ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Harvard John A. Paulson
School of Engineering
and Applied Sciences

Master Thesis

Deep learning methods to detect methane plumes in Sentinel-2 satellite imagery: application to the Permian oil and gas basin

Atmospheric Chemistry Modeling Group - Harvard University

Spring Term 2023

Supervised by:
Prof. Konrad Schindler
Daniel J. Varon

Author:
François Martin-Monier

Declaration of Originality

I hereby declare that the written work I have submitted entitled

Deep learning methods to detect methane plumes in Sentinel-2 satellite imagery: application to the Permian oil and gas basin

is original work which I alone have authored and which is written in my own words.[1]

Author

François	Martin-Monier

Student supervisor

Daniel	Varon

Committee members

Daniel	Varon
Konrad	Schindler

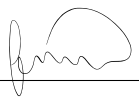
Supervising lecturer

Marco	Hutter

With the signature I declare that I have been informed regarding normal academic citation rules and that I have read and understood the information on 'Citation etiquette' (`https://www.ethz.ch/content/dam/ethz/main/education/rechtliches-abschluesse/leistungskontrollen/plagiarism-citationetiquette.pdf`). The citation conventions usual to the discipline in question here have been respected.

The above written work may be tested electronically for plagiarism.

Paris, 12/09/2023
Place and date


Signature

[1] Co-authored work: The signatures of all authors are required. Each signature attests to the originality of the entire piece of written work in its final form.

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I am grateful towards Daniel J. Varon, for supervising my work and offering me the possibility to work with him in the Jacob Group at Harvard University. I am particularily thankful for the time he dedicated to this project and the insightful feedback I received during our weekly meetings. Together with Jack Bruno and Marc Watine, with whom I was also grateful to work with, we held weekly meetings which where key in advancing this work and completing the objectives I had set myself.
I would also like to thank Professor Daniel J. Jacob for hosting me in his research group for 6 months, and allowing me this opportunity. Along with the rest of the Jacob Group, he always made me feel welcome and included in the lab.
I am also deeply appreciative of Professor Konrad Schindler's supervision throughout this project. He has always been available and provided key insights from his expertise in computer vision and remote sensing.
Furthermore, I would like to thank Professor Marco Hutter for enabling me to work on the topic I was passionate about.
Finally, I am thankful towards Nick Balasus for allowing me to discover Cambridge and Boston by bike, Marc Watine and Elfie Roy for being fantastic office companions and Emma Darniche for supporting me in writing my Master Thesis on the other side of the Atlantic.
I also place on record, my sense of gratitude to one and all, who directly or indirectly, have lent their hand in this venture.

François Martin-Monier

Abstract

Anthropogenic methane emissions from large industrial facilities can be detected by remote sensing instruments. In this study, we develop a building block for a systematic monitoring pipeline of methane point sources. A dataset of Sentinel-2 observations of oil and gas facilities in the Permian basin is created, with simulated methane plumes embedded in the images. A dataset of over 9000 images, labelled according to standard computer vision formats is delivered, with capabilities of extending the methodology to other regions and with a larger variety of synthetic methane plumes. We also develop a U-Net based methane plume detector operation from raw radiances with minimal preprocessing. We show that introducing Normalized Difference Methane Index (NDMI) as an input feature to our network increases $F-0.5$ score on the validation set by 10.1%. Furthermore, preprocessing the NDMI according to Z-score normalization improves $F-0.5$ score on the validation set by 23.4%. When evaluating on real plumes from the Permian basin, 42% of plumes found by existing physics based methods are detected. This demonstrates the viability of training on synthetic methane plumes for deployment on real data but also highlights the need for further optimization of our algorithm and input data. We also find that training on NDMI as the only methane sensitive feature leads to a higher detection threshold but no false positives, a promising feature for a high throughput facility monitoring computer vision pipeline.

Symbols

Symbols

$F_{0.5}$	F_β-score with $\beta = 0.5$
F_1	F_β-score with $\beta = 1$

Acronyms and Abbreviations

ETH	Eidgenössische Technische Hochschule
ACMG	Atmospheric Chemistry Modeling Group
GWP	Global Warming Potential
CH4	Methane
CO2	Carbon Dioxide
H2O	Water
IPCC	Intergovernmental Panel on Climate Change
SWIR	Short Wave Infrared
NIR	Near Infrared
RGB	Red-Green-Blue
S2	Sentinel-2
MSI	Multi-Spectral Instrument
L1C	Level 1C data
WRF	Weather Research and Forecasting Model
LES	Large Eddy Simulation
TOA	Top Of Atmosphere
TOASR	Top Of Atmosphere Spectral Radiance
VCD	Vertical Column Density
SZA	Solar Zenith Angle
SAA	Solar Azimuth Angle
VZA	Viewing Zenith Angle
VAA	Viewing Azimuth Angle
AMF	Air Mass Factor
GEE	Goole Earth Engine
API	Application Programming Interface
NDI	Normalized Difference Indices
NDMI	Normalized Difference Methane Index
NDVI	Normalized Difference Vegetation Index

NDBI	Normalized Difference Built-up Index
BSI	Bare Soil Index
Bx	Sentinel-2's x^{th} band
OGIM	Oil and Gas Infrastructure Mapping Database
od	optical depth
MBMP	Multi Band Multi Pass
SNR	Signal to Noise Ratio
COCO	Common Objects in COntext
CNN	Convolutional Neural Network
FCN	Fully Convolutional Network
GAN	Generative Adversial Network
IoU	Intersection over Union/Jaccard score
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negatives

Chapter 1

Introduction

1.1 Motivation

Atmospheric methane accounts for about one third of anthropogenic greenhouse gas warming since pre-industrial times [19]. Given methane's short lifetime in the atmosphere compared to CO2 [41], reducing methane emissions is a strong lever for effective near-term reduction of climate warming.

Recent approaches to reducing methane emissions have leveraged the gas' absorption in the short wave infrared (SWIR) spectrum and the presence of this absorption band in numerous remote sensing instruments to detect methane emissions from Space. Given anthropogenic methane emissions can be traced back to individual point sources from large industrial facilities, improving our ability to monitor methane point sources with satellite instruments would open new opportunities for climate change mitigation and targeted policymaking.

Amongst the largest sources of anthropogenic methane emissions, the oil and gas sector represents a compelling target for methane emissions reduction. In this sector, emissions could be reduced by over 75% with existing technoloigies [19] with around 50% of the total emissions abatement coming at no additional cost due to methane emission mitigation technologies' ease of access and inherent value of the gas [37].

The Permian basin, seing some of the largest increases in production in the last decade [49], provides a unique opportunity to interpret regional emission trends in terms of facility-level emissions. This region also features moderately complex ground surfaces that have so far impeded large-scale methane plume detection with satellite instruments.

1.2 Aim

A compelling aspect of satellite observations of atmospheric methane is the potential for global spatial and temporal completeness [24]. The Sentinel-2 twin satellites provide just this capability, with frequent revisit (¡5 days), fine resolution (20m pixel resolution) and open access [50].

Systematic monitoring of emitting facilities using such instruments would offer new capabilities for targeted emissions reduction. For this, artificial intelligence approaches to detecting methane point sources are gaining traction, yet still struggle

with complex backgrounds and require expensive preprocessing.

We aim to build on these previous AI-based methods to develop a methane plume detector for Sentinel-2 data, requiring no expensive preprocessing and targeted for the more complex Permian basin. Given the lack of labeled data in this region to train a machine learning algorithm on, an important first step is to create a dataset of synthetic, labeled methane plumes that can be shared with other researchers. Hence, our goal is threefold:

- Creating a dataset of real Sentinel-2 observations of oil and gas facilities in Permian basin, containing simulated methane plumes
- Developping an algorithm to detect methane plumes from raw radiances in a heterogeneous environment
- Evaluating the simulation to reality gap by testing the detector on real plumes from the Permian basin

Chapter 2

Background and Literature Review

2.1 Methane: A Significant Contributor to Global Warming

Since pre-industrial times, anthropogenic greenhouse gas warming can be mainly attributed to methane (CH4) and carbon dioxide (CO2) emissions. Atmospheric methane accounts for about one third of such warming while CO2 accounts for over half [19]. The influence of various greenhouse gases on the climate hinges on two fundamental attributes: their atmospheric persistence and their capacity to trap energy. Unlike carbon dioxide (CO2), which lingers for centuries, methane has a significantly shorter atmospheric lifespan, approximately 12 years. Nevertheless, during its presence in the atmosphere, methane exhibits far greater energy-absorbing capability [41]. Reported to short term global warming potential (20 year GWP), methane has approximately 84 times the 20 year GWP of CO2 [19]. Hence, given methane's short lifetime in the atmosphere and its significant impact on radiative forcing, reducing methane emissions is a important lever for effective near-term reduction of climate warming.

2.2 Methane: a Compelling Target for Climate Change Mitigation

Unlike CO2 emissions, which mainly arise from diffuse urban hotspots and a relatively small number of large point sources (power plants), anthropogenic methane emissions can be traced back expansive industrial facilities. The crux of effective mitigation strategies lies in discerning the precise source of these emissions. In figure 2.1, we observe that methane emissions mostly come from industries such as oil and gas, waste management, agricultural operations, and other industrial activities. All of these industries operate large scale facilites/plants whose locations can be accurately reported.

In this work, we elect to focus on detecting emissions from the oil and gas sector for several reasons. First of all, those emissions are not a byproduct of this sector's operations but the product itself. In this sector, emissions could be reduced by over 75% with existing technoloigies [19] with around 50% of the total emissions

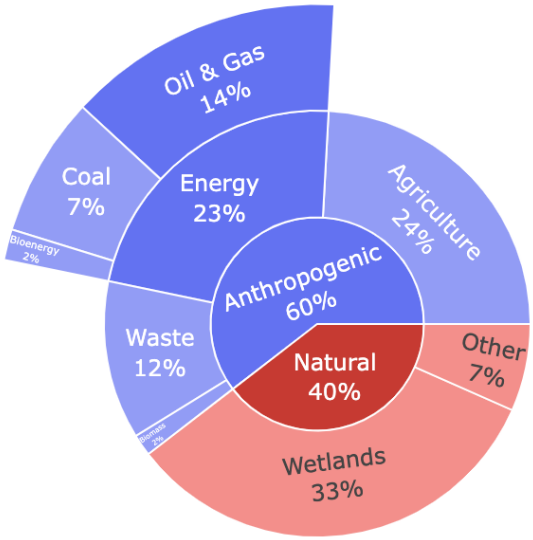


Figure 2.1: Global methane emissions breakdown by source [44]

abatement coming at no additional cost due to methane emission mitigation technologies' ease of access and inherent value of the gas [37]. Hency why for this sector, mitigating methane emissions has an inherent value [19].
In addition, methane emissions from the oil and gas sector are usually large plumes that can be attributed to a specific location. Compared with agricultural our waste management sources where emissions are spread over multiple points in a facility, this makes methane emissions from the oil and gas sector easier to detect. Finally, the oil and gas sector is well studied and numerous sources of data exist on emissions, actors and production facilities ([39], [2], [49], [8]. This facilitates building a plume detection pipeline.

2.3 Permian Oil and Gas Basin: a Significant Methane Emissions Hotspot

As the largest natural gas producing country in the world [1], working on technologies mitigating methane emissions for the oil and gas sector in the United-States is key key to reducing emissions worldwide. In the United States, the Permian basin is the second largest natural gas production basin after the Appalachia basin as shown in figure 2.2. Its bright desertic terrain and relatively uniform topography make it an easier basin to study than the Appalachia basin. Indeed, previous studies have shown the correlation between terrain complexity and methane plume detection limits from satellites [17].

The 2010s United States Shale Boom is another motivation for focusing on the Permian basin [49]. As shown in 2.3, the rapid increase in shale gas extraction in the Permian basin coincides with the launch of the Sentinel-2 constellation [7]. Having the right sensor at the right time motivates us to produces tools that could provide a better picture of the impact of the natural gas extraction on methane emissions.
Finally, previous studies and surveys of methane emissions in the Permian basin provide a complete picture of where methane plumes are emitted from in the region

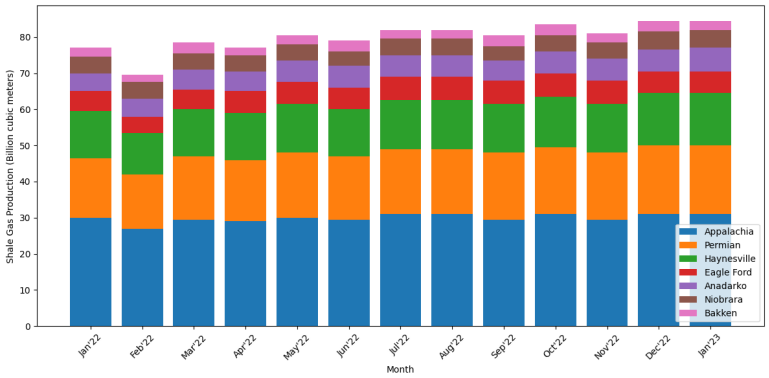


Figure 2.2: Production of shale gas in the United States by basin, Jan 2022-Jan 2023 [47]

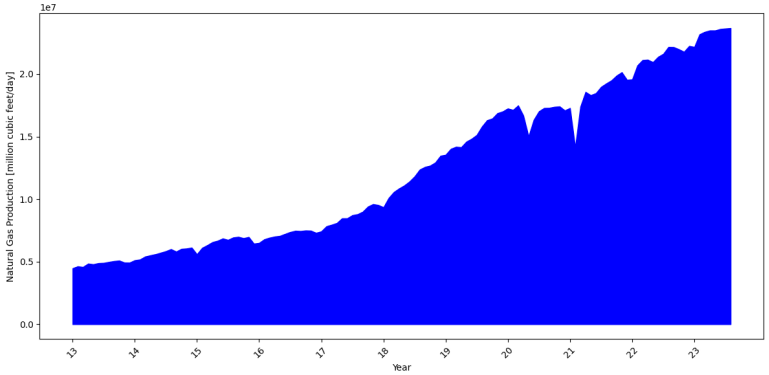


Figure 2.3: Natural gas prodcution in the Permian basin 2013-2023 [49]

[8]. Thus, we can target the training of a machine learning algorithm towards such facilities and the terrain in their surroundings.

2.4 Remote Sensing to Tackle Methane Emissions

National greenhouse gas emission inventories are built using bottom-up estimates in which emissions are tied to underlying processes. These inventories inform climate policy but are highly uncertain [24]. Satellite observations of atmospheric methane can provide top-down estimates of emissions, complementary to the bottom-up ones [40].

To establish top-down estimates of methane emissions, there exists a myriad of possible sensors, as shown in figure 2.4.
We can split these instruments in two categories:

- **Area flux mappers**: To measure methane emissions on a regional to global scale. Area flux mappers combine satellite measurements of methane concentrations with atmospheric modeling to estimate the amount of methane being emitted from different regions on the Earth's surface
- **Point source imagers**: To measure emissions from specific point sources by detecting a gas plume. Point source imaging instruments are sensitive to

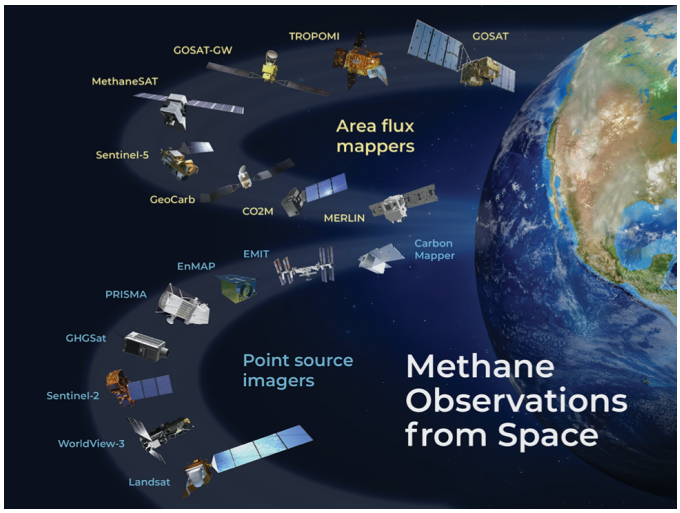


Figure 2.4: Remote sensing instruments for observation of methane from Space [24]

methane concentrations and can pinpoint areas where methane emissions are significantly higher than the surrounding background levels. In the context of this thesis, these are the instruments of interest

Zooming in on point source imagers, we distinguish two types of instruments:

- **Hyperspectral imagers**: To capture a very large number of narrow and contiguous spectral bands across a wide range of the electromagnetic spectrum. This high spectral resolution allows them to provide highly detailed information about the spectral signatures of different gases, including methane. Each individual band corresponds to a specific wavelength, and by analyzing the intensity of light at each wavelength, we can identify the unique spectral "fingerprint" of methane. In the context of methane point source detection, hyperspectral instruments can help distinguish methane from other gases more accurately due to their ability to detect specific absorption features associated with methane. These instruments are often use for targeted observations, to focus on specific areas with high spectral detail and come with higher costs. PRISMA is an example of an open-access hyperspectral instrument

- **Multispectral imagers**: To capture data in a few select and wider spectral bands rather than the large number of fine bands that hyperspectral instruments record. These instruments are designed to capture data in specific bands that are known to be sensitive to the absorption or reflection of certain gases, including methane. Capturing data over a wider spectrum introduces more noise when trying to capture a unique spectral "fingerprint". Multispectral instruments often have global coverage, providing few-days revisit time for any point in its given latitude band. The Sentinel-2 constellation and Landsat 8 are examples of multispectral sensors with an open-data access policy

Short and consistent revisit time is key in establishing a systematic facility level monitoring pipeline, which is why we focus on multispectral instruments.
The Sentinel-2 twin satellites provide just this capability, with frequent revisit (≤ 5 days), fine resolution (20m pixel resolution) and open access [50]. Methane column enhancements of individual plumes can be derived from the SWIR bands of the Multi-Spectral Instrument (MSI) onboard both satellites.

2.5 Methane Point Sources

Methane gas absorbs in the Short Wavelength Infrared spectrum (1.4 μm - 3 μm). For a sensor equiped with a SWIR instrument, methane rich pixels will exhibit reduced reflectance in the absorption bands. The specific absoprtion spectrum of methane can be seen in figure 2.5.

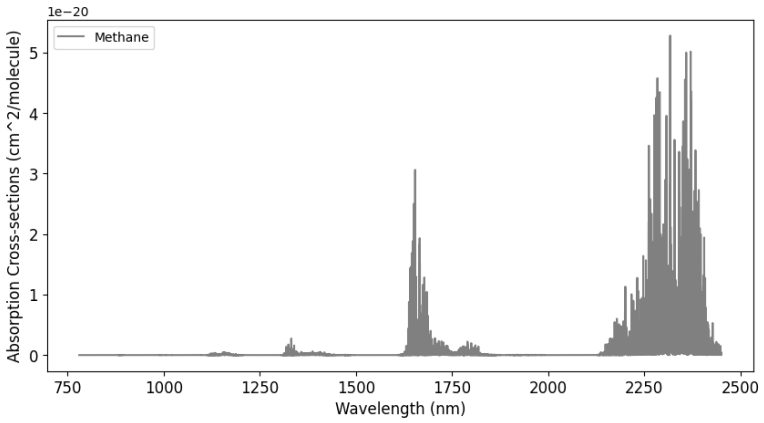
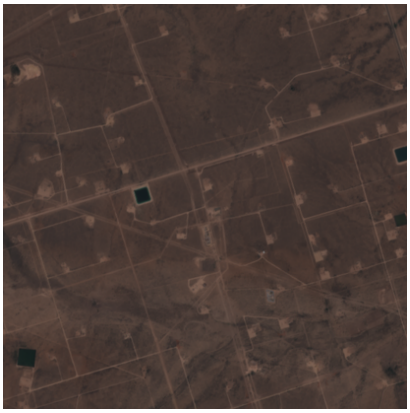


Figure 2.5: Methane absoprtion spectrum in NIR/SWIR wavelengths [16]

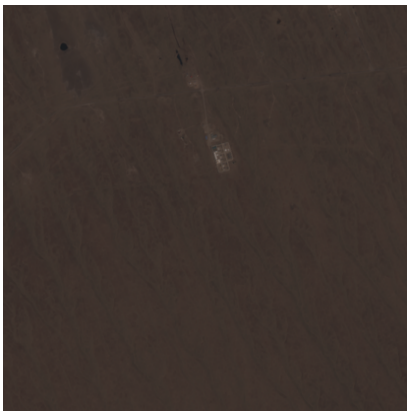
2.5.1 Remote Sensing for Observing Methane Point Sources

The first time an orbiting remote sensing instrument measured methane emissions from a point source was in 2016, with the Aliso Canyon blowout [48]. Since the feasibility of detecting methane emissions from space was proven, several methane point source imaging missions have been planned. Further research by Varon et al. [50] has also developped methods to detect methane plumes in existing satellite constellation such as Landsat and Sentinel-2. These unintended capabilities from existing sensors have opened up the field of methane detection from Space based sensors. Such works on Sentinel-2 data have shown that physics-based retrievals can empirically detect point sources down to about 2 t/h, with a strong dependency on surface properties, with vastly better performances over bright, homogeneous scenes such as deserts [17]. Subsequent works, including by Ehret et al., 2023 [**?**] have improved the accuracy of such physics-based retrievals although they still require some manual work and may be inaccurate [12].

