma_c^{-2}} \mathcal{T}\left[\tilde{\mathrm{CO}}_2\right](\boldsymbol{f})\right], \tag{12}$$

which is also known as a 'Wiener (deconvolution) filter'. Operators $\mathcal{T}$ and $\mathcal{T}^{-1}$ are 3D wavelet transforms and their inverses, which project the image pixel space ($\boldsymbol{x}$) into frequency domains ($\boldsymbol{f}$). Compared to eq. (4), BM3D works with a scalar quantity rather than a vector quantity for each frequency, and the observation operator $\boldsymbol{H}$ is simply replaced by 1. The factor $C_{nn}^{-1}(f)$ is given by $C_{nn}^{-1}(f) = |\mathcal{T}[\mathrm{CO}_2](\boldsymbol{f})|^2$. Thus, C_{nn}^{-1} is the energy of the true (noisefree) CO_2 signal in the wavelet transformed domain. High spectral energy implies low noise and vice versa. Of course, the noisefree signal is not available, so the Wiener

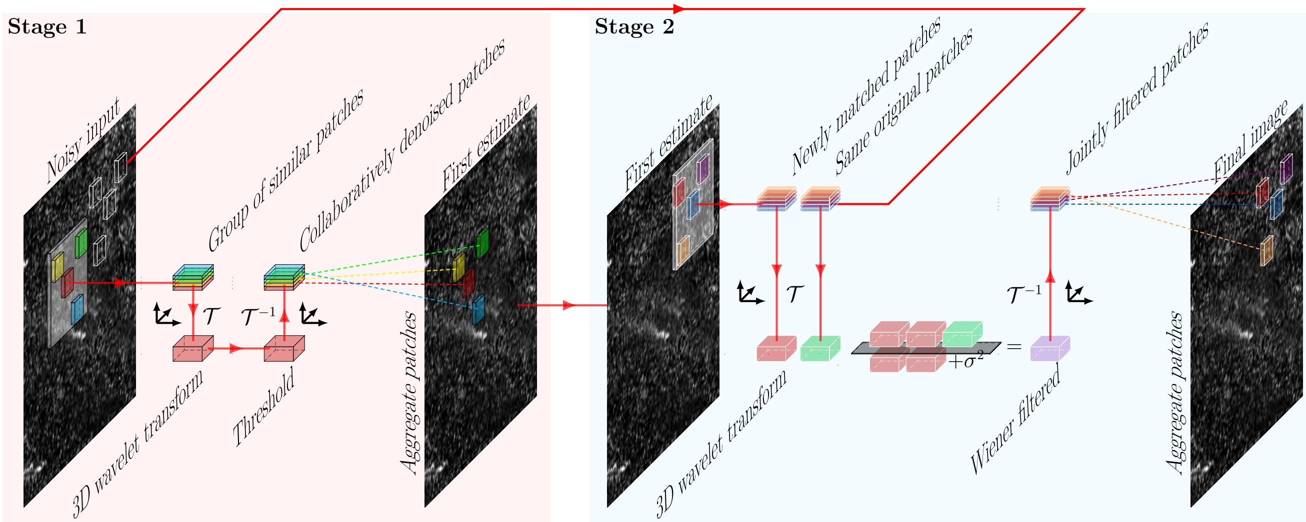

Figure 2. A schematic explanation of BM3D. In stage 1, similar looking patches are collaboratively denoised to produce a first denoised estimated image. In stage 2, similar looking patches are selected from the first estimate, and corresponding patches from the original input, form two blocks. Using a Wiener filter, the original image patches are denoised, leading to the final denoised image. The steps are carried out for all possible patches in the image.

filter of eq. (12) is not actually computable. BM3D circumvents this problem by first obtaining an estimate of the noisefree CO_2 data through an initial filtering step, which is used instead of the true noisefree signal in eq. (12).

BM3D manages to achieve good performance using the assumption of *image self-similarity* (i.e., small patches of similar-looking data repeat throughout an image). If one can find several of such similar-looking patches in the image, and takes their mean, then random noise should be attenuated (this is called ‘non-local means’, Buades et al., 2005, 2011). The first estimate in BM3D is obtained in a similar manner. More precisely, first, an 8×8 image patch is selected and N similar patches are found in the image. Second, an $N \times 8 \times 8$ ‘3D block’ is formed of these patches. Third, the 3D blocks are transformed into the wavelet domain using a 3D wavelet transform $\mathcal{T}$ and denoised using a hard thresholding step (i.e., frequency components with low energy are removed). Fourth, after an inverse wavelet transform $\mathcal{T}^{-1}$, the N denoised patches are moved back to their respective spots in the image. This process is repeated for each image patch. The fifth step is to repeat the entire process, except that the denoising now uses Wiener deconvolution of eq. (12) with C_{nn}^{-1} defined by the first denoised estimate, yielding the MMSE of the final image. The method is sketched in Figure 2.

BM3D denoises *color images* by forming a composite channel that contains the summed red, green, and blue image data. This composite channel is used for patch selection (step 1, above). For the remaining channels, the same patches are used, but each channel is denoised individually. We propose the same to make the process work for CO_2 and NO_2 images: we normalize the CO_2 and NO_2 images, and then form one channel of $(CO_2 + NO_2)/2$ and one channel of just NO_2. Patch selection is carried out on the first channel (the mean of the normalized CO_2 and NO_2 images), but denoising of the patches is carried out on both channels individually. By subtracting the second channel from the first, we end up with a new CO_2 image, which was helped by the higher signal-to-noise ratio of the NO_2 image during patch selection and denoising. A reference implementation in Python can be used that is called 'bm3d' on pypi by Mäkinen et al. (2020).

2.3 Sequential denoising using the two presented methods

As the two methods (joint MMSE and BM3D) are sufficiently different in the structural features they use to denoise the data, it stands to reason that an application of both BM3D (to provide an initial cleaned up version of the data) followed by the joint MMSE (to further enhance the signal) will have the potential to further denoise the data. In this paper, we also test this sequential denoising method.

3 Results

3.1 Performance metrics

We score the performance of the methods where the truth is available using the two most common metrics in computer vision. The first is the peak signal-to-noise ratio (PSNR) in units of decibel, i.e. a higher value means a better performance,

$$\text{PSNR} = 10\log_{10}\left(\frac{(\max(c)-\min(c))^2}{\frac{1}{n_x n_y}\sum_{i_x}\sum_{j_y}(\hat{c}_{i_x,j_y}-c_{i_x,j_y})^2}\right), \tag{13}$$

where $c \equiv c_{i_x,j_y}$ is the true (noise-free) signal and $\hat{c} \equiv \hat{c}_{i_x,j_y}$ is the estimated signal, indexed over all 2D pixels i_x and j_y. The second metric is the structural similarity index measure (SSIM; Wang et al., 2004, again, a higher value means a better performance),

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{14}$$

where x and y are 7×7 tiles/patches from images c and $\hat{c}$ respectively, μ_x and μ_y are their sample averages, σ_x and σ_y are the sample standard deviations, σ_{xy} is their covariance, and $c_1 = (0.01(\max(c) - \min(c)))^2$ and $c_2 = (0.03(\max(c) - \min(c)))^2$. The PSNR is very sensitive to random noise, while the SSIM is very sensitive to image artifacts such as blurring. Consequently, we want the PSNR and the SSIM to improve simultaneously.

We can make a noise estimate using the algorithm described in Immerkaer (1996), which compares a grid-aligned Laplacian estimate with a diagonal Laplacian estimate, to estimate the noise standard deviation for Gaussian (i.e., white or random) noise as

$$\sigma_{\text{est}} = \sqrt{\frac{\pi}{2}}\frac{1}{6(W-2)(h-2)}\sum_{\text{pixels}}|I(x,y) * N| \tag{15}$$

where W and H, respectively, are the width and height of the trace gas image I, and where $*$ denotes a spatial convolution with the 2-D kernel

$$N = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}. \tag{16}$$

3.2 Application to synthetic joint CO_2 and NO_2 images

In this section, we will apply the algorithms presented above to synthetic CO2M CO_2 and NO_2 satellite images from the SMARTCARB dataset (Kuhlmann et al., 2020a, b). The SMARTCARB dataset is a synthetic quasi-Level 2 product at the CO2M spatial resolution (roughly 2×2 km^2) and swath length (roughly 250 km) over primarily Germany and surrounding regions, for the year 2015. As it is essential that plume signals look 'similar' for input to the MMSE methods, we will use column-averaged dry-air mole fractions of CO_2 (XCO_2) data (in ppmv) and tropospheric NO_2 column densities (in molecules/cm^2). The reason for using XCO_2 rather than CO_2 column densities is that the latter are strongly susceptible to surface topography variations. One could also use XNO_2 images, but the topographic effect on NO_2 images is typically negligible. Hence, we will essentially use the 'standard' data products. When denoting the results with the joint MMSE, the parameter T refers to the window size used to compute expected values within the joint MMSE method. For example, $T = 5$ means that we select a 5×5 region centered on a pixel, which for CO2M is a region of about 10×10 km.

