Automated detection of methane emissions in TROPOMI data using Neural Networks

Berend J. Schuit

MSc. Thesis Report
MSc. Aerospace Engineering - Spaceflight
Faculty of Aerospace Engineering
Delft University of Technology

March 12, 2021

Automated detection of methane emissions in TROPOMI data using Neural Networks

by

Berend J. Schuit

to obtain the degree of Master of Science in Aerospace Engineering, Spaceflight
at the Delft University of Technology,
to be defended publicly on Friday March 19, 2021 at 10:00 AM.

Grade: / 10

Student number:	4415523	
Project duration:	May 11, 2020 – March 19, 2021	
Thesis committee:	Dr. D. M. Stam	TU Delft, Thesis supervisor
	Dr. J. D. Maasakkers	SRON, Main supervisor
	Prof. Dr. I. Aben	SRON, Supervisor
	Dr. W. van der Wal	TU Delft, Committee chair
	Dr. I. C. Dedoussi	TU Delft, Committee member

TU Delft

"Now that it was visible, it was no longer acceptable."

- David Attenborough

Preface

The research project which is documented in this report took place in the context of my Master Thesis in Spaceflight, Aerospace Engineering at Delft University of Technology. The research project took place at The Netherlands Institute for Space Research (SRON) and as an intern I was part of the Earth Science Group, more specifically the TROPOMI CH_4 Level 4 team, from May 2020 until March 2021. Eventhough nobody could be physically present at the research institute due to the corona pandemic, the scientists at SRON did their best to make the most out of the remote situation for all of the interns by providing online group meetings, close supervision and regular student meetings. I am very grateful to the scientists at SRON for the opportunity to have been part of this inspiring scientific environment in which I was able to learn a lot during this project.

I would like to thank my supervisors, Bram & Ilse (SRON) and Daphne (TU Delft) without whom this project would not have been possible. The very thorough feedback they gave on draft versions of the literature study report and this final thesis report allowed me to further improve it and bring it to a level I could not have achieved without them. I want to thank Bram & Ilse for setting out the general outlines of the project before the start and for their excellent guidance and supervision throughout the project. I would like to thank Ilse for setting up a group meeting where I could present my work for the scientists at SRON.
I also want to thank all members of the committee in advance for the defense and for taking the time to read my report.

My special thanks go out to Bram, who as my main supervisor provided close guidance during our weekly meetings and also outside of those meetings whenever I was stuck with a question, which I am grateful for. Bram really took the extra step as a supervisor and I am really grateful for his great supervision. For instance he set up a mock defense for me in order to prep for the real one, in which several fellow interns and colleagues did participate. I also would like to thank all colleagues who participated in this.

I would also like to thank all my colleagues, most of whom did an internship at the same time at SRON, this past year: Lodewijck, Gourav, Allard, Jurian, Gijs, Stijn, Brian and Marius. It would have been great to have been able to work together in the student's room at SRON and have gone out for beers on Friday afternoons. I would like to especially thank Lodewijck for showing me the ropes with the code at the start of my project.
I want to thank Lodewijck, Gourav & Bram for the great collaboration on the TROPOMI-GHGSat project. I really enjoyed being able to take part in this amazing project as part of my time at SRON. This was a very valueable additional experience. My thanks also go out to the GHGSat team for the three-weekly meetings and for the opportunity to present my research during one of those.
I also would like to thank Sander and Bram Albers, a student supervised by Sander with a research project with close similarities to mine, for the Machine Learning student meetings in which we shared insights and information on both our projects.
I would also like to thank Dr. Yasjka Meijer, who kindly provided me with two figures from his presentation on CO2M at SRON for in my report.

At last I want to thank my roommates for tolerating me overloading our home wifi-network with gigabytes of satellite data. I also want to thank my parents, Willy & Rob and my brother Wouter, for their support and genuine interest in my project. And ofcourse Linda, for not getting tired of me talking about plumes, plumes and plumes all the time and of me sending her news articles about TROPOMI and GHGSat, but most of all for her understanding and support.

Berend Schuit

March 12, 2021
Delft

Abstract

Methane (CH_4) is one of the major greenhouse gases in our atmosphere driving climate change, responsible for about one quarter of the warming since pre-industrial times. Anthropogenic sources constitute a significant portion of global methane emissions. A relatively small fraction of anthropogenic point sources contributes a disproportionally large amount to total CH_4 emissions. The TROPOspheric Monitoring Instrument (TROPOMI) has been observing atmospheric total column CH_4 concentrations with daily global coverage at a spatial resolution of 5.5×7 km^2 since October 2017.
Rapid detection of point source emissions is vital to allow identification and mitigation. However, manual inspection of the very large TROPOMI dataset is infeasible. In order to automatically detect methane plumes originating from point sources with an as small as possible delay after measurement, a Convolutional Neural Network (CNN) has been designed in this study. The CNN consists of 9 layers in total, which was found to be optimal. This CNN model was trained using data from the vast archive of TROPOMI observations. The training dataset consists of 407 scenes containing a plume, obtained by manual inspection of the TROPOMI data, and over 2000 scenes with no plumes. The CNN is capable of detecting CH_4 plumes originating from point sources, above the detection threshold, based on single-overpass data. The CNN has learned the generalized features of a CH_4 plume signal in TROPOMI data based on this dataset. Subsequently, an accuracy of 97% was obtained on the test set.
In order to further improve the automated detection, the most difficult to classify scenes were mined after which the model was further improved using this additional data. Several algorithms were developed to extract additional metrics (Feature Engineering) from the scenes potentially containing a plume. These metrics were used as input for a second model, a Fully-connected Neural Network (FcNN) which further separates real plumes and scenes which look like plumes in the CH_4 field but are in fact artefacts. This FcNN correctly classifies 88% of the plumes out of the scenes flagged by the CNN in the second test set in an automated way.
These two Neural Networks combined correctly identified 86 out of 97 plumes with 29 false detections out of a total of 17,243 scenes. The Feature Engineering algorithms also provide a plume mask and rough estimates of the source location, the total emissions and the source rate for each detected plume.
This automated process of plume detection is applied to all available TROPOMI CH_4 data of 2020, something which would be unfeasible to do manually. This analysis provides insight into the spatial and temporal distribution of point source emissions and provides an estimate on the total amount of CH_4 emissions resulting from these point sources for the year 2020. The detected point source emissions together are estimated to account for 1.82 Tg of CH_4 emissions. When extrapolating this, while taking the ratio of detections over valid retrieval scenes into account, we arrive at an estimate of 17.2 Tg a^{-1}. This is respectively 0.55% and 5.2% of global annual anthropogenic CH_4 emissions.
The developed model can also be applied to recent measurements, with the potential to automatically detect a CH_4 plume only 20 minutes after the TROPOMI CH_4 data becomes available. The plumes which are detected in single overpass measurements using this approach can be further analysed in order to attribute them to a specific source. This can be done through further analysis of prior TROPOMI data or by targeting a high resolution satellite instrument (e.g. GHGSat) based on the location of the identified plume.

Key Points

- A Convolutional Neural Network was designed which is capable of detecting CH_4 plume-like features by morphology.
- A Fully-connected Neural Network further distinguishes between plumes and artefacts.
- Estimates of source location, total emissions, source rate and a plume mask are automatically generated.
- A processing pipeline is used to automatically detect plumes unsupervised in new - recently measured - TROPOMI CH_4 data, effectively providing global monitoring.
- An inventory was created of all plumes detected in TROPOMI CH_4 data for the year 2020. The detected plumes are estimated to account for 1.82 Tg of CH_4 emissions in total.

Keywords

TROPOMI, Methane, CH4, Plume, Super-emitter, Point source emissions, Machine Learning, Convolutional Neural Network, CNN, Feature Engineering, Fully-connected Neural Network, FcNN

Nomenclature

Physics Constants

M_{air}	mean molecular weight of air	28.96×10^{-3} kg mol^{-1}
M_{CH_4}	molecular weight of methane	16.043×10^{-3} kg mol^{-1}
g	gravitational acceleration	9.80665 m s^{-2}
η	Avogadro constant	$6.02214076 \times 10^{23}$ mol^{-1}
R_{Earth}	Radius of the Earth	6.378×10^{6} m
p_0	Sea level standard pressure	101,325 Pa = 1013.25 hPa

Neural Network symbols

n_H	size of the height of the image
n_W	size of the width of the image
n_C	number of channels
p	padding
s	stride
I	input image
K	Kernel, sometimes referred to as filter
Ψ	activation function
b	bias
a	arbitrary layer in the CNN
CONV	convolutional operator
POOL	pooling operator
ϕ	pooling function
$\hat{y}$	prediction of a certain sample
y	true value, also referred to as label
J	loss function
L	cost function
m	size of the training dataset
θ	model parameters ((filter)weights+biases)
G	backpropagation function
κ	Cohen's Kappa (not to be confused with K)
TP	true positive
TN	true negative
FP	false positive
FN	false negative

Other symbols

CH_4	methane	
CO_2	carbon dioxide	
CO	carbon monoxide	
μ	mean value of a dataset	
σ	standard deviation of a dataset	
R^2	Coefficient of determination	
δm	a change in the atmospheric mass of a certain greenhouse gas	kg

Definitions

a posteriori	"from the later"
a priori	"from the earlier"
GWP	Global Warming Potential , "The time-integrated Radiative Forcing due to a pulse emission of a given component, relative to a pulse emission of an equal mass of CO_2." [1]
RF	Radiative Forcing [W m^{-2}] , "The resulting flux imbalance in the radiative budget for the Earth system, caused by a change δm in the atmospheric mass of a greenhouse gas X." [2]
xch4	atmospheric methane total column dry mixing ratio [ppb] , "The total number of CH_4 molecules in a vertical column above a unit surface divided by the corresponding total number of molecules of dry air in that column." [3]

Abbreviations

AI	Artificial Intelligence 24
CAM	Class Activation Map 51, 74
CAMS	Copernicus Atmosphere Monitoring Service 42
CH_4	Methane i–iii, 1, 3, 14, 17–23, 25, 26, 37–42, 44–46, 49, 51, 52, 55, 60, 61, 63, 64, 66–68, 71–77, 84
CNN	Convolutional Neural Network iii, 1, 2, 24–28, 30, 32, 33, 40, 42, 44–58, 62, 65–69, 74, 75
CO	Carbon monoxide 18, 41, 76
CO_2	Carbon dioxide 3, 14
CV	Computer Vision 24, 26
DL	Deep Learning 24, 25, 38, 47, 51, 53
EDGAR	Emission Database for Global Atmospheric Research 10
EPA	Environmental Protection Agency 72
ESA	European Space Agency 15, 17, 20
FcNN	Fully-connected Neural Network iii, 2, 25, 32, 48, 50, 65, 66, 68, 69, 75, 76
GCM	General circulation model 5
GMAO	Global Modeling and Assimilation Office 40
GOSAT	Greenhouse Gases Observing Satellite 18
GWP	Global Warming Potential 4–6
IME	Integrated Mass Enhancement 60, 61, 67, 83
KNMI	Royal Netherlands Meteorological Institute 15, 18
LES	Large Eddy Simulations 61
ML	Machine Learning 1, 24, 37
NASA	National Aeronautics and Space Administration 40
NIR	Near infrared 41, 63
NN	Neural Network iii, 2, 24–26, 28, 29, 32, 33, 38, 40–42, 44, 46, 51, 58, 65–68, 72, 75, 76
NPP	Suomi National Polar-Orbiting Partnership 15, 16
PCA	Principal Component Analysis 64
ResNet	Residual (convolutional neural) Network 53, 54, 56, 74
RF	Radiative Forcing 5
S5P	Sentinel 5 Precursor 15, 16
SNR	Signal-to-noise ratio 18
SRON	The Netherlands Institute for Space Research ii, 1, 12, 15, 17–21, 37, 68
SWIR	Short-wave infrared 13–16, 18, 19, 37, 41, 42, 63
SZA	Solar zenith angle 18, 41
TCCON	Total Carbon Column Observing Network 18
TIR	Thermal infrared 13–15
TM5	Tracer Model 5, The global chemistry transport model 46
TROPOMI	TROPOspheric Monitoring Instrument ii, iii, 1, 2, 14–21, 23, 25, 26, 37–42, 44, 46, 49, 51, 52, 55, 56, 63, 66–69, 71, 73–77, 83

VIIRS Visible/Infrared Imager and Radiometer Suite 15, 18, 19
VZA Viewing zenith angle 18, 41

Contents

cover image credits:
KNMI, https://www.knmi.nl/kennis-en-datacentrum/uitleg/tropomi, last access: 10-03-2021
GHGSat, https://www.ghgsat.com/our-platforms/iris/, last access: 10-03-2021

1
Introduction

This report documents a research study performed at The Netherlands Institute for Space Research (SRON) as part of the TROPOMI CH_4 Level 4 team. The Netherlands Institute for Space Research (SRON) is the institute responsible for the scientific management of the TROPOspheric Monitoring Instrument (TROPOMI) Methane (CH_4) product. This research project was set out by Dr. J.D. Maasakkers and Prof. Dr. I. Aben. The context of this project is the TROPOMI-GHGSat collaboration, in which SRON takes part. This collaboration is discussed in detail in Section 2.6.

The goal of this research project is to design a machine learning approach to automatically detect Methane (CH_4) plume emissions, originating from point sources, in TROPOMI CH_4 Level 2 data with sufficient accuracy. The research questions are stated in Section 3.4.
The context and theoretical background of this research study will be discussed in more detail in the first few chapters.

An example of what a large methane plume looks like in TROPOMI data is shown in Figure 1.1. A plume is characterised by high atmospheric methane concentrations (ppb, parts per billion) which are transported downwind from the location of emission. An example of a methane plume is already shown here for convenience and for clarity, before discussing other aspects of the research.

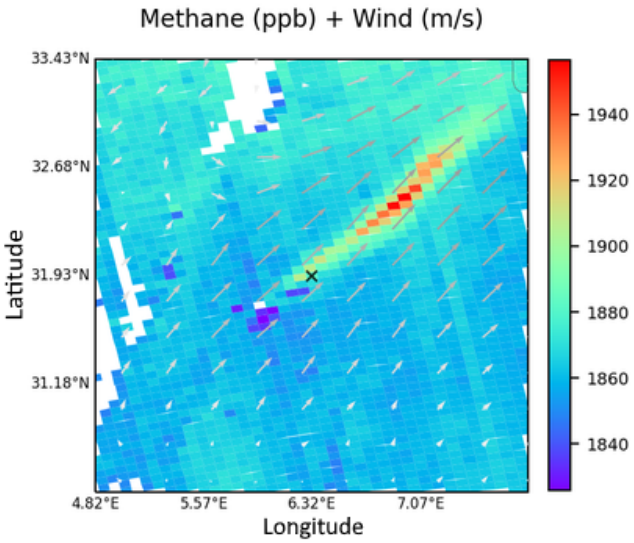


Figure 1.1: An example of a large methane plume in TROPOMI measurement data. The arrows indicate the direction and speed of the wind.