Artificial intelligence approaches to detecting and quantifying methane point sources are also gaining traction. In a preprint study, Bruno et al. [4] use a U-Net based architecture [43] to detect and quantify methane plumes in methane retrieval fields (L2) extracted from a hyperspctral instrument. In a separate contribution, Joyce et al. [27], introduce a multi-tiered deep learning approach to detecting methane plumes and quantifying their concentration and emission rate. Their approach takes as input raw radiances (L1C) of PRISMA, a hyperspectral sensor. Both of the above approaches train on real scenes with synthetic methane plumes embedded in the bands. The synthetic plumes are generated using the WRF-LES model [54].
In the context of multispectral instruments, Vaughan et al. [52] develops a plume detector operating on Sentinel-2 L1C data (raw radiances). Once again, they use a U-Net skeleton and train on 925 confirmed methane plumes from Turkmenistan. This novel approach bypasses the necessity for methane retrieval fields when train-



(a) Park compressor station, Permian basin



(b) Korpezhe compressor station, Turkmenistan

Figure 2.6: Summer month observation of scenes in Turkmenistan and Permian basin

ing on multispectral data, consequently eliminating the need for potential manual preprocessing work on the data.

2.6 Significance and Innovation

We develop a deep learning pipeline to indentify synthetic methane plumes from Sentinel-2's L1C product over the Permian basin, using the WRF-LES model to simulate synthetic plumes. The significance of this work is in:

- Training a methane plume detector on multispectral L1C data with embedded synthetic plumes.
  While training on synthetic plumes necessarily introduces a simulation-to-reality gap, this allows us to train on much larger volumes of data, learn deeper feature representations. We also establish a methodology for detecting plumes in under-studied geographies

- Focusing on the Permian oil and gas basin, where the heterogeneous background tends to produce noisier retrieval fields.
  We know from previous studies that the detection threshold for Sentinel-2 is highly dependant on the background hetorogeneity and reflectivity [17]. Figure 2.6 displays an example of a natural compressor station for both regions. While the Turkmensitan facility is located in a homogeneous desertic background, the Permian basin one is surrounded by fields, roads, other buildings and different types of soils. We expect more clutered backgrounds to introduce challenges with artifacts

Chapter 3

Creating a dataset of synthetic methane plumes

Training computer vision algorithms to represent features in an image requires large amounts of labelled data. Due to significant differences between satellite sensors, this labelled data should come from the satellite of interest of this study - Sentinel-2. There is no such large open-source dataset of labelled methane plumes. In this section, we cover the process of creating a dataset of synthetic methane plumes. This dataset should:

- Cover the variability of scenes in the Permian basin
- Focus on methane emitting infrastructure in the Permian basin
- Include a diverse and realistic representation of methane plumes, including source rate and shape
- Correspond to a standardized computer vision format to encourage usability of the dataset by other researchers

The process for building a dataset of synthetic methane plumes is presented in figure 3.16. In this chapter, we cover the various data sources, steps and design choices outlined in the pipeline. At the end of the chapter, we demonstrate that this method is successfull at producing synthetic data with a controlled concentration of methane embedded in Sentinel-2 raw radiances.

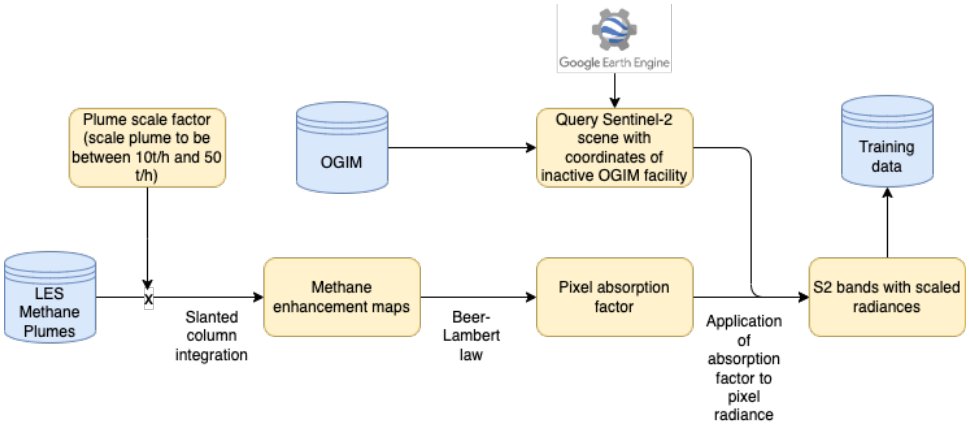


Figure 3.1: Pipeline for embedding synthetic methane plumes in Sentinel-2 imagery

3.1 Feature Selection

3.1.1 Selecting a Sensor

Several satellites have proven capabilities of observing methane point sources in the shortwave infrared. We can divide point sources imagers in two categories: tasked and global sensors. Tasked sensors cover a specific area at the request of a user. Global sensors overpass any given site at a given revisit rate. The goal of this thesis being the development of an automated workflow for detecting methane plumes, we need to strike a balance between:

- **Spectral and spatial resolution**: Higher resolutions translate to lower detection thresholds [24]
- **Revisit rate**: The greater the frequency of revisits, the increased number of observations we collect at a specific location, thereby enhancing our likelihood of detecting emissions
- **Open access to data**: For the development of a machine learning algorithm, vast quantities of data are necessary. Using free data is the most realistic option for this master thesis

The only two point source imagers that have an open access policy and a global coverage are Landsat-8 and Sentinel-2. While both have similar spatial and spectral resolutions, Sentinel-2 has a much lower revisit rate than Landsat-8 (5 days versus 16 days) as Sentinel-2 is a 2 satellite constellation. This is why we choose Sentinel-2 for this study.

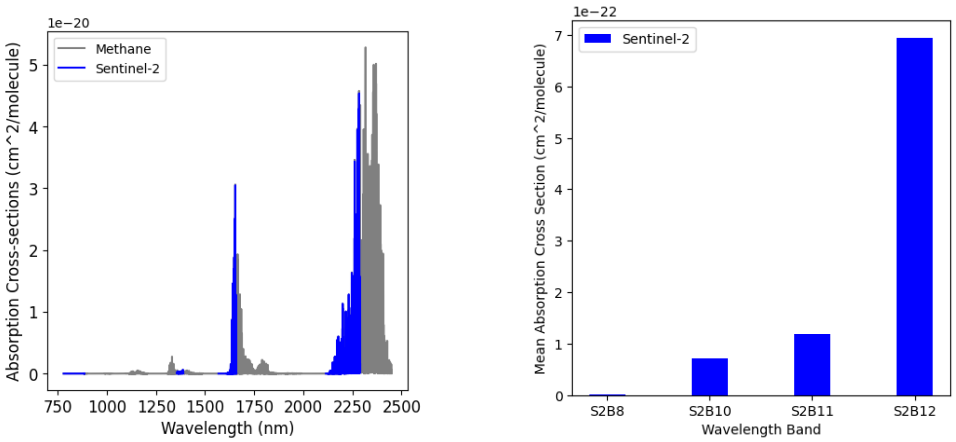
3.1.2 Selecting Sentinel-2 Methane Bands

Sentinel-2 is a multispectral instrument with a total of 13 bands. We give an overview of those bands in table 3.1.

Band	Resolution [m]	Central Wavelength [nm]	Description
B1	60	443	Ultra Blue
B2	10	490	Blue
B3	10	560	Green
B4	10	665	Red
B5	20	705	Visible and Near Infrared
B6	20	740	Visible and Near Infrared
B7	20	783	Visible and Near Infrared
B8	10	842	Visible and Near Infrared
B8a	20	865	Visible and Near Infrared
B9	60	940	Short Wave Infrared
B10	60	1375	Short Wave Infrared
B11	20	1610	Short Wave Infrared
B12	20	2190	Short Wave Infrared

Table 3.1: Sentinel-2 bands [14]

To embed simulated methane plumes in a given band, we need to calculate the per-pixel absorption in this band using the Beer-Lambert law and a radiative transfer model. As this is computationally expensive, we select a subset of the bands where we can expect a significant methane absorption and only embed plumes in these bands.



(a) S2-CH4 absorption spectrum overlap in infrared

(b) Mean SWIR S2-CH4 absorption cross section

Figure 3.2: Absorption cross sections between CH4 and Sentinel-2 absorption spectrums

To get a first sense of interesting Sentinel-2 bands, we plot the methane and Sentinel-2 absorption spectrums in the infrared domain. The resulting plot is in figure 3.2 a). From this figure, we deduct that bands B12 and B11 are good candidates to embed methane plumes, confirming findings in previous works [50]. There also seems to be an overlap between Sentinel-2's band B10 and a portion of the methane absorption spectrum centered around 1375 nm.
To further investigate this observation, we compute the mean absorption cross section between Sentinel-2 bands and the methane absorption spectrum. These results can be found in figure 3.2 b). These findings confirm the potential of band B10. While band 12 has the highest absoprtion by a factor of approximately 6 compared to band B11, band B10's absorbtion is 60% of band B11. This motivates us to embed simulated methane plumes in band B10 in addition to bands B11 and B12. When downloading S2 data, we also select bands B1, B2, B3, B4 and B8. B2, B3 and B4 are selected to get the visible information for a given scene. This should help to eliminate visible artifacts from a methane enhacement field. B8 is selected to calculate usefull ratios, covered in section 3.1.3. Data is downloaded using the Google Earth Engine (GEE) API.

3.1.3 Deriving Features

From the Sentinel-2 bands downloaded through the GEE API, we can compute useful normalized difference indices (NDI). NDIs are commonly used in remote sensing studies for several reasons:

- **Enhancing feature discrimination**: NDIs are designed to enhance the contrast between certain features that can be difficult to distinguish in individual spectral bands. This can make it easier to identify and differentiate specific land cover types, vegetation health, and other relevant attributes
- **Removing atmospheric and illumination effects**: NDIs mitigate effects of variations in atmospheric or illuminations conditions such as sun angle, atmospheric haze or aerosols
- **Compatibility with various sensors**: By computing a relative difference

between two bands, NDIs reduce the effect of central wavelength shift for a given band. This helps in comparing scenes across different sensors. This can be useful in the event were this study is generalized to other satellite constellation but key here are differences within a constellation. Sentinel-2 is a 2 satellite constellations and as shown in figure 3.3, S-2A and S-2B have slight differences in their instruments.

- **Analysis across time**: By normalizing differences between bands, NDIs reduce the overall variation of any given scene which in turn encourages comparison of data across time

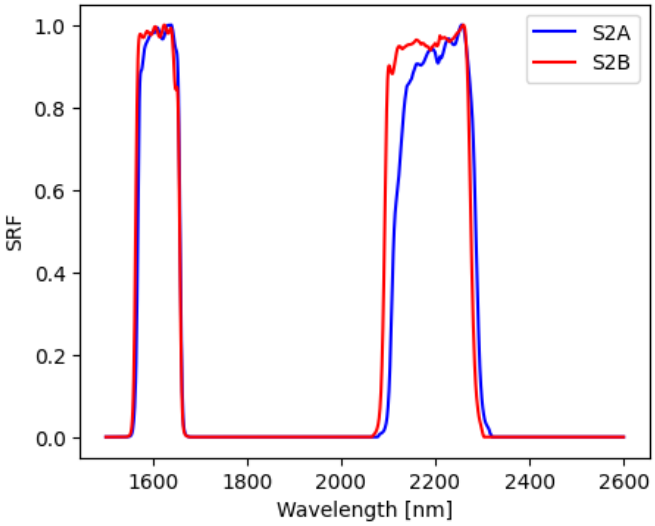


Figure 3.3: Spectral response functions for both Sentinel-2 satellites in SWIR domain [7]

With these benefits in mind, we compute the following NDIs for inclusion in our dataset:

- **Normalized Difference Vegetation Index (NDVI)**: Indicator for vegetation content. A high NDVI translates to high absorption in the near-infrared spectrum which in turn translates to higher vegetation density and health [23]

$$NDVI = \frac{B8 - B4}{B8 + B4} \tag{3.1}$$

- **Normalized Difference Built-up Index (NDBI)**: Indicator used to emphasize built up area [55]

$$NDBI = \frac{B11 - B8}{B11 + B8} \tag{3.2}$$

- **Bare Soil Index (BSI)**: Indicator used to characterize soil variations. Bands B11 and B4 represent soil mineral composition. Bands B8 and B2 indicate presence of vegetation [10]

$$BSI = \frac{(B11 + B4) – (B8 + B2)}{(B11 + B4) + (B8 + B2)} \tag{3.3}$$

- **Normalized Difference Methane Index (NDMI)**: Indicator comparing Sentinel-2's 2 SWIR bands and emphasizing methane. A feature appearing strongly in band 12 but not in band 11 will be brought forward by this ratio. As both bands have close central wavelengths, they should have similar features and the ratio should be close to zero. From figure 3.2 (b), we know that methane absorbs strongly in band 12 but not as much in band 11. Hence, a deviation from 0 in this ratio is a prime candidate for a methane plume.
  Sentinel-2 does however have coarse spectral resolution. Species such as CO2 and water vapor have similar absorption spectrums to methane and could therefore be artifacts. Varon et al. [50] shows that this effect is negligible as CO2 and water vapor are rarely co-emitted with methane point sources. They demonstrate that these two chemical species can be considered uniform across a scene.

$$NDMI = \frac{B12 - B11}{B11 + B12} \tag{3.4}$$

  NDMI is first introduced in Webber et al. [53] for application to HyTES, a hyperspectral instrument. NDMI is also applied to multispectral data (Landsat) in He et al., (submitted). Unlike NDVI, where high values indicate higher likelihood of vegetation, low and high values of NDMI can both indicate presence of methane depending on wether a plume emits more thermal energy than it absorbs [53].

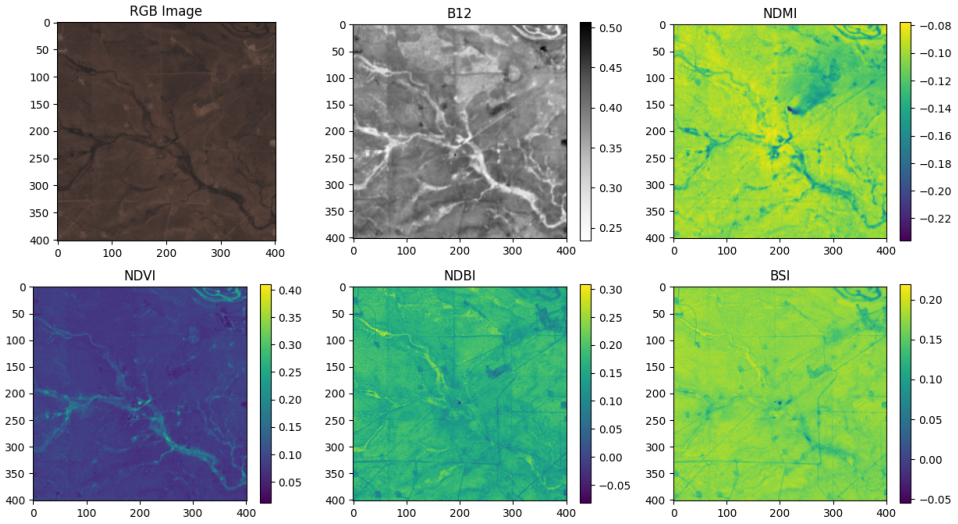


Figure 3.4: Normalized difference indices

3.2 Selecting Scenes

Anthropogenic methane emissions can be traced back to many individually small point sources from numerous industrial sectors, including fossil fuel production and distribution facilities, agricultural operations and waste treatment facilities. Monitoring these facilities offers a unique opportunity for methane emissions abatement. Several studies have assembled databases of methane emitting facilities. METER-ML [57] consists of close to 90'000 georeferenced images of methane emitting facilities across the United States. These facilities range across various industries such as

energy, waste water treatment of landfills. The Oil and Gas Infrstructure Mapping database (OGIM) [39] is a global geospatial database focusing on the oil and gas sector. OGIM uses data from over 450 publicly available datasets. This dataset is particularly suited for the study of oil and gas production in North America as approximately 85% of the data points are in Northern American countries. Moreover, close to 100% of the data for North America is obtained from government sources, making for more reliable data. These factors render the OGIM database well suited for our study of the Permian basin, spanning across Texas and New Mexico.

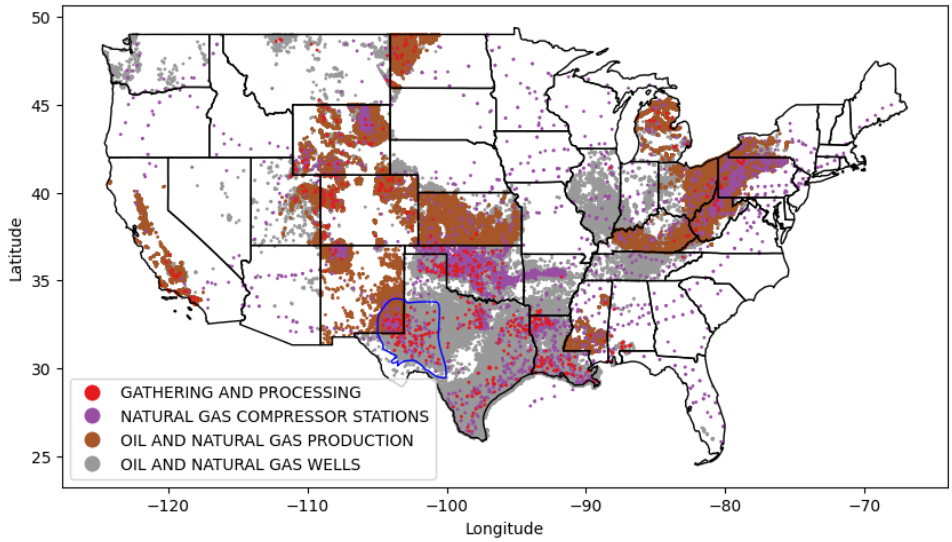


Figure 3.5: Production, compressor and processing OGIM facilities in the contiguous United States. Permian basin outlined in blue [39]

OGIM covers upstream (production and gathering), midstream (transport and storage) and downstream (product transformation) facilities. For the purpose of narrowing the focus of this study, we decide to exclude pipelines from the training data. Table 3.2 provides a breakdown of the facilities identified in the Permian basin.

Category	Stream	Count	Fraction of total
Oil and natural gas wells	upstream	200363	72.9%
Oil and natural gas production	upstream	73973	26.9%
Gathering and processing	midstream	100	0.05%
Natural gas compressor stations	midstream	261	0.1%
Petroleum terminals	midstream	28	0.05%
Crude oil refineries	downstream	2	∼0%

Table 3.2: OGIM facility count by type

From table 3.2, we observe a strong bias towards upstream facilities[1]. While we could sample from the OGIM dataset at random, this would mean most of the sampled facilities would be upstream facilities. Recent surveys of the Permian

[1]The OGIM database seperates oil and natural gas wells and production facilities. Upon further inspection of the data, we find that both categories represent similar facilities with different naming across state borders: all facilities falling in the oil and natural gas production category are located in New Mexico while most of the oil and natural gas wells facilities are located in Texas (for the Permian basin).

basin provide information on the sources of large emission events [8]. Results from a 2019 airborne survey by Cusworth et al. can be found in table 3.3 and figure 3.6.