We will refer to results from the joint MMSE as 'jMMSE' and from the BM3D method as 'BM3D'. Additionally, we show the results from applying a simple 5×5 pixel mean filter (denoted as '5×5 mean filter' or '5×5 filter') to purely the CO_2 data, which was proposed in Hakkarainen et al. (2022) as a simple but effective method to prepare the noisy SMARTCARB data for the divergence method. Figures 3–4 show an example of the denoising methods applied to synthetic CO2M CO_2 and NO_2 images. The examples use the 'high noise' scenario of the SMARTCARB dataset with random errors of $\sigma_{\mathrm{VEG50}} = 1$ ppm for XCO_2 (the VEG50 scenario uses vegetation albedos and solar zenith angle of 50°; Buchwitz et al. (2013)) and $\sigma = 2 \times 10^{15}$ molecules/cm^2 for NO_2. We select the simulation day 2015-10-23, and focus on the coal-fired power plants Prunéřov[1] and Počerady[2]. These mid-sized power plants were selected as their emissions produce only weak plume enhancements compared to the CO_2 measurement noise level. Figure 3 shows that the simulated high noise on the CO_2 signal largely obscures the signal of the power plants, while the high noise on the NO_2 signal does not cause considerable changes with respect to the 'true' simulated NO_2 field. We can see that the 5×5 px mean filter does not manage to recover much of the CO_2 signal. Conversely, applying the joint MMSE to the two noisy input fields recovers much of the CO_2 images for a window size $T = 9$ – see ~~Appendix~~ Supplementary section E for images of other window sizes. The BM3D method (panel c) performs roughly equal to the joint MMSE method with $T = 9$ (panel b). We obtain the highest objective score by sequentially applying the joint MMSE method with $T = 9$ to the BM3D results (panel d), with a visibly good fit to the noisefree CO_2 signal, as well as an eightfold improvement of the SSIM and an increase in PSNR by +~~44.9~~

[1] 50.42°N 13.26°E; simulated with emissions of 11.4 Mt CO_2/yr and 11.3 kt NO_2/yr.
[2] 50.43°N 13.68°E; simulated with emissions of 9.3 Mt CO_2/yr and 9.2 kt NO_2/yr.

dB 45.1 dB when going from the noisy image to the denoised image. To put this into context, consider that for Gaussian white noise, averaging X images with noise variance σ_{noisy} yields $\sigma^2_{\text{denoised}} = \sigma^2_{\text{noisy}}/X$. Their PSNR improvement in dB may be expressed as $10\log_{10}\left(\sigma^2_{\text{noisy}}/\sigma^2_{\text{denoised}}\right) = 10\log_{10}(X)$, correspondingly we . Using this relation, we we obtain here that $X = 10^{44.9/10} \approx 31\,000$. In other words, $X = 10^{45.1/10} \approx 32\,000$, i.e., the image denoised with BM3D and jMMSE for $T = 9$ has the same noise characteristics as if we would have averaged 3132 000 images with these Gaussian independent high noise characteristics. Thus, the joint information content in CO_2 and NO_2 images is very large.

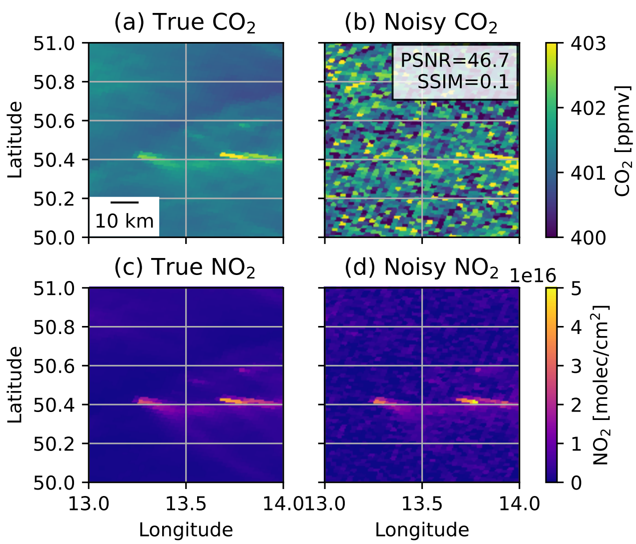

Figure 3. An example of a synthetic CO2M satellite image on 23 October 2015 for a 'high noise' scenario zooming on the emission plumes of the coal-fired power stations near Prunéřov and Počerady.

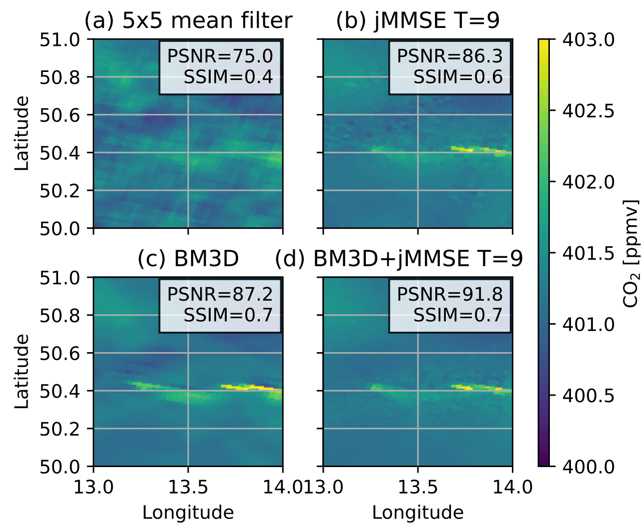

Figure 4. The noisy data from Figure 3(eb) denoised using the jMMSE methods. Visually, it is clear that the plumes originally obscured by noise become visible again. The PSNR and SSIM scores have increased (indicating improvement). The best denoising performance is obtained by the combination of BM3D and the jMMSE method for $T = 9$, with a +44.9 45.1 dB improvement.

3.3 Application to joint SO_2 and NO_2 TROPOMI images

The method is tested on real observations from also evaluated using Level-2 trace gas products from Sentinel-5P / TROPOMI, which provides trace gas images at approximately 7×3.5 near-daily global coverage at a spatial resolution of approximately 7×3.5 km^2 at nadir. Note that this resolution is coarser than that of CO2M. Two of the measured quantities are NO_2 tropospheric columns and We use reprocessed (RPRO) SO_2 data (processor version 02.04.01) and NO_2 data (processor version 02.04.00) for the year 2021 (European Space Agency (ESA), 2020, 2021). We perform the denoising using a qa_value > 0.35 to retain some more data for the denoising methods to work with, but after denoising, we analyse and plot the results using the standard recommended threshold of qa_value > 0.75. The SO_2 product used here corresponds to the SO_2 total

~~columns~~pre-Covariance-Based Retrieval Algorithm (COBRA) product. The more recent operational RPRO product is obtained using the COBRA algorithm, which has significantly lower noise and a corrected bias in the retrieval. As a consequence, the denoising performance demonstrated here for SO_2 should be interpreted in the context of this earlier processor version; results may differ for COBRA-based products due to their improved signal-to-noise ratio and potentially different error correlation structure. To better represent surface emissions, an air mass factor correction is applied to the SO_2 images, dividing the total column by the average of the three lowest averaging kernel weights[3]. While a more detailed investigation of air mass correction factors would benefit accurate emission estimates, that is beyond the scope of this study.