In Chapter 2 the theoretical background on the scientific fields which are part of this study is discussed. The basics on atmospheric methane are covered and the influence of methane on global warming is discussed. Next it is discussed how methane is put into and taken out of the atmosphere (sources & sinks) and which part anthropogenic (human) factors have in this. Next the physical basic principles of measuring atmospheric methane are discussed, with special attention to how satellites are used for this. The TROPOMI and also the GHGSat instruments are discussed in more detail, as those are the instruments used in the TROPOMI-GHGSat project. More details about this collaboration and how the instruments complement each other are discussed as well.

The context of the research project, the opportunities leading to this project and the relevance are discussed in Chapter 3. This chapter is concluded with the research questions in Section 3.4 which this research project aims to answer.

Chapter 4 elaborates on the Machine Learning (ML) methodology used in this study and why this approach was selected. It provides a conceptual description on the selected ML methodology, the Convolutional Neural Network (CNN), and also a more in-depth mathematical description on the different components of a CNN. The performance evaluation metrics which are later used to infer classification performance are defined here.

Chapter 5 focuses on how the TROPOMI CH_4 Level 2 dataproduct can be converted to a suitable format for utilisation in a ML problem. The different atmospheric parameters included are discussed and the training data which is used to train the CNN in order to detect CH_4 plumes is discussed.

Chapter 6 is about the development of the architecture of the model. First a simple baseline model is made to compare performance results against, then the different CNN models which were designed are discussed and compared in terms of performance. After a functioning model was created, it is also discussed how further improvements were

realized and how the optimal model was selected.

In addition to the CNN model, several feature engineering algorithms were developed in order to obtain valueable information about a potential plume and in order to infer whether the potential plume is a real plume or an artefact. The standardized format to which the TROPOMI data is converted (as discussed in Chapter 5) is key here. Those feature engineering algorithms are elaborated in Chapter 7. In Chapter 8 it is elaborated how these metrics, resulting from those algorithms, are used by a second Neural Network (NN), a Fully-connected Neural Network (FcNN) in order to further automate the detection process.

Figure 1.2 shows how the different parts of the model are related to one another and in which Chapter those are described.

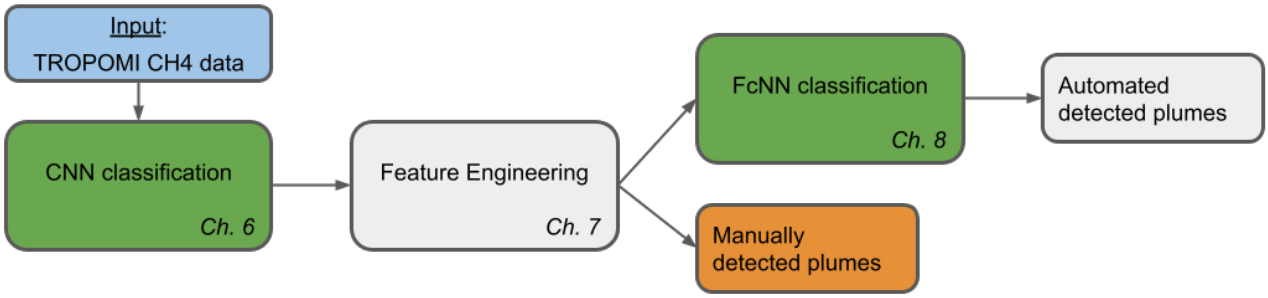


Figure 1.2: A schematic overview of the different components of the model which is developed.

The applications of the workflow and data pipeline which is developed during this research project are discussed in Chapter 9. The main elements here are the applications within the TROPOMI-GHGSat project and an inventory of detected plumes with high emission rates in the year 2020, which would not have been feasible to perform without the developed NNs. Based on the detected plumes a rough estimate is made of the total emissions related to these detected point source emissions.
The report is concluded with a chapter stating the conclusions and opportunities for further research.

2
Theoretical Background

2.1. Methane in the Earth's atmosphere

Methane (CH_4), which is an invisible and odorless gas, is one of the most important greenhouse gasses. After Carbon dioxide (CO_2), methane has the largest contribution to human induced global warming [4]. However the fraction of the atmosphere consisting of methane molecules is much smaller compared to the fraction of carbon dioxide, the effect per additional unit of mass of methane is much stronger. Before moving on to the relevance of methane as a greenhouse gas, the basics of atmospheric methane are covered first.

The Earth's atmosphere is the layer of air covering our rocky planet. It is the part of the biosphere humanity lives in and most transportation phenomena of the natural planet occur in, such as weather. 99.999% of the particles (and thus mass) of the atmosphere are located below an altitude of 80 km above sea level [2]. Relative to the radius of the Earth (6378 km) this is just a very tiny layer. There is no upper boundary of the atmosphere, the air content decreases further with altitude until it levels with the empty space around the Earth in outer space.
The mean pressure exerted by the Earth's atmosphere at sea-level is $1.013 \cdot 10^5$ Pa, which is equal to 1013 hPa or 1 atm (atmosphere). The global mean pressure at the surface of Earth (which is on average a few kilometers above sea level) is 984 hPa. From this the total mass of the atmosphere over the entire globe can be computed to be roughly $5.2 \cdot 10^{18}$ kg [2].

The atmospheric pressure decreases exponentially with altitude (US standard atmospheric model, [5]), this explains why the mass of the atmosphere is concentrated at the bottom. Figure 2.1 displays this relation. Three main "zones" within the atmosphere can be distinguished when we consider the domain below 80 km altitude, based on the temperature-regimes. The temperature does not decrease exponentially with altitude, but the gradient of the atmospheric temperature inverts twice with respect to altitude within the domain below 80 km. This flipping gradient causes the layers to be separate, mixing does not occur between the separate layers as easily as it does within a layer. This results in three altitude-domains called the Troposphere, the Stratosphere and the Mesosphere, their rough altitude ranges are indicated in Figure 2.1. The "boundaries" where the temperature gradient inverts are called the boundary layers, those are the Tropopause and the Stratopause.

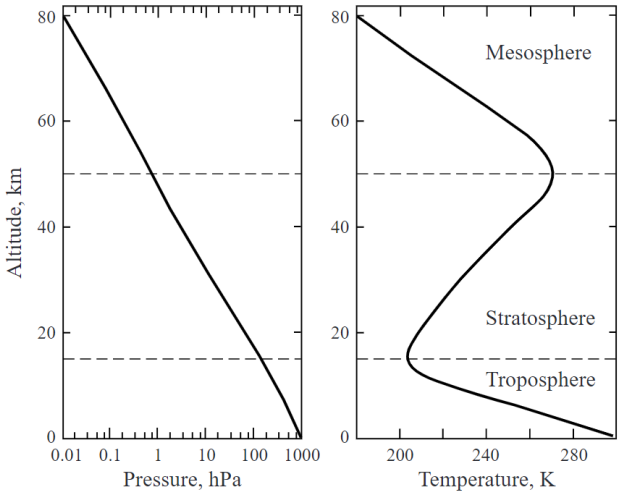


Figure 2.1: General model of the pressure and temperature profiles versus altitude in the exponential atmosphere model (latitude of 30°N, March). [2]

Most of the atmospheric methane is located in the troposhere, this will be discussed in more detail in the following sections. This is a relevant property for remote sensing as we will see in Figure 2.8 in Section 2.4.1.

The gases which are most abundant in the atmosphere are N_2, O_2, Ar and H_2O. In dry air the mixing ratios of these abundant gases are 0.78, 0.21 and 0.0093 respectively. Large fluctuations in the amount of water vapor (H_2O) are present in the atmosphere. The dry mixing ratio denotes which fraction of the total amount of molecules in the atmosphere (excluding H_2O) is of the particular gas (i.e. 21% of the atmosphere is oxygen (O_2)). The mixing ratio remains constant when variable atmospheric properties such as temperature or pressure change, this makes it a robust measure of the gas-contents in the atmosphere. [2]. This principle is illustrated in Figure 2.9a.

Methane is one of the so-called trace gasses present in the Earth's atmosphere. The mixing ratio of methane in the atmosphere is roughly speaking $1.8 \cdot 10^{-6}$ which is equivalent to 1.8 ppm (parts per million) or 1800 ppb (parts per billion) [6] [2]. The concentration of methane in the atmosphere is usually denoted as parts per billion in literature.

To be more exact, the average mixing ratio of methane was 1803.2 ± 2 ppb in 2011, which is 250.4% of the level before the industrial era [4] (750 ppb is most commonly used as the baseline). The methane content in the atmosphere has increased enormously since the beginning of the industrial era.
Palaeoclimatic records are a source for methane contents in the earlier ages of the Earth. Ice cores go back 740 kyr, deep oceanic sediments go back several million years. The air trapped in those records has been thoroughly analysed by experts to find the atmospheric methane contents during those ages. The earth has gone through several glacial-interglacial cycles the last half a million years, about 20% of these cycles is spent in the warm interglacial part. Currently we are in an interglacial period. [7]

Methane levels have fluctuated between about 400 ppb and 700 ppb during glacial and inter-glacial periods [8]. At pre-industrial times the methane content in the atmosphere was 720 ppb [6]. In 2011 the atmospheric methane concentration was around 1800 ppb. Now, in 2021 the average concentration is getting closer to 1900 already in the northern hemisphere.
The mean molecular weight of air (M_{air}) is $28.96 \cdot 10^{-3}$ kg mol^{-1} [2]. As stated earlier, the total mass of the atmosphere is $5.2 \cdot 10^{18}$ kg. Taking into account the mixing ratio of methane (xch4) in the Earth's atmosphere (approximately 1800 ppb) and the molecular weight of methane (M_{CH_4}) is $16.043 \cdot 10^{-3}$ kg mol^{-1}, an estimate of the total mass of methane in the atmosphere can be computed. The total mass of methane in the atmosphere is about $5.2 \cdot 10^{12}$ kg, 5.2 Gt, 5200 Mt or 5200 Tg.
This value helps to put in perspective yearly methane emissions in terms of magnitude compared to the current composition.

2.2. The relevance of Methane as a greenhouse gas

Methane is a well known greenhouse gas, after CO_2 it has the largest contribution to global warming. Over a period of 20 years (1 kg of) CH_4 has a ≈ 84 times higher Global Warming Potential (GWP) (more on GWP later in this section) compared to CO_2. [4]

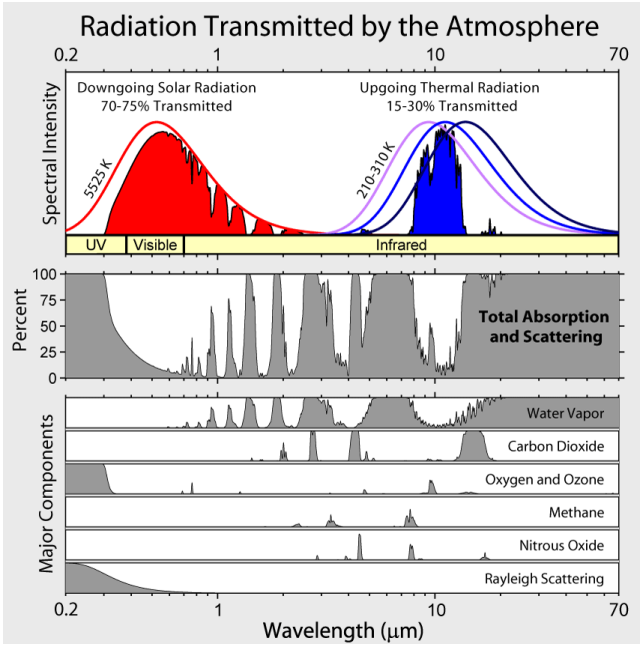


Figure 2.2: The atmospheric absorption bands of CH_4 and the total atmosphere as a whole. The atmospheric window is clearly indicated in blue.[1]

Greenhouse gasses are gasses that absorb terrestrial radiation emitted by the Earth (ranging from 5 to 50 μm). Without these gasses the radiation would dissipate to space, but since the gasses are present the radiation is being trapped in the Earth system. Figure 2.2 shows the radiation emitted by the sun and the Earth (with the lines indicating the theoretical black bodies radiation for the best fitting temperature) and the absorption of the atmosphere at particular wavelengths by specific greenhouse gases. The absorption wavelengths are schematised quite roughly in this figure. The shown CH_4 absorption feature at about 8 μm is the major absorption wavelength, but there are several. Methane also has absorption features at 1.65 μm and 2.3 μm [6] (the absorption feature at 2.3 μm is also visible in Figure 2.2). Those two are in the near-infrared (between visible and infrared), which means they can be measured quite well by several instruments as will be discussed in Section 2.4.1. CH_4 absorbs radiation in these particular wavelengths because only this specific radiation can be used by the molecule to increase the internal energy level since this is a quantized process [2].

The most important absorption band however is the band at about 8 μm. This band is within the atmospheric window, which ranges roughly speaking from 8 to 13 μm. Radiation from the surface of the Earth in this wavelength can escape the atmosphere relatively easily, which is very important for the temperature balance of the Earth's surface [2]. Figure 2.2 shows the atmospheric window and the CH_4 absorption band at around 8 μm.

While CO_2 is present in much larger quantities and has a higher radiative forcing in total compared to methane (compared to pre-industrial times), CH_4 is also a very important greenhouse gas because of the absorption band

[1] https://commons.wikimedia.org/wiki/File:Atmospheric_Transmission.png last access: 19-05-2020

within the atmospheric window. The absorption bands of CO_2 are almost entirely opaque already, which means simply speaking, an increase in atmospheric CO_2 content does less incremental damage than an increase in CH_4 (in the atmospheric window) because additional methane has to potential to block a yet "open" part of the atmospheric window. Of course this is overly simplified, and both CH_4 and CO_2 are also part of other chemical reactions that influence the greenhouse effect. But still the atmospheric window is one of the reasons CH_4 is an important greenhouse gas.

2.2.1. Radiative forcing of atmospheric methane

An increase in greenhouse gas concentration leads to a radiative perturbation of the Earth system which leads to an initial warming. This effect is refered to as Radiative Forcing (RF) in literature. A General circulation model (GCM) can be used to estimate the effect of changes in the atmospheric composition to the Earth's atmosphere system, based on a greenhouse effect model. However these models can be quite complex, and consist of different kinds of positive and negative feedback loops, almost all of them show a linear relation between changes in the surface temperature and initial RF [2].