Source type	Stream	Contribution to emissions $> 500kg/h$
Well	upstream	20.6%
Compressor	midstream	25.6%
Processing	midstream	12.4%
Tank	midstream	41.4%

Table 3.3: Emission contributions by source type [8]

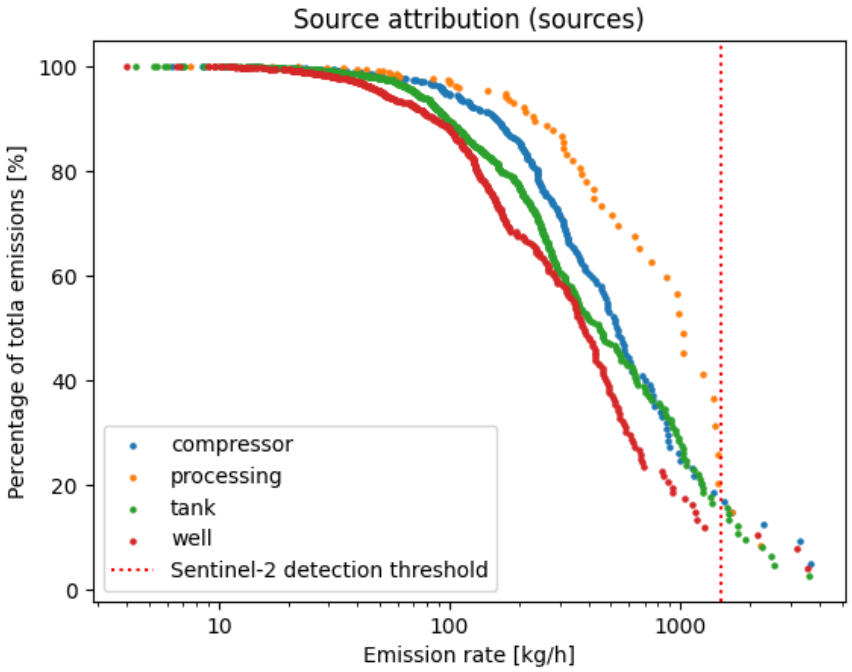
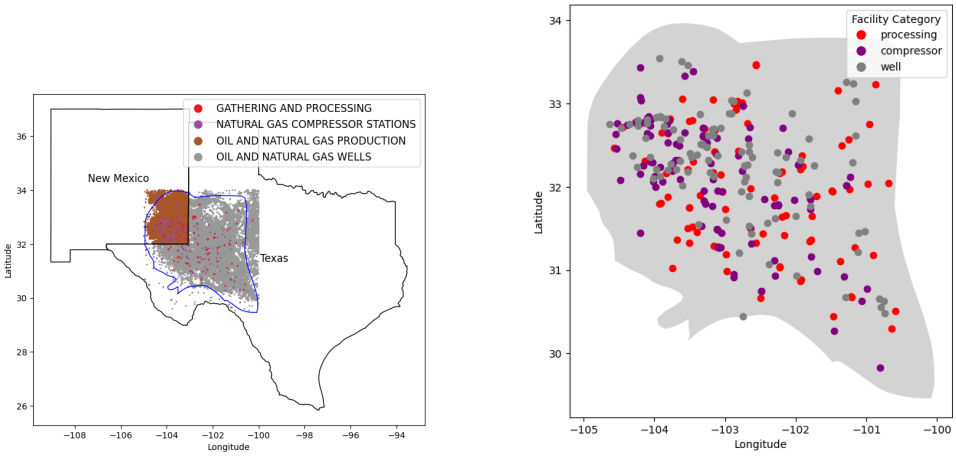


Figure 3.6: Distribution of CH4 emissions by source type and emission rate, quantified during a 2019 aerial campaign [8]

Table 3.3 and figure 3.6 demonstrate that emissions orginate mainly from midstream facilities. Tanks are a source type that is not included in the OGIM database but that are often colocated with well pads, as seen in figure 3.14 a). While this study does provide insights into the contribution of facility types to the Permian basin emissions, this is an incomplete picture. This study was done over a subset of the Permian and for 3 months. Furthermore, the subset of plumes with an emission rate greater than Sentinel-2's detection limit is not a representative sample. With this in mind, we chose to sample facilities from the OGIM database evenly across categories. This corrects the strong bias towards upstream facilities in OGIM, while keeping in mind the limits of the 2019 Permian basin airborne campaign. Our sampling from the OGIM database can be found in table 3.4. We limit ourselves to 300 locations for the first iteration of the synthetic plume dataset to encourage rapid development of a detection algorithm.
A spatial breakdown of the selected locations can be found in figure 3.7 b). Sampling gathering and processing facilities and natural gas compressor stations leaves little room for randomness due to the high proportion of the sampled facilities toward the total amount of facilities in each category. On the other hand, sampling from

Category	Stream	Count	Fraction of sampled data	Fraction of category
Oil and natural gas wells	upstream	67	22%	∼0%
Oil and natural gas production	upstream	33	11%	∼0%
Gathering and processing	midstream	100	33%	100%
Natural gas compressor stations	midstream	100	33%	38.3%

Table 3.4: OGIM facility sampling



(a) Total OGIM facilities in the Permian basin

(b) Sampled OGIM facilities in the Permian basin

Figure 3.7: Sampling production, compressor and processing facilities from the full OGIM database in the Permian basin

wells and production facilities is highly variable. We sample randomly from each upstream category in order to obtain a spatially distributed dataset.

3.3 Querying scenes

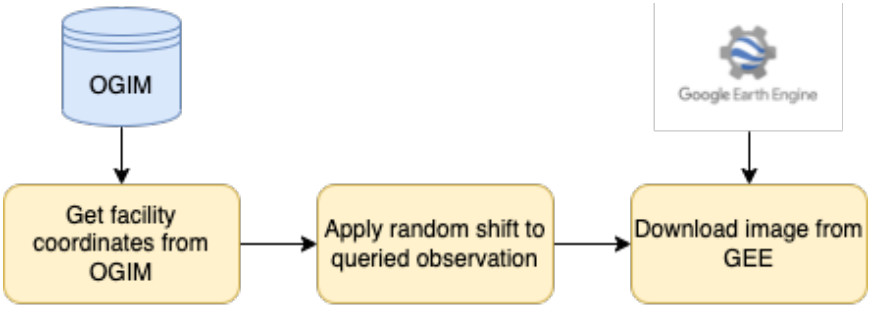
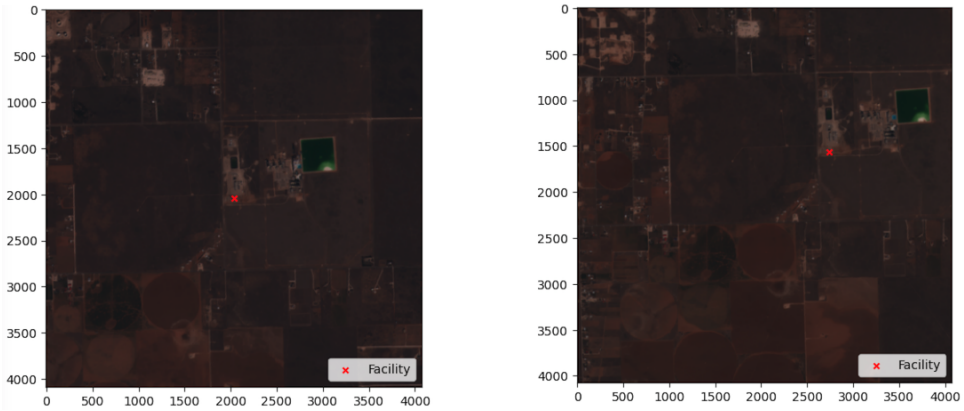


Figure 3.8: Querying scenes from Google Earth Engine using shifted OGIM coordinates

For each facility, we query all Sentinel-2 observations over a year. Sampling over a whole year allows us to account for seasonal scene variability. We only download observations where cloud cover is lower than 1%. When taking into account the Sentinel-2 constellation revisit rate, this yields an average of 30 observations per location per year in the Permian basin. As discussed in section 3.2, we sample from 300 locations. Thus, our dataset contains approxiamtely 9000 images. We chose the number of images first in order to build a dataset small enough to facilitate rapid iterations while ensuring there is a reasonable amount of data to learn from.
To query observations, we use Google Earth Engine (GEE) and query 4km x 4km scenes, the maximum image size allowed by the GEE API at 10m resolution[2]. The process is summarized in figure 3.8. For each facility, we apply a random shift within [-1km, 1km] to the queried coordinates. This ensures points of interest are not systematicaly in the center of an image, removing a potential learnable bias from the dataset. After shift, we keep a 1km buffer zone between the facility location and the edge of the image. This is to avoid cutting off parts of a facility. Figure 3.9 shows an example of the random shifting.



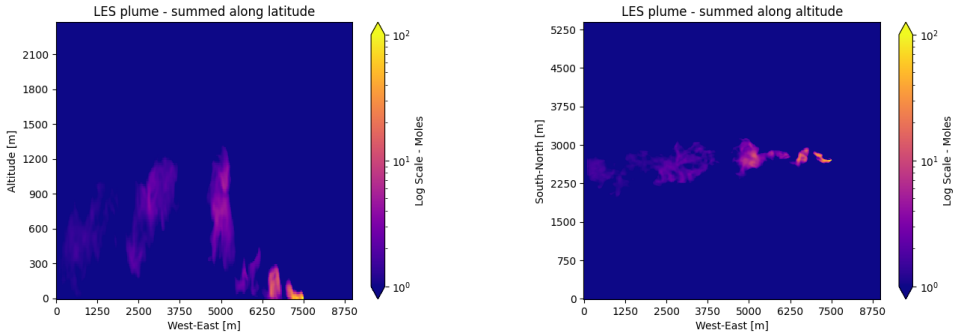
(a) Centered scene (b) Shifted plume

Figure 3.9: Scene shift applied when querying a scene from Google Earth Engine

3.4 Embedding Synthetic Plumes

To generate a dataset of Sentinel-2 scenes with embedded synthetic methane plumes, we first need to generate said methane plumes. For this, we use the Weather Research and Forecast (WRF) model configured with the Large Eddy Simulation (LES) package. This atmospheric transport model can simulate plumes originating from a point source. As they are expensive to run, LES simulations are obtained internally from the Atmospheric Chemistry Modeling Group. Each LES simulation contains 3 hours of simulation, with snapshots taken every 5 minutes. The first hour is discarded to account for spin-up of the model. We use a total of 5 simulations, thus resulting in 600 different plumes to embed in the remotely sensed scenes. Each simulation is named after an airport and has varying parameters which are summarised in table 3.5. The airport name specifies the location where the simulation was done as topographic features have an influence on the simulations. The source rate of 12 unit/s is equivalent to 693 kg/h. We want to create a dataset of large

[2]Sentinel-2's reslution varies by band. SWIR bands have a 20m pixel resolution. The finest resolution is for the RGB bands and stands at 10m. This is the limiting factor when querying observatins from GEE



(a) Side view

(b) Top view

Figure 3.10: Snapshot of a LES plume

plumes that can be encountered in the Permian basin. For this, we aim to embed plumes with a source rate between 10 t/h and 50 t/h. Thus, for every plume, we scale it by a random factor so that its source rate is in the above interval.

Name	Delrio	Dodge	Oakland	Peachtree	Reno
U wind $[m.s^{-1}]$	-5	-9	-7	-1	-3
V wind $[m.s^{-1}]$	0	0	0	0	0
Heat flux $[W.m^{-2}]$	200	150	100	250	300
Source rate $[unit.s^{-1}]$	12	12	12	12	12
Horizontal resolution $[m]$	25	25	25	25	25
Vertical resolution $[m]$	15	15	15	15	15

Table 3.5: Overview of WRF-LES simulations

3.4.1 Integrating Large Eddy Simulations

LES simulations are atmospheric transport models and therefore produce 3D fields of methane concentration expressed in $unit/m^3$. A satelite only sees a top of atmosphere (TOA) concentration which is why synthetic LES plumes must be converted into vertical methane column densities in $unit/m^2$. The side and top view of a LES plume snapshot is shown in figure 3.10.
While vertical column densities could be obtained by summing along the z-axis, this doesn't take into account the viewing configuration and the path taken by light going through the methane plume. When considering a satellite observation of a given scene, viewing angles have a strong influence on the reflectance and absorption in given wavelengths [13].
To illustrate the impact of the viewing configuration, we consider a combination of zenith and azimuth angles likely to be encountered in a Sentinel-2 observation. These angles are:

- Solar Zenith Angle (SZA) = 30° → angle between solar rays and vertical direction
- Solar Azimuth Angle (SAA) = 105° → horizontal angle with respect to north of the Sun's position
- Viewing Zenith Angle (VZA) = 5° → angle between light path to satellite and vertical direction

- Viewing Azimuth Angle (VAA) = 45° → horizontal angle with respect to north of the satellite's position

In such a configuration, we find non-negligeable differences between the downwelling and upwelling columns, as illustrated in figure 3.11. In this figure, VCD stands for Vertical Column Density and is the actual integrated plume concentration.

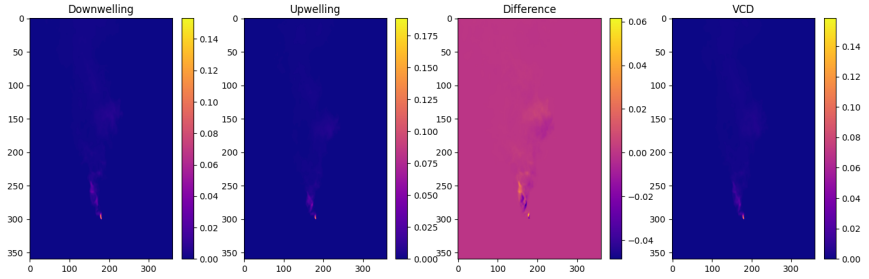


Figure 3.11: Difference between downwelling and upwelling methane columns

Instead of vertical integration, we perform slanted column integration as illustrated in figure 3.12.

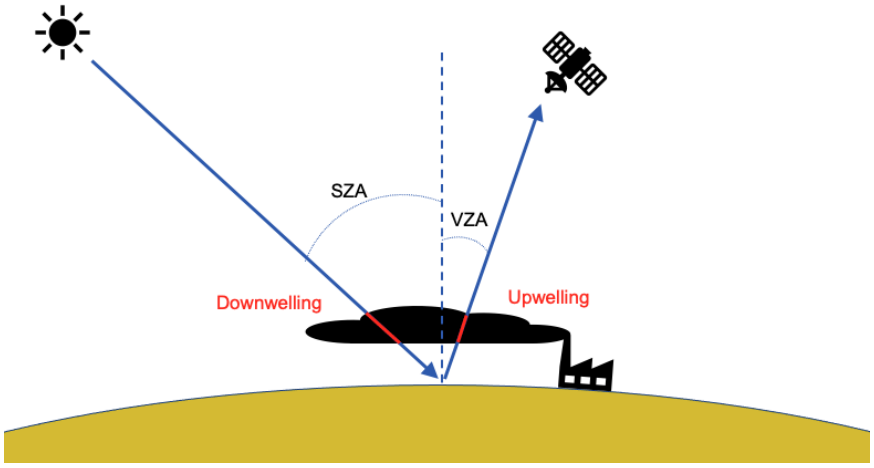


Figure 3.12: Light path considered for slanted column integration

To correct for the light path, we apply the following formula:

$$VCD = \frac{Downwelling + Upwelling}{AMF} \tag{3.5}$$

where

$$AMF = \frac{1}{cos(SZA)} + \frac{1}{cos(VZA)} \tag{3.6}$$

and

- VCD = Vertical Column Density
- AMF = Air Mass Factor

As the viewing configuration differs for each observation, we integrate LES plumes on the fly when embedding it in an image. For each image, a synthetic plume is sampled randomly from the WRF-LES simulations, with the following parameters also randomized:

- **Simulation location**: One of the dodge, oakland, peachtree or reno simulations for the train set. The delrio simulation is kept for the validation set
- **Simulation file**: second or third hour of simulation
- **Time step**: between 0 and 119 (one snapshot every 30 second of the simulation)
- **Source rate**: All simulations are run with a 693 kg/h source rate. We scale this source rate between 14 and 72 to have a range of source rates between 10 t/h and 50 t/h
- **Wind direction**: All plume are simulated with a constant wind direction at the source. To simulate varying wind directions, we rotate the plume at random with an angle between 0°and 360°

An overview of the randomized parameter distribution can be found in figure 3.13. The randomized wind angle can be found in the appendix, figure A.1

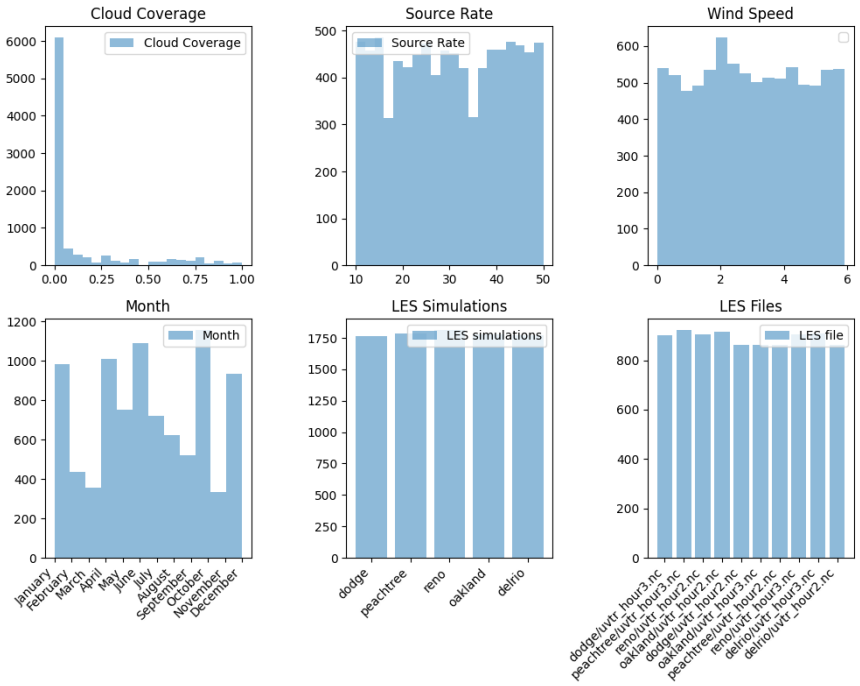


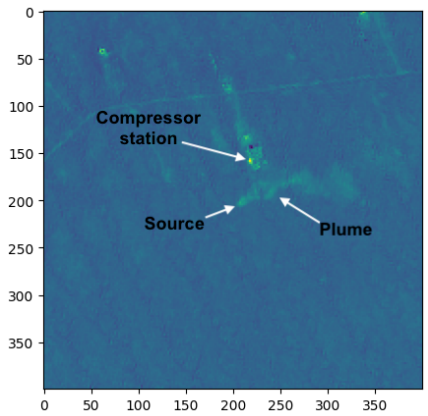
Figure 3.13: Occurence of randomly sampled plume parameters

3.4.2 Plume Placement

Once integrated for a specific viewing configuration, we position a given methane plume in the corresponding scene. This positioning introduces a bias which we carefully consider. While we could choose to embed a plume randomly in a scene, this doesn't take into account the colocation of the emissions and emitting facility. On the other hand, positioning a plume directly on the facility ignores the fact that an emission may emanate from around the facility. A prime example are storage tanks. From the 2019 Permian basin emissions survey by Cusworth et al. presented in table 3.3, we know that around 40% of emissions come from storage tanks. Tanks are usually located on a well pad but not right next to the well itself

as seen in figure 3.14 a). Similarly, emissions might originate from a pipeline block station located around an emitting facility or from an unlit flare, usually located near an emitting facility as illustrated in figure 3.14 b). To take into account the variability of methane sources in the vicinity of an emitting facility, we choose to place plumes within a given distance of the point location of the studies facility. We set this distance to 500 meters. This forms a bounding box around the facility, in which the plume source is randomly placed.