Figure 5(a-d) shows a TROPOMI overpass image over South Africa centered at Johannesburg for 3 March 2021, along with the ratio of the reported column precision to column values. This latter aspect is the inverse SNR (or, equivalently, the noise-to-signal-ratio). The NO_2 image has substantially larger regions of low inverse SNR values (indicative of a good SNR) than that found for SO_2. Hence, the SNR for NO_2 is much better than that for SO_2. Figure 5(e) shows that the SO_2 signal can be improved using the NO_2 signal with the multichannel BM3D method. The noise reduction becomes greater when the combination of the BM3D and jMMSE method is applied (panel f). Note that we use the jMMSE using $T = 5$, which effectively implies a window of 25×24.5 km^2 at nadir, which is slightly larger than what we used for the SMARTCARB test. Figure 5(g-h) illustrates the changes compared to the original TROPOMI image, indicating substantial noise reduction. What is notable is that values close to 0 in Figure 5(g-h) are exactly those regions with good SO_2 SNR as shown in Figure 5(d). Hence, denoising has primarily removed 'bad' signal while preserving 'good' signal. Furthermore, the removed noise as shown in 5(g-h) is essentially feature-less and consists of random speckles. Had we seen plume-like features in Figure 5(g-h) we would know that we were subtracting signal and not just noise, but this is clearly not the case. Thus, subtracting the denoised image from the original image provides an easy way to check if signal was added or destroyed.

If we apply the noise estimation method from Immerkaer (1996) to our image, we find that the original SO_2 image has $\sigma_{\text{est,original}} = 6.8$ mol/cm^2 (which is 1.7 times the mean reported column precision for this overpass), while $\sigma_{\text{est,BM3D}} = 2.1$ mol/cm^2 and ~~$\sigma_{\text{est,BM3D+jMMSE T=5}} = 1.2$~~ $\sigma_{\text{est,BM3D+jMMSE T=5}} = 1.0$ mol/cm^2; in other words, a relative improvement of about 70% or ~~82~~86%, respectively.

When ~~averaging over~~ considering a full year of observations (only selecting observations with a qa value larger than 0.75) we find that the original SO_2 image has a mean noise estimate following the method of Immerkaer (1996) of $\overline{\sigma}_{\text{est,original}} =$ 7.0 mol/cm^2, the BM3D average estimated noise is ~~$\overline{\sigma}_{\text{est,BM3D}} = 4.35$~~ $\overline{\sigma}_{\text{est,BM3D}} = 3.8$ mol/cm^2, and ~~$\overline{\sigma}_{\text{est,BM3D+jMMSE T=5}} = 2.7$~~ $\overline{\sigma}_{\text{est,BM3D+jMMSE T=5}} = 2.4$ mol/cm^2 – that is, a ~~38% and 62~~47% and 66% improvement in noise characteristics, respectively.

To further illustrate the advantage of this methodology, we present annual SO_2 divergence maps in Figure 6, i.e., computations of $\nabla \cdot (\boldsymbol{u}_{\text{eff}} \mathrm{SO}_2)$ averaged over a full year, where $\boldsymbol{u}_{\text{eff}}$ is the 2-D vector containing the effective horizontal wind. In this case, wind fields were computed by vertically averaging ERA-5 reanalysis fields using the GNFR-A emission profile ~~.~~ as weights. *(We present an identical analysis for the SMARTCARB dataset in Supplementary material D, where we show that the*

[3]Following equation ~~$\mathrm{SO}_{2,\text{new}} = (\sum_l x'_l / \sum_l A_l x'_l)\mathrm{SO}_2$~~ $\mathrm{SO}_{2,\text{new}} = (\sum_l x'_l / \sum_l A_l x'_l)\mathrm{SO2}_{\text{old}}$ with $\boldsymbol{x}' = [0, \ldots, 0, 1, 1, 1]$ from the top of the atmosphere to the surface and $\boldsymbol{A}$ is the averaging kernel. The reasoning is that we look only at power plant emissions, so we expect all the SO_2 mass to be present close to the surface.

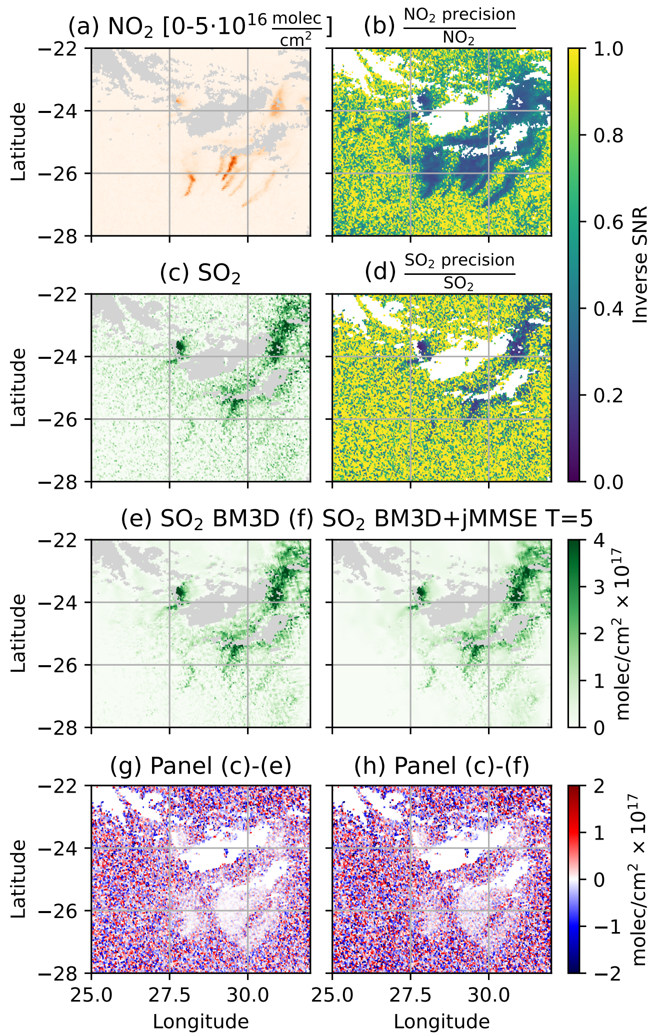

Figure 5. Single TROPOMI overpass image on 2021-03-03 of (a) NO_2 and (c) SO_2. All pixels with a qa value larger than 0.75 are plotted. Panels (b) and (d) plot the inverse of an estimate of the SNR, where values close to zero are assumed to be of high SNR. Panels (e) and (f) show an application of the (multichannel) BM3D and additionally the jMMSE. Panels (g) and (h) show the difference between the estimates and the original SO_2 image.

denoising methods, with the exception of the 5 × 5 pixel mean filter, do not introduce any biases, supporting the validity of the denoising approach for emission quantification.) The divergence was computed on the TROPOMI overpass coordinate system, and then remapped to a common 0.03-degree grid. We consider the divergence map after applying BM3D to the overpasses in Figure 6(b), a 5×5 pixel mean filter to each TROPOMI overpass as suggested by Hakkarainen et al. (2022) in Figure 6(c), and the BM3D+jMMSE T=5 approach in Figure 6(d). The 5×5 pixel mean filter reduces noise but considerably smears the signal; in contrast, the other two methods better suppress noise while retaining source sharpness. Using the method of Immerkaer (1996), we observe a 94% noise reduction with the 5×5 pixel mean filter, while the BM3D approach reduces noise by ~~37~~49% and the BM3D+jMMSE approach by ~~60~~68%.

To demonstrate that the denoising minimally impacts the signal structure, we compare emission estimates from different divergence fields in Figure 7 (a more detailed examination of the estimates is provided in ~~Appendix~~ Supplementary section E, where we also present a 'difference' plot between the ~~original~~ 'noisy' divergence image and the denoised estimates). We focus on the ratio of the original 'noisy' TROPOMI data estimates to those obtained with the denoising approaches. Emission estimates were obtained by masking data more than 0.15 degrees from the point source location, corresponding to an integration radius of 16.7 km longitudinally and 15 km latitudinally. It is clear that the 5×5 pixel mean filter estimates significantly deviate from those based on noisy data. In contrast, estimates using the BM3D and BM3D+jMMSE T=5 approaches are closer to the noisy data estimates, although some deviation is present. Sources where these two methods significantly differ from the noisy data are weakly represented in the SO_2 and/or NO_2 dataset. This indicates that the noisy data may not accurately capture their presence and thus may not serve as reliable ground truth for comparison. In fact, the major deviations in Figure 7, the Newcastle Steel Works and Camden Power Station, yield *negative* emissions when using the original noisy data. Therefore, the divergence method does not provide a reliable estimate of these source emissions in these cases. This may be due to a (combination of a) variety of reasons, e.g., there was not enough data to produce a robust average; there was a temporal bias in the sampling (the divergence method needs satellite images with plumes being blown in all directions to produce a 'point' on a map; otherwise 'dipoles' with seeming positive and negative sides appear – so if our overpass consistently samples cases where the wind is blowing in one direction, we will get artifacts); the steady state assumption was invalid; our chosen effective wind is not appropriate for the terrain and/or plume.