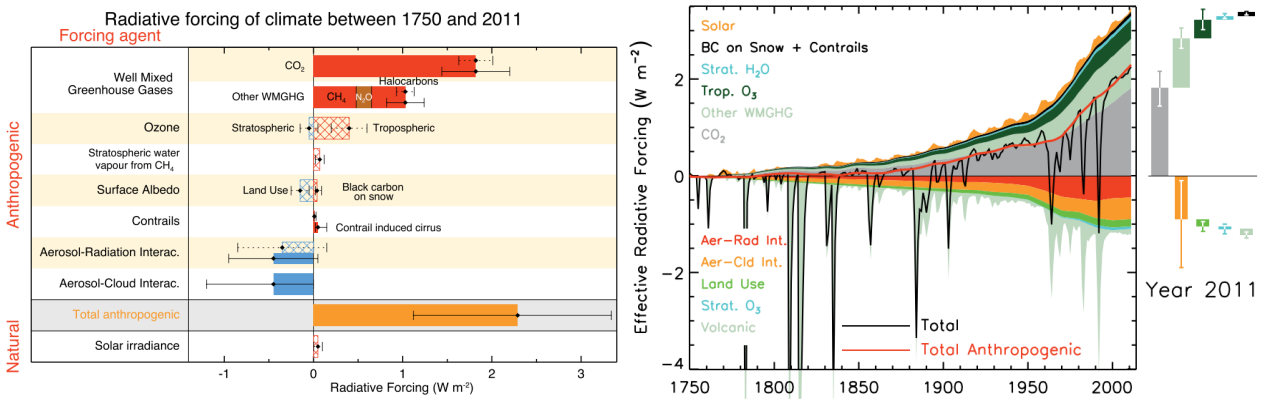
The Radiative Forcing [W m^{-2}] is defined as: The resulting flux imbalance in the radiative budget for the Earth system, caused by a change δm in the atmospheric mass of a certain greenhouse gas. [2]

By using a GCM with radiative forcing equilibrium, an addition of δm of a certain greenhouse gas leads to more absorption of the outgoing radiation, which leads to a higher temperature due to the addition of the given amount of mass of the greenhouse gas. A more standardised metric of the radiative forcing is the Global Warming Potential (GWP), this is the RF resulting from an addition of 1 kg of a certain greenhouse gas injected into the atmosphere, relative to 1 kg CO_2. [2]
In Chapter 8 of the fifth Assessment Report by the IPCC [1], the Global Warming Potential (GWP) is defined as: "The time-integrated Radiative Forcing due to a pulse emission of a given component, relative to a pulse emission of an equal mass of CO_2".

Table 2.1 shows that the effect of an emission of 1 kg CH_4 is very large compared to 1 kg of CO_2. In the medium to short term of 20 years, 1 kg of emitted CH_4 leads to 84-86 times as much radiative forcing as 1 kg of CO_2 [4].
It should be noted that the lifetime of CO_2 is simplified in Table 2.1. In reality CO_2 is mostly absorbed by the surface ocean and after 1000 years only 60 to 85% of the original emissions are absorbed, although much uncertainty remains about this as of yet ([9] TFE7 & TFE7-Fig1). Several different processes exist which remove CO_2 from the atmosphere, with timescales ranging from years to decades to centuries and even millennia. CO_2 removal from the atmosphere is a very complex process. A rough estimate about the short term CO_2 absorption via land uptake and ocean invasion is a removal of about 55% in 100 years ([10] Box 6.1 Fig 1). The main difficulty with removing CO_2 from the atmosphere is that it has to go back into rocks eventually, which is a very slow process.
In order to express the relative lifetime of atmospheric CH_4 to atmoshperic CO_2 (which is used as a sort of standard unit to compare other GHG to) the value presented in Table 2.1 can be used an a very rough assumption. It is clear that methane has a lower atmospheric residence time due to the reaction with the OH radical in the atmosphere. [10] [9]

GHG	Atmospheric lifetime* [yr]	GWP** [20 yr]	GWP** [100 yr]
CO_2	$\approx 100-1000$ ***	1	1
CH_4	12.4 ± 1.4	84-86	28-34

Table 2.1: The Global Warming Potential of CH_4 by the IPCC's Fifth Assessment Report Chapter 8 [1].
* The Perturbation lifetime is used in these metrics. [1]
** The ranges indicate whether or not climate-carbon feedback is taken into account. [1]
*** It should be noted that the lifetime of CO_2 is hugely simplified here, this is explained in the text below. [9] [10]



(a) Radiative forcing budget. From [1]

(b) Radiative forcing over time. From [1]

Figure 2.3: Change in radiative forcing between 1750 and 2011. From the IPCC's Assessment Report 5 (2014) [1]

Figure 2.3a shows the radiative forcing in 2011 compared to 1750 (pre-industrial times) as a baseline of 0. It is clearly visible that CO_2 is the major contributor to the anthropogenic forcing, CH_4 is the largest contributor of the other well-mixed greenhouse gases.
Figure 2.3b shows the development of the different forcing factors over time. It is visible that the anthropogenic GHG emissions increase heavily and that the aerosols which are emitted by anthropogenic activities do have a negative effect on the radiative forcing which is smaller.

It can be concluded from the IPCC report AR5 [1], from Table 2.1 and Figure 2.3 that anthropogenic CH_4 emission is one of the largest contributing factors to the radiative forcing budget of the Earth system, and thus one of the major accelerators of climate change. Measured since pre-industrial times, the emission based radiative forcing of CH_4 is 0.97 W m^{-2} (including effects on stratospheric water vapor and tropospheric ozone). Which is about 60% of that of CO_2 [9]. The relatively short lifetime and high GWP shown in Table 2.1 indicate that less methane emission on the short term could help lower radiative forcing, and thus global warming, in the following decades. The radiative forcing metric was introduced in its current form in 2007, which was thereafter used in the IPCC reports, as described in Ramaswamy et al. 2019. [11]

Figures 2.3a and 2.3b show the effects of CO2 and CH4 since the beginning of the industrial revolution (1750). The effect of current annual emissions shows a different relative importance.
When related to the actual annual anthropogenic emissions of both CH_4 and CO_2 while also taking into account their atmospheric lifetimes and GWP their relative effects on temperature increase on the short term are much closer as shown in Figure 2.4 . This figure shows that while the total mass of antrhopogenic methane emissions are only roughly 10% of carbon dioxide emissions, the effects on short- to medium-term warming are similar. Carbon dioxide is a larger problem on the long-term, but on the short-term annual CH_4 emissions are just as pressing.

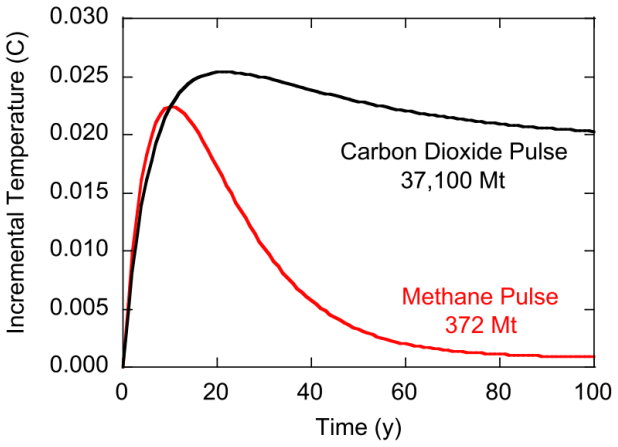


Figure 2.4: Global mean surface temperature increase following a one year pulse of carbon dioxide (black) or methane (red) in year 0 equal to the annual global anthropogenic emission of each based on the year 2017. Source: [12] and [2]

Methane emissions originate from a wide variety of sources as will be discussed in Section 2.3.3. Detecting the few point-source super-emitters early on would allow for quick action to shut down those emitters [6]. This has a large potential to reduce the total mass of methane emitted, which scales almost linear with the radiative forcing as was described before. More on this in Section 2.3.4.

[2] https://www.energypolicy.columbia.edu/research/commentary/nowhere-hide-implications-policy-industry-and-finance-satellite-bas
last access: 12-01-2021

2.3. The global methane budget

On the human timescale the Earth can be assumed to be a closed system, and thus all particles circulating trough different reservoirs of the Earth as well. Out-gassing of the upper layers of the atmosphere to space is a very small flux compared to the other fluxes taking place between different spheres of the Earth system. The total sum of all fluxes into and out of different spheres, also taking into account chemical reactions, has to add up to 0. This analogy allows for budgets to be drafted, this is the common approach to quantify emissions in climate science and allows for linking results from different disciplines of science together.

2.3.1. Sources and sinks

When looking at the methane cycle from the atmosphere perspective, sources are regions and categories which contribute a net emission of methane into the atmosphere, resulting in an influx (positive flux) into the atmosphere. Whereas sinks are regions or categories which take methane from the atmosphere, resulting in an outflux (negative flux) of atmospheric methane. If this influx out-weights the outflux of atmospheric methane, the total amount of methane in the atmosphere will increase. The atmosphere can be seen as a buffer zone which absorbs all the methane emitted by the sources and provides the methane used by the sinks. A global overview of categories of sources and sinks will be given below.

The two main methods used in methane quantification are top-down and bottom-up estimations of the CH_4 sources and sinks. In the paper Three decades of global methane sources and sinks by Kirschke et al. [13] the writers describe the kind of research they have performed into the changes in the methane sources and sinks over the 30 years between 1980 and 2010.

Both sources and sinks occur naturally and can also be induced by humanity, either regularly (mostly on well known in advance as a by-product of human activities, such as flaring) or accidentally (i.e. as a blowout at a natural gas processing facility). These different kinds of sources and sinks will be described in more detail in the following sections.

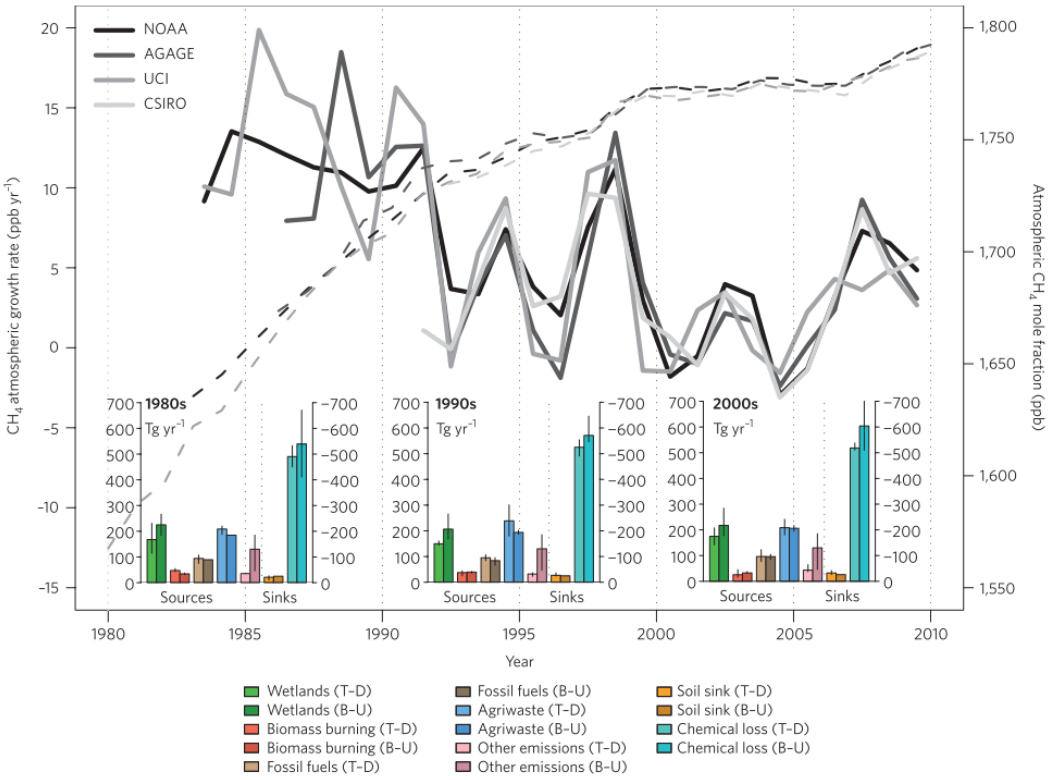


Figure 2.5: The sources, sinks, atmospheric mixing ratios and growth rate of Methane over the decades between 1980 and 2010. From [13]

An overview was presented by Kirschke et al. [13] which relates the bottom-up and top-down methods to the change in atmospheric methane mixing ratios and growth rate from 1980 to 2010. Figure 2.5 is a very powerful visualization, since it provides a total overview of the relation between the sources, sinks and imbalance between those and the effect the imbalance has on the atmosphere. Small changes in the larger sources and sinks do make up the growth of the methane concentration in the atmosphere.

The general categories of sources and sinks shown in Table 2.2 are identified by Kirschke et al.

Both Figure 2.5 and Table 2.2 show the relative magnitude of the largest sources and sinks of atmospheric methane. Both show that the sum of anthropogenic sources is larger than the sum of natural sources. Fossil fuels has a large contribution to the total of sources, roughly speaking 1/6. The largest sink by far is the chemical reaction with the radical OH in the Troposphere.