(a) Colocated tank and well



(b) Methane plume from an unlit flare near a compressor station

Figure 3.14: Colocation but not superposition of potential emission sources and OGIM facilities

Figure 3.15: Plume source bounding box

3.4.3 Modifying Radiances

With 2D methane column densities, knowledge of which bands we want to embed methane in and a strategy of where to embed a plume in a given Sentinel-2 observation, we can proceed to embedding the plumes in scenes. By embedding, we hear scaling the appropriate Sentinel-2 radiance by an appropriate factor on a pixel by pixel basis. This pixel by pixel process is outlined in figure 3.16.

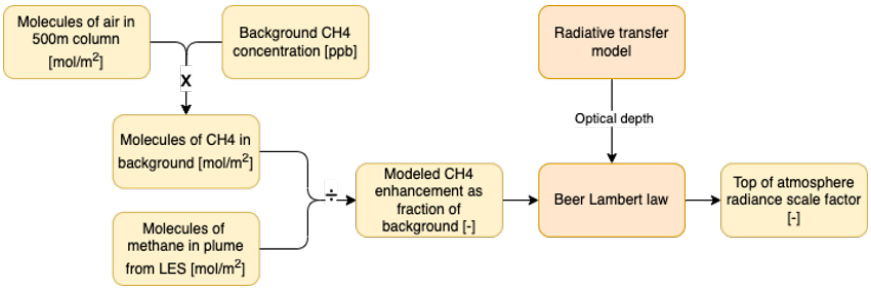


Figure 3.16: Pipeline for embedding methane plumes in Sentinel-2 imagery

To estimate a radiance scaling factor, we must first estimate the quantity of methane the synthetic plume adds to the background methane concentration. Indeed, methane is an important trace gas in the Earth's atmosphere, making up about 0.00019% of the atmosphere, or 1900ppb [26]. Our goal is to detect a methane plume, not all the methane gas in a given scene. Hence why we start by computing the ratio of methane concentration in the plume over the background methane concentration. To estimate the background methane concentration, we assume that our embedded methane plume is mainly in the bottom 500m of the atmosphere and compute the background methane concentration in these first 500m.
With the ratio of plume methane versus background methane, we compute the top of atmosphere spectral radiance (TOASR) in $W.m^{-2}.m^{-1}.sr^{-1}$, for the scaled methane concentration ($TOASR_{plume}$), for a given band in which we embed methane.
The TOASR is computed using the Beer-Lambert law for the slant column optical depths of CH4, CO2 and H2O, over the entire atmospheric column (up to satellite altitude). We only consider optical depths of CO2 and H2O in addition to CH4 as these are two main species that have strong absoprtion bands in the SWIR spectrums we consider and that have significant concentratations in the atmosphere. We split the CH4 optical depth in a lower layer optical depth and an upper layer optical depth. The lower layer CH4 optical depth is the one scaled by the plume to background methane ratio. The equations for TOASR is given by equation 3.7:

$$\begin{aligned} TOASR_{plume} = \exp(-((1 + ch4\_scale) * odCH4_{lower} + odCH4_{upper} \\ + odCO2 + odH20) * solar\_spectrum) \end{aligned} \tag{3.7}$$

where odX is the optical depth of the specie, $solar\_spectrum$ is the upwelling solar spectrum and $ch4_{scale}$is the methane enhancement as a fraction of background.

The scale factor we use to scale the given band's radiance is then given by:

$$scale\_factor = \frac{TOASR_{plume} - TOASR_{background}}{TOASR_{background}} \tag{3.8}$$

and the band scaling is given by:

$$scaled\_band = band * (1 + scale\_factor) \tag{3.9}$$

In equation 3.8, $TOASR_{background}$ is computed using a 100 layer, clear-sky radiative transfer model from Varon et al. [50]. From the same radiative transfer model, we obtain the optical depths for CH4, CO2 and H2O as well as the upwelling solar spectrum used to cumpute $TOASR_{plume}$.

As the radiative transfer model accounts for surface altitude, solar zenith angle and viewing zenith angle, we repeat the pipeline outlined in figure 3.16 for every scene.

3.5 Testing Embedded Plumes

Having embedded LES plumes in raw Sentinel-2 radiances, we now want to verify our methodology is embedding the correct concentrations of methane in said radiances.
To verify our method, we retrieve a methane plume using the Multi Band Multi Pass (MBMP) method [50]. This method retrieves a methane concentration for each pixel by considering at the the difference between a scaled B12 and B11 ratio from two different observations of the same scene. The results from this method can be found in figure 3.17.

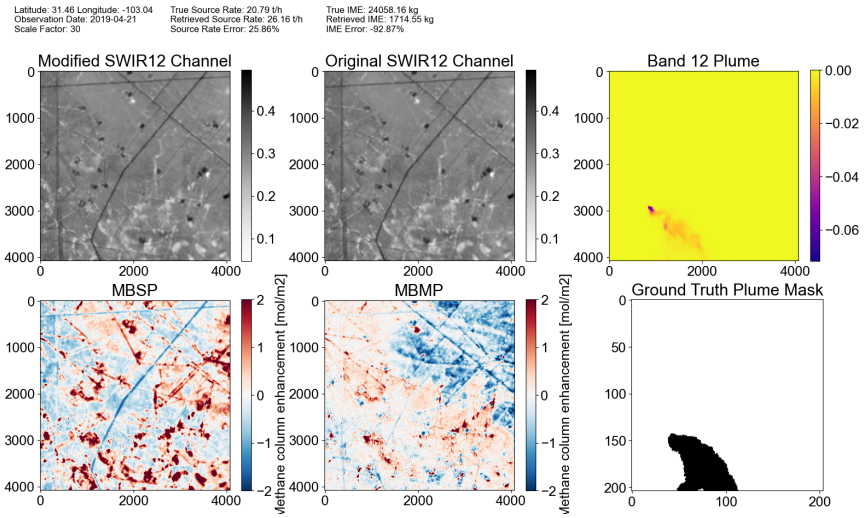
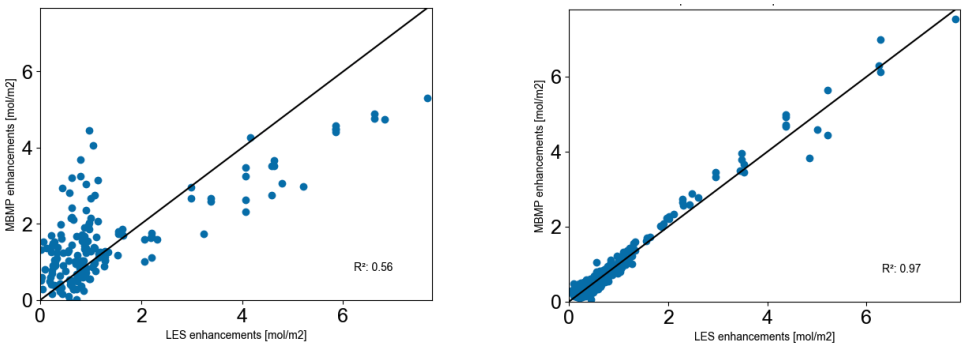


Figure 3.17: MBMP retrieval for a synthetic plume embedded in a Permian basin scene

As we know the pixel by pixel concentration of the MBMP retrieval and of the 2D methane enhancement map, we can compare both as seen in figure 3.18 a).
In this figure, we observe a correlation between the retrieved and embedded plumes but with large residuals. While this could mean our method is biased, this could also be due to the noise in heterogenous scenes in the Permian basin. To verify this, we embed a plume in a homogenous scene and perform the same retrieval. We choose to embed this plume over a desert scene in Algeria, another methane emissions hotspot. The results from this retrieval are presented in figure 3.18 b).
In this figure, we observe a strong correlation between the retrieved and embedded plumes. This confirms the weak correlation in figure 3.18 a) is due to the scene noise and provides confidence in our embedding methodology.

3.6 Generating Training and Validation Sets

When training a machine learning algorithm, we usually train on a specific set and verify how well the algorithm has learned on a held out set, the validation set. The algorithm is never trained on the validation set. This way, we can check if the algorithm is learning meaningful and generalizable patterns from the training data or if its overfitting to the training data. Importantly, we must avoid data leakage. Data leakage occurs when information from the training set can be found in the validation set or vice versa. The validation performances would then be overly optimistic.



(a) Permian basin (b) Hassi Messaoud

Figure 3.18: MBMP retrieved plume versus LES plume methane column enhacements

In our case, there is a risk of data leakge if we embed the same plumes in the validation and training sets or if we validate on backgrounds that the algorithm learns from in the training steps.
To avoid the facility data leakage risk, we split the scenes in a test and validation set by facility. In other words, all observations of a single facility will always be in a single set. The split ratio is based on the total number of scenes, not facilities, and set at 80%. In other, words, 80% of the scenes are in the training set and 20% are in the validation set. To avoid the LES data leakage risk, we leverage our five distinct simulations as explained in section 3.4. We embed plumes from four simulations in the training scenes and keep the fifth simulation to embed in the validation scenes.

3.7 Data Normalization

We create a dataset of synthetic methane plume and train a deep learning algorithm. Normalizing input data for deep learning applications is crucial as machine learning algorithms rely on consistent and normalized data [18]. Normalization helps with:

- **Avoiding numerical instability**: Deep learning models often involve computations that are sensitive to the scale of the input data. If the input features have vastly different scales, it can lead to numerical instability during training. By normalizing the input data, all features are brought to a similar scale, reducing the chances of numerical instability and ensuring a more stable and efficient training

- **Improved generalization**: Deep learning models aim to learn general patterns from the training data that can be applied to unseen data. Normalizing the input data helps the model focus on the intrinsic patterns and relationships within the data rather than being biased by differences in feature scales. It can prevent certain features from dominating the learning process simply due to their larger scales. In our case, this could be a metallic storage tank that has much a higher reflectance than the rest of the scene or a water body that obsorbs strongly in the infrared domain

- **Regularization effects**: Normalization can have a regularization effect on the model. By enforcing a constraint on the input data, it reduces the model's capacity to fit noise or irrelevant variations in the training data. This regularization can help prevent overfitting

Our dataset is composed of two types of features. First of all, normalized difference indices consisting of dimensionless quantities, bounded between -1 and 1. The training dataset also has Sentinel-2 L1C data which consists of reflectances. Unlike RGB images that are encoded in 256bit, reflectances are bounded by 0 but unbounded in the positive direction. They are also long tailed as illustrated in figure 3.19 where the 98% of the data is in a [0, 0.44] interval but the maximum value is 2.17 .

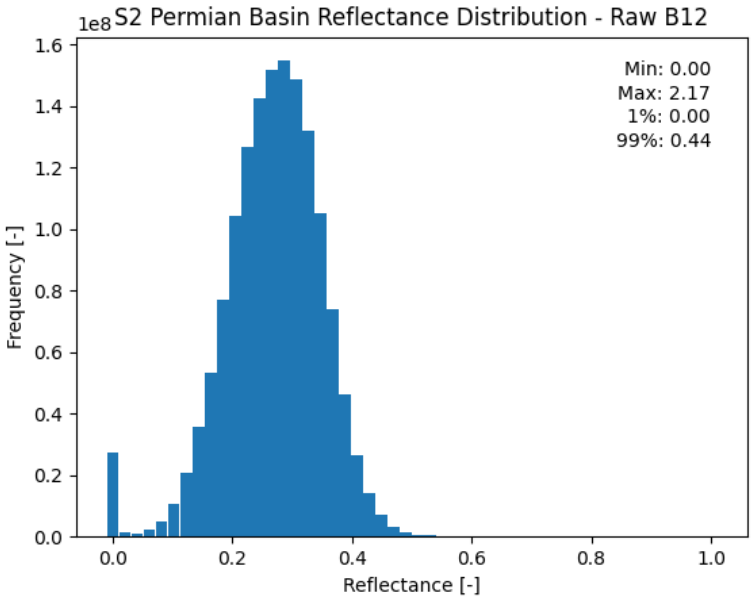


Figure 3.19: Long tailed distribution of B12 reflantances L1C Sentinel-2 data

To train on these types of data, we need a normalization method that bounds these quantities and controls their distribution. We focus on the following normalization methods:

- **Min-max scaling**: Remapping of a given input range (bounded or unbounded) according to the following formula:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.10}$$

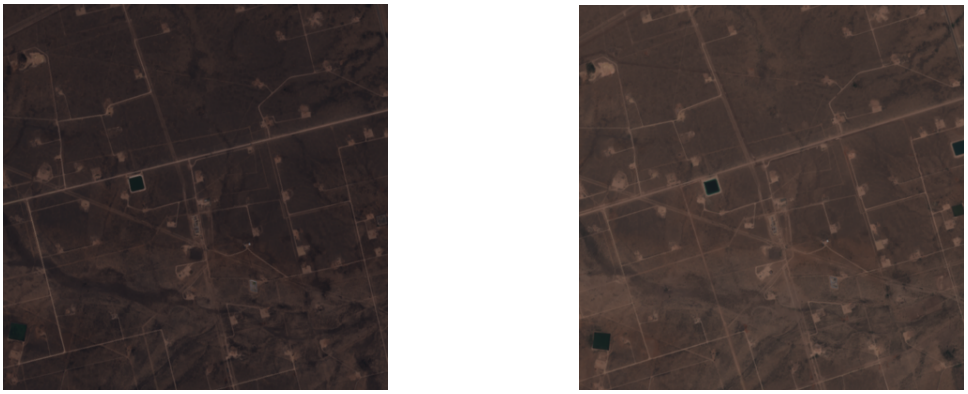
  The result will be bounded by 0 and 1

- **Percentile scaling**: When the input range contains outliers, such as long tailed satellite imagery reflectances, min-max scaling may be sensitve to extreme data points. In this case, min-max percentile scaling may be appropriate and is given by the following formula:

$$x_{scaled} = \frac{x - x_1}{x_{99} - x_1} \tag{3.11}$$

  where x_1 is the 1st percentile of the input data and x_{99} is the 99th percentile of the input data. The result will be bounded but not by 0 and 1

- **Z-score normalization (standardization)**: Transforms the input data so that it has a mean of 0 and a standard deviation of 1. We refer to this method



(a) Parks compressor station in January

(b) Parks compressor station in July

Figure 3.20: Illumination variations over a same location

as standardization. This is achieved with the following formula:

$$x_{stan} = \frac{x - \mu}{\sigma} \tag{3.12}$$

Where μ is the mean of the image and σ is the standard deviation of the image.
Because standardization not only rescales but also centers data around 0, it is particularly useful when dealing with features that have different scales. In our case, this can mitigate the effect of illumination changes across seasons or regions. This issue is illustrated in figure 3.20, where the observation of the same facility is much brighter in July than in January. In subfigure a), the mean value for B12 is 0.303 while it's 0.395 for subfigure b), captured in July. By standardizing the bands, the reflectance distribution for both observations would be centered around 0, with 1 standard deviation, mitigating impacts of illumniations changes.

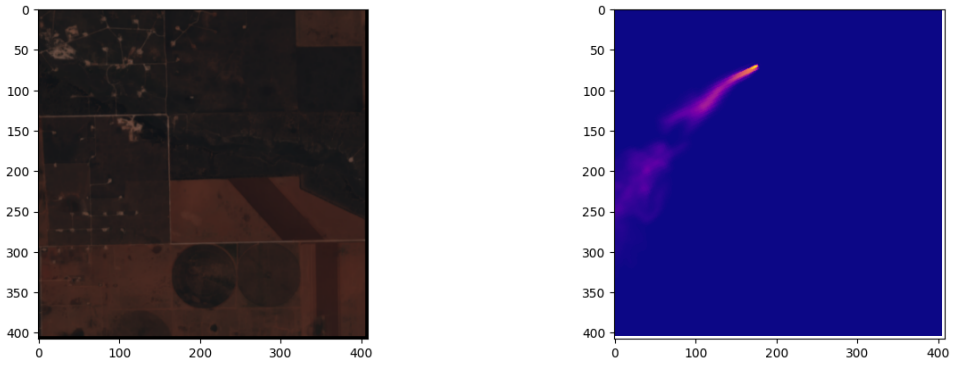
Different pre-processing methods can be used for both types of data (radiances and NDIs). In various tests covered in section 4.4.4, we find that standardizing (Z-score normalization) both types of data without scaling yields the best results.

3.8 Creating Labels

Training a methane plume detection algorithm requires labelled data. For a binary segmentation task (plume/no plume segmentation), the labelled data must be in the form of binary masks. In this section, we cover the process of generating binary masks for the embedded plumes.

3.8.1 Masking Plumes

In this section, we will study the scene and its associated plume illustrated in figure 3.21
Masking plumes is the first step towards creating labels for our dataset. As shown in figure 3.22 a), creating naïve masks including the entire embedded plume can cover a significant portion of the scene, even where the plume is not discernable from the background noise.



(a) Studied scene

(b) Studied plume

Figure 3.21: Scene and associated plume studied over the course of section 3.8.1

We need a thresholding strategy in order to converge towards a global minimum when training. When devising a masking strategy, we need to take into account the following:

- **Minimise noise**: We aim to minimise the noise included in the masked scene to increase the network's performance
- **Maximise methane plume**: The primary goal of a mask is to encompass the largest discernable portion of the studied methane plume
- **Consistency**: A masking strategy must mask the smallest plumes while keeping the larger plume masks to a reasonable fraction of the image
- **Physical Sense**: The masks must correspond to a physical reality. A mask for a large plume should be significantly larger then one for a small plume. Without physical grounding, a network risks being biased towards the mean source rate of the dataset

To fulfill the above criteria, we introduce masking methods based on vertical column density (VCD) and signal to noise ratio (SNR). VCD is the integrated methane column intorduced in section 3.4.1. VCD is measured in mol/cm^2 and represents the ground truth methane plume. SNR is obtained by dividing a given band by this band's standard deviation. SNR is dimensionless and represents how easily interpretable a given pixel is compared to the background. A SNR greater than 1 indicated more signal than noise. The masking strategies used are listed bellow and illustrated for the scene in figure 3.21.

- **VCD percentile masking**: We compute a k-th percentile of the VCD bellow which a given percentage of the methane plume frequency distribution falls. For example, a 95th percentile mask will reveal the strongest 5% of the VCD pixels. This should be the pixels neighbouring the source of the plume. This has the advantage of directing the plume segmentation algorithm on detecting the source of the plume but lacks physical grounding. The 5% of pixels with highest methane concentration of a 10t/h plume will have a smaller methane concentration than for a 50t/h plume. Shown in figure 3.22 c)
- **VCD threshold masking**: In this method, we only include VCD pixels over a given threshold in the mask. The threshold becomes a tunable hyperparameter. For example, a natural choice would be the mean methane background

concentration in the Permian, estimated to be around 0.67 $mol/m2$ [26]. This is illustrated in figure 3.22 b)

- **NDMI SNR masking**: This method doesn't rely on the ground truth plume (ie, the VCD) but on the embedded plume. We compute the NDMI for a given scene and isolate the plume from the NDMI. To do so, a "no methane" NDMI is calculated from bands 11 and 12 before the plume is embedded. By substracting the NDMI without methane from the NDMI with methane, we obtain the methane plume as it is embedded. Next, we compute the standard deviation of the NDMI with methane and divide the above difference by the standard deviation. This highlights wich parts of the plume should be visible in the NDMI ratio. We include in the mask all pixels above a given threshold in the SNR map. The natural choice for this threshold is 1 as this indicates the plume signal is greater than the background. However the threshold remains a tunable hyperparameter. The python code for this is in listing B.1 (in Annex) and the result is illustrated in figure 3.22 d)

- **NDMI focused SNR masking**: The focused SNR method extends the above described SNR method. Instead of computing the standard deviation across the whole scene, we compute this standard deviation within the plume. Here, we define the plume as all the VCD pixels above a given threshold. This method is more robust to noisy artefacts which could influence the standard deviation of the scene. However, by definition, the methane plume should have the highest pixel values in the NDMI. Therefore the focused standard deviation will usually be higher then the standard deviation calculated over the entire scene. Hence the SNR could be lower for plume pixels and the plume mask would cover a smaller amount of the plume

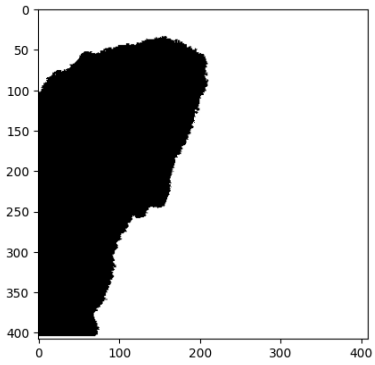
Overall, we elect to mask using SNR masking of the NDMI illustrated in figure 3.22 c). We choose this method as it produced the more robust and accurate results on the validation set during training.