4 Discussion

The joint MMSE and BM3D approaches aim to denoise data while preserving the signal. The examples demonstrate the effectiveness of these methods, but they also have limitations.

Firstly, these methods assume independent and identically distributed (i.i.d.) noise with zero mean. Consequently, structural noise patterns, such as stripes, may not be denoised and could be misinterpreted as meaningful signals by BM3D due to the self-similarity of structural noise. While tuning BM3D for such cases is possible with prior knowledge of noise distribution in the 2D wavenumber domain (Mäkinen et al., 2020), this knowledge is often absent for satellite data. Another violation of the i.i.d. assumption may show up in the form of correlated errors in both satellite products due to shared dependencies in the radiative

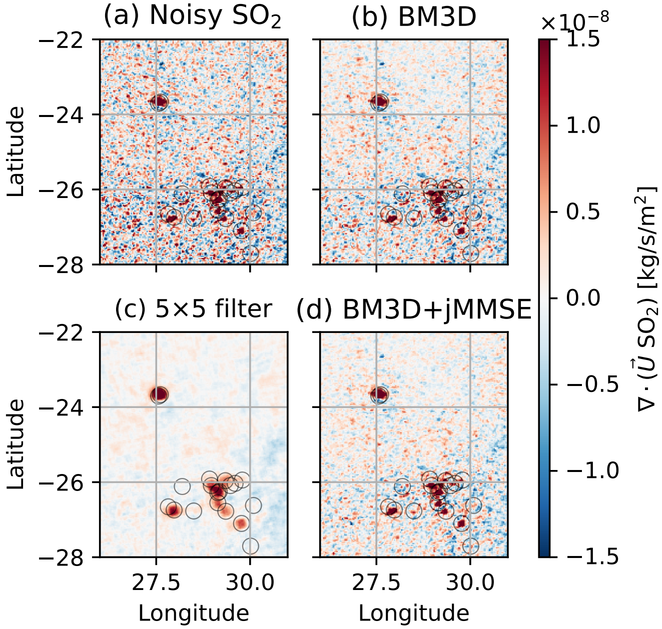

Figure 6. Annual divergence maps for a region in South Africa. (a) shows the original ‘noisy’ TROPOMI SO_2 divergence map, (b) the estimate from applying BM3D to the SO_2 and NO_2 field prior to taking the divergence, (c) the estimate from a 5×5 mean filter to the noisy SO_2 prior to taking the divergence, (d) the estimate from the optimal SO_2 image using the proposed BM3D+jMMSE T=5 method prior to taking the divergence. A number of known sources are encircled.

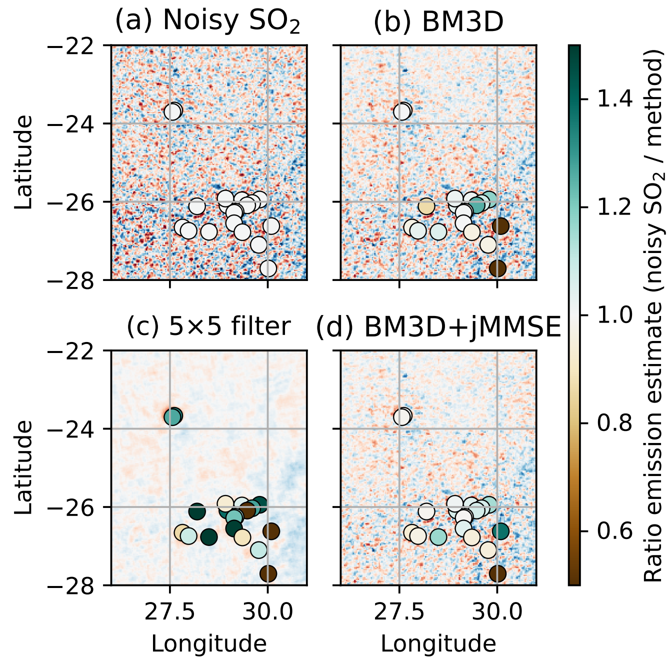

Figure 7. SO_2 emission estimate *ratios* from integrating along a $\sim$15 km radius around point source locations across the divergence maps shown in Fig. 6. A ‘blue’ color in the ratio columns corresponds to an *underestimation* compared to the ‘noisy’ SO_2 divergence map, while a ‘red’ color in the ratio columns corresponds to an *overestimation*. Note, that the ‘noisy’ SO_2 divergence map is not necessarily a ground truth, and this plot is shown for illustrative purposes only.

transfer (such as albedo, cloud contamination or aerosol loading). These factors enter both retrievals, thereby introducing a common-mode error component. Mitigation strategies could include restricting the analysis to scenes with low sources of such errors, or extending the statistical model presented here to include such correlated errors or biases. Consequently, users should interpret the denoised image with caution in regions where strong scene-dependent retrieval artifacts are expected, as these features, shared between the low and high SNR images, may be preserved or even reinforced in the denoised output.

Secondly, the practical implementation of jMMSE requires spatial stationarity, meaning the ratio of NO_2:CO_2 column densities (or similarly chosen trace gases) should be approximately constant within a window of $T \times T$ pixels. It is clear that there is no globally fixed NO_2:CO_2 ratio, and although this is approximately true when we focus on a small region, NO_x chemistry inside plumes will change ratios inside plumes (Meier et al., 2024; Krol et al., 2024). Hence, the value for T must be kept as low as possible. The competing interest, of course, is that for robust statistics, T should be chosen as large as possible. This raises the question of how to appropriately choose T. This can be based on subtracting the original image from the denoised image. Whatever shows up as structureless features there is noise, while whatever shows up as structured noise is likely the removal of real signal (e.g., this might happen if the difference shows something that looks like a plume and is indeed coincident with a plume on the original data). If one can grow T, but at some value of T the noise rejection does not improve anymore, then one has found the optimal T for noise rejection. Conversely, if one can grow T but at some point one is starting to reject also signal, then one can say they have found the optimal T to retain the signal. This argument suggests that spatial stationarity is best satisfied over small regions and indicates that denoising will be more applicable in high-resolution rather than coarse-resolution satellite images. The authors believe that rather than prescribing a value of T in this paper, it should remain an open hyperparameter for end users to decide. However, in the supplementary material we demonstrate two example workflows for selecting the parameters based on a simple grid search – either by minimizing an objective function that maximizes noise reduction while minimizing signal bias, or by performing a simple grid search optimization for the remaining hyperparameters. Once optimal parameters are found to work for a handful of images, they generally work well for a full dataset.

5 Conclusions

We presented two minimum mean square error (MMSE) estimators that enhance the signal-to-noise ratio (SNR) of noisy CO_2 or SO_2 images using co-registered NO_2 images from the upcoming GOSAT-GW and CO2M satellites, as well as from Sentinel-5P. These methods enhance the visibility of plumes that are hard to discern in the noisy images. The first method, joint MMSE, preserves plumes with good SNR (i.e., where the signal is strong or highly correlated with the NO_2 field) while subtracting noise elsewhere. The second method, BM3D, leverages image self-similarity by denoising a linear combination of normalized CO_2 or SO_2 and NO_2 images. The best outcomes result from combining both estimators, initially applying BM3D for denoising, followed by joint MMSE for further refinement of the CO_2 or SO_2 image.

We demonstrate the effectiveness of these techniques in two case studies. In synthetic data tests, the denoising process improves peak SNR by more than 40 decibels. When applied to TROPOMI SO_2 and NO_2 images over South Africa, or their annual divergence maps, we observe that a 30-60% reduction in noise levels is possible, while leaving plume structures intact.

The proposed denoising methods can enhance plume detection for single-overpass images and averaged satellite datasets. These techniques improve plume visibility and may assist in plume emission quantification methods, such as Gaussian plume inversion, cross-sectional flux methods, or the divergence method, by providing cleaner input data. Therefore, by systematically reducing noise in total column images, this approach strengthens satellite capabilities for monitoring atmospheric emissions with greater precision.

Code and data availability. A C++ implementation of BM3D may be obtained from https://github.com/gfacciol/bm3d, although in this paper we use a Python implementation from https://pypi.org/project/bm3d/. The code which implements the joint MMSE is added as a supplement to this paper, along with example data (one example from the SMARTCARB dataset and one example from the TROPOMI dataset), which can be used to reproduce Figures 1, 3-5.

Appendix A: Python implementation

In order to implement eq. (11), we need to compute the quantities $\mathbf{C}_{nn}$, $\mathbf{C}_{dd}$ and $\mathbb{E}[\boldsymbol{M}]$ with sufficient accuracy. In this section, we give some possible ways to obtain these quantities.