Top level Source or Sink	Lower level Source or Sink	Flux [Tg CH_4 yr^{-1}]	uncertainty range
Natural sources		**283**	[179-484]
	Natural wetlands	196	[142-284]
	Other natural sources*	87	[37-200]
Anthropogenic sources		**333**	[273-409]
	Agriculture and waste	204	[180-241]
	Biomass burning	33	[24-45]
	Fossil fuels	96	[77-123]
Sinks		**606**	[514-785]
	Soils	30	[9-47]
	Tropospheric OH	510	[483-738]
	other chemical loss	66	[13-84]
imbalance		**10**	[-4 - 19]

Table 2.2: The CH_4 sources and sinks as given by [13]. The fluxes in this table are based on the yearly averages between 2000 and 2010 (the average was taken between top-down and bottom-up, or interpolated due to missing data).
However this data is slightly outdated (2013, just like IPCC AR5) and the fluxes vary slightly from year to year, this does give a good view of the order of magnitude of the relative magnitude of the different contributing categories, which was the intention of this table. Do note this data should not be interpreted as exact values
*Other natural sources include: Fresh water (lakes and rivers), Wild animals, Wildfires, Termites, Geological (incl. oceans), Hydrates and Permafrost (excl. lakes and wetlands)

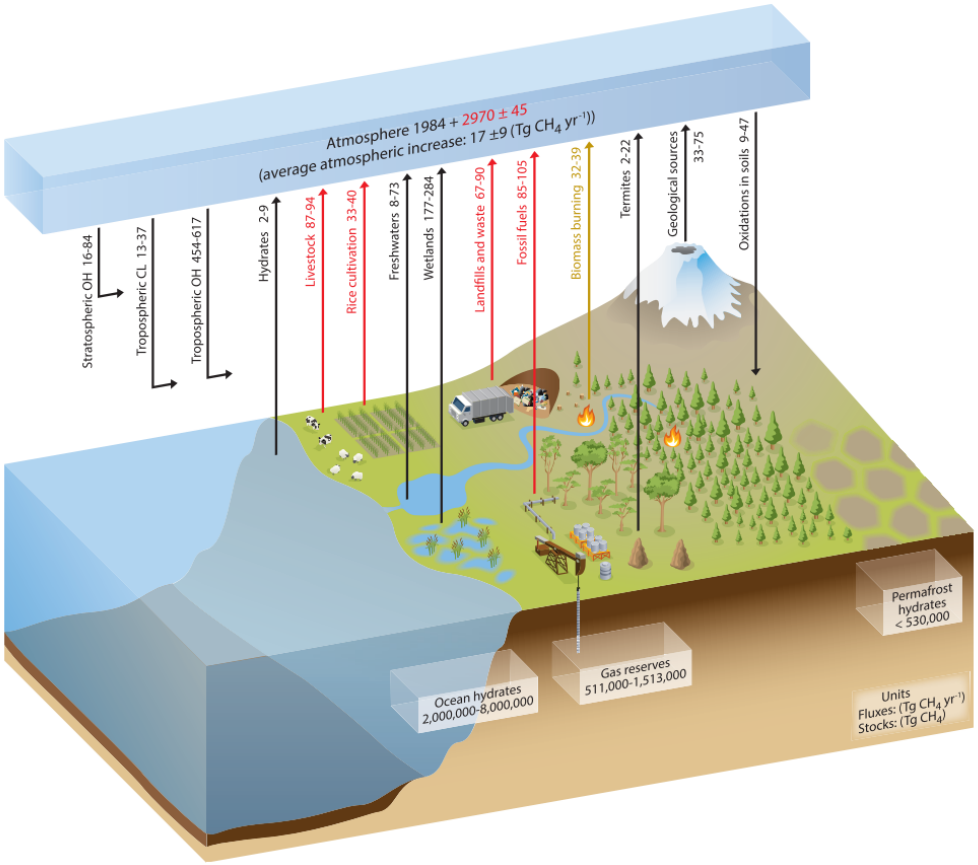


Figure 2.6: A schematic overview of the global cycle of CH_4 indicating major sources, sinks and reservoirs. From the fifth Assessment Report by the IPCC [10]

Another schematic overview of the global cycle of CH_4 is presented in the fifth Assessment Report by the IPCC. This inventory shows a comparable fossil fuel source, but a higher yearly total atmospheric influx. Both inventories show quite good agreement on the fossil fuel related sources of CH_4. The fossil fuel source will be investigated in more detail with a more recent source [14] in the following sections.
The fifth Assessment Report [10] was published in 2013, in the same year as Kirschke et al. [13], but uses a slightly different approach and is based on a slightly different time period, but the conclusions and magnitudes of the fluxes agree quite well. In late 2021 the sixth Assessment Report of the IPCC will be published, with an updated inventory of the global CH_4 sources and sinks, combining all the research performed in the past few years to form an as strong consensus as possible.

In the following subsections the different types of sources (& natural sinks) will be discussed in more detail.

2.3.2. Natural sources and sinks

In this section the naturally occurring sources and sinks will be described to get a better understanding of the methane fluxes taking place in the Earth system, and how these relate to the atmospheric methane concentration. Some of the major sources of atmospheric methane are natural. Most of the sinks can also be seen as natural, since little to no human interaction takes place to influence the sinks. The amount of OH and CH_4 in the atmosphere heavily influence the rate in which CH_4 is reacting with OH (more on this later in this section), it can be argued a human effect on this is present, but this is a passive relation.
From the natural sources wetlands are by far the largest contributor, but uncertainties in the exact relative magnitudes of the sources and sinks have been found to be present in earlier inventories (such as Kirschke et al. (2013)) in more recent years [6]. In newer inventories estimates for wetlands are better. The influence of wetlands is visible in satellite data (SCIAMACHY and GOSAT) data, where the northern part of South America and central Africa clearly stand out with their known abundance of wetlands [15]. Other natural sources such as fresh water, termites, biomass burning and geological sources also emit quite large fluxes, but all of those are dwarfed by wetlands.

The atmospheric methane sinks consist of the categories soils and loss due to chemical reactions. The soils can be influenced by human activity and land-use, but this flux is small compared to chemical reactions taking place in the atmosphere. Most reactions take place in the troposphere, since most of the particles of the atmosphere are located there (Section 2.1). Another reason the troposphere is of most importance is because gases are usually emitted at the surface [2]. About 10% of the loss occurs in the stratosphere, the other 90% in the troposphere. The major chemical reaction taking place in the troposphere is the reaction of CH_4 with OH radicals. This reaction is listed as a sink of CH_4, but this does not mean the methane simply disappears, the atoms continue to be present in the atmosphere in the form of CH_3 and H_2 and ultimately CO_2.
This is different from when for example CO_2 is used by trees for photo-synthesis, this comes closer to the actual "consumption and long-term storage", or removal of the greenhouse gas. Methane is not actually removed from the atmosphere, just converted through chemical reactions.

When grouped together the natural sources provide about 40-45% of the total source of CH_4, the other 60-55% is due to anthropogenic sources. [13]

2.3.3. Anthropogenic sources

Anthropogenic emissions are relatively largest in the rich and developed western world (North America + Europe) and China together with the fossil-fuel producing countries of central Asia and eastern Europe [13]. This is one of the reasons why the atmospheric concentration of CH_4 is higher in the northern hemisphere compared to the southern hemisphere. Agriculture and waste emissions are highest in emerging economies, fossil fuel emissions are highest in richer, more developed countries.

Agriculture and waste are the largest contributing categories, emissions related to these categories are quite stable and constant, they do not heavily fluctuate over the years, but they do within a year because of the periodic nature of agriculture (due to the harvesting season et cetera). The main sources within the category of agriculture are rice cultivation and cattle related. The combustion of fossil fuels emits a relatively small fraction of total CH_4 emissions but more of other greenhouse gasses. This is why the fossil fuel related part is relatively small in the total global yearly CH_4 emissions budget (Figure 2.2). When natural gas (of which the main component is methane) is burned before release into the atmosphere, or flared, with a high combustion efficiency only little CH_4 is emitted, but instead more CO_2.

Of all types of anthropogenic CH_4 emissions, emissions related to fossil fuel exploitation are the most interesting for mitigation. However the Agriculture and waste category is larger in the inventory of [13], emissions within this category are way more difficult and less desirable to reduce. This category is way more spatially scattered, and mostly consists of area sources instead of point-sources. This means the emissions per area are much lower, but the area over which they occur is much larger.
Emissions related to fossil fuel exploitation are the opposite, usually large emissions arise from small areas or even point-sources. Figure 2.7 shows an overview of different kinds of methane emissions related to fossil fuel exploitation, the stream of the production process, and related losses for that phase, are indicated in the figure for the three main categories of fossil fuels. In the study performed by Scarpelli et al. [14] the oil&gas related methane emissions were subdivided into those categories in a resolution of 0.1° by 0.1°, similar to the Emission Database for Global Atmospheric Research (EDGAR) bottom-up emissions inventory, but more detailed. The authors of the paper created the gridded overview, but the data was obtained from the UNFCCC reported emission inventory. The relative sizes of the categories oil, gas and coal are shows in Table 2.3 (from [14]). These fluxes can be seen in the bigger context by comparing it to the category Fossil fuels in the Anthropogenic sources in Table 2.2, which is based on [13], (note that the data does not describe the same year, meaning it is only suitable as a rough estimation).

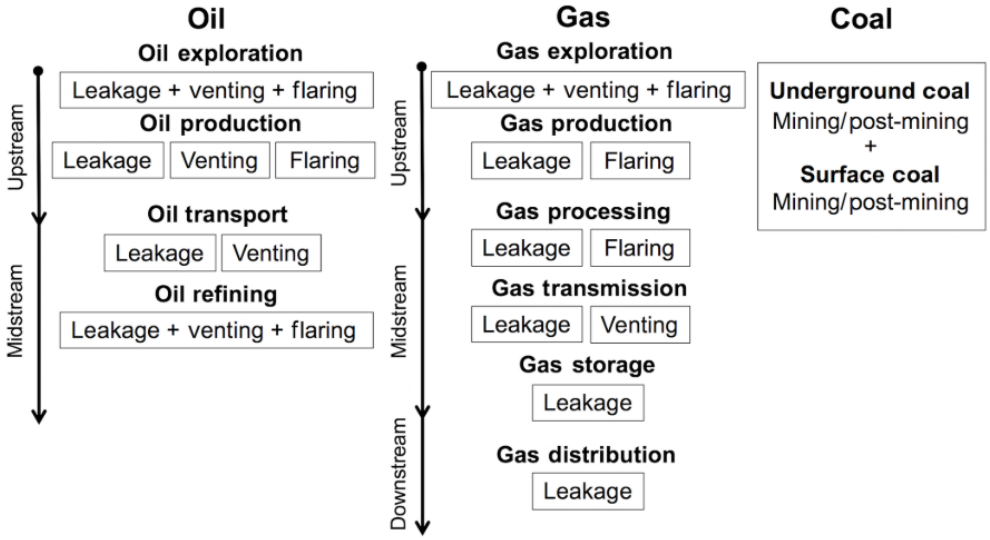


Figure 2.7: A schematisation of the different categories of methane emissions related to fossil fuel exploitation. From [14]

Leakage is self-explanatory, but can be very different for different processes in oil&gas exploitation and for different activities along the production stream. Flaring is a process where excess gas, or unwanted gas is burned off to get rid of it [16]. Venting is the process where gas is released without burning it, this has significantly stronger negative effects compared to emissions from burning gas. Fracking is the process where hydraulic pressure is used to force open cracks and fissures in the ground in order to extract oil or gas. This method has been performed increasingly since 2005 in the United States, and due to the uncontrolled and open nature of the exploitation process it is related with high additional emissions such as leakage. For example, a large increase in emissions has been observed in Permian Basin, the largest oil&gas production basin in the United States, likely due to fracking and a relatively high ratio of flaring and venting compared to other basins and because of the focus on oil. [16]

Sector	Subsector	CH_4 emission [Tg yr^{-1}]
Oil		**41.5**
	Exploration	1.4
	Production (leakage)	17.8
	Production (venting)	21.6
	Production (flaring)	0.5
	Refining	0.1
	Transport (leakage)	< 0.1
	Transport (venting)	< 0.1
Gas		**24.4**
	Exploration	< 0.1
	Production (leakage)	7.4
	Production (flaring)	< 0.1
	Processing (leakage)	2.3
	Processing (flaring)	0.1
	Transmission (leakage)	7.1
	Transmission (venting)	0.6
	Storage (leakage)	1.0
	Distribution (leakage)	5.7
Coal		**31.3**
total		**97.2**

Table 2.3: The global fossil fuel related CH_4 emissions for 2016 for the main categories of fossil fuels (no uncertainty ranges were provided in the source). From [14]

Table 2.3 shows which parts of the production processes of oil&gas production shown in Figure 2.7 result in most CH_4 emissions. For oil the largest emissions occur during production, for natural gas extraction this is more evenly spread across the production process and for coal emissions are mostly related to the mining activities, so also on the production site. When searching for point-sources related to fossil fuel production this is valueable information to know in which parts of the production stream one might expect an emission event in the future.

Jacob et al. [6] state that within the anthropogenic fossil fuel related emissions, a small fraction of the point-sources emits disproportionally large volumes of methane. These few large point-sources lead to the majority of emitted methane [6]. This was also the result of a study by Zavala-Araiza et al. [17], which will be discussed more in the next section. Because of the disproportionally large contribution of the few largest events it is worthwhile to investigate these point-source super-emitters in more detail. Adressing methane point-soures is also a top-priority in the EU methane strategy, released by the European Commission in October 2020 [18], this is discussed in more detail in Section 3.3.1.

2.3.4. Super-emitter events

Jacob et al. [6] found that the top 1% of the point source emissions listed in the Greenhouse Gas Reporting Program (GHGRP) in 2012 contributed 29% of the total point source emissions in the US of the GHGRP inventory. This supports the assumption that the few largest point-source leaks contribute a dis-proportionally large amount of the total methane emissions. Such super-emitters can fall either under the category of fossil fuels or agriculture and waste, depending on what the source is. In the 2012 inventory the super emitters were part of the categories waste, coal mining and natural gas processing.

Zavala-Araiza et al. [17] performed research specifically into the topic of the classification of super-emitters. Natural gas is in essence a cleaner alternative to conventional fossil fuels when combusted properly, but when the gas itself is released into the atmosphere this advantage is reduced or even removed. extreme CH_4 emissions events which occur accidental during processing in oil&gas facilities are usually not quantified and remain unreported to inventories such as EDGAR, and the responsible company is not held accountable, especially when the facilities are located at a remote area [19].

Recently two very large super emitters were identified early with the new TROPOMI instrument. One was a gas well blowout in Ohio in Februari / March 2018 (≈ 60 kt) Pandey et al. [19]. The other was a leak at a gas compressor station in Turkmenistan (≈ 142 kt) [20] which was detected by GHGSat. Another well known event was the Alison Canyon gas leak in the fall and winter of 2015 (≈ 97 kt). The emission amounts are quite significant on a global scale. At the time of writing another paper is in review by scientists from SRON and other authors about a blowout event in Louisiana. [21]
These events and their emissions metrics are summarized in Table 2.4.

Event	total CH_4 [kt]	duration	peak rate [t hr^{-1}]	source
Alison Canyon	≈ 97	4 months	≈ 60	[20], [22], [23]
Turkmenistan	142 ± 34	11 months*	43.3 ± 12.0	[20]
Ohio blowout	60 ± 15	20 days	120 ± 32	[19]
Louisiana blowout	43 (21 – 63)	38 days	101 (49 – 127)	[21]

Table 2.4: A non-exhaustive overview of some of the super-emitter point-source events in recent years.
* 11 months is the duration observed by GHGSat, corresponding to the emitted amount of CH_4 listed here. In TROPOMI data the source can be found from November 2017 onwards (1.5 years), which could mean it was active even before was TROPOMI launched. Prof. Dr. Aben noted a signal might even have been present in SCIA data already.

Rough indication of order of magnitude of super-emitter events compared to global annual CH_4 emissions

In order to be able to compare these emissions to the tables presented before (Table 2.2 & 2.3) describing the annual global fossil fuel CH_4 emissions, the values in the table can be converted as 1 Teragram [Tg] = 1000 Kiloton [kt].
If we compare these super-emitter events to the total global 2016 fossil fuel emissions from Table 2.3 (97.2 Tg) we can conclude, for illustrative purposes only, that such a single super emitter event (assumed average of 100 kt) is roughly responsible for about 0.1 % of the global annual fossil fuel related CH_4 emissions, to indicate the order of magnitude of such an event.

These are the kind of super-emitters that contribute a disproportionally large amount of the anthropogenic methane emissions into the atmosphere. The anthropogenic CH_4 emission can be lowered tremendously if such incidents in remote areas could be identified rapidly for mitigation purposes [6]. The second entry in the table above was detected using satellite measurements of atmospheric methane contents (by GHGSat, accidentally when inspecting nearby mud-volcanoes), the first and third event listed were discovered from the ground.