3.8.2 Defining a Standard for Labels

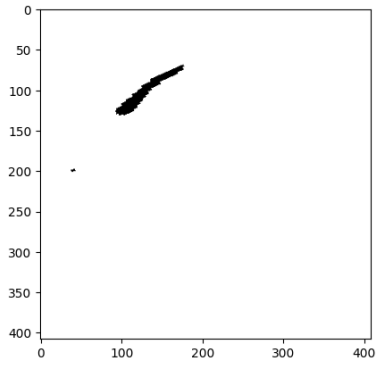
Without a large, open-source dataset of Sentinel-2 scenes with labeled methane plumes available for the research community, sharing our dataset is one of our objectives. To encourage other researchers to build computer vision pipelines to detect and quantify methane plumes in Sentinel-2 data, we format our data according to the Common Object in COntext (COCO) format [32]. This format is well documented and used by several foundation models.
More specifically, we use the format for panoptic segmentation described in listing 3.1.

Listing 3.1: COCO panoptic segmentation label format

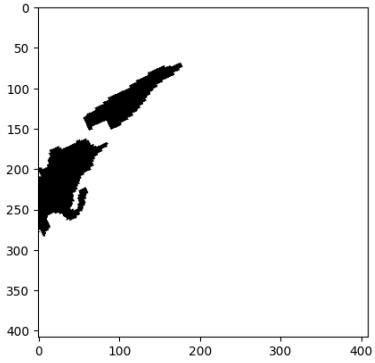
```
image{
"id": str,
"license": int,
"width": int,
"height": int,
"file_name": str,
"date_captured": str
}

annotation{
"image_id": int,
"file_name": str,
```



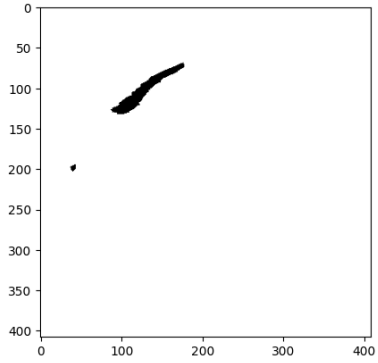
(a) Threshold mask with threshold = 0



(b) VCD threshold mask with threshold = 0.67



(c) VCD percentile mask with percentile = 95%



(d) NDMI SNR mask with threshold = 1

Figure 3.22: Illustrations of various masking methods

```
    "segments_info": [segment_info],
    }

    segment_info{
    "id": int,
    "category_id": int,
    "area": int,
    "bbox": [x,y,width,height],
    "iscrowd": 0 or 1,
    }

    categories[{
    "id": int,
    "name": str,
    "supercategory": str,
    "isthing": 0 or 1,
    "color": [R,G,B],
    }]
```

In listing 3.1, segment_info is a list of length equal to the number of distinct objects in an image. In our case, this may be 1 if there is a plume or 0 if the image doesn't contain a plume. image[file_name] corresponds to the file name of the Sentinel-2 scene while annotation[file_name] corresponds to the file name of the mask. In

COCO annotations, objects are identified by a unique RGB encoding. In our case, there are two possible labels which are:

- Plume: represented by red pixels: [R, G, B] = [255, 0, 0]
- No plume: represented by black pixels [R, G, B] = [0, 0, 0]

segment_info[id] identifies to which category a given label is. This id is determined by the RGB code according the following formula:

$$id = R + G * 256 + B * 256^2 \tag{3.13}$$

Furthermore, we set "iscrowd" and "isthing" to 0.
image[id] is the same as annotation[image_id] and is the concatenation of the OGIM facility's unique ID and the date of capture of the Sentinel-2 scene in YYYY-MM-DD format. For example, the ID for the observation of OGIM facility 2571247 on the 2019-12-17 is 257134720191217. To convert such a formatted COCO label to a simple binary mask, we use the code snippet in listing B.2 (in Annex).

Chapter 4

Algorithm Development

4.1 Background on Image Segmentation

Image segmentation consists of splitting an image in distinct segments where each pixel of the image corresponds to an object we are trying to segment. Image segmentation tasks are usually broken down in instance segmentation, panoptic segmentation and semantic segmentation.

- **Semantic segmentation**: Consists of segmenting parts of an image that belong to the same class, for example a bulding or a road. Image segmentation can be seen as an image classification task at a pixel level
- **Instance segmentation**: Extends semantic segmentation to segmenting distinct instance within a class, for example, building A or building B
- **Panoptic segmentation**: Combination of both semantic segmentation and instance segmentation. Panoptic segmentation assigns a pair of semantic and instance labels to each pixel in the image

In the case of detecting methane plumes, we are dealing with semantic segmentation. A plume may include several distinct segments but we don't have an interest in classifying these segments distinctly. All segments belong to the same source in our case.

4.1.1 A Brief History of Image Segmentation Algorithms

Since the advent of Convolutional Neural Networks (CNN) in computer vision [29] [30], traditional image segmentation techniques (edge detectors [5], thresholding [46], etc) have been replaced by CNN based image segmentation methods. These have proven successfull in solving common problems faced by traditional methods such as variations in illumination and texture or translations.
Fully Convolutional Networks [34] (FCN) first adapted CNNs to image segmentation tasks but lacked precise localization capabilities due to pooling and upsampling operations loosing the global semantic context of the image.
Deconvnet [36] introduced the concept of an encoder-decoder architecture to attempt to recover spatial information lost during pooling operations. Deconvet did however struggle with capturing fine details and creating artifacts.
Building on top of the encoder-decoder architecture, the U-Net architecture [43] added skip connections from matching encoder and decoder bloc. Those connections were a key factor in helping retain fine spatial information by fusing low level

and high level features. On top of this, U-Nets have proven capable to handle limited amounts of data. This makes them particularly suitable for the remote sensing community, where data is scarce and expensive.
There have since been several variations of the U-Net architecture such as U-Net++ (nesting mutiple U-Net architectures within each other) [56], Attention U-Net (incorporating attention mechanisms in the network) [38] or Residual U-Net (introducing residual connections to mitigate the vanishing problem gradient) [9].

Beyond CNNs, image segmentation tasks have also been tackled by more recent architectures. Generative Adversial Networks (GAN) [15] have been trained to generate realistic looking segmentation masks. Examples include CycleGAN [58] or Pix2Pix [22].
Recently, the transformer based approach has been applied to computer vision [11] and specifically to image segmentation. Swin Transformer [33] or DETR [6] are examples of vision transformers that have been successfully applied to image segmentation tasks. Their ability to capture long-range dependencies produced competitive results.

We chose to investigate methane plume segmentation using a U-Net architecture. We set aside GANs as they are not considered state of the art for image segmentation tasks and may be challenging to train and stabilize [45]. As for transformer based architectures, while they have produced state of the art results in image segmentation challenges [31], they are constrained by requiring large volumes of data to learn meaningful representations. Many transformer applications build on top of foundation models by fine-tunning them to a specific task. Open-source pre-trained foundational models for geospatial data, such as the one provided by NASA [25], have begun to emerge, though they remain a very new advancement. This is why we elect to create an initial version of our methane plume detector using the well-established and proven U-Net architecture.

Focusing on methane plume segmentation, most previous approaches have also used a U-Net architecture as backbone to their pipeline ([4], [27], [52]).

4.2 U-Net Architecture

U-Net networks are fully convolutional networks characterised by a U-shaped architecture comprised of an encoder and a decoder [43]. The encoder and decoder both have 4 blocks and are connected by a bridge. The contracting path (encoder) doubles the filters and halves the spatial dimension while the expansive path (decoder) halves the filters and doubles the spatial dimension. One of the unique features of a U-Net are skip connections. Each encoder bloc has a skip connection to the equivalent decoder bloc. This allows high level features with semantic information to be combined with low level features which have a detailed spatial representation. For image segmentation tasks where precise localisation but also contextual understanding are key, U-Nets with skip connections are well suited.
While the number of encoder/decoder blocs usually stays the same, the complexity of the network can be adjusted with the number of filters. In figure 4.1, the first encoder bloc has 64 filters and each subsequent encoder bloc doubles the number of filters. The same can be said for the decoder blocs but the other way around. Choosing this first bloc filter number is a key parameter in defining the network depth. For example, a 5 channel 512x512 image fed in a U-net with stride of 1, kernel size of 3x3, padding of 1 and 64 base filters has 31 million parameters. The same network with 32 base filters has 7.8 million parameters.

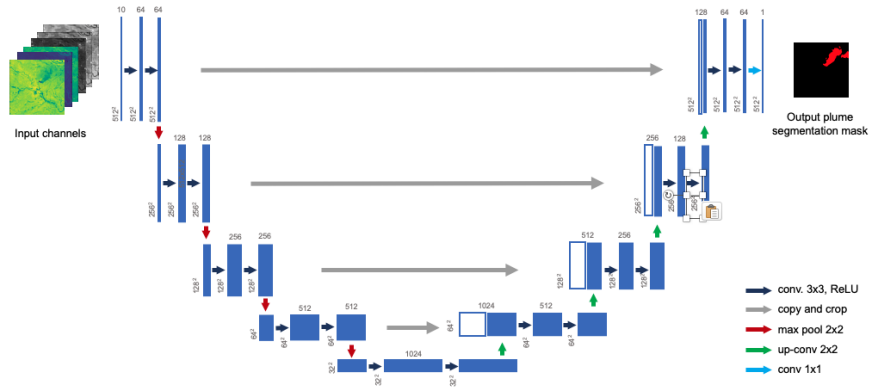


Figure 4.1: U-Net architecture (inspired from [43])

4.3 Training Setup

4.3.1 Loss Function and Metrics

We use the following loss function [20]:

$$L = E - logJ \tag{4.1}$$

where E is the binary cross entropy loss and J is the Jaccard index (also called Intersection over Union or IoU).
We choose cross entropy loss (BCE) as this is the common loss function for classification tasks [42]. BCE penalizes the model for misclassifying a label proportionaly to the deviation in probability. We opt for BCE over the more usual categorical cross entropy as we face a binary classification problem (plume/no plume). BCE is defined as follows [20]:

$$E = \frac{1}{N}\sum_{i=1}^{N} -(y_i * log(p_i) + (1 - y_i) * log(1 - p_i)) \tag{4.2}$$

where p_i is the probability of class i and $(1 - p_i)$ the probability of class 0.
In equation 4.1, J is the jaccard index, defined as:

$$J(P,T) = \frac{|P \cap T| + smooth}{|P \cup T| + smooth} \tag{4.3}$$

$$0 \leq J(P,T) \leq 1 \tag{4.4}$$

where P defines the prediction and T the target labels. We add "smooth" as a smooting parameter to ensure differentiability of the Jaccard index. We set $smooth = 1$ as we get the best training performances with this hyperparameter (figure 4.2).
The Jaccard index is critical in our application. Indeed, in the case of imbalanced datasets, where one class is significantly larger than the other, the Jaccard index is less influenced by the prevalence of the majority class. Our dataset contains much more "plume" pixels than "no plume" pixels. In fact, plume pixels represent only 0.6% of the dataset (8M plume pixels versus 1465M no plume pixels). By focusing on the proportion of the intersection of two sets relative to their union, the IoU measurement inherently accounts for the class distribution. BCE loss doesn't account for class imbalance which is why we give a stronger weight to the Jaccard index through the application of the log function (Jaccard index being bounded by

0 and 1).

To track model development, we rely on the the Jaccard index evaluated on the validation set.

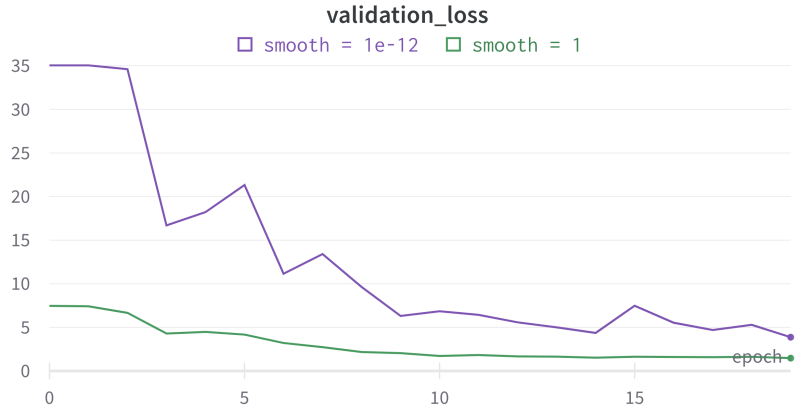


Figure 4.2: Impact of Jaccrard smoothing parameter on validation loss

4.3.2 Hyperparameters

We train our U-Net with an AdamW optimizer, learning rate of 10^{-4} and clip gradients to 0.1. In this section, we cover the choice of these hyperparameters.

Optimizer

We trained with an AdamW Optimzer [35], during 20 epochs. We also tested training with an Adam optimizer [28]. Adam is an optimization algorithm for the backpropagation step through the neural network. It dynamically adjusts the learning rate for each parameter depending on a moving average of previous gradients and squared gradients. The goal is to efficiently converge to the global minimum of the loss function during training. Adam mitigates overfitting to the training data by including weight decay as a regularization method. Weight decay adds a penalty to the loss function based on the magnitude of weights, encouraging smaller weight values. It affects both the weights and the moving averages of past gradients. With AdamW, weight decay only affects the weights themselves and not the moving averages of past gradients, stabilizing training. In addition, decoupling the weight decay strength from learning rate adjustements makes it easier to tune the learning rate. For both optimizers, the initial decay rates are set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and weight decay is 0.01
As reported in figure 4.3, AdamW stabilizes the validation loss in our training and leads to a slightly better final validation loss.

Learning Rate

We trained with a learning rate of 10^{-4}. We also tested different learning rates and learning rate schedules. A learning rate schedule modifies the learning rate over the course of the training. In addition to the 10^{-4} schedule, we tested a multi-step learning rate schedule of 10^{-3} for the first 10 epochs and 10^{-4} for the last 10 epochs, a multi-step learning rate schedule of 10^{-4} for the first 10 epochs and 10^{-5} for the

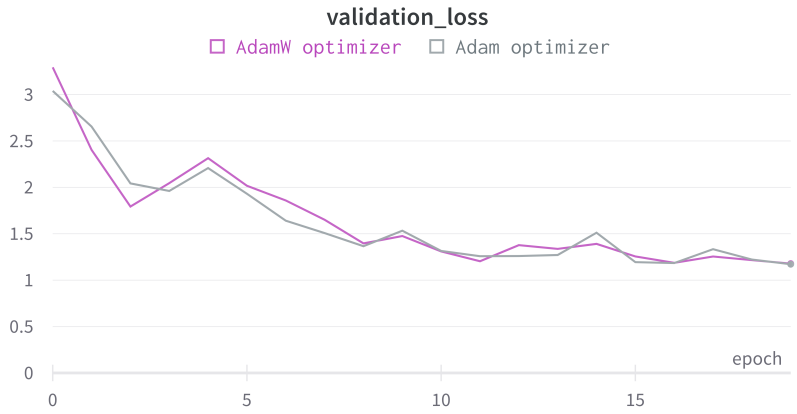
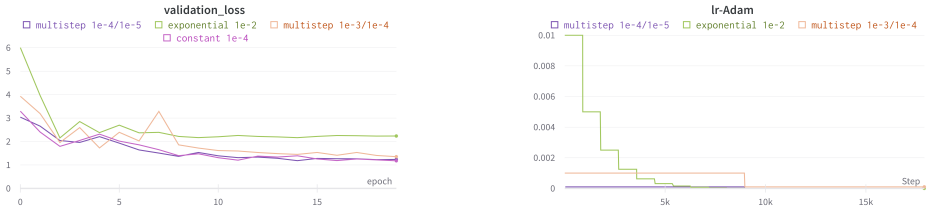


Figure 4.3: Training performances for Adam and AdamW optimizers

last 10 epochs and an exponential learning rate schedule starting at 10^{-2}.
we observe that the performances for each schedule are reported in figure 4.4.



(a) Loss

(b) Learning rate schedule

Figure 4.4: Training performances for varying learning rate schedules

Gradient Clipping

Clipping the gradients to a threshold value helps avoid exploding gradient issues. When training deep neural networks, the gradients of the loss function with respect to the model's parameter may become large, causing the updates to the network weights to also become very large, causing instability in the training. Gradient clipping limits the magnitude of the gradients to a given threshold during back-propagation.
Figure 4.5 illustrates the improvement in the training with gradient clipping set to 0.1.

Regularization

During training, we don't observe an increase in the validation loss as shown in figure 4.6. This suggests we aren't overfitting to the training data. Hence, no dropout is applied in the U-Net architecture.

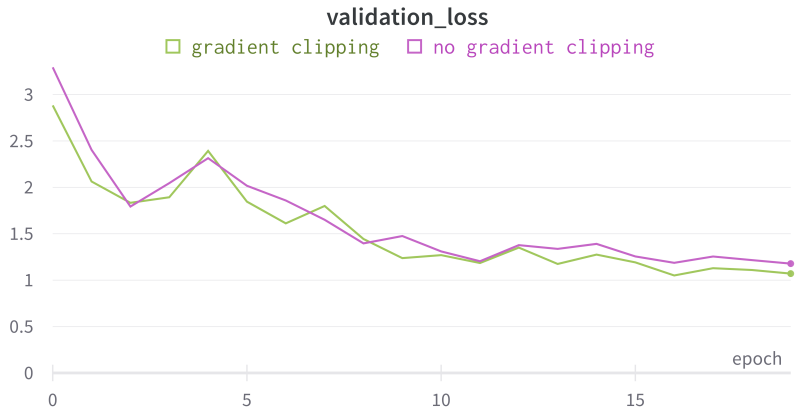
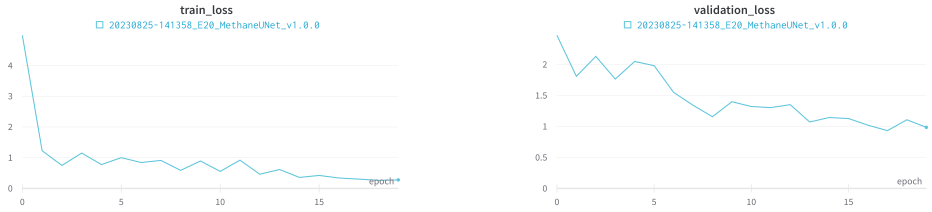


Figure 4.5: Training performances when training with gradient clipping



(a) Training loss

(b) Validation loss

Figure 4.6: Training performances over training and validation sets

4.4 Results

As mentioned in section 3.6, we train on a given selection of scenes from the Permian (80% of the total scenes) and validate on held out scenes (20% of total scenes). We also embed LES plumes from 4 disctinct simulations in the training scenes and validate on a fifth LES simulation. In this section, we present an overview of the performances of the plume segmentation algorithm on the validation dataset.

4.4.1 Metrics

For the quantitative analysis of the validation results, we use results of the confusion matrix on a pixel by pixel basis. In other words, we evaluate the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) when comparing pixels in the predicted masks and pixels in the ground truth masks. To analyse the results of the confusion matrix, we introduce the following metrics:

- **Recall**: Measures how many retrieved plume pixels are indeed methane. This is an appopriate metric if the consequences of false negatives are high

$$Recall = \frac{TP}{TP + FN} \tag{4.5}$$

- **Precision**: Measures how many plume pixels are indeed predicted as methane.