1. The expected column $\mathbb{E}[\boldsymbol{M}]$ ~~is~~ can be defined as the ~~average (here: median)~~ median value of a $T \times T$ patch around any given pixel.

2. The data covariance matrix $\mathbf{C}_{dd}$ is defined by eq. (5). To estimate it from the data (the 'sample covariance matrix'), we form two row vectors of size $1 \times T^2$ – for example, $\boldsymbol{C\tilde{O}_2}$ as a $1 \times T^2$ row vector containing all $C\tilde{O}_2$ observations in a $T \times T$ patch around the pixel, and $\boldsymbol{N\tilde{O}_2}$ as a $1 \times T^2$ row-vector containing all $N\tilde{O}_2$ observations around the pixel. Defining the sample deviation matrix as a $2 \times T^2$ array, $\mathbf{M} = [\boldsymbol{C\tilde{O}_2}^T \quad \boldsymbol{N\tilde{O}_2}^T]^T - \mathbb{E}[\boldsymbol{M}]$, we may compute the 2×2 data covariance matrix,

$$\mathbf{C}_{dd}|_{\text{sample}} = \frac{1}{T^2 - 1}\mathbf{M}\mathbf{M}^T. \tag{A1}$$

   As we want to use a small value for T, it makes sense to use covariance shrinkage operators, which makes the estimation more robust (Ledoit and Wolf, 2022). A simple operation is to get the eigenvector decomposition $\mathbf{C}_{dd}|_{\text{sample}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$, and reconstruct it with modified eigenvalues as

$$\mathbf{C}_{dd}|_{est} = \mathbf{U}\boldsymbol{\Lambda}'\mathbf{U}^T, \tag{A2}$$

   where $\boldsymbol{\Lambda}' = (\alpha\boldsymbol{\Lambda} + (1-\alpha)\text{Diag}(n_{CO_2}, n_{NO_2}))$ contains the modified eigenvalues based on the expected noise characteristics of the data, and $\text{Diag}(\cdot,\cdot)$ forms a diagonal matrix with the elements given. For $\alpha = 0.5$ and n_{CO_2} and n_{NO_2} as diagonal entries of $\mathbf{C}_{dd}|_{\text{sample}}$ we, for example, mix the sample covariance matrix and a diagonalized covariance matrix.

3. The noise covariance matrix $\mathbf{C}_{nn}$ corresponds to the precision of the instrument. If the noise is uncorrelated and known as σ^2_{CO2} and σ^2_{NO2} for the two measurements, that simply corresponds to $\mathbf{C}_{nn}|_{\text{instrument}} = \text{Diag}(\sigma^2_{CO2}, \sigma_{NO2})$. Alternatively, if this type of data is not available, we may estimate it as the median covariance of a single overpass (or image), the idea being that a typical image contains primarily 'noise' and only a limited amount of 'signal' (i.e., hot spot enhancements) such that the median covariance matrix of the data is representative for the noise.

We can give a straightforward example of an implementation of this algorithm in about 100 lines of Python code. Special care is taken of missing data through use of the numpy 'mask' feature. Along with the robust estimator for $\mathbf{C}_{dd}$, we also include the capacity to make sure $\mathbf{C}_{dd}$ has a stable inverse by clipping the conditioning number and by adding small values along its diagonal. The code given here is given as an example of how the quantities used in the theory could be computed in practice.

```
def covariance_shrinkage(Cdd, alpha, max_cond, min_eig):
    var_CO2 = np.ma.median(Cdd[:,:,0,0])
    var_NO2 = np.ma.median(Cdd[:,:,1,1])
```

```
    , U = np.linalg.eigh(np.nan_to_num(Cdd))
    [:,:,0] = alpha * [:,:,0] + (1-alpha) * var_CO2
    [:,:,1] = alpha * [:,:,1] + (1-alpha) * var_NO2
    eig_max = np.maximum([:,:,1], min_eig)
    eig_min_allowed = eig_max / max_cond
    [:,:,0] = np.maximum([:,:,0], eig_min_allowed)
    recon = (U * [..., None, :]) @ U.T((0,1,3,2))
    recon = (U * [..., None, :]) @ U.transpose((0,1,3,2))
    recon[np.isnan(Cdd)] = np.nan
    return recon

def ridge_regularization(Cdd, max_cond, , ):
    cond_num = np.linalg.cond(Cdd.filled(1.0))  # shape (lon, lat)
    p = 2.0  # curvature of ramp
    cond_factor = np.clip(np.log10(cond_num) / np.log10(max_cond), 0, 1)
    nugget_strength =  + cond_factor**p * ( - )

    # Apply proportional nugget to each variance term
    Cdd[:,:,0,0] += nugget_strength * np.ma.median(Cdd[:,:,0,0])
    Cdd[:,:,1,1] += nugget_strength * np.ma.median(Cdd[:,:,1,1])

    variance_floor = np.ma.median( Cdd[:,:,0,0] )
    Cdd[:,:,0,0] = np.maximum(Cdd[:,:,0,0], variance_floor)
    return Cdd

def covariance(D, i, j):
    count = (D[i,...]*D[j,...]).count(axis=-1) - 1
    return np.ma.sum(D[i,...]*D[j,...], axis=-1)/count

def nanaverage(A,W,axis=-1):
    return np.nansum(A*W,axis=axis)/((~np.isnan(A))*W).sum(axis=axis)

def MMSE_estimate_fixed(
    in_arr1, in_arr2, T, CNN=None, alpha=0.5,
    method='median', max_cond=1e7, min_eig=1e-13,
```

```
=1e-3, =1e0):

# --- Pad data (s.t. our windows catch all the data)
CO2 = np.pad(in_arr1, ((T, T), (T, T)), 'symmetric')
NO2 = np.pad(in_arr2, ((T, T), (T, T)), 'symmetric')

# --- Extract overlapping patches of size WxW
CO2_tiles = view_as_windows(CO2                , (T,T))
NO2_tiles = view_as_windows(NO2                , (T,T))
mask_tile = view_as_windows(~np.isnan(CO2+NO2), (T,T))

# --- Reshape
mask_tile = mask_tile.reshape(*mask_tile.shape[:-2], -1)
X = np.stack((CO2_tiles.reshape(*CO2_tiles.shape[:-2], -1),
              NO2_tiles.reshape(*NO2_tiles.shape[:-2], -1)))

# --- Generate masked array
X = np.ma.array( X, mask=~np.stack([mask_tile] * 2, axis=0) )

# --- Compute expected value of the dataset
av_field = {"mean": np.nanmean(X,-1,keepdims=1),
            "median": np.nanmedian(X,-1,keepdims=1)}[method]

# --- Compute sample covariance matrix
D = X - av_field
Cdd = np.ma.zeros((*D.shape[1:3], 2, 2)) * np.nan
Cdd[...,0,0] = covariance(D, 0, 0)
Cdd[...,0,1] = covariance(D, 0, 1)
Cdd[...,1,0] = covariance(D, 1, 0)
Cdd[...,1,1] = covariance(D, 1, 1)
Cdd = covariance_shrinkage(Cdd, alpha, max_cond, min_eig)
Cdd = ridge_regularization(Cdd, max_cond, , )

# --- Compute noise covariance matrix
Cnn = np.zeros_like(Cdd)
```

```
    Cnn[:,:,0,0] = np.ma.median(Cdd[...,0,0])
    if CNN is None:
        pass
    elif type(CNN) is np.float32:
        Cnn[:,:,0,0] = Cnn[:,:,0,0]*0.5 + CNN*0.5
    else:
        CNN = np.pad(CNN, ((T, T), (T, T)), 'symmetric')
        CNN = view_as_windows(CNN, (T,T))
        CNN = CNN.reshape(*mask_tile.shape[:-1], -1)
        CNN = np.nanmedian(CNN, -1)
        Cnn[:,:,0,0] = Cnn[:,:,0,0]*0.5 + CNN*0.5

    # --- Apply filter
    wCddICnn = np.linalg.solve(Cdd.filled(np.nan),Cnn.filled(np.nan))
    wCddICnnEM = np.einsum('ijk,kijl->ijl',wCddICnn[...,0].squeeze(),D)

    est_gather = np.zeros((*CO2.shape, T**2)) * np.nan
    for i in range(T**2):
        y, x = np.mod(i,T)-T//2, i//T-T//2
        xs, xe = max((T//2)+x,0), min(CO2.shape[0]-(T//2)+x, CO2.shape[0])
        ys, ye = max((T//2)+y,0), min(CO2.shape[1]-(T//2)+y, CO2.shape[1])
        est_gather[xs:xe, ys:ye, i] = wCddICnnEM[:,:,i]

    # --- Generate filter grid coordinates
    y, x = np.meshgrid(np.arange(-T//2+1, T//2+1), np.arange(-T//2+1, T//2+1))
    weights_2dgauss = np.exp(-(x**2 + y**2) / (2 * 4**2))

    # --- Compute final output
    pred = CO2 - nanaverage(est_gather,W=weights_2dgauss.flatten())
    est = np.where( np.isfinite(CO2), pred, np.nan )
    est = np.where( np.isnan(NO2) & np.isnan(est), CO2, est )

    return est[T:-T,T:-T]
```

Appendix B: Alternative derivation of jMMSE estimator

The jMMSE estimator derived in the main text provides a straightforward route to obtaining an estimator of the noisefree CO_2 data based on the joint observational model of eq. (2). That is, we first derive the maximum a posteriori solution, and rewrite this to the linear minimum mean square estimator. The original derivation, however, was done along the following lines – it may be considered to be an alternative derivation of the method, for which the generalization to other setups might be obtained more easily.