In the next section the different methods to measure atmospheric methane will be discussed.

2.4. Measuring atmospheric methane

Atmospheric concentrations of methane can either be measured using using in-situ measurements of air samples or by remote sensing techniques. Remote sensing techniques are most commonly applied, since in-situ measurements only provide insight into local concentrations.
The other option is remote sensing, this is different from direct measurements as the entire CH_4 column is measured, either from an aircraft, from a satellite or from a ground based instrument. [13]. Regional mapping campaigns such as AVIRIS-NG use an aircraft to inspect regions, like oil&gas production areas. Drones are also used for more local and small scale inspections.
For methane two major ground-based networks exist, these are the Total Carbon Column Observing Network TCCON[3] and Network for the Detection of Atmospheric Composition Change NDACC[4]. Both networks are affiliated to one another. TCCON consists of about 30 ground-based Fourier Transform Spectrometers, which take measurements in the near-infrared spectral region, located all over the world. NDACC is a network of more than 70 ground based station which performs a lot of different measurements, including atmospheric trace gasses. Those networks are used to verify the measurements of (new) satellites, and therefore also play and important role in satellite remote sensing. [15]

[3] https://tccondata.org/ last access: 24-06-2020
[4] http://www.ndaccdemo.org/about last access: 24-06-2020

2.4.1. The physical basic principles

In Section 2.2 the absorption bands of methane (1.65 µm, 2.3 µm and 8 µm) were already discussed, as well as their relation to the atmospheric window and global warming. However these absorption bands are troublesome for the thermal balance of the Earth system in terms of global warming since these cause the gasses to absorb terrestrial radiation, they can also be used in order to detect the presence of the gas using remote sensing, or more specifically, absorption spectroscopy.

A dip in the measured backscattered solar radiation by the Earth's surface at the wavelength of the absorption bands of methane can indicate the presence of enhanced quantities of methane in the Earth's atmosphere (Figure 2.9b. This is the fundamental principle which makes remote sensing of atmospheric tracegasses such as methane possible.

The 1.65 µm and 2.3 µm absorptionbands are within the SWIR domain, the absorptionband at wavelength 8 µm is within the TIR domain of the electromagnetic spectrum. Both require different measurement techniques to be detected.

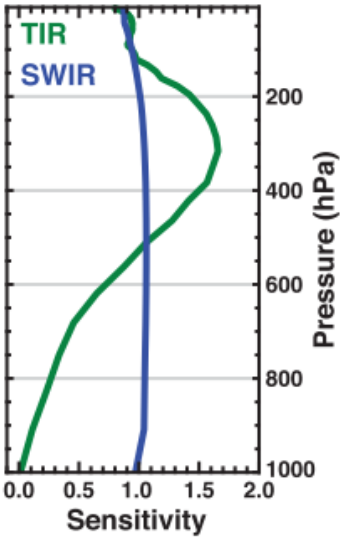
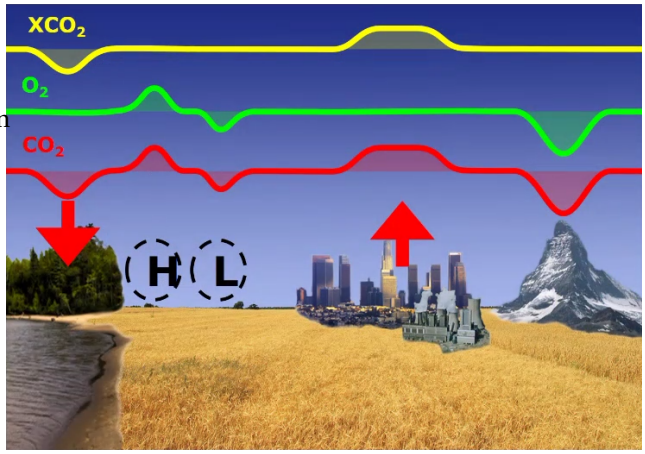


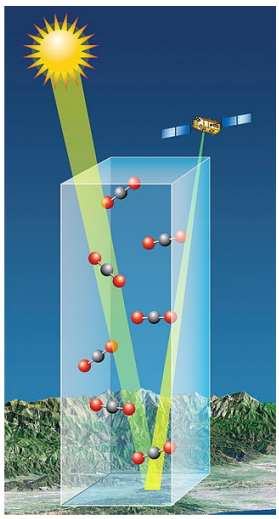
Figure 2.8: The general sensitivity of Short-wave infrared (SWIR) and Thermal infrared (TIR) instruments versus atmospheric pressure. From [6]

SWIR instruments detect solar radiation which is backscattered by the Earth and the Earth's atmosphere. TIR instruments measure the re-emitted radiation by the atmosphere. The sensitivity profiles versus atmospheric pressure (and thus altitude) is shown in Figure 2.8. The figure shows that if one wants to measure the methane content in the troposphere (Figure 2.1), then SWIR is the better option for satellite based observations since TIR cannot measure accurately extending to the surface.

The troposphere is of most interest since the majority of the mass of the atmosphere is located within the troposphere and emissions are usually located at the surface. A combined TIR measurement however is very useful to deduct a vertical profile from a SWIR measurement. For SWIR instruments it is important to pass over a given location at the same local solar time, to ensure consistency in illumination and viewing geometry conditions. This ensures measurements at different moments in time are inter-comparable. This can be achieved by launching the satellite carrying the instrument in a polar sun-synchronous low Earth orbit. [6]



(a) An illustration of the advantages of using the dry-air mole fraction given the sensitivity of the quantity to sources and sinks, but an indifference to high- and low-pressure zones (surface pressure) and surface topography. This example shows XCO_2, but the principle is similar for XCH_4. [3]



(b) The total column mixing ratio principle visualized. Source: [5]

Figure 2.9: The principles of atmospheric absorption spectroscopy and the unit in which atmospheric concentrations are expressed illustrated.

An atmospheric trace gas concentration obtained by satellite remote sensing is most often expressed as "total column dry air atmospheric mixing ratio". It takes the total air column into account, from the surface all the way up to where the pressure equalizes with outer space at multiple hundreds of kilometers above the Earth's surface (Figure 2.9b). The advantage of this expression for the methane content is that it is insensitive to differences in surface alti-

[4] https://commons.wikimedia.org/wiki/File:Artist_rendition_of_the_CO2_column_that_OCO-2_will_see.jpg last access: 03-02-2021

tude and high and low atmospheric pressure zones (Figure 2.9a). In other words: The dry-air mole fraction of CH_4 is defined as "The total number of CH_4 molecules in a vertical column above a unit surface divided by the corresponding total number of molecules of dry air in that column." [3]. The air column is retrieved from the total number of O_2 molecules. The unit of the atmospheric mixing ratio is parts per billion, or ppb.

An important detail to note is that satellite absorption spectroscopy instruments measure concentrations, not emissions. The background concentration is always present, which makes it difficult to isolate emissions. This makes CH_4 emission detection more difficult than NO_2. Furthermore concentrations resulting from emissions mix with surrounding air volumes with lower a concentration, which means emission plumes do not just linger in the atmosphere after emission, but diffuse which spatially equalizes the local enhanced concentrations over time. The mechanisms which drives plume dissipation is turbulent diffusion. [24]

2.4.2. Satellite remote sensing

The only viable method to provide global coverage in a reasonable temporal scale is satellite remote sensing, therefore this is the most commonly applied remote sensing technique. For methane, two techniques exist, which are based on detecting methane in either the SWIR or TIR absorptionband.

Multiple instruments capable of measuring the radiation spectrum at the wavelengths of the absorption bands of methane are currently in orbit. Jacob et al. [6] provided an overview of the history and near future of atmospheric methane measurement techonologies with satellite instruments in 2016. The authors summarized this in a table with all instruments and their details. From this a few categories and trends can be identified. Full details can be found in [6] Table 1.
Measurements of methane in the atmosphere began in 1996 with a TIR instrument (IMG). The first SWIR instrument was operated in 2003 (SCIAMACHY). Most earlier SWIR instruments measured in the 1.65 µm band, since this allows also for CO_2 detection. GOSAT was another important instrument in the context of atmospheric CH_4 detection. It was launched in 2009 and has been monitoring atmospheric CO_2 and CH_4 concentrations since then.

As technology improves with time, trends in the decreases of the pixel size and return time and an increase in the global coverage can be identified. Most instruments focused on global coverage, in line with the desire from the scientific community in the past 15-10 years to be able to create accurate global inventories of atmospheric methane. [8]

In 2017 the TROPOMI was launched, currently the state of the art instrument for performing global detections of methane. It measures methane in the 2.3 µm band. TROPOMI provides daily coverage of the entire Earth, and thus has a repeat time of 1 day. The pixel size is 7×5.5 km^2 in nadir, which is a much higher resolution compared to previous satellites providing full-Earth coverage. TROPOMI is therefore currently the state of the art instrument for performing global detections of methane. [25]

In 2016 GHGsat-D was launched, this instrument is different from previous instruments which focused on global coverage, resulting in a large pixel size. GHGSat focuses on a 12x12 km^2 field of view of which it takes measurements with a 50x50 m^2 resolution. This is an unprecedented level of detail from space, but comes consequentially with the downside of the small observed area and large return time.[26]

MethaneSAT[6], which is scheduled for launch in 2022, will be in between TROPOMI and GHGSat in terms of spatial resolution and coverage with a swath width of 260km and a pixel spatial resolution of 400x100 m, but lower revisit times compared to TROPOMI. This will be a valuable addition, but for the purposes of global monitoring and exact point-source detections TROPOMI and GHGsat will still be more suitable.

The CO2M mission in the context of ESA's Copernicus Program is scheduled for launch in 2026. The main focus of this mission is CO_2, but methane will be a supporting product. The absorptionbands for methane and carbondioxide overlap in the SWIR-1 band (1590-1675 nm) (Figure 2.2). With a spatial resolution of 4km^2 and a revisit time of a week above 40°latitude CO2M fits between the specifications of TROPOMI and MethaneSAT. [27]

Sentinel 5 is scheduled for launch in 2021/2022 and is intended to fill in the void left by TROPOMI after it's operational lifetime.

The following sections will provide details specifically on TROPOMI and GHGSat.

[6] https://www.methanesat.org/fit-with-other-missions/, last access: 14-01-2020

2.5. TROPOMI

2.5.1. Spacecraft and Instrument details

The TROPOspheric Monitoring Instrument (TROPOMI) [25] is the instrument on the Sentinel 5 Precursor (S5P) satellite. S5P is part of the European Space Agency (ESA) Sentinel program and focuses particularly on the atmosphere. S5P bridges the gap in continuity of observations between Envisat and Sentinel-5 (launch scheduled for 2021).
TROPOMI is a partnership between Airbus Defense and Space, Royal Netherlands Meteorological Institute (KNMI), SRON and TNO, commissioned by the NSO and ESA. SRON and KNMI are responsible for the scientific management. TROPOMI is funded by the Dutch government.

parameter	value [unit]
Satellite Name	Sentinel-5P
Payload	TROPOMI
NORAD-ID	42969
orbit type	near-polar, sun-synchronous
mean local solar time	13:30 [h]
semimajor axis	7205 [km]
mean altitude	834.0 [km]
eccentricity	≈ 0 (near circular) [-]
inclination	98.7 [°]
orbital period	101.4 [minutes]
orbital repeat cycle	227 (≈16) [orbits (days)]
Launch date	October 13, 2017
Design lifetime	7 [years]

Table 2.5: The orbital properties of the Sentinel 5P satellite. Details obtained from: [7] and https://www.n2yo.com/satellite/?s=42969

In 2017 the TROPOMI was launched, currently the state of the art instrument for performing global detections of methane. TROPOMI has daily coverage of the entire Earth, and thus a repeat time of 1 day. The pixel size is 7×5.5 km^2, which is a much higher resolution compared to previous satellites providing full-Earth coverage. This result is achieved by the push-broom mechanism of the instrument. TROPOMI performs measurements in the 2.3 µm band (SWIR), in contrast to all previous TIR instruments which used the 1.65 µm band.

S5P orbits the Earth in a near-polar frozen sun-synchronous orbit. The orbital altitude is about 834 km, the inclination 98.7° and the mean local solar time at the ascending node is approximately 13:30h. The orbital period is 101 minutes, resulting in a bit over 14 orbit per day. The repeat cycle is 227 orbits, which is about 17 days [28]. A more elaborate overview is presented in Table 2.5.

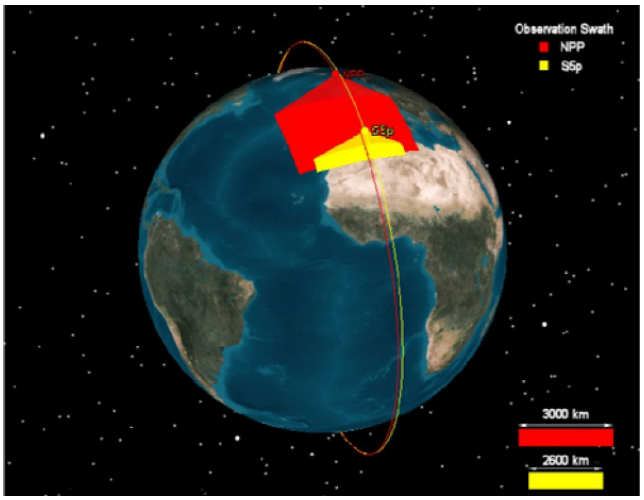


Figure 2.10: The orbital geometry of S5P (yellow) and NPP (red) [8]

The 13:30h orbit is optimal for the S5P mission for a number of reasons. The two most important ones being the similarity to measurements of prior instruments (only OMI, SCIAMACHY had a morning-orbit), which allows for comparison and validation, and the synergy with the Visible/Infrared Imager and Radiometer Suite (VIIRS) aboard the Suomi National Polar-Orbiting Partnership (NPP) satellite. The VIIRS instrument provides high spatial resolution data on clouds and aerosols, which can be used to optimize the measurements performed by TROPOMI, such clouds hinder some of the measurements. S5P's orbit is designed in such a way that it trails behind NPP with a delay of 5 minutes. Figure 2.10 shows the orbital geometry and swath widths of both satellites. The figure clearly shows the trailing behaviour of S5P.