This is an appopriate metric if the consequences of false positives are high

$$Precision = \frac{TP}{TP + FP} \tag{4.6}$$

- F_β: Is the adjusted harmonic mean of recall and precision. The β parameter is introduced to weight recall or precision more stronly than the other. $\beta \leq 1$ favors precision while $\beta \geq 1$ favors recall

$$F_\beta = \frac{1 + \beta^2}{Precision^{-1} + Recall^{-1}} \tag{4.7}$$

We use both $\beta = 1$ (precision and recall have the same weight) and $\beta = 0.5$ (precision is weighted higher than recall). Indeed, false positives are dangereous for two reasons:

  - Hinder automation of a plume detection pipeline
  - Damages overall reputation of the field of plume detection from Space

4.4.2 Scores

The trained network has the following scores:

IoU	Recall	Precision	F1	F0.5
0.510	0.682	0.649	0.633	0.634

Table 4.1: Results on the validation set for the trained network

We also include a few predictions made by the network in figure 4.7
In these examples, we observe that the network usually finds the plume correctly. The predicted mask may not fully cover the ground truth maks, but the most expressive part of the plume is usually found, with the exception of the 10.39 t/h plume.
As we can see from the 13.86 t/h plume or the 49.20 t/h plume, the network seems robust to infrastructure artifacts. These scenes contain many well pads yet none of them are confused for a plume.
Other plume like features such as the bottom right bright yellow streak in the 49.20 t/h plume don't appear as plumes. This highlights the importance of using other bands that can discriminate plume-like features in the NDMI from actual plumes.
From this selection, we observe that artifacts are usually topographic features, for example in the 32.57 t/h plume. For other artifacts such as the ones in the 10.39 t/h plume, it is not clear from the NDMI band what is confused as a methane plume.

4.4.3 Predictor Variables

With a trained network yielding satisfactory predictions on the validation set, we want to understand which variables have an influence of the plume prediction ability of the network. For this, we consider two types of variables: scene dependant parameters and input features. Scene dependant parameters are intrinsic charcteristics of the plume/observation at hand such as solar zenith angle, source rate, source altitude, etc. Input features are the channels on which we train the neural network (NDMI, B12, B4, etc).

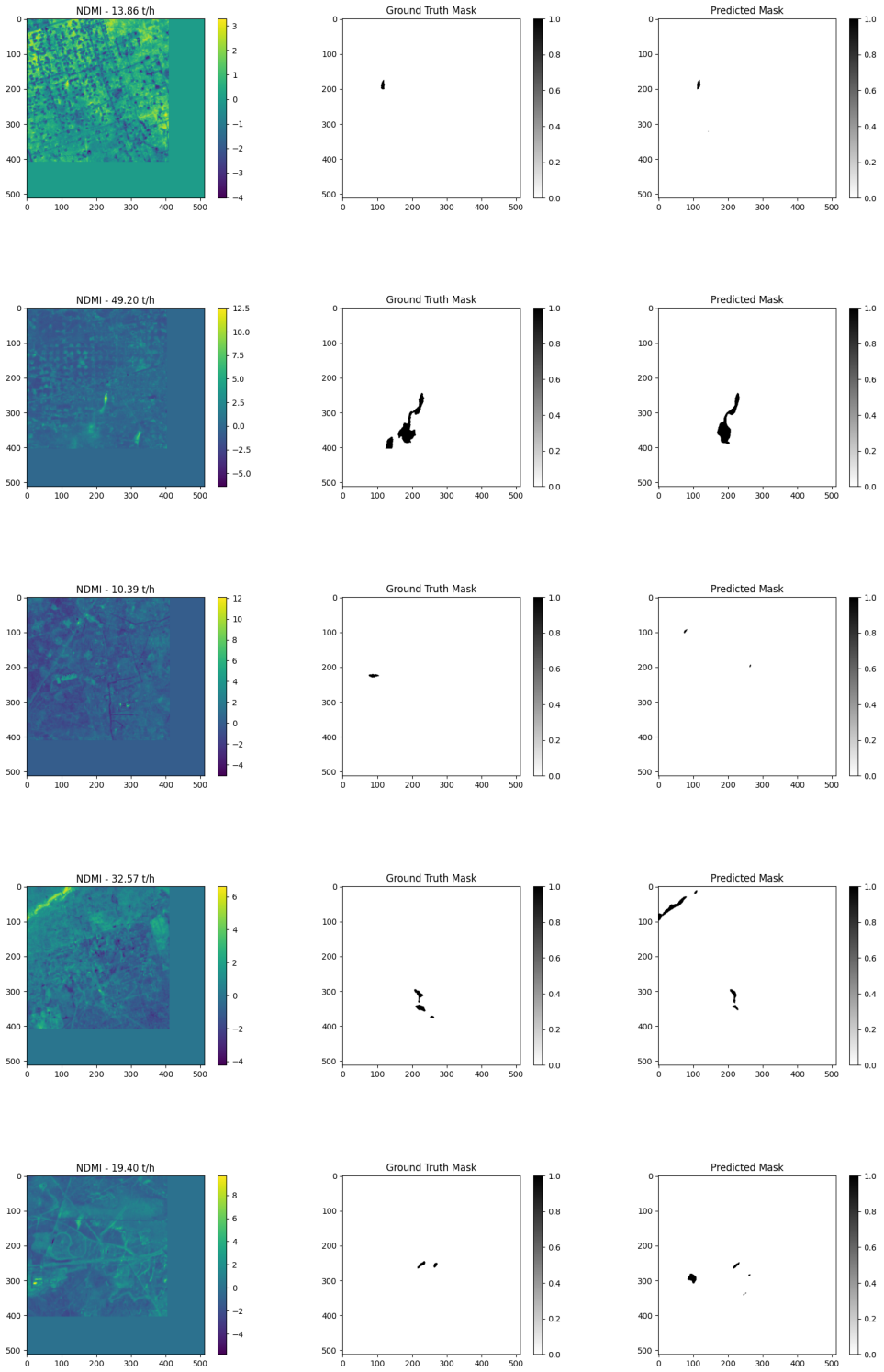
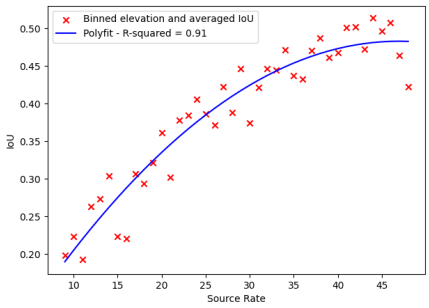


Figure 4.7: Predictions from the validation set

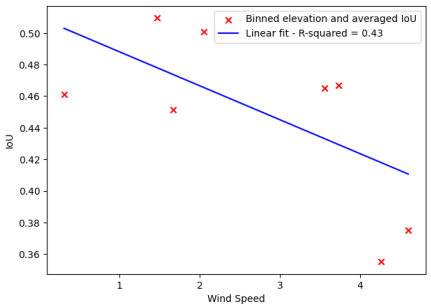
Scene dependant parameters

In figure 4.8, the IoU between predicted and ground truth masks are binned in equal length bins and the bin's average source rate and IoU are reported on the x and y axis respectively.

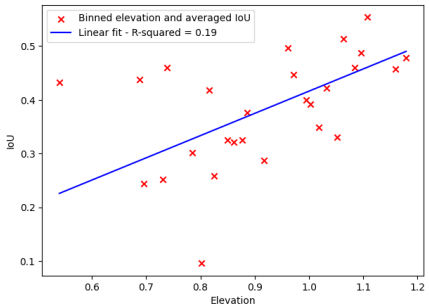
As is apparent from figure 4.8 (a), source rate is a strong predictor of the detectability of a source. This is to be expected as the source rate dictates how much methane is present in the scene. The relationship between source rate and IoU seems to plateau towards 0.5, which suggests there is a maximum IoU reached by the network.
We also notice a relationship between wind speed and detectability, as suggested by figure 4.8 (b). This is also to be expected as the stronger the wind, the more likely the plume will be dissipated in the atmosphere, rendering the plume indistinguishable from the background methane concentration.
We also notice a suprising correlation between source altitude and detectability in figure 4.8 (c). While we didn't investigate this dependancy further, we believe this could be the subject of further research and could help correct for biases in quantification of methane point sources across regions.



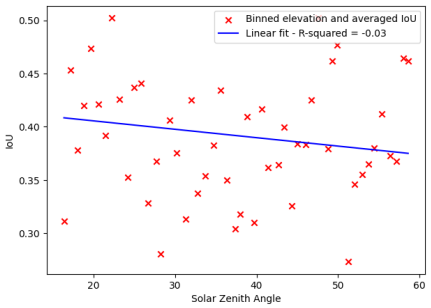
(a) IoU vs Source Rate



(b) IoU vs Wind Speed



(c) IoU vs Elevation



(d) IoU vs Solar Zenith Angle

Figure 4.8: Correlations between scene dependant variables and source detectability

Input features

To evaluate the importance towards detecting a plume of the features we feed in our network, we perform a channel importance analysis through channel ablation. In our chosen configuration, a scene has 11 channels (B2, B3, B4, B8, B10, B11, B12, NDMI, NDBI, BSI, NDVI). To measure the importance of each channel, we set all the pixel values for a single channel to 0 over the entire validation set, run

a pass over the validation set and calculate the average IoU. We do this iteratively over all channels and produce the results presented in figure 4.9.

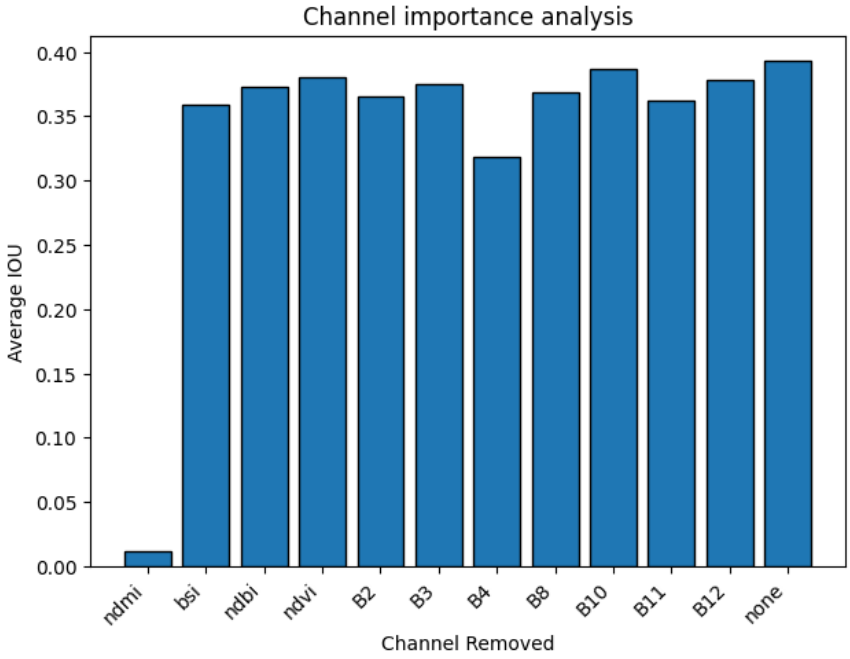


Figure 4.9: Channel importance analysis over the validation set

The results from this study demonstrate the importance of NDMI towards detecting methane plumes in Sentinel-2 imagery and B4's role (red channel) in filtering out false positives.
We also observe that the 3 SWIR bands in which we embedded methane (B10, B11, B12) aren't being considered much by the network for predicting plume locations. This can be expected for B12 and B11 as they are already taken into account in the NDMI. B10's lack of influence on the results rejects the hypothesis we made in section 3.1.2. We observed that B10 had a significant mean absorption cross section with methane's absorption cross spectrum and hypothesised that this band could be useful for detecting methane plumes. Its lack of influence could be linked to its greater spatial resolution. At 60m per pixel, it is significantly greater than all other bands (10m or 20m) and could lack the details necessary to differentiate a point source. In fact, a satellite spatial resolution of less than 60 is usually required to observe a plume [3].

The channel importance analysis offers an incomplete look at the significance of input features on the detectability of methane plumes. We check which band has the most impact on the predicition from the subset of bands we trained on. However, we don't check the changes in performances when training on different subset of bands. For this, we run three different trainings:

- A training on all bands: NDI, RGB, NIR and SWIR bands
- A training without the SWIR bands (train on NDI, RGB, NIR)
- A training without NDMI (train on all NDIs but NDMI, RGB, NIR, SWIR)

The goal here is to understand which role the different methane signals (SWIR bands or NDMI) have in the detection of synthetic methane plumes. The results

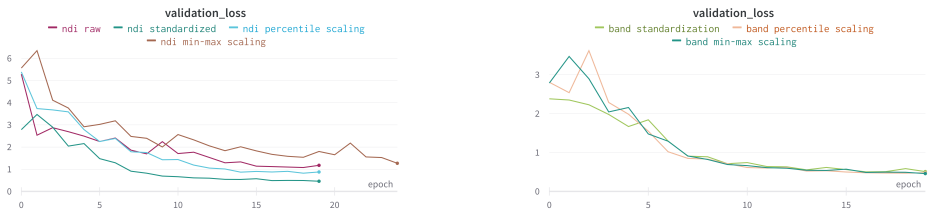
from these three trainings can be found in table 4.2.
From these results, we confirm the importance of NDMI towards detecting methane plumes. We also observe that the SWIR bands have a negative impact on performances on the validation set, across all metrics. While it could be surprising that adding more data has a negative impact on performances, we need to circle back to the training labels. Those are generated on the basis of what methane signal is distinguishable from the background in the NDMI signal. This means that the labels are tailored to the NDMI. As we can expect the methane signal in B12 to be weaker than NDMI, this means a label covers the distinguishable B12 methane plume but also artifacts that may be around the plume, adding noise to the detections. Rather than highlighting the negative impact of the SWIR bands on training, we believe this highlights the limits of our labelling strategy presented in section 3.8.1.

Training Features	IoU	Recall	Precision	F1	F0.5
All bands	0.510	0.682	0.649	0.633	0.634
All but B10, B11, B12	0.571	0.738	0.703	0.693	0.693
All but ndmi	0.448	0.646	0.594	0.575	0.576

Table 4.2: Impact of training features on Recall, Precision and F_β metrics

4.4.4 Influence of Normalization

In section 3.7, we outlined several normalization methods for both Sentinel-2 radiances and NDIs. We observe that choosing the right normalalization is key to achieving optimal training performances. Both NDIs and Sentinel-2 bands having different bounds and distributions, we study normalization for both seperately.
First, we fix preprocessing for the Sentinel-2 bands to min-max scaling and vary the preprocessing for the NDIs. From the results, presented in figure 4.10 a), we observe that standardizing the NDIs yields the best results. The differences in performances here are significant, with a 61% decrease in validation loss when standardizing NDIs compared to keeping the raw form.
With these results in mind, we fix preprocessing for the NDIs to standardization and vary the preprocessing for the Sentinel-2 bands. From the results, presented in figure 4.10 b), we don't notice significant differences between the three different preprocessing methods. While the path to convergence varies, final validation losses are almost identical. To mitigate potential impacts of illumination variations and help our model generalize to different regions, we decide to also standardize the Sentinel-2 bands.



(a) Varying NDIs preprocessing methods

(b) Varying Sentinel-2 bands preprocessing methods

Figure 4.10: Impact of preprocessing on training performances

In figure 4.11, we plot the mean NDMI and their average IoU in bins (the values are binned in bins of unequal sizes, hence the outliers). We observe that the greater

the deviation of the mean NDMI from 0, the poorer the performance becomes. This confirms the importance of standardizing the NDIs. This performance gap is also visible in the confusion matrix metrics, displayed in table 4.3.

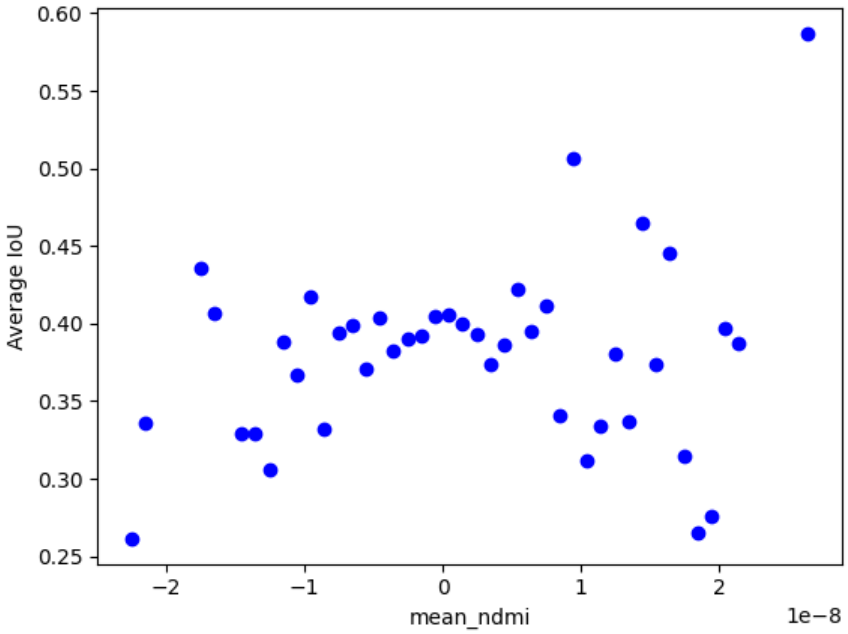


Figure 4.11: Iou vs Mean NDMI

Network	IoU	Recall	Precision	F1	F0.5
No standardization of NDIs	0.389	0.518	0.624	0.513	0.557
Standardized NDIs	0.510	0.682	0.649	0.633	0.634

Table 4.3: Impact of standardization on Recall, Precision and F_β metrics

4.4.5 Detectability of Methane Plumes in Real Images

This work's overarching goal is to train a computer vision alogorithm to detect synthetic methane plumes and deploy it on real Sentinel-2 data. Thus, evaluating our algorithms's performances on real plumes is key to understanding its actual performances.

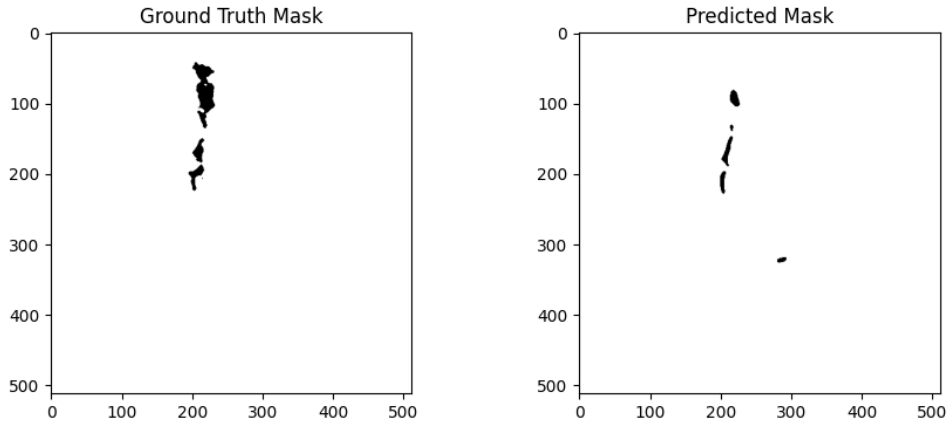
Test dataset

We build a test dataset of known plumes from public records of methane plume detections in Sentinel-2 imagery.
To build this dataset, we first collect a record of open sourced methane plumes in the Permian basin. The plume locations and source rates are obtained from Varon et al. [51], Irakulis-Loitxate et al. [21] and PermianMAP [49], a database of methane plumes in the Permian basin published by EDF (Data from U. Arizona, NASA-JPL, and EDF). Next, we filter out methane plumes that do not have a Sentinel-2 overpass on the detection date. Finally, we filter out plumes with a source rate inferior to 4 t/h. Although the demonstrated detection limit for Sentinel-2 is 2 t/h, we trained our algorithm on plumes with source rates between 10 t/h and 50

t/h. Thus, we filter out plumes which we don't expect to detect and would make differentiating performance variations more challenging. With all these filters, we obtain 19 observations of scenes with known methane plumes. We also include 6 scenes with no known methane plumes to test for false positives. Of these 19 scenes, 18 are observations of the same facility, further limiting the statistical significance of the test data.
For the labels to evaluate detections, we use the MBMP method from Varon et al. [50] to create plume masks. Due to the methodology employed in constructing the test dataset, the failure to detect any plume within it using our detector would indicate that our algorithm does not match the current state-of-the-art physics bases approaches.

Method

Due to the small sample size of the test dataset, we don't use the metrics introduced in 4.4.1 but the confusion matrix itself (TP, FP, TN, FN). Furthermore, we establish the confusion matrix on a distinct plume by plume basis instead of a pixel by pixel basis.
To illustrate the difference between both methods, we base ourselves on figure 4.12. With the pixel by pixel method, we find 572 true positives, 210 false positives and 1555 false negatives. On the plume by plume method, we find 1 true positive, 1 false positive and 0 false negatives. This method encourages a more qualitative approach to avoid drawing false conclusions from a non statistically significant dataset.