The linear minimum mean square error problem can be formulated as

$$\arg\min_{\boldsymbol{h},b} \mathbb{E}\left[\left(\boldsymbol{h}^T\tilde{\boldsymbol{M}}+b-c\right)^2\right],$$

i.e., we try to estimate c by $\boldsymbol{h}^T\tilde{\boldsymbol{M}}+b$. Note that $\boldsymbol{h}$ acts on the *noisy data*, while before $\boldsymbol{H}$ acted on the *noise-free column*, thus these are entirely different vectors. In the ideal noise-free case we have that $\boldsymbol{h}=\boldsymbol{w}=[1 \quad 0]^T$. We take a derivative with respect to b and equate the result to zero,

$$\frac{\partial\mathbb{E}\left[\left(\boldsymbol{h}^T\tilde{\boldsymbol{M}}+b-c\right)^2\right]}{\partial b} = 2\mathbb{E}\left[\boldsymbol{h}^T\tilde{\boldsymbol{M}}+b-c\right] \equiv 0,$$

$$\iff \hat{b} = \mathbb{E}[c]-\boldsymbol{h}^T\mathbb{E}[\tilde{\boldsymbol{M}}]$$

Substituting this in eq. (??) we get

$$\mathbb{E}\left[\left(\boldsymbol{h}^T\tilde{\boldsymbol{M}}+b-c\right)^2\right] = \mathbb{E}\left[(\boldsymbol{h}^T(\tilde{\boldsymbol{M}}-\mathbb{E}[\tilde{\boldsymbol{M}}]))^2+(c-\mathbb{E}[c])^2-2\boldsymbol{h}^T(\tilde{\boldsymbol{M}}-\mathbb{E}[\tilde{\boldsymbol{M}}])(c-\mathbb{E}[c])\right],$$

$$= \underbrace{\boldsymbol{h}^T\left(\mathbb{E}[(\tilde{\boldsymbol{M}}-\mathbb{E}[\tilde{\boldsymbol{M}}])(\tilde{\boldsymbol{M}}-\mathbb{E}[\tilde{\boldsymbol{M}}])^T]\right)\boldsymbol{h}}_{\boldsymbol{h}^T\mathbf{C}_{dd}\boldsymbol{h}} + \underbrace{\mathbb{E}[(c-\mathbb{E}[c])^2]}_{\sigma_c^2}$$

$$-\underbrace{2\boldsymbol{h}^T\mathbb{E}[(\tilde{\boldsymbol{M}}-\mathbb{E}[\tilde{\boldsymbol{M}}])(c-\mathbb{E}[c])]}_{2\boldsymbol{h}^T\boldsymbol{C}_{dc}},$$

with $\mathbf{C}_{dd}$ the data covariance matrix as given in eq. (5), σ_c^2 the variance of the prior, and $\boldsymbol{C}_{dc}$ is given as following, assuming signal-independent noise $\mathbb{E}[c\boldsymbol{n}] = \mathbb{E}[c]\mathbb{E}[\boldsymbol{n}]$, and recalling the covariance rule $\mathbb{E}[(a-\mathbb{E}[a])(b-\mathbb{E}[b])] = \mathbb{E}[ab] - \mathbb{E}[a]\mathbb{E}[b]$,

$$\boldsymbol{C}_{dc} = \mathbb{E}[(\tilde{\boldsymbol{M}} - \mathbb{E}[\tilde{\boldsymbol{M}}])(c - \mathbb{E}[c])],$$

$$= \mathbb{E}[\tilde{\boldsymbol{M}} c] - \mathbb{E}[\tilde{\boldsymbol{M}}]\mathbb{E}[c],$$

$$= \mathbb{E}[(\boldsymbol{H}c + \boldsymbol{n})c] - \mathbb{E}[\boldsymbol{H}c + \boldsymbol{n}]\mathbb{E}[c],$$

$$= \left(\boldsymbol{H} \underbrace{\left(\mathbb{E}\left[c^2\right] - \mathbb{E}\left[c\right]^2 \right)}_{\sigma_c^2} + \underbrace{\mathbb{E}\left[\boldsymbol{n}c\right] - \mathbb{E}\left[\boldsymbol{n}\right]\mathbb{E}\left[c\right]}_{=0} \right) \underbrace{\boldsymbol{H}^T \boldsymbol{w}}_{=1}$$

$$= \boldsymbol{H}\boldsymbol{H}^T \sigma_c^2 \boldsymbol{w}$$

$$= (\mathbf{C}_{dd} - \mathbf{C}_{nn})\, \boldsymbol{w},$$

where we used eq. (8) to find a convenient model-free expression.

Differentiating eq. (??) with respect to $\boldsymbol{h}$ and equating the result to zero yields

$$\frac{\partial \mathbb{E}\left[\left(\boldsymbol{h}^T \tilde{\boldsymbol{M}} + b - c\right)^2\right]}{\partial \boldsymbol{h}} = 2\mathbf{C}_{dd}\boldsymbol{h} - 2\boldsymbol{C}_{dc} \equiv 0,$$

$$\iff \hat{\boldsymbol{h}} = \mathbf{C}_{dd}^{-1}(\mathbf{C}_{dd} - \mathbf{C}_{nn})\,\boldsymbol{w} = \left(\mathbf{I} - \mathbf{C}_{dd}^{-1}\mathbf{C}_{nn}\right)\boldsymbol{w}.$$

The obtained least-squares optimal values for $\hat{\boldsymbol{h}}$ and $\hat{b}$ yield the jMMSE estimate for the denoised CO_2 column,

$$\hat{c} = \hat{\boldsymbol{h}}^T \tilde{\boldsymbol{M}} + \hat{\boldsymbol{b}},$$

$$= \hat{\boldsymbol{h}}^T (\tilde{\boldsymbol{M}} - \mathbb{E}[\tilde{\boldsymbol{M}}]) + \mathbb{E}[c],$$

$$= \boldsymbol{w}^T (\mathbf{I} - \mathbf{C}_{nn}\mathbf{C}_{dd}^{-1})(\tilde{\boldsymbol{M}} - \mathbb{E}[\tilde{\boldsymbol{M}}]) + \mathbb{E}[c],$$

which we recognize is the same as eq. (11).