The data of the cloud coverage is used during the on-ground processing of the data. The solar zenith angle and the viewing zenith angle are both important properties of atmospheric retrievals, therefore it is desireable to perform new measurements under similar conditions as measurements taken by previous instruments. [25] [28]

[7] https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-5p/orbit last access: 03-06-2020
[8] https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-5p/orbit last access: 03-06-2020

The TROPOMI instrument is one of the most advanced spectrometers currently in use. The instrument is a nadir viewing grating UV-VIS-NIR-SWIR spectrometer and therefore has spectral bands in multiple wavelength-regions including the shortwave infrared (SWIR, 2305 to 2385 nm) which is used to measure methane. The spectral resolution is 0.25 nm for SWIR. The instrument observes the nadir direction in a "push-broom" manner, covering 2600 km in the cross-track direction every second, while moving about 5.5 km (7 prior to early 2019) along track. This results in about 216 spectra per second. All ground pixels in the cross-track direction are measurement at the same time. This results in a spatial resolution of 7×5.5 km^2 in the nadir direction (directly 'underneath' the spacecraft), to the far sides of the swath the surface area covered by a single measurement increases by a factor of up to about 5. [25]

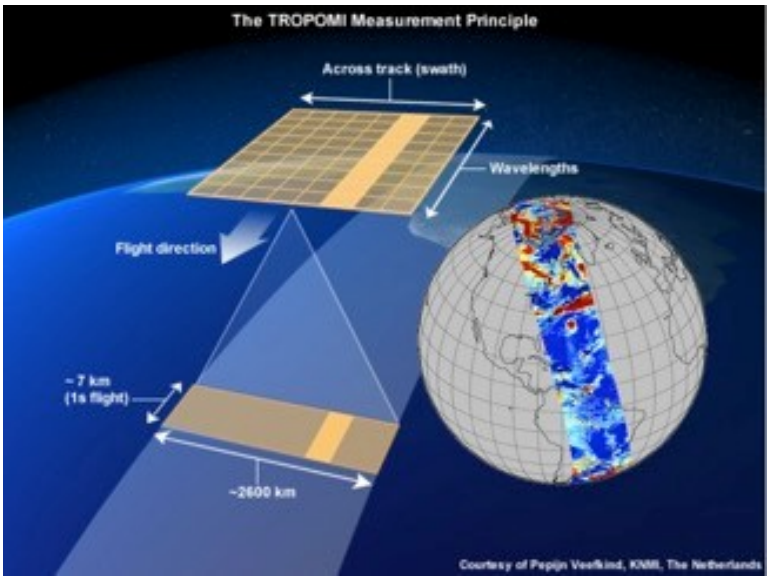


Figure 2.11: A schematisation of the geometry of the TROPOMI measurement principle. Credits shown within image.

A schematisation of the measurement geometry is shown in Figure 2.11.

The science objectives of the mission are cited from Veefkind et al. [25]:

1. To better constrain the strength, evolution, and spatiotemporal variability of the sources of trace gases and aerosols impacting air quality and climate.
2. To improve upon the attribution of climate forcing by a better understanding of the processes controlling the lifetime and distribution of methane, tropospheric ozone, and aerosols.
3. To better estimate long-term trends in the troposphere related to air quality and climate from the regional to the global scale.
4. To develop and improve air quality model processes and data assimilation in support of operational services including air quality forecasting and protocol monitoring

The first three science objectives have relations to the CH_4 measurements performed by the instrument.

For observing the troposphere, and more specifically methane in the troposphere, it is important the pixels are cloud-free, since clouds obscure the line of sight of the measurement resulting in a failed measurement. This is why the formation flying with NPP was implemented. The varying nature of the troposphere has been a driving factor in the design of the instrument and orbit, resulting in the daily global coverage and high spatial resolution. The high requirements for the TROPOMI instrument led to many of the Sentinel-5 design requirements being already met with the S5P/TROPOMI design.

The part of the instrument which is used to detect methane consists of four modules. The telescope captures the incident light, the SWIR spectrometer and relay optics grate the lightbeam. Then there are the instrument cooling unit and the cooler, which are required to limit self-emission in the band of the SWIR detector of the instrument. The SWIR spectrometer is thermally isolated from the other modules. The SWIR spectrometer consists of special filters, a slit, mirrors and a immersed grating part. The immersed grating part is specifically developed for the TROPOMI instrument. [25] [28]
The SWIR spectrometer instrument is expected to degrade over time in the space environment, this can be monitored publicly at the TROPOMI Monitoring Portal[9]. A Quartz Tungsten Halogen lamp is incorporated in the design in order to be able to calibrate the instrument on-board during the mission. This is done at the start of the mission. Illumination from the Sun, Sun and Earth spectra and laser diodes will also be used in order to guarantee the quality of the measurements. The design lifetime of the mission is 7 years, the instrument is designed in order to function properly during this timeframe. [25]
How these measurements have been used in prior studies, with the TROPOMI instrument or both the TROPOMI and GHGSat instruments together, are further processed and used in the TROPOMI-GHGSat project will be elaborated in Sections 2.5.4 and 2.6.2.

[9] http://mps.tropomi.eu/monitoring last access: 08-09-2020

2.5.2. Methane retrievals

As in common in a ESA satellite mission, the data is made available as a Level 2 dataproduct free to download. The Level 2 Methane dataproduct is developed by SRON.
The Sentinel 5 precursor / TROPOMI KNMI and SRON level 2 Input Output Data Definition [29] provides a detailed overview of the inputs and outputs of the Level 2 data (not methane specific).
Section 5.7 of the Algorithm Theoretical Baseline Document for Sentinel-5 Precursor Methane Retrieval [30] provides a coverage of the algorithm, inputs and outputs in full detail including flowcharts (Figure 7 and 8 of [30]) which illustrate the data-flow from measurement to level 2 data.

As described in Sections 2.4.1 and 2.5.1 the measurement principle used by the TROPOMI instrument is to measure backscattered spectra of sunlight from the Earth's surface and atmosphere. This is done in the SWIR spectral range, where the absorption features of CH_4 molecules will leave a measureable sign [31]. In Figure 2.8 it is shown that in the absence of clouds and high aerosol concentrations, SWIR measurements can penetrate towards to surface of the Earth through the atmosphere, where the majority of sources are located.

To turn this radiance spectrum into a column atmospheric concentration of methane (XCH_4) a retrieval algorithm is used. Clouds, which are optically thick scatterers are relatively easy to filter out, whereas optically thin scattering also occurs in the atmosphere but is much harder to detect. This optically thin scattering makes the measured XCH_4 value deviate from the true XCH_4 value, and thus should be compensated for [32].
One option for retrieval is the proxy method. This method measures CO_2 at the same time as CH_4 and assumes the scattering effects are similar, which enables the XCH_4 to be decuded from the ratio of CH_4 / CO_2. However this method requires simultanious measurement of the CO_2 column and therefore is not suitable for application in the TROPOMI mission. [31]
The alternative is the so-called physics-based method. This method consists of simultaneously measuring the CH_4 concentration and the physical scattering properties of the atmosphere, such as aerosols and cirrus clouds. The scattering properties of the atmosphere have a large influence on the perceived XCH_4. The perceived XCH_4 can differ quite significantly from the actual XCH_4 due to the influence of the physical properties of the atmosphere. This method makes use of the O_2 A-band in the NIR and the absorption band of CH_4, this doesn't require any prior information on the CO_2 concentrations. It should be noted however that the physics based model is more complicated and is limited by measuring capabilities of aerosols. This method can also be affected by forward model errors in describing the aerosol properties. The proxy method cannot be applied due to the wavelength bands in which TROPOMI measures, therefore the physics based method is used for TROPOMI XCH_4 retrievals. [31] [30]

The operational CH_4 retrieval algorithm was based on RemoTeC [33]. This physics based algorithm was developed for GOSAT and OCO observations, and has been fine-tuned for the use on TROPOMI during the development phase of the mission. Put simply, the algorithm uses a model that computes a concentration given an atmospheric state. The model should be able to make a good estimate of the measurement based on the atmospheric state. This estimated measurement is compared to the actual measurement which is taken. The first estimate is made based on a priori knowledge of the CH_4 and CO profiles, temperature, humidity, scattering patterns, surface albedo and pressure amongst others. (A full overview of the parameters taken into account in the model is presented by Hu et al. [31] in Table 1). This first estimate is compared with the model to make the best possible fit which gives the XCH_4 in 12 vertical layers, and then this procedure is iterated to find the best estimate. [30] [31]
Methane retrieval algorithms can be sensitive to atmospheric parameters such as surface albedo and aerosol optical thickness, as is discussed in the paper by Ayasse et al. [34], however this is mostly compensated for by the retrieval algorithms.
Given the way the atmospheric models are structured, the output methane product is given as a methane concentration over 12 sub-layers. In the actual measurement this has little meaning however, therefore this is converted to the column-averaged dry air mixing ratio XCH_4 [31]. This is done by using a column averaging kernel A_{col}, this gives an indication of the methane concentrations in the different sub-layers of the atmosphere. This process also yields a standard deviation of the retrieval noise σ_{XCH_4}, which is a measure of the precision of the measurement. [31]
XCH_4, A_{col} and σ_{XCH_4} are part of the level 2 dataset.

An overview of the parameters from the TROPOMI Level 2 CH_4 dataproduct which are being used are listed in Table A.1 in the Appendix.

Several factors can influence the quality of a methane mixing ratio observation. An overall bias was found in the data when it was compared with GOSAT [15], and with the TCCON and NDACC ground-based networks. This was then corrected for, resulting in an average bias of only 0.3%.
The factors and conditions which influence the quality of the measurements are summarized in a quality score, the qa_value parameter, for easy filtering without the requirement for full knowledge about the details behind it.

qa_value	Condition	Remark
1.0	Convergence, clear-sky	Highest quality
0.8	Failed deconvoluted irradiance spectrum	Not pixel specific but row specific
0.4	Not confidentially clear-sky SZA < 70°, VZA < 60° Surface albedo > 0.02 Fraction of good spectral pixels > 70% AOT < 0.3 CH_4 noise related error < 10 χ^2 < 100 SNR SWIR < 50	
0.0	No convergence	

Table 2.6: The conditions of the qa_values of the XCH_4 Level 2 data product with their conditions. [35]

TROPOMI CH_4 data is only available for Viewing zenith angle (VZA) below 60° and over land. Measurements too far away from the sub satellite point yields problems with the illumination angles. Sea surfaces also pose problems with reflections, those are filtered out during pre-processing. Oceans are often just too dark, but measurements from sun-glint could be included in future dataproducts. [35]. The different levels of qa_values are shown in Table 2.6.

The Sentinel 5 precursor / TROPOMI KNMI and SRON Level 2 Input Output Data Definition [29] provides a detailed overview of the inputs and outputs of the Level 2 data (not methane specific).
Section 5.7 of the Algorithm Theoretical Baseline Document [30] for Sentinel-5 Precursor Methane Retrieval [30] provides a coverage of the algorithm, inputs and outputs in full detail including flowcharts (Figure 7 and 8 of [30]) which illustrate the data-flow from measurement to Level 2 data.

The TROPOMI Level 2 CH_4 product is processed offline, which means that the measurements are heavily processed before release. The data is generally made available in 48-72 hours. The reason for this delay is the demanding full-physics retrieval model and the need for accurate cloud data from VIIRS. Other TROPOMI dataproducts, such as Carbon monoxide (CO) are available in ±3 hours.

2.5.3. Other retrieved parameters next to methane

As discussed in the sections above, the TROPOMI instrument retrieves a range of other atmospheric parameters next to CH_4. Several other dataproducts exist, such as Nitrogen dioxide NO_2, sulfur dioxideSO_2, ozone O_3 and carbon mono-oxide CO. Within the CH_4 dataproduct itself other relevant parameters are included, such as the aerosol optical thickness, cloud fraction, surface pressure and standard deviation of the CH_4 retrieval noise. Some of those are a priori parameters, other are retrieved from the measured spectra as described in the previous section.

2.5.4. Prior studies using the TROPOMI instrument

To indicate the capabilities of the TROPOMI instrument and in particular the scientific value of the Level 2 CH_4 dataproduct a few important prior studies using TROPOMI CH_4 data will be briefly discussed.

In March 2018 the first paper about atmospheric methane concentration inferred with TROPOMI was published [15]. This paper shows comparisons with the Greenhouse Gases Observing Satellite (GOSAT) (and thus indirectly with TCCON) and shows that TROPOMI successfully captures large area enhancement expected from wetlands.
In 2019 a paper was published about a gas well blowout in Ohio on February 15, 2018 [19]. This paper demonstrates TROPOMI can be used to identify individual large point-source emitters and that such events are even visible in data from a single overpass. The paper also demonstrates that TROPOMI CH_4 data can be used to detect and monitor a blowout event until repair and to make an estimate about the total amount of emissions. This clearly highlights the value of daily global continuous monitoring as opposed to targeted instruments.
In April 2020 a study on methane emissions in the Permian Basin, which is located in Texas, United States showed that averaged data of a year can be used to quantify emissions of an entire oil&gas production basin and how the standard Level 2 data can improved using further custom processing [16] [36].
A paper which is at the time of writing still in review describes how TROPOMI was used to quantify methane emissions from a natural gas well blowout in Lousiana, United States. The paper describes the blowout event where part

of the event a flare was lit in order to prevent further methane emissions, TROPOMI data was combined with VIIRS data which was used to infer when the flare was lit and the volume of methane being flared. This paper demonstrated that emission estimates could be made for the full blowout duration based on single overpass data, and that the officially reported amount of methane leaked was incorrect. [21]

These studies highlight that TROPOMI not only can be used to detect and quantify such events over a longer time span using averaged data, but that single orbit observations could potentially also be used in order to detect such events when they happen already at the first overpass. It is also demonstrated that it is feasible to quantify large point source emissions using single overpass TROPOMI CH_4 data.
Studies in which the TROPOMI and GHGSat instruments were used together are briefly discussed in Section 2.6.2.

2.6. The TROPOMI-GHGSat project

2.6.1. The GHGSat instrument

GHGSat Inc.[10] is a private company based in Montreal, Canada which is planning to operate a constellation of microsatellites and which aims to quantify GHG emissions (including CH_4) from individual point sources. Their instruments are optimized towards a very high resolution for a small coverage area, in order to zoom in on point-source emissions and attribute those to exact facilities. GHGSat-D [26] (demonstration, nicknamed Claire) has been operational since 2016 and has been involved in multiple successful observations [20], [19]. GHGSat-C1 (nicknamed Iris) was launched on September 2, 2020. At the time of writing GHGSat-C2 (nicknamed Hugo) had just been launched on Janauary 24, 2021, performing the first successful measurement[11] on January 30, 2021. Another successful observation (Figure 9.3) was made by GHGSat-C2 in early February, based on a location originally recommended by SRON based on a detection in November (Figure 9.2) by an early version of the model developed in this study.
GHGSat aims to expand the current fleet into a constellation of about 10 satellites by the end of 2022 for better coverage and lower revisit times.