(a) Ground truth plume (b) Predicted plume

Figure 4.12: Ground truth plume and its corresponding prediction

Results

The confusion matrix obtained from running our detector on the test dataset can be found in table 4.4. A selection of predictions can be found in figure 4.14.

		Predicted Value	
		Positive	Negative
Actual Value	Positive	8	14
	Negative	11	5

Table 4.4: Confusion matrix for the test dataset

From these results, we observe that our network correctly detected 8 of the 19 plumes and correctly predicted negatives for 5 out of the 6 scenes without plumes. Furthermore, figure 4.13 doesn't highlight the relationship between detectability and source rate highlighted in section 4.4.3, suggesting source rate cannot be the only predictor of detectability.
Furthermore, we fail to detect over half of the test dataset's plumes, indicating subpar performances when compared to state of the art MBMP methods [50].

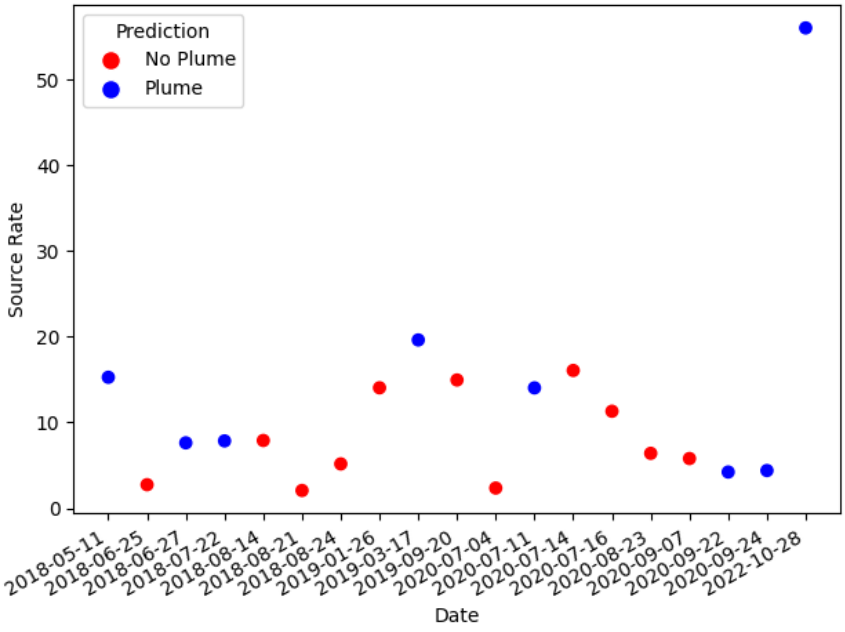


Figure 4.13: True positives and false negatives in real plume dataset

Influence of training data

As with performances during training, we observe varying performances depending on the channels we train on. When testing the three variants of our network presented in section 4.4.3, we get the results in table 4.5:

	All bands	No SWIR bands	No NDI bands
True positives	8	4	11
False positives	14	0	26
True negatives	5	6	4
False negatives	11	16	8

Table 4.5: Performances over test dataset for varying training features

These results suggest only training on NDIs (NDMI, NDVI, NDBI, BSI) leads to highly confident real plume detections. There are no false positives and all scenes without plumes are correctly identified as such. Introducing SWIR bands in the training data, whether with or without NDIs, leads to more plume detected but also much more false positives. While we remain cautious when drawing conclusions from this test data, we can envision two versions of this network designed for two different purposes:

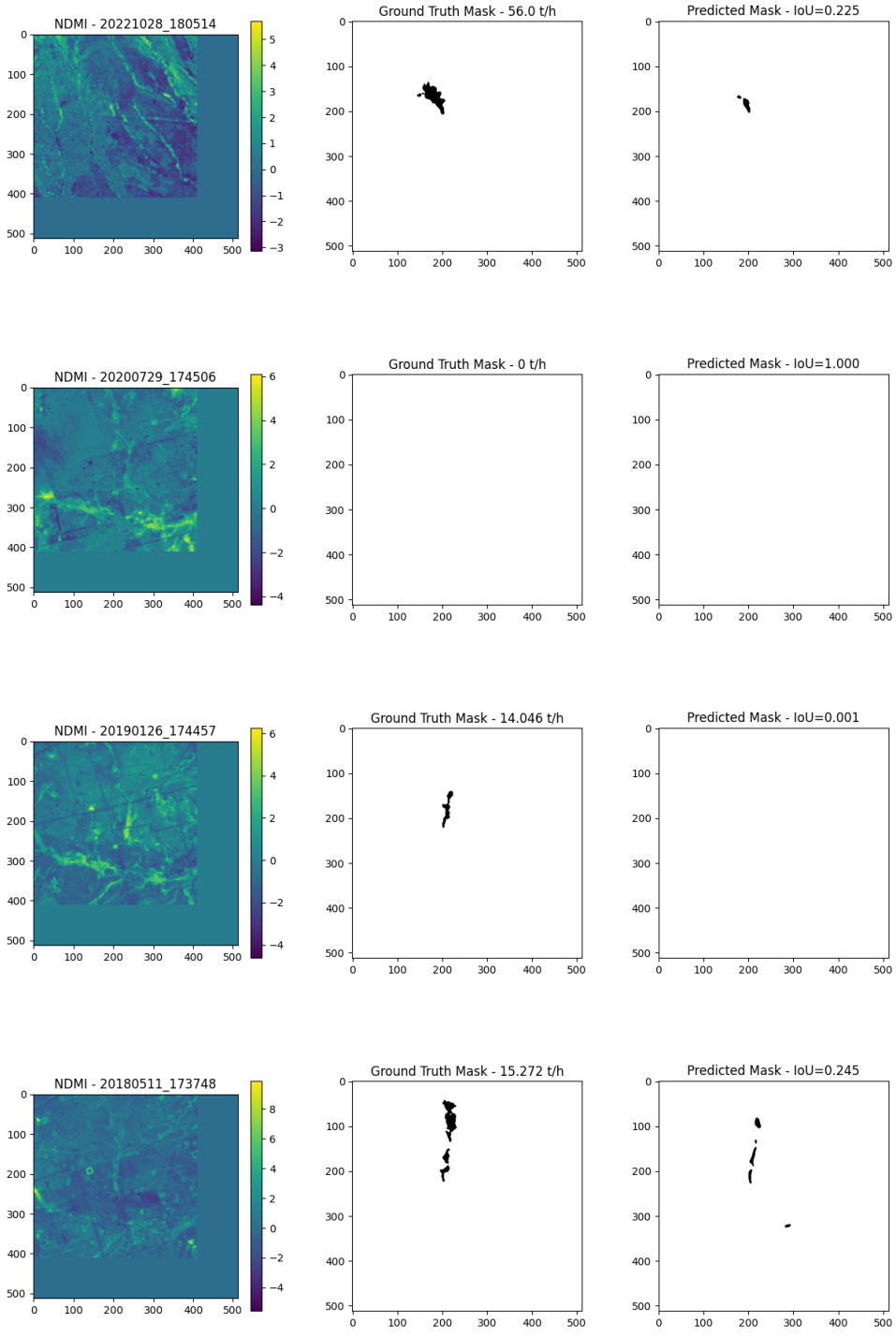


Figure 4.14: Predictions from the test set

- **No SWIR bands**: A version trained without SWIR bands when deploying the detection algorithm on large amounts of data. For example if we deploy this algorithm on all Sentinel-2 observation of OGIM facilities in the Permian basin. This would produce hundreds of thousands of observations per week.

Any percentage of false positives, even small, would call into question the automated nature of this pipeline

- **No NDIs**: A version trained without the normalized difference indices when searching for plumes in scenes where we have an a priori knowledge of the presence of a plume or not. For example if we use this algorithm to find the exact source of a TROPOMI detection (area flux mapper whith daily revisit, with proven capabilities of detecting large methane plumes [24]. During TROPOMI overpass, plumes may have already travelled downwind and locating the exact source is not trivial)

Extension to other oil and gas basins

While we trained a methane plume detector on scenes from the Permian basin, we want to study if such a targeted training can generalize to detecting methane plumes in other oil and gas basins. For this, we collect dates and coordinates of proven methane plumes in other regions. We focus on the Hassi Messaoud oil field in Algeria, and the Korpezhe oil and gas field in Turkmenistan. These two regions are extensively studied by the research community [50], offering a priori information on methane plumes. For the Korpezhe location, we select all observations between 2018-12-01 and 2019-03-31. According to Varon et al. 2021, there are 8 plume scenes and 2 no plume scenes in the period. For the Hassi Massaoud point source, we select all proven plumes between 2019-12-01 and 2019-12-30. According to Varon et al. 2021, there are 11 plume scenes and 1 no plume scenes in the period. The results, computed in the same manner as in 4.4.5, can be found in table 4.6.

	Korpezhe oil/gas field	Hassi Messaoud oil field
True positives	6	7
False positives	3	9
True negatives	2	1
False negatives	2	4

Table 4.6: Performances over test dataset for varying training features

Figure 4.15 and 4.16 present a selection of detected and missed plumes from these two regions.
From these results, we can see our methodology of training solely on Permian scenes can generalize to other regions although some fine tuning could improve performances.

4.5 Discussion and Interpretation

The goal of this thesis is to train a plume detection algorithm on synthetic plumes embedded in multispectral satellite imagery of the Permian oil and gas basin and use this algorithm to detect real plumes in the Permian basin. A key challenge here is to learn meaningful representations of synthetic plumes that port well to real plumes. In other words, reducing the simulation to reality gap as much as possible.

A challenge when searching for methane plumes in the Permian basin is the heterogeneity of the background. Roads, buildings, fields and topography create many potentiel artifacts that can be confused with methane plumes. To improve plume-artifact discrimination, we use the Normalized Difference Methane Index as an input feature to our network. By looking at the normalized difference between two neigboring bands we aim to enhance the contrasts between features that can be difficult

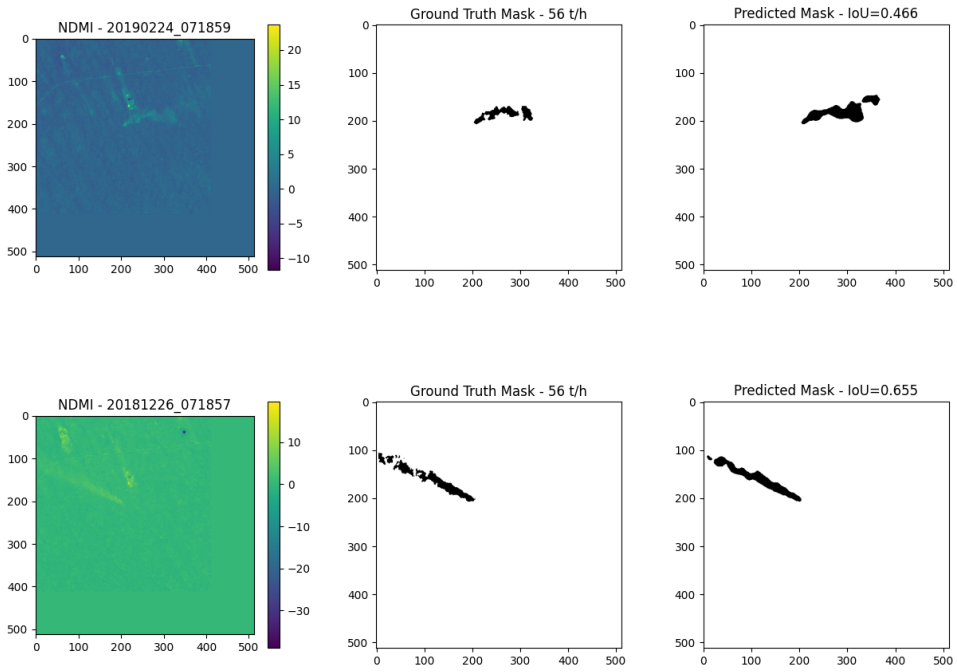


Figure 4.15: Detecting plumes in the Korpezhe oil and gas field (Turkmenistan, 38.4939°N, 54.1977°E)

to distinguish in one of the spectral bands. This is specifically the case of methane, which absorbs strongly in Sentinel-2's B12 and less in B11, two neighboring bands. With NDMI, we want to increase the contrast between the methane plume and its background to better distinguish it and filter out above mentionned artifacts.
Our results indeed show inclusion of NDMI as a feature plays an important role towards lowering the number of false positives and increasing the number of true positives. From the results in table 4.2, we see that including NDMI in the training data increases recall (an indicator of how few false negatives are produced) by 5.6% and increases precision (an indicator of how few false positives are produced) by 9.3%.
Furthermore, key to NDMI's ability to produce more true positives is its standardization. Standardizing individual NDMI bands to have 0 mean and standard deviation of 1 leads to a 31.7% increase in recall and 4% increase in precision. Finding the right preprocessing was a long process of trial and error where we tried various combinations of standardization and normalization of NDIs and Sentinel-2 raw radiances.

We initially set the objective of detecting methane plumes from raw radiances instead of methane retrieval fields. The reasoning behind this objective is the price of obtaining such retrieval fields (computational cost and need of occasional manual intervention). Although the significant feature in predicting the presence of methane plumes is not a raw radiance, we still manage to detect such plumes using an indicator that can be computed on the fly, at cheap computational cost and without the need for selecting a reference scene.

For training our network, we observed good performances from the loss function

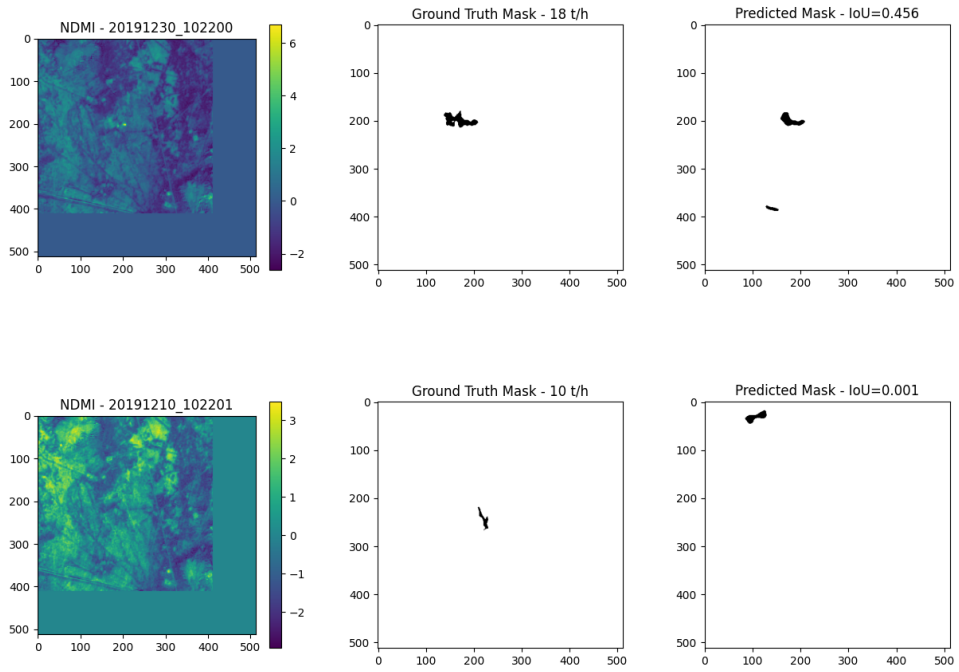


Figure 4.16: Detecting plumes in the Hassi Messaoud oil field (Algeria, 31.6585°N, 5.9053°E)

presented in equation 4.1. This loss function enabled us to deal with a strongly imbalanced dataset (0.5% of "plume" pixels) and avoid overfitting to the training dataset.

With the plume detector having reached satisfactory performances on the synthetic data, we probed the simulation to reality gap by evaluating it on knowm occurences of large plumes (greater than 4 t/h) in the Permian basin. Here, we were able to demonstrate detection capabilities on real plumes. This proves the viability of the method and opens up possibilities of training algorithms to detect plumes in regions where we have low numbers of real plumes.
Moreover, we observed that training on NDMI without SWIR bands had a positive impact on false positives. A high number of false positives is the most significant hurdle in deploying a plume detector to survey methane point sources. The reduced occurrence of false positives suggests a robust ability to filter out artifacts. However, there remains a significant gap between performances on simulated data and on real data. Our model only finds 42% of plumes from the dataset. We identify a few factors with potential influence on the simulation to reality gap:

- **Source rate interval**: We embedded plumes with a source rate between 10 t/h and 50 t/h. The goal was to target the ultra large emitters in the Permian basin. However, empirical data from figure 3.6 and source rates in the test dataset suggest we could lower the minimum source rate to 2 t/h. This is the empirical detection treshold of Sentinel-2 for methane plumes and would cover a more realistic interval of source rates

- **Variability of LES methane plumes**: We create a dataset of synthetic plumes from over 600 different plumes that are scaled up to the desired source

rate. However, the plumes come from only five simulations and snapshots of simulations are temporally correlated. A plume at time=t is correlated with a plume at time=t+5. Moreover, there is only one simulation topography per wind speed, meaning both are correlated in our training data. Increasing the overall variability of the embedded plumes could help bridge the observed simulation to reality gap

Chapter 5

Conclusion

For this thesis, we worked towards a high frequency monitoring of methane point sources in the Permian basin. For this, we set out to produce three final products: a dataset, an algorithm and an application to real plumes.

Contributions

First of all, we produced a dataset of close to 9000 Sentinel-2 observations of oil and gas facilities with synthetic plumes embedded in Sentinel-2's SWIR bands (B10, B11, B12). We envisioned a pipeline fetching data from WRF-LES simulations, the OGIM database and Google Earth Engine to produce this synthetic data. To increase the variability of the plumes, we randomized plume placement, wind direction, plume shape and source rate. Furthermore, we embedded plumes in scenes with less than 1% cloud coverage, selected over a whole year. We selected the facilities for which we embed methane plumes according to prior knowledge of emission source in the Permian basin. The dataset is produced using standardized COCO labels and common file types to encourage reusability by other researchers. The pipeline itself can be reproduced in other geographies, with more LES data and more Sentinel-2 scenes.

With a large dataset of synthetic methane plumes, we trained a deep learning image segmentation algorithm based on the U-Net architecture. For high frequency monitoring of emitting facilities, we want to operate on data that can be streamed from the Sentinel-2 API with minimal preprocessing an especially, without costly reference scene selection. For this, we leverage NDMI, a normalized difference index between Sentinel-2's methane sensitive SWIR bands that can be computed on the fly. On the synthetic data, we show that NDMI improves detectability of methane plumes in the heterogeneous Permian basin by 10.1% and is particularily effective at filtering out false positives. Furthermore, we find that Z-score normalization of NDMI increases the $F_{0.5}$ score by 23.4% compared to using raw NDMI. Overall, we successfully train a plume detection algorithm that can detect synthetic methane plumes from the above dataset.

Finally, we evaluate the simulation to reality gap when training on synthetic methane plumes. For this, we assemble a dataset of proven plumes in the Permian basin and run our plume detector on these scenes. We manage to detect 42% of the real plumes, showing the feasibility of training an algorithm on synthetic methane plumes to detect real plumes, without the need of costly preprocessing. We also find that training with NDMI but excluding SWIR bands leads to a higher detection threshold but no false positives. This is a promising characteristic for a high

troughput computer vision pipeline.
We stop short of making definite claims on the simulation to reality gap due to the low statistical significance of our test data.

Limitations

While we indicate transferability of an algorithm trained on synthetic plumes to real plumes, a gap in performances remains. The detection threshold of our algorithm is higher than non machine learning methods and reducing it remains an important barrier to deployment.

Our plume detector has also been trained on a low variability of plume shapes. Increasing the variability of embedded plumes through using more LES simulations could help bridge the simulation to reality gap.

We also fail to reliably measure performances on real plumes due to a small number of such publicly known real plumes. Building a larger test dataset would be necessary to validate the effectiveness of the method.

Outlook

Overall, our method demonstrates the possibility of training a plume detector for a specific region, without the need for collecting true labelled data. Deployed, such a dataset would then be reinforced by actual detections of real methane plumes. This opens the possibility of observing methane emissions in understudied regions.