Appendix B: Explicit form of the Joint MMSE model

We can simplify eq. (11) by using the fact that $\boldsymbol{w}^T = [1 \quad 0]$,

$$\hat{c} = \tilde{CO}_2 - [\mathrm{Cov}(n_{CO_2}, n_{CO_2}) \quad \mathrm{Cov}(n_{CO_2}, n_{NO_2})]\, \mathbf{C}_{dd}^{-1}(\boldsymbol{M} - \overline{\boldsymbol{M}}),$$

and we may furthermore invert the data covariance matrix to write eq. (??) as

$$\hat{c} = \tilde{CO_2} - \frac{1}{1-\frac{\text{Cov}(\tilde{CO_2},\tilde{NO_2})^2}{\text{Cov}(\tilde{CO_2},\tilde{CO_2})\text{Cov}(\tilde{NO_2},\tilde{NO_2})}} \frac{\text{Cov}(n_{CO_2},n_{CO_2})}{\text{Cov}(\tilde{CO_2},\tilde{CO_2})}(\tilde{CO_2}-\overline{CO_2})$$

$$+\frac{1}{\frac{\text{Cov}(\tilde{CO_2},\tilde{CO_2})\text{Cov}(\tilde{NO_2},\tilde{NO_2})}{\text{Cov}(\tilde{CO_2},\tilde{NO_2})^2}-1} \frac{\text{Cov}(n_{CO_2},n_{CO_2})}{\text{Cov}(\tilde{CO_2},\tilde{NO_2})}(\tilde{NO_2}-\overline{NO_2}),$$

$$-\frac{1}{\frac{\text{Cov}(\tilde{CO_2},\tilde{CO_2})\text{Cov}(\tilde{NO_2},\tilde{NO_2})}{\text{Cov}(\tilde{CO_2},\tilde{NO_2})^2}-1} \frac{\text{Cov}(n_{CO_2},n_{NO_2})}{\text{Cov}(\tilde{CO_2},\tilde{NO_2})}(\tilde{CO_2}-\overline{CO_2}),$$

$$+\frac{1}{1-\frac{\text{Cov}(\tilde{CO_2},\tilde{NO_2})^2}{\text{Cov}(\tilde{CO_2},\tilde{CO_2})\text{Cov}(\tilde{NO_2},\tilde{NO_2})}} \frac{\text{Cov}(n_{CO_2},n_{NO_2})}{\text{Cov}(\tilde{NO_2},\tilde{NO_2})}(\tilde{NO_2}-\overline{NO_2}).$$

Assuming no noise correlation between the CO_2 and NO_2 data, $\text{Cov}(n_{CO_2},n_{NO_2})=0$, that simplifies to

$$\hat{c} = \tilde{CO_2} - \frac{\text{Cov}(n_{CO_2},n_{CO_2})}{\text{Cov}(\tilde{CO_2},\tilde{CO_2})-\frac{\text{Cov}(\tilde{CO_2},\tilde{NO_2})^2}{\text{Cov}(\tilde{NO_2},\tilde{NO_2})}}\left((\tilde{CO_2}-\overline{CO_2})-\frac{\text{Cov}(\tilde{CO_2},\tilde{NO_2})}{\text{Cov}(\tilde{NO_2},\tilde{NO_2})}(\tilde{NO_2}-\overline{NO_2})\right).$$

Appendix B: Details for the derivation

We provide some more detail to some equations in the main body of the text, to aid a reader in reproducing the derivation of the (joint) MMSE model.

A1 Sherman-Morrison-like matrix inversion identity

The Sherman-Morrison formula is typically given as

$$\left(\mathbf{A}+\boldsymbol{u}\boldsymbol{v}^T\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\boldsymbol{u}\boldsymbol{v}^T\mathbf{A}^{-1}}{1+\boldsymbol{v}^T\mathbf{A}^{-1}\boldsymbol{u}}.$$

By pre-multiplying with $\mathbf{A}$ we obtain

$$\mathbf{A}\left(\mathbf{A}+\boldsymbol{u}\boldsymbol{v}^T\right)^{-1} = \mathbf{I} - \frac{\boldsymbol{u}\boldsymbol{v}^T\mathbf{A}^{-1}}{1+\boldsymbol{v}^T\mathbf{A}^{-1}\boldsymbol{u}}.$$

Furthermore, if we substitute $\boldsymbol{u}=\boldsymbol{v}=f\boldsymbol{w}$ for some constant factor f, we can rewrite the left- and right-hand side into

$$\mathbf{A}\left(\mathbf{A}+\boldsymbol{w}f^2\boldsymbol{w}^T\right)^{-1} = \mathbf{I} - \frac{\boldsymbol{w}f^2\boldsymbol{w}^T\mathbf{A}^{-1}}{1+f^2\boldsymbol{w}^T\mathbf{A}^{-1}\boldsymbol{w}}.$$

Dividing the numerator and denominator of the fraction by f^2 finally yields

$$\mathbf{A}\left(\mathbf{A}+\boldsymbol{w}f^2\boldsymbol{w}^T\right)^{-1}=\mathbf{I}-\frac{\boldsymbol{w}\boldsymbol{w}^T\mathbf{A}^{-1}}{f^{-2}+\boldsymbol{w}^T\mathbf{A}^{-1}\boldsymbol{w}}.$$

In the main text we used $\mathbf{A}=\mathbf{C}_{nn}$, $\boldsymbol{w}=\boldsymbol{H}$, and $f^2=\sigma_c^2$, i.e.,

$$\mathbf{C}_{nn}\underbrace{\left(\mathbf{C}_{nn}+\boldsymbol{H}\boldsymbol{H}^T\sigma_c^2\right)^{-1}}_{\mathbf{C}_{dd}^{-1}}=\mathbf{I}-\frac{\boldsymbol{H}\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

A1 Moving from (9) to (10)

We started with eq. (9), also stated above,

$$\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}=\mathbf{I}-\frac{\boldsymbol{H}\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}},$$

where $\mathbf{C}_{dd}=\mathbf{C}_{nn}+\boldsymbol{H}\boldsymbol{H}^T\sigma_c^2$. We have as our sole goal to rewrite this expression into a form that equals eq. (4).

We start by bringing the identity matrix to the left-hand side and multiplying with -1,

$$\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}=\frac{\boldsymbol{H}\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

We then pre-multiply with $\boldsymbol{w}^T=[1\quad 0]$ which satisfies $\boldsymbol{w}^T\boldsymbol{H}=1$ to find

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}},$$

and we post-multiply with $\tilde{\boldsymbol{M}}$ to get

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\tilde{\boldsymbol{M}}=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

Compared to eq. (4) we now only lack a factor $\sigma_c^{-1}\mathbb{E}[c]/(\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H})$. The simplest way to gain this factor is to simply add $\mathbb{E}[c]$ to both sides,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\tilde{\boldsymbol{M}}+\mathbb{E}[c]=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}+\mathbb{E}[c],$$

and realizing we can make $\mathbb{E}[c]$ part of the fraction by multiplying it with the denominator,

$$\begin{aligned}\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\tilde{\boldsymbol{M}}+\mathbb{E}[c]&=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}+\left(\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}\right)\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}},\\&=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}+\sigma_c^{-2}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}+\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.\end{aligned}$$

We can see on the right-hand side that we got our desired term with $\sigma_c^{-2}\mathbb{E}[c]$ and an extra, unwanted term. There is a convenient expression for this extra unwanted term which we can deduce from eq. (??), namely,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\boldsymbol{H}\mathbb{E}[c]=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

Subtracting (??) from (??) gives

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\boldsymbol{H}\mathbb{E}[c]\right)+\mathbb{E}[c]=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}+\sigma_c^{-2}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

Finally, we can see that $\boldsymbol{H}\mathbb{E}[c]=\mathbb{E}[\boldsymbol{H}c]=\mathbb{E}[\boldsymbol{M}-\boldsymbol{n}]$, which we can use as a substitution on the left-hand side of the equation,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]+\mathbb{E}[\boldsymbol{n}]\right)+\mathbb{E}[c]=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}+\sigma_c^{-2}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

Under the assumption of zero mean noise $\mathbb{E}[\boldsymbol{n}]=0$ which underpins the Bayesian solution of eq. (4), we obtain the final expression,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right)+\mathbb{E}[c]=\frac{\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\tilde{\boldsymbol{M}}+\sigma_c^{-2}\mathbb{E}[c]}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}}.$$

As a sidenote, we remark that we can obtain the posterior covariance also from eq. (??), namely,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\mathbf{C}_{nn}\boldsymbol{w}=\frac{1}{\sigma_c^{-2}+\boldsymbol{H}^T\mathbf{C}_{nn}^{-1}\boldsymbol{H}},$$

see, e.g., eq. 6.9 in Fichtner (2021). The left-hand portion here allows for a simplification, e.g.,

$$\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\mathbf{C}_{nn}\boldsymbol{w}=\sigma_{CO2}^2-\boldsymbol{w}^T\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\mathbf{C}_{nn}\boldsymbol{w}$$

A1 Moving from (10) to (11)

Finally, we can simplify eq. (10), also present in the previous subsection, to a simplified expression,

$$\begin{aligned}\hat{c}&=\boldsymbol{w}^T\left(\mathbf{I}-\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right)+\mathbb{E}[c],\\ &=\boldsymbol{w}^T\mathbf{I}\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right)-\boldsymbol{w}^T\left(\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right)+\mathbb{E}[c],\\ &=\underbrace{\boldsymbol{w}^T\tilde{M}}_{=\tilde{CO}_2}\underbrace{-\boldsymbol{w}^T\mathbb{E}[\boldsymbol{M}]+\mathbb{E}[c]}_{=0}-\boldsymbol{w}^T\left(\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right),\\ &=\tilde{CO}_2-\boldsymbol{w}^T\left(\mathbf{C}_{nn}\mathbf{C}_{dd}^{-1}\right)\left(\tilde{\boldsymbol{M}}-\mathbb{E}[\boldsymbol{M}]\right).\end{aligned}$$

Appendix B: Additional figures

A1 Synthetic case – jMMSE for different window sizes

The noisy data from which Figure 3(c) was drawn, though we now consider a larger area, denoised using the jMMSE with $T = 3$ to $T = 13$ leads to increasing PSNR and SIM scores. This indicates enhanced performance with larger window sizes. Same as Figure **??**, but now we show the difference with respect to the noise-free case. We see that we remove increasingly more noise, but do not improve our recovery of some plumes (visible as 'blue' features, i.e., where our recovery underestimates the true source strength).