The instrument on GHGSat-D is a wide-angle fixed-cavity Fabry-Perot (F-P) imaging spectrometer which also operates in the SWIR. The GHGSat instruments are, in contradiction to TROPOMI designed to measure methane column plumes (XCH_4 concentrations) over pre-selected 12×12 km^2 domains at a spatial resolution of less than 50×50 m^2 at a measurement precision of 1-5%. [26]
This allows for fine scale investigation of facility-level point-source emissions. Compared to other currently operational or planned CH_4 observation missions, this very detailed spatial resolution is un-precedented [6].
The GHGSat-D instrument has been demonstrated to be applicable to longer term average emission quantification from individual coal mine vents [37], though this is a different application than how it is used within the TROPOMI-GHGSat project.

2.6.2. Prior studies using both the TROPOMI and GHGSat instruments

The TROPOMI and GHGSat instruments complement each others strengths and are therefore very well suited to collaborate (more on this in the next section). This was first illustrated in the paper by Varon et al. [20].
Varon et al. [20] describe how the TROPOMI and GHGSat instruments worked together to detect and quantify a persistent CH_4 emission piont-source from an oil&gas production site in Turkmenistan. The location was discovered by accident by GHGSat and was then confirmed by SRON based on TROPOMI data. Using the archive of daily global data which is available for TROPOMI, the authors were able analyse the location earlier in time, to confirm that the source had been active for months already, ranging as long as TROPOMI had been operational, and possibly even longer.
This study underlines the strength of the synergy between both instruments and it highlights the value of the daily global measurements archive of TROPOMI data, together with the capabiliy to detect point-source emissions in single overpass TROPOMI data.

[10] https://www.ghgsat.com/ (last access: 13-01-2021)
[11] https://www.ghgsat.com/hugo-delivers-first-methane-plume-image/ last access: 03-02-2021

2.6.3. SRON & TROPOMI in the TROPOMI-GHGSat project

Successful applications of the synergy between TROPOMI and GHGSat is described in the paper [20], several news articles [12] [13] [14] [15] [16] and during a conference by Maasakkers [38] & Zehner [39].

The TROPOMI and GHGSat instruments are very well suited for a collaboration given their different specifications. Figure 2.12 shows an artist impression (physically inaccurate, but clear and illustrative) of the synergy between TROPOMI and GHGSat. TROPOMI scans the Earth for identification of potential points of interest, GHGSat investigates those locations in high resolution.

TROPOMI has daily global coverage with a pixel size of up to 7×5.5 km^2 which is optimized for daily global monitoring (Section 2.5.1), but it is difficult to detect facility-level point sources with this resolution. The spatial resolution of TROPOMI is high enough to distinguish e.g. the city center of Rotterdam from the harbour of Rotterdam, but not to further detail [25] [6]. It has been demonstrated that it is possible to use TROPOMI data to detect and locate persistent sources up to a kilometer accuracy, but this relies on longer term averaging, and is therefore not possible with single-overpass data.

The GHGSat instrument on the other hand is optimized towards high-resolution inspection of a small field of view. GHGSat focuses on a 12x12 km^2 field of view of which it takes measurements with a 50x50 m^2 resolution. This is an unprecedented level of detail from space, but comes consequentially with the downside of the small observed area and large return time [26].

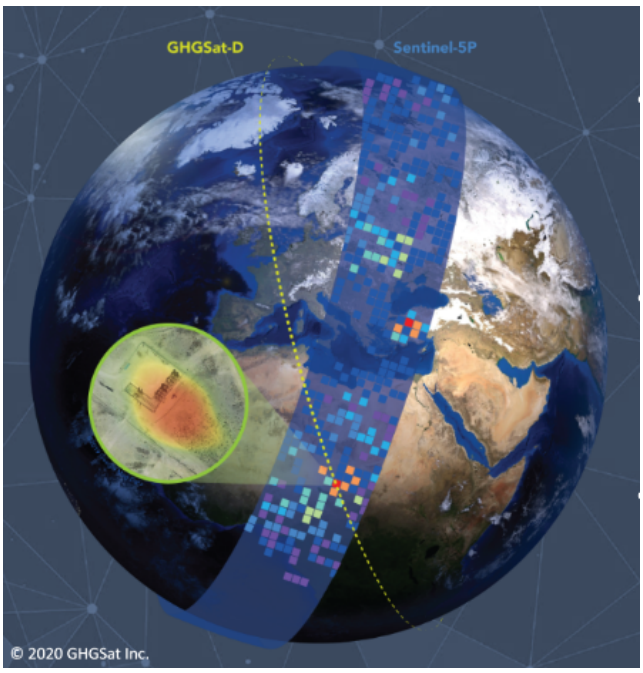


Figure 2.12: An artistic impression of the TROPOMI-GHGSat project collaboration. Credits shown in image.
Do note this image is an artistic impression, to show in a schematic way how the two satellites cooperate. It should be noted that in reality both satellites are in a more similar sun-synchronized near-polar orbit. The displayed TROPOMI pixel size is not accurate.

This is where the synergy with the SRON Level 4 CH_4 Science Team working with TROPOMI-data comes in. Using TROPOMI, the locations of interest can be identified based on the daily global medium-sized pixel coverage, which GHGSat can then observe in more detail in order to identify the exact facility responsible for the emissions.

ESA (and thus indirectly SRON) and GHGSat have signed a Memorandum of Intent[17] in 2019 in order to further strengthen the collaboration.

The unprecedented performance of TROPOMI in daily global coverage together with the high-resolution capabilities of GHGSat for a selected field of view allow for a workflow where point-sources which emit large amounts of methane which is detectable in TROPOMI data can be linked to an exact source with the GHGSat instrument. This exact source attribution allows for holding the polluter responsible and for corrective measures and mitigation in order to seal off the source of emission and thereby lowering anthropogenic CH_4 emissions (Section 2.3.3).
The sources which can be detected using this approach are the type of sources called super-emitters in the research by Zavala-Araiza et al. [17] and those are also identified in the EU Methane Strategy (Section 3.3.1) by the European Commission [18] as top-priority in reducing methane emissions.

[12] https://www.economist.com/science-and-technology/2020/02/01/using-satellites-to-spot-industrys-methane-leaks last access: 12-03-2021
[13] https://www.nytimes.com/2020/11/11/climate/methane-leaks-satellite-space.html last access: 12-03-2021
[14] http://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-5P/Detecting_methane_emissions_during_COVID-19 last access: 12-06-2020
[15] https://nos.nl/artikel/2311530-groot-methaan-lek-ontdekt-vanuit-de-ruimte-en-aangepakt.html last access: 12-03-2021
[16] https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-5P/New_Space_satellite_pinpoints_industrial_methane_emissions last access: 29-07-2020
[17] https://www.esa.int/ESA_Multimedia/Images/2019/09/Working_together_to_monitor_greenhouse_gases last access: 31-01-2021

3
Research Definition

3.1. Opportunities for point-source detection

Since point sources contribute a disproportionally large fraction to global methane emissions [17] it is desirable to know more about the spatial and temporal distribution of CH_4 point sources. In Section 2.3.4 several super-emitting events were discussed. With a fully automated global monitoring model, such events might potentially be detected automatically in TROPOMI data in less than an hour after the data becomes available (which is 48-72 hours after measurement). Therefore such an automated global monitoring model would be very valueable.

There are three main enabling developments which lead to this project:

- **The large archive of TROPOMI data**. TROPOMI was launched in October 2017 and has been measuring the atmosphere eversince. This means over three years of observations are available at this point.
- **The known locations where persistent emissions occur**. Given that the TROPOMI-GHGSat project has been on-going since 2019, and because hot-spots were identified in other contexts, these locations can be used to extract numerous scenes with plume signals from the archive of TROPOMI data.
- **Recent developments in the field of Machine Learning** provide useful tools which can be utilized.

3.2. Opportunities in the TROPOMI-GHGSat project

The SRON side of the TROPOMI-GHGSat project is mostly focused on detecting potential CH_4 emission hotspots, which can then be communicated to GHGSat for local high-resolution observations.

Before the start of this project, mostly longer-term averages and 'jumps' of a few dozen ppb from one day to another were used to identify potential hotspots in the recent TROPOMI CH_4 data. The longer-term averages work really well to detect persistent sources, but the downside is that such an average can only be taken after data from months of measurements is available. Also single large events tend to be suppressed by a larger number of lower concentrations at that location, if the source is not persistent or only is persistent for e.g. a few days.
The other detection method which was in place before the start of this project was aimed at the detection of large instantaneous increases in atmospheric CH_4 mixing ratio relative to the previous day with valid measurements. This method worked well to identify very large instantaneous emission events, but has limited success in detecting smaller emissions and does not take into account the morphology of a plume, only high values. It also does not differentiate between plumes and artefacts.

An approach which autonomously scans TROPOMI CH_4 single-overpass data for plume-features as soon as the data becomes available can contribute to the TROPOMI-GHGSat project. If it is possible to classify scenes in a standardized, autonomous way, with a sufficient level of confidence this can decrease the workload in performing detections, speed up the process and potentially lead to detection of more sources.

The TROPOMI-GHGSat project has been ongoing since the collaboration on the study by Varon et al. [20] in 2019. At the start of this research study, multiple locations with consistent CH_4 emissions had been detected already. Together with the vast archive of TROPOMI CH_4 data this enables the creation of a large dataset of plumes, without the need to manually scan three years of TROPOMI CH_4 data covering the entire Earth, which would definitely be an infeasible strategy to locate combinations of location and date which can be exploited for a dataset of plumes as training data for a machine learning approach.

The aim of this study is to meet these opportunities within the context of the TROPOMI-GHGSat project. In the Section 3.4 the research questions which form the basis of this study are discussed.

3.3. Relevance of this research project

Sections 2.2 and 2.3 and especially Subsections 2.3.3 and 2.3.4 illustrate the importance of research into methane point-sources. Atmospheric methane is the second largest driver of global warming [4] and a large portion is caused by anthropogenic activities, mostly fossil fuel exploitation or landfill related. Point-source emitters contribute a disproportionally large fraction of the global anthropogenic CH_4 emissions, the distribution is known to be heavy tailed [17]. New satellite technologies for remote sensing of the atmospheric methane concentration provide important data to detect and quantify point-sources, allow for rapid mitigation and hold polluters accountable (Sections 2.4.1, 2.4.2).
The TROPOMI-GHGSat project is especially well-suited for this purpose given the synergy between the characteristics of both instruments (Section 2.6.3). The TROPOMI-GHGSat project can therefore help to lower global anthropogenic CH_4 emissions.

The expansion plans of GHGSat in the coming few years increases their observation capacity, which calls for more and faster information on possible locations of interest provided from the TROPOMI data. In the next few years other satellites will be launched as well which could be suitable for a similar synergy (Section 2.4.2). The current workflow functions very well in the current set-up, but due to the required manual effort in performing detections with the current project structure this approach is unfeasible to meet the growth of the collaboration expected over the next few years. Parts of the detection chain have to be automated, and Machine Learning is a very good tool to do so. There are also opportunities to lower the threshold of which emissions are detected..
As stated in Section 3.2, the biggest opportunity for improvement is detection of plumes in single-orbit data. This approach contains some challenging aspects (discussed in Section 4.2.1), but could, when successfully applied, also add a lot of value to the project by speeding up detections in a more autonomous way.

3.3.1. The EU Methane Strategy as part of the European Green Deal

The European Commission released [1] the EU Methane Strategy [18] in the context of the European Green Deal on October 14th, 2020 (about halfway this project, as this project ranges from May 2020, to March 2021).
In the EU Methane Strategy [18] the European Commission stresses the significance of methane emissions to global warming and presents a strategy on how to reduce methane emissions. As a technical leader on the world stage with the Copernicus program the EU plans to take a leading role in remote sensing focused addressing of CH_4 emissions. In the EU methane strategy it is recognized that not just emissions within the borders of the EU, but also emissions related to imported products like oil & gas need to be adressed.
The EU methane strategy is part of the European Green Deal [2].

In the Roadmap [3], leading up to the publication of the EU Methane Strategy [18], published by the European Commission on July 8, 2020 the plans to decrease anthropogenic methane emissions are broadly outlined:

"At global level, at least half of the reduction in energy-related methane emissions is possible at no net cost to industry. Methane can leak from coal, oil and gas installations, or be vented into the atmosphere. On average, 5% of the sources contribute to 50% of emissions (“super-emitters”). A key challenge is also to improve actual measurement, reporting and verification at private entity level. Finally, the EU imports most of the gas it consumes and the majority of methane emissions associated with this gas are emitted before reaching the EU’s borders"
"In the energy sector, leak detection and repair programs, as well as finding and addressing ‘superemitters’ can be a very effective action. The EU imports most of the gas it consumes and the majority of methane emissions associated with this gas are emitted before reaching the EU’s borders, making it important for the EU to tackle methane emissions throughout the energy supply chain"

These passages, published by the European Commission on July 8, 2020, clearly indicate the relevance of the global search for point-sources by the TROPOMI-GHGSat project to which this Thesis research project aims to contribute.

[1] https://ec.europa.eu/commission/presscorner/detail/en/IP_20_1833 last access: 12-01-2021
[2] https://ec.europa.eu/energy/topics/oil-gas-and-coal/methane-gas-emissions_en last access: 27-07-2020
[3] https://ec.europa.eu/info/law/better-regulation/ last access: 27-07-2020

3.4. Research Questions

The main research question of this thesis is:

> **"What is the most suitable machine learning algorithm to automatically detect plumes in TROPOMI atmospheric methane data, and how can this model be optimized, implemented and utilized?"**

This main research question will be answered by first answering the following lower-level research questions, of which the answers together will provide the answer to the main research question.

- What is the most suitable type of machine learning model based on the kind of input data and the type of the problem?
  - Which potentially suitable machine learning models do exist?
  - Which kinds of machine learning models are applied in similar scientific problems?
  - What is a sufficient amount of training data? Can the input data be pre-processed for better results?
- What is the most suitable kind of input data (including training, validation and test data) to train the model on?
  - What is the best way to pre-process the TROPOMI CH_4 data?
- How can the selected model be optimized?
  - Which supporting data can be used to further optimize the model?
  - How does the model perform, evaluated using the manual workflow and known results, in terms of the amount of false positives and false negatives?
  - How can the model be improved after a proof-of-concept version is created.
- How can the model be incorporated into the TROPOMI-GHGSat project in a sustainable, future-proof way?
  - How can the model be incorporated into the TROPOMI-GHGSat project in an automated way, to reduce manual work and visual inspections.
  - How can the model be further improved on future newly detected plumes to enlarge the trainingset and increase the performance of the model (if possible in an automated way)?

4

Machine Learning methodology

Even though the field is almost as old as the computer itself, Artificial Intelligence has seen an increase in recent years in the application in scientific works and in attention of the general public. Artificial Intelligence (AI) is the name of the entire field of knowledge. Parts of it are closely related to the field of Computer Vision (CV), but not necessarily all. Close ties also exist with computer science, statistics and mathematics for the entire field of AI.