Through training on observations of facilities catalogued in the OGIM database, we explore systematic surveying of methane emitting facilities. A next step for this project would be to stream Sentinel-2 observations of OGIM facilities and run our plume detector on them.

Bibliography

[1] The World Factbook 2021. Technical report, Central Intelligence Agency, Washington, DC, 2021.

[2] Energy Information administration. Drilling Productivity Report. Technical report, February 2023.

[3] Alana K. Ayasse, Philip E. Dennison, Markus Foote, Andrew K. Thorpe, Sarang Joshi, Robert O. Green, Riley M. Duren, David R. Thompson, and Dar A. Roberts. Methane Mapping with Future Satellite Imaging Spectrometers. *Remote Sensing*, 11(24):3054, January 2019. Number: 24 Publisher: Multidisciplinary Digital Publishing Institute.

[4] Jack Bruno, Dylan Jervis, Daniel Varon, and Daniel Jacob. U-Plume: Automated algorithm for plume detection and source quantification by satellite point-source imagers. *EGUsphere*, pages 1–24, August 2023. Publisher: Copernicus GmbH.

[5] John Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers, May 2020. arXiv:2005.12872 [cs].

[7] Copernicus. Sentinel-2 Spectral Response Functions (S2-SRF).

[8] Daniel H. Cusworth, Riley M. Duren, Andrew K. Thorpe, Winston Olson-Duvall, Joseph Heckler, John W. Chapman, Michael L. Eastwood, Mark C. Helmlinger, Robert O. Green, Gregory P. Asner, Philip E. Dennison, and Charles E. Miller. Intermittency of Large Methane Emitters in the Permian Basin. *Environmental Science & Technology Letters*, 8(7):567–573, July 2021. Publisher: American Chemical Society.

[9] Foivos I. Diakogiannis, François Waldner, Peter Caccetta, and Chen Wu. ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162:94–114, April 2020. arXiv:1904.00592 [cs].

[10] Sanne Diek, Fabio Fornallaz, Michael E. Schaepman, and Rogier De Jong. Barest Pixel Composite for Agricultural Areas Using Landsat Time Series. *Remote Sensing*, 9(12):1245, December 2017. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021. arXiv:2010.11929 [cs].

[12] Thibaud Ehret, Aurélien De Truchis, Matthieu Mazzolini, Jean-Michel Morel, Alexandre d'Aspremont, Thomas Lauvaux, Riley Duren, Daniel Cusworth, and Gabriele Facciolo. Global Tracking and Quantification of Oil and Gas Methane Emissions from Recurrent Sentinel-2 Imagery. *Environmental Science & Technology*, 56(14):10517–10529, 2022. _eprint: https://doi.org/10.1021/acs.est.1c08575.

[13] R. Escadafal and A. R. Huete. Influence of the viewing geometry on the spectral properties (High resolution visible and NIR) of selected soils from Arizona. Courchevel, January 1991.

[14] GIS Geography. Sentinel 2 - Bands (Combination). https://gisgeography.com/sentinel-2-bands-combinations/.

[15] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks, June 2014. arXiv:1406.2661 [cs, stat].

[16] I. E. Gordon, L. S. Rothman, C. Hill, R. V. Kochanov, Y. Tan, P. F. Bernath, M. Birk, V. Boudon, A. Campargue, K. V. Chance, B. J. Drouin, J. M. Flaud, R. R. Gamache, J. T. Hodges, D. Jacquemart, V. I. Perevalov, A. Perrin, K. P. Shine, M. A. H. Smith, J. Tennyson, G. C. Toon, H. Tran, V. G. Tyuterev, A. Barbe, A. G. Császár, V. M. Devi, T. Furtenbacher, J. J. Harrison, J. M. Hartmann, A. Jolly, T. J. Johnson, T. Karman, I. Kleiner, A. A. Kyuberis, J. Loos, O. M. Lyulin, S. T. Massie, S. N. Mikhailenko, N. Moazzen-Ahmadi, H. S. P. Müller, O. V. Naumenko, A. V. Nikitin, O. L. Polyansky, M. Rey, M. Rotger, S. W. Sharpe, K. Sung, E. Starikova, S. A. Tashkun, J. Vander Auwera, G. Wagner, J. Wilzewski, P. Wcisło, S. Yu, and E. J. Zak. The HITRAN2016 molecular spectroscopic database. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 203:3–69, December 2017.

[17] J. Gorroño, D. J. Varon, I. Irakulis-Loitxate, and L. Guanter. Understanding the potential of Sentinel-2 for monitoring methane point emissions. *Atmospheric Measurement Techniques*, 16(1):89–107, 2023.

[18] Lei Huang, Jie Qin, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. Normalization Techniques in Training DNNs: Methodology, Analysis and Application, September 2020. arXiv:2009.12836 [cs, stat].

[19] IEA. Global Methane Tracker 2023 – Analysis. https://www.iea.org/reports/global-methane-tracker-2023.

[20] Vladimir Iglovikov, Sergey Mushinskiy, and Vladimir Osin. Satellite Imagery Feature Detection using Deep Convolutional Neural Network: A Kaggle Competition, June 2017. arXiv:1706.06169 [cs].

[21] Itziar Irakulis-Loitxate, Luis Guanter, Yin-Nian Liu, Daniel J. Varon, Joannes D. Maasakkers, Yuzhong Zhang, Apisada Chulakadabba, Steven C. Wofsy, Andrew K. Thorpe, Riley M. Duren, Christian Frankenberg, David R. Lyon, Benjamin Hmiel, Daniel H. Cusworth, Yongguang Zhang, Karl Segl, Javier Gorroño, Elena Sánchez-García, Melissa P. Sulprizio, Kaiqin Cao, Haijian Zhu, Jian Liang, Xun Li, Ilse Aben, and Daniel J. Jacob. Satellite-based

survey of extreme methane emissions in the Permian basin. *Science Advances*, 7(27):eabf4507, June 2021. Publisher: American Association for the Advancement of Science.

[22] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks, November 2018. arXiv:1611.07004 [cs].

[23] Kriegler F. J. Preprocessing transformations and their effects on multspectral recognition. *Proceedings of the Sixth International Symposium on Remote Sesning of Environment*, pages 97–131, 1969.

[24] D. J. Jacob, D. J. Varon, D. H. Cusworth, P. E. Dennison, C. Frankenberg, R. Gautam, L. Guanter, J. Kelley, J. McKeever, L. E. Ott, B. Poulter, Z. Qu, A. K. Thorpe, J. R. Worden, and R. M. Duren. Quantifying methane emissions from the global scale down to point sources using satellite observations of atmospheric methane. *Atmospheric Chemistry and Physics*, 22(14):9617–9646, 2022.

[25] Johannes Jakubik, Linsong Chu, Paolo Fraccaro, Carlos Gomes, Gabby Nyirjesy, Ranjini Bangalore, Devyani Lambhate, Kamal Das, Dario Oliveira Borges, Daiki Kimura, Naomi Simumba, Daniela Szwarcman, Michal Muszynski, Kommy Weldemariam, Bianca Zadrozny, Raghu Ganti, Carlos Costa, Campbell Edwards, Blair & Watson, Karthik Mukkavilli, Hendrik Schmude, Johannes & Hamann, Parkin Robert, Sujit Roy, Christopher Phillips, Kumar Ankur, Muthukumaran Ramasubramanian, Iksha Gurung, Wei Ji Leong, Ryan Avery, Rahul Ramachandran, Manil Maskey, Pontus Olofossen, Elizabeth Fancher, Tsengdar Lee, Kevin Murphy, Dan Duffy, Mike Little, Hamed Alemohammad, Michael Cecil, Steve Li, Sam Khallaghi, Denys Godwin, Maryam Ahmadi, Fatemeh Kordi, Bertrand Saux, Neal Pastick, Peter Doucette, Rylie Fleckenstein, Dalton Luanga, Alex Corvin, and Erwan Granger. Prithvi-100M, August 2023.

[26] Dylan Jervis, Jason McKeever, Berke O. A. Durak, James J. Sloan, David Gains, Daniel J. Varon, Antoine Ramier, Mathias Strupler, and Ewan Tarrant. The GHGSat-D imaging spectrometer. *Atmospheric Measurement Techniques*, 14(3):2127–2140, March 2021. Publisher: Copernicus GmbH.

[27] P. Joyce, C. Ruiz Villena, Y. Huang, A. Webb, M. Gloor, F. H. Wagner, M. P. Chipperfield, R. Barrio Guilló, C. Wilson, and H. Boesch. Using a deep neural network to detect methane point sources and quantify emissions from PRISMA hyperspectral satellite images. *EGUsphere*, 2022:1–22, 2022.

[28] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017. arXiv:1412.6980 [cs].

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[30] Yann Lecun, Koray Kavukcuoglu, and Clement Farabet. Convolutional Networks and Applications in Vision. In *ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems*, pages 253–256, May 2010.

[31] Feng Li, Hao Zhang, Huaizhe xu, Shilong Liu, Lei Zhang, Lionel M. Ni, and Heung-Yeung Shum. Mask DINO: Towards A Unified Transformer-based Framework for Object Detection and Segmentation, December 2022. arXiv:2206.02777 [cs].

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context, February 2015. arXiv:1405.0312 [cs].

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, August 2021. arXiv:2103.14030 [cs].

[34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation.

[35] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, January 2019. arXiv:1711.05101 [cs, math].

[36] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning Deconvolution Network for Semantic Segmentation, May 2015. arXiv:1505.04366 [cs].

[37] Ilissa B. Ocko, Tianyi Sun, Drew Shindell, Michael Oppenheimer, Alexander N. Hristov, Stephen W. Pacala, Denise L. Mauzerall, Yangyang Xu, and Steven P. Hamburg. Acting rapidly to deploy readily available methane mitigation measures by sector can immediately slow global warming. *Environmental Research Letters*, 16(5):054042, May 2021. Publisher: IOP Publishing.

[38] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y. Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention U-Net: Learning Where to Look for the Pancreas, May 2018. arXiv:1804.03999 [cs].

[39] M. Omara, R. Gautam, M. O'Brien, A. Himmelberger, A. Franco, K. Meisenhelder, G. Hauser, D. Lyon, A. Chulakadaba, C. Miller, J. Franklin, S. Wofsy, and S. Hamburg. Developing a spatially explicit global oil and gas infrastructure database for characterizing methane emission sources at high resolution. *Earth System Science Data Discussions*, 2023:1–35, 2023.

[40] Paul I. Palmer, Liang Feng, Mark F. Lunt, Robert J. Parker, Hartmut Bösch, Xin Lan, Alba Lorente, and Tobias Borsdorff. The added value of satellite observations of methane forunderstanding the contemporary methane budget. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2210):20210106, 2021. _eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2021.0106.

[41] Michael J. Prather, Christopher D. Holmes, and Juno Hsu. Reactive greenhouse gas scenarios: Systematic exploration of uncertainties and the role of atmospheric chemistry. *Geophysical Research Letters*, 39(9), 2012. _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2012GL051440.

[42] Shukla Pratik. How did Binary Cross-Entropy Loss Come into Existence? *Towards AI*, March 2023. Publisher: Towards AI Co.

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, May 2015. arXiv:1505.04597 [cs].

[44] M. Saunois, A. R. Stavert, B. Poulter, P. Bousquet, J. G. Canadell, R. B. Jackson, P. A. Raymond, E. J. Dlugokencky, S. Houweling, P. K. Patra, P. Ciais, V. K. Arora, D. Bastviken, P. Bergamaschi, D. R. Blake, G. Brailsford, L. Bruhwiler, K. M. Carlson, M. Carrol, S. Castaldi, N. Chandra, C. Crevoisier, P. M.

Crill, K. Covey, C. L. Curry, G. Etiope, C. Frankenberg, N. Gedney, M. I. Hegglin, L. Höglund-Isaksson, G. Hugelius, M. Ishizawa, A. Ito, G. Janssens-Maenhout, K. M. Jensen, F. Joos, T. Kleinen, P. B. Krummel, R. L. Langenfelds, G. G. Laruelle, L. Liu, T. Machida, S. Maksyutov, K. C. McDonald, J. McNorton, P. A. Miller, J. R. Melton, I. Morino, J. Müller, F. Murguia-Flores, V. Naik, Y. Niwa, S. Noce, S. O'Doherty, R. J. Parker, C. Peng, S. Peng, G. P. Peters, C. Prigent, R. Prinn, M. Ramonet, P. Regnier, W. J. Riley, J. A. Rosentreter, A. Segers, I. J. Simpson, H. Shi, S. J. Smith, L. P. Steele, B. F. Thornton, H. Tian, Y. Tohjima, F. N. Tubiello, A. Tsuruta, N. Viovy, A. Voulgarakis, T. S. Weber, M. van Weele, G. R. van der Werf, R. F. Weiss, D. Worthy, D. Wunch, Y. Yin, Y. Yoshida, W. Zhang, Z. Zhang, Y. Zhao, B. Zheng, Q. Zhu, Q. Zhu, and Q. Zhuang. The Global Methane Budget 2000–2017. *Earth System Science Data*, 12(3):1561–1623, 2020.

[45] Divya Saxena and Jiannong Cao. Generative Adversarial Networks (GANs Survey): Challenges, Solutions, and Future Directions, April 2023. arXiv:2005.00065 [cs, eess, stat].

[46] Linda Shapiro and George Stockman. *Computer Vision*. Prentice Hall, 2022.

[47] Statista. U.S.: shale gas monthly production by basin 2022-2023. https://www.statista.com/statistics/1369661/monthly-shale-gas-production-in-the-united-states-by-basin/.

[48] D. R. Thompson, A. K. Thorpe, C. Frankenberg, R. O. Green, R. Duren, Luis Guanter, Andre Hollstein, E. Middleton, L. Ong, and S. Ungar. Space-based remote imaging spectroscopy of the Aliso Canyon CH4 superemitter. *Geophysical Research Letters*, 43(12):6571–6578, 2016. Publisher: Wiley Online Library.

[49] NASA-JPL U. Arizona and EDF. Operator Performance Dashboard.

[50] D. J. Varon, D. Jervis, J. McKeever, I. Spence, D. Gains, and D. J. Jacob. High-frequency monitoring of anomalous methane point sources with multispectral Sentinel-2 satellite observations. *Atmos. Meas. Tech.*, 14(4):2771–2785, April 2021. Publisher: Copernicus Publications.

[51] Daniel J. Varon, Daniel J. Jacob, Benjamin Hmiel, Ritesh Gautam, David R. Lyon, Mark Omara, Melissa Sulprizio, Lu Shen, Drew Pendergrass, Hannah Nesser, Zhen Qu, Zachary R. Barkley, Natasha L. Miles, Scott J. Richardson, Kenneth J. Davis, Sudhanshu Pandey, Xiao Lu, Alba Lorente, Tobias Borsdorff, Joannes D. Maasakkers, and Ilse Aben. Continuous weekly monitoring of methane emissions from the Permian Basin by inversion of TROPOMI satellite observations. *Atmospheric Chemistry and Physics*, 23(13):7503–7520, July 2023. Publisher: Copernicus GmbH.

[52] Anna Vaughan, Gonzalo Mateo-García, Luis Gómez-Chova, Vít Růžička, Luis Guanter, and Itziar Irakulis-Loitxate. CH4Net: a deep learning model for monitoring methane super-emitters with Sentinel-2 imagery. *EGUsphere*, pages 1–17, May 2023. Publisher: Copernicus GmbH.

[53] Cody M. Webber and John P. Kerekes. An Examination of Enhanced Atmospheric Methane Detection Methods for Predicting Performance of a Novel Multiband Uncooled Radiometer Imager. preprint, Gases/Remote Sensing/-Data Processing and Information Retrieval, April 2020.

[54] Takanobu Yamaguchi and Graham Feingold. Technical note: Large-eddy simulation of cloudy boundary layer with the Advanced Research WRF model. *Journal of Advances in Modeling Earth Systems*, 4(3), 2012. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2012MS000164.

[55] Y. Zha, J. Gao, and S. Ni. Use of normalized difference built-up index in automatically mapping urban areas from TM imagery. *International Journal of Remote Sensing*, 24(3):583–594, January 2003. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01431160304987.

[56] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. UNet++: A Nested U-Net Architecture for Medical Image Segmentation, July 2018. arXiv:1807.10165 [cs, eess, stat].

[57] Bryan Zhu, Nicholas Lui, Jeremy Irvin, Jimmy Le, Sahil Tadwalkar, Chenghao Wang, Zutao Ouyang, Frankie Y. Liu, Andrew Y. Ng, and Robert B. Jackson. METER-ML: A Multi-Sensor Earth Observation Benchmark for Automated Methane Source Mapping, August 2022. arXiv:2207.11166 [cs].

[58] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, August 2020. arXiv:1703.10593 [cs].

Appendix A

Figures

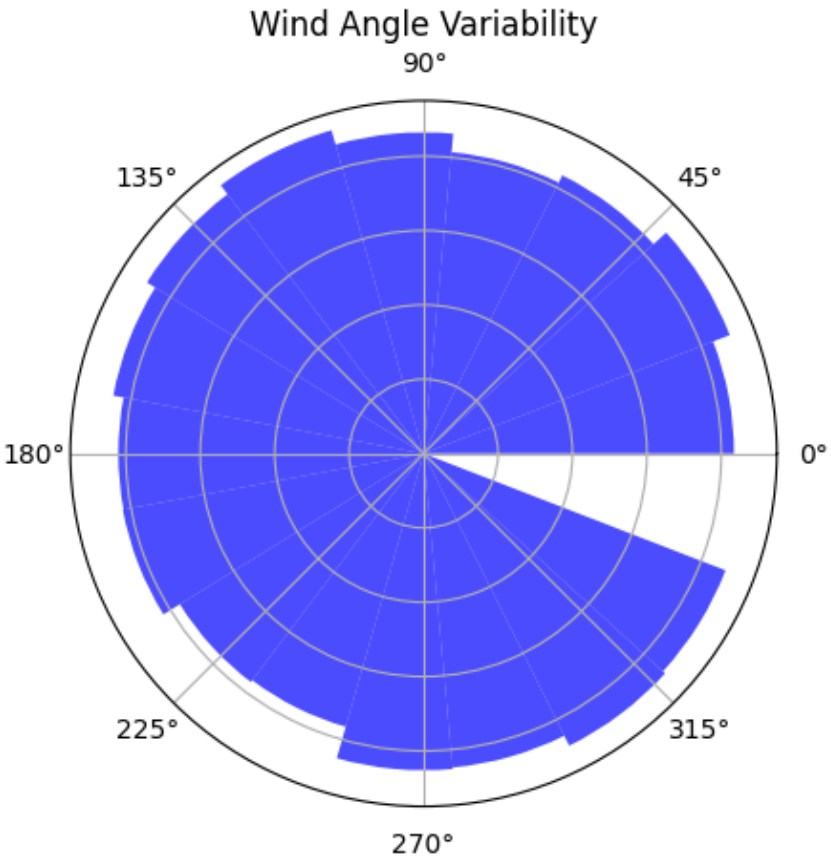


Figure A.1: Count of wind angle in synthetic plume dataset

Appendix B

Code Snippets

Listing B.1: NDMI SNR code

```
import numpy as np

ndmi_methane = (b12_synthetic - b11_synthetic) / (b12_synthetic
    + b11_synthetic)
ndmi_no_methane = (b12 - b11) / (b12 + b11)
ndmi_plume = ndmi_methane - ndmi_no_methane

threshold = 1
snr = ndmi_methane / np.nanstd(ndmi_methane)
mask = np.where(snr >= threshold, 1, 0)
```

Listing B.2: COCO label to binary mask code

```
import torch
import numpy as np
from PIL import Image
catid2color = {
    1: [255, 0, 0],
}
binary_mask = np.zeros((image_width, image_height), dtype=np.
    uint8)
mask_filename = annotation["file_name"]
mask_filepath = f"{mask_directory}/{mask_filename}"
category_id = 1

mask_image = Image.open(mask_filepath)
mask_array = np.array(mask_image)

mask_check = np.all(mask_array == catid2color[category_id],
    axis=-1)
binary_mask[mask_check] = category_id
binary_mask = torch.tensor(binary_mask).to(torch.float32)
```