A1 TROPOMI SO_2 case – jMMSE for different window sizes

The noisy data from Figure 5(c) denoised using the jMMSE with $T = 3$ to $T = 13$ (without an application of BM3D). We can see that the data becomes increasingly less noisy. Same as Figure **??**, but now we show the difference with respect to the original SO_2 input data. We notice diminishing returns regarding noise removal as T grows.

A1 SO_2 emission estimates

In the main body of the paper, we display the emission estimate ratios between the 'noisy' divergence and denoised SO_2 divergence estimates. Here, we provide the actual estimates we obtained in the form of a heatmap. Some emission estimates are negative, specifically for Newcastle steel works, Camden power station, and Kelvin power station. This indicates that the integration radius may be inadequate or that an important nearby sink was overlooked. As noted in the main body, a more thorough study, with improved AMF corrections and carefully chosen integration ranges for each source, would likely yield more reliable numbers. However, this is beyond the scope of the current paper.

Heatmap of SO_2 emission estimates. The abbreviation SW stands for 'steel works' and 'PS' stands for power station.

Difference plot of the divergence, by subtracting the SO_2 divergence map, reproduced in panel (a), from the denoised estimates shown in Figure 6. We can see that the 5×5 filter shows considerable loss of signal, while the BM3D and BM3D+jMMSE methods, in their difference plots, essentially just show a noise reduction without affecting the sources (apart from some residual 'red' spots, which indicate that the signal has been *boosted* by the denoising process).

Author contributions. EK derived and implemented the jMMSE algorithm and BM3D method modification. EK, GK, and DB all contributed equally to the writing process.

Competing interests. Two of the co-authors are a member of the editorial board of *Atmospheric Measurement Techniques*. The contact author has declared that none of the authors has any other competing interests.

Acknowledgements. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 958927 (CoCO2) and the European Union's Horizon Europe programme under grant agreement no. 101082194 (CORSO), with additional funding by the Swiss State Secretary for Education, Research and Innovation (SERI) no. 22.00422.

References

Beirle, S., Borger, C., Dörner, S., Li, A., Hu, Z., Liu, F., Wang, Y., and Wagner, T.: Pinpointing nitrogen oxide emissions from space, Science advances, 5, eaax9800, 2019.

Beirle, S., Borger, C., Jost, A., and Wagner, T.: Improved catalog of NO_x point source emissions (version 2), Earth System Science Data, 15, 3051–3073, 2023.

Buades, A., Coll, B., and Morel, J.-M.: A review of image denoising algorithms, with a new one, Multiscale modeling & simulation, 4, 490–530, https://doi.org/10.1137/040616024, 2005.

Buades, A., Coll, B., and Morel, J.-M.: Non-local means denoising, Image Processing On Line, 1, 208–212, https://doi.org/10.5201/ipol.2011.bcm_nlm, 2011.

Buchwitz, M., Reuter, M., Bovensmann, H., Pillai, D., Heymann, J., Schneising, O., Rozanov, V., Krings, T., Burrows, J., Boesch, H., et al.: Carbon Monitoring Satellite (CarbonSat): Assessment of atmospheric CO 2 and CH 4 retrieval errors by error parameterization, Atmospheric Measurement Techniques, 6, 3477–3500, 2013.

Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K.: Image denoising by sparse 3-D transform-domain collaborative filtering, IEEE Transactions on image processing, 16, 2080–2095, https://doi.org/10.1109/TIP.2007.901238, 2007.

European Space Agency (ESA): Copernicus Sentinel-5P TROPOMI Level 2 Sulfur Dioxide Total Column Products, https://doi.org/10.5270/S5P-74eidii, 2020.

European Space Agency (ESA): Copernicus Sentinel-5P TROPOMI Level 2 Nitrogen Dioxide total column products, https://doi.org/10.5270/S5P-9bnp8q8, 2021.

Fichtner, A.: Lecture Notes on Inverse Theory, Tech. rep., ETH Zurich, Switzerland, https://doi.org/10.33774/coe-2021-qpq2j, 2021.

Hakkarainen, J., Ialongo, I., Koene, E., Szeląg, M. E., 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, 878 731, 2022.

Immerkaer, J.: Fast noise variance estimation, Computer vision and image understanding, 64, 300–302, 1996.

Koene, E. F. M., Brunner, D., and Kuhlmann, G.: On the theory of the divergence method for quantifying source emissions from satellite observations, Journal of Geophysical Research: Atmospheres, 129, https://doi.org/https://doi.org/10.1029/2023JD039904, 2024.

Krol, M., van Stratum, B., Anglou, I., and Boersma, K. F.: Evaluating NO_x stack plume emissions using a high-resolution atmospheric chemistry model and satellite-derived NO_2 columns, Atmospheric Chemistry and Physics, 24, 8243–8262, https://doi.org/10.5194/acp-24-8243-2024, 2024.

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, Atmospheric Measurement Techniques, 12, 6695–6719, https://doi.org/10.5194/amt-12-6695-2019, 2019.

Kuhlmann, G., Clément, V., Marshall, J., Fuhrer, O., Broquet, G., Schnadt-Poberaj, C., Löscher, A., Meijer, Y., and Brunner, D.: SMARTCARB – Use of satellite measurements of auxiliary reactive trace gases for fossil fuel carbon dioxide emission estimation, https://doi.org/10.5281/zenodo.4034266, 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://doi.org/10.5281/zenodo.4048228, 2020b.

Kuhlmann, G., Henne, S., Meijer, Y., and Brunner, D.: Quantifying CO_2 emissions of power plants with CO_2 and NO_2 imaging satellites, Frontiers in Remote Sensing, p. 14, 2021.

Ledoit, O. and Wolf, M.: The power of (non-)linear shrinking: A review and guide to covariance matrix estimation, Journal of Financial Econometrics, 20, 187–218, 2022.

Mäkinen, Y., Azzari, L., and Foi, A.: Collaborative filtering of correlated noise: Exact transform-domain variance for improved shrinkage and patch matching, IEEE Transactions on Image Processing, 29, 8339–8354, 2020.

Meier, S., Koene, E. F., Krol, M., Brunner, D., Damm, A., and Kuhlmann, G.: A lightweight NO 2-to-NO_x conversion model for quantifying NO_x emissions of point sources from NO_2 satellite observations, Atmospheric Chemistry and Physics, 24, 7667–7686, 2024.

Miller, C., Crisp, D., DeCola, P., Olsen, S., Randerson, J., Michalak, A., Alkhaled, A., Rayner, P., Jacob, D. J., Suntharalingam, P., et al.: Precision requirements for space-based X_{CO_2} data, Journal of Geophysical Research: Atmospheres, 112, https://doi.org/10.1029/2006JD007659, 2007.

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, Atmospheric Chemistry and Physics, 19, 9371–9383, 2019.

Veefkind, J. P., Aben, I., McMullan, K., Förster, H., De Vries, J., Otter, G., Claas, J., Eskes, H., De Haan, J., Kleipool, Q., et al.: TROPOMI on the ESA Sentinel-5 Precursor: A GMES mission for global observations of the atmospheric composition for climate, air quality and ozone layer applications, Remote sensing of environment, 120, 70–83, 2012.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P.: Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing, 13, 600–612, 2004.

Yahya, A. A., Tan, J., Su, B., Hu, M., Wang, Y., Liu, K., and Hadi, A. N.: BM3D image denoising algorithm based on an adaptive filtering, Multimedia Tools and Applications, 79, 20 391–20 427, https://doi.org/10.1007/s11042-020-08815-8, 2020.