AI is a wide term and entails a wide variety of different algorithms aimed at different applications, such as natural language processing , autonomous robotics and knowledge representation. The hierarchy of the different subfields within AI is shown in Figure 4.1.
ML is one of the most widely applied subfields of AI and focuses on scenarios where a sufficient amount of data is available in order to develop a model which learns the statistical properties of different classes in the data in order to be able to make predictions about future, new data.
The Neural Network (NN) is one of the most frequently applied subfields of ML due to its capability to deal with highly non-linear data. Deep Learning (DL) is a more modern, and complex version of NN. Deep Learning is in essence not different from regular Neural Networks, it is a modern version of designing, training and applying Neural Networks, but it is very similar to "regular" or "shallow" Neural Networks. In most cases the same software libraries are used to design NNs and DL models. The clearest demarcation, stating the difference between NN and DL models found, is from Ma et al. [40] stated as:
"Layers between the input and output are often referred to as “hidden” layers. A neural network containing multiple hidden layers is typically considered as a “deep” neural network—hence, the term “deep learning”".

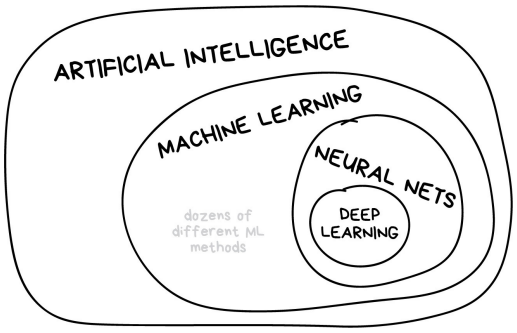


Figure 4.1: A basic overview of the field of Artificial Intelligence (AI) and the hierarchical structure of the subfields Machine Learning (ML), Neural Networks (NN) and Deep Learning (DL).[1]

The main reason AI (and it’s subfields) is on the rise is the advance of powerful and relatively inexpensive graphical processing units (GPU) [41]. Regular computers have CPUs, a central processing unit, which are optimized for serial tasks. GPUs on the other hand are optimized for parallel tasks, exactly what AI and ML algorithms needs for learning.

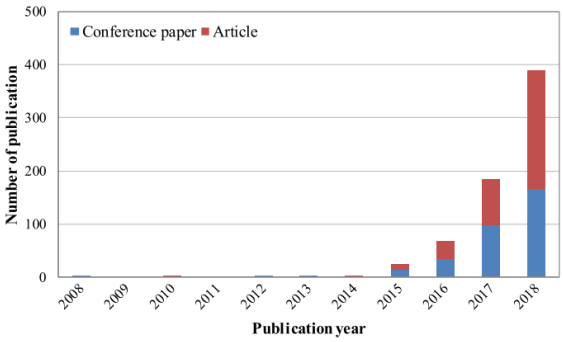


Figure 4.2: Indication for the rise of usage of Deep Learning techniques in scientific publications in the domain of Remote Sensing. Figure from: Ma et al. [40].

Another factor influencing the rise in ML applications is the availability of data. In recent times large datasets are more and more available. On one hand this is due to cheaper and larger storage solutions, as well as cloud-based storage and processing systems. Within the field of Remote Sensing, this trend is also clear. The TROPOMI instrument for example produces 11 million valid observations per month. The amount of data gathered by modern satellite instruments has increased a lot over recent years, due to increased measurement resolution, higher downlink bandwidth capabilities, larger on-board storage capacity, and other innovations.
Another factor, which is both a cause and an effect of the rise of AI and ML in a positive feedback loop, is the development of tools and software libraries which makes these techniques more accessible for a larger group of scientists and researchers [41].
Two main survey papers about Deep Learning in remote sensing provide a firm background on this sub-field of study. Ball et al. [41] (2017) and Ma et al. [40] (2019), the former being more elaborate, but also a bit more outdated. Ma et al. [40] have provided a clear recent historic context on the shift from Machine Learning approaches to CNN and DL in recent years in their survey paper.
The rise of DL in other fields, and successes obtained with these approaches in the years following 2014 has led to

[1] https://vas3k.com/blog/machine_learning/ last access: 15-07-2020

a shift to NN and DL within the field of Remote Sensing (see Figure 4.2). Numerous successes were obtained with image analysis, object detection and land cover classification problems with DL in recent years. The most commonly applied NN / DL techinique is the Convolutional Neural Network (CNN).

4.1. Convolutional Neural Networks, a conceptual description

The Convolutional Neural Network (CNN) [42] is a kind of Neural Network. A CNN can be applied for various different purposes which is why CNN are the most frequently applied type of DL model within the field of remote sensing [41] [40]. CNNs allow a computer to extract meaning from an image using feature maps in an automated way.
The CNN was designed for multiple channels of data and it was based on the human visual neocortex, which is the part of the brain which processes visual input from the eyes [41]. The history of the development of CNN and the relation to neuroscience is described in detail by Lindsay [43]. The way CNNs are similar to the brain on processing images makes this kind of artificial neural network suitable for multi-band remote sensing data processing, which is in format similar to visual input to the brain through the eyes (real-time rgb images) to the neocortex of the brain.

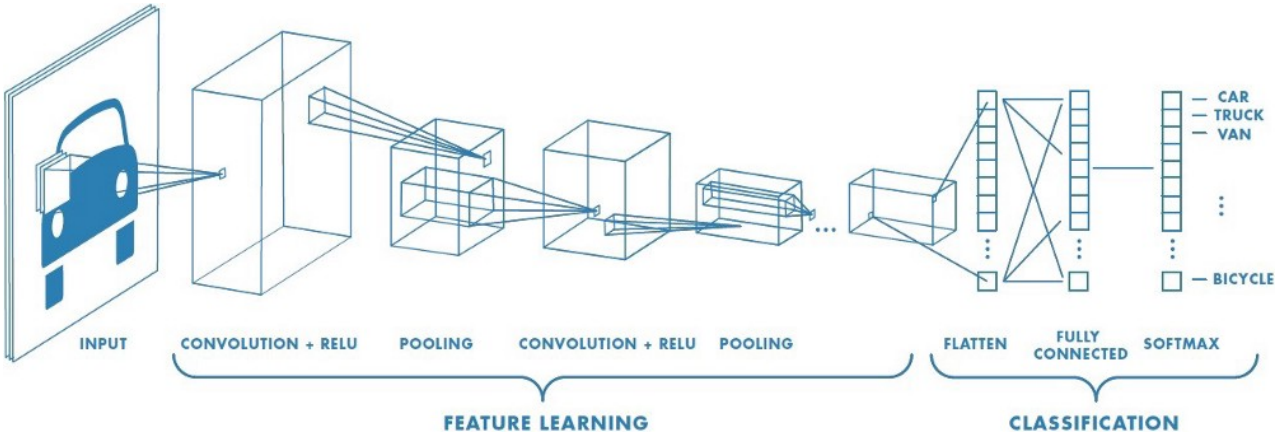


Figure 4.3: An overview of a frequently applied type of DL network, the Convolutional Neural Network (CNN). This example shows an input image of a car, which is labeled as one of multiple classes of means of transportation. [2]

Figure 4.3 gives a schematic representation of a CNN. A CNN typically consists of three types of hierarchical structures which are: **convolutional layers**, **pooling layers** and **fully-connected layers** (terms discussed later in more detail are boldfaced). In the convolutional layers pixels are aggregated with their neighbouring pixels to generate a **feature map**, those features are pooled and passed on to a next layer. Fully connected layers are "regular" NN layers. The distinction between the convolutional and pooling layers is shown in Figure 4.3. **Pooling** reduces the dimensionality of features by only propagating a statistical feature of feature groups.
Multiple **convolutional layers** with different aggregation schemes can be added in sequence. At the end of the model sequence a **fully-connected NN** performs the classification based on the outputs of the feature maps. Due to the **convolution** structure of the model, abstract features present in the data can be learned. Another advantage of grouping the abstract features is that the training is more efficient compared to a fully connected network since there are less parameters, because the parameters are shared since a given kernel is applied to the entire image.

It was concluded that the CNN is likely to be the best approach in the context of this project based on the literature study. The challenge within this project is for the model to detect the morphology of CH_4 plumes, a CNN is very suitable for this objective. Once pre-processed, the TROPOMI data can be suitable as input for a CNN. The relatively low resolution of the images will pose to be challenging though [44], as will be discussed later on. Many different types of CNN exist, but the core building blocks of which the models are constructed are similar for standard models and for custom models alike.

The main advantages of the CNN in the context of detecting CH_4 plumes in TROPOMI data are:

- **Rotational and translational indifference** Where a certain feature is located in an input image is does not matter, due to the convolutional structure features are detected anywhere in an image. This is important since the plumes will not always be e.g. in the center of an image. FcNN do not have this property.
- **Parameter sharing** The kernels used in the convolutional layers are applied all over the image. Complex feature structures can be learned by a kernel, which can then be detected all over input image.
- **Transparancy** It is possible to visualize which parts of the input image the CNN bases the predictions on.

The next sections describe the mathematical foundation of CNN. This theoretical framework will be referred to in Chapter 6 where the model which was developed in this study is discussed.

4.2. From TROPOMI data to input Image for the CNN

Before we proceed to the mathematical description of the CNN, first the input data is discussed in order to be able to relate the general equations and layers discussed in the next section to the context of methane plumes in TROPOMI data.

Figure 1.1, in the Introduction, showed an example of a very clear plume in TROPOMI CH_4 data. Figure 4.4A shows another example of a clear plume and Figure 4.4B shows the same plume after it is re-projected to the format which is used for the input samples. 4.4C shows a zoomed in version of subfigure B, which shows that the measurements of TROPOMI can be converted to a matrix of values, suitable for Computer Vision algorithms, like Neural Networks.

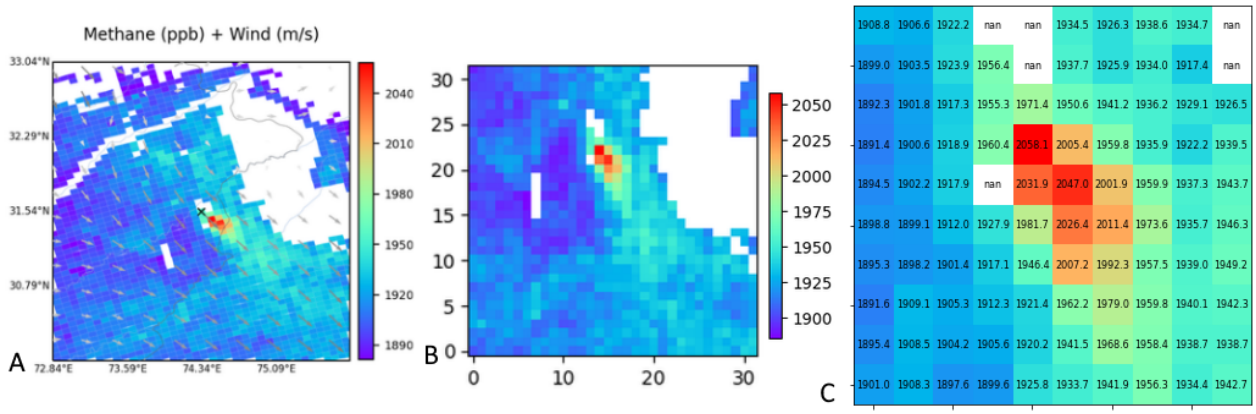


Figure 4.4: Subfigure A shows an example of a methane plume in TROPOMI data in latitude-longitue space. Subfigure B shows the same scene as an input sample. Subfigure C shows a zoomed in area of Subfigure B. Later in this chapter this is referred to as Image in a genral context.

This results in a 2d-matrix with values representing the atmospheric mixing ratio of CH_4. If only the atmospheric mixing ratio is taken into account this is the only input, when other atmospheric parameters are passed as input several of these 2D matrices are stacked in the third dimension.
The details of the data preparation are elaborated in Chapter 5, but an example is shown here for convenience and clarity.

4.2.1. Challenging aspects of methane plume detection using CNN

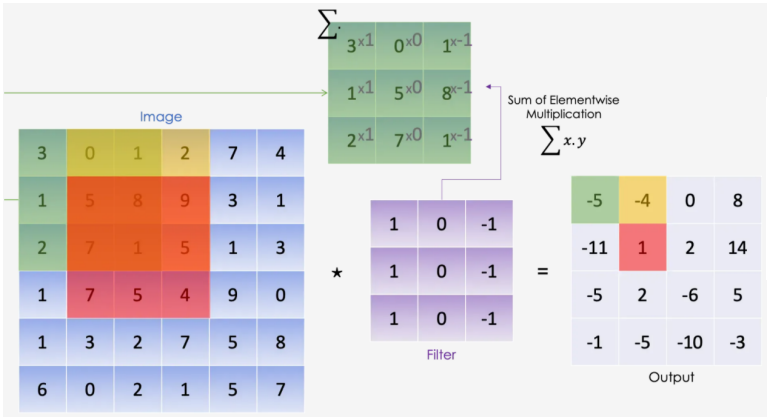
The automated detection of methane plume signals in TROPOMI data is a challenging problem. Figure 4.4 illustrates several of the challenging aspects related to this problem. First of all we have the low resolution, generally a CH_4 plume signal in TROPOMI data is only represented by 10-20 pixels (Figure 4.4C). This mean the information about a plume signal is represented by only a very small number of datapoints.
Furthermore the physical phenomenon of CH_4 point source emissions is not a binary phenomenon. A plume signal in TROPOMI data has to stand out above the measurement noise. Point source emissions with small source rates, which cause weak singal, are much harder to distinguish from the background noise. Manually labeling the training-data also poses challenges, as it is only possible to apply a binary label on samples which clearly contain a plume or clearly do not contain a plume signal. There are also in-between cases, which are even difficult for a human expert to label. This is different for example when one wants to design a classifier which separates apples from tomatoes. This is a truly binary problem, for which in theory a classifier with 100% accuracy can be designed. These photographs are also in a higher resolution most likely.
With methane plumes there are also multiple atmospheric parameters which can have influenced the methane retrieval (Section 2.5.2). This results in artefacts which can cause false signals in the xch4 field, which means the xch4 field on itself does not contain all information on itself. This is issue will also have to be addressed, and this is described further in Section 6.5 and Chapters 7 & 8.

4.3. Mathematical foundation of the Convolutional Neural Network (CNN)

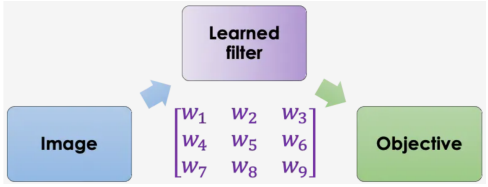
The images used in this section are obtained from this source [45]. The notation convention in those images is sticked to throughout this section in order to maintain consistency between the equations and the figures. The math was verified with other sources [46], [47], [42].
The images show a 3-channel dataset but the equations hold for any number of channels, three is most convenient for visualisations and is most commonly applied in general since visual imagery like photographs are made up of a red, green and blue channel.

4.3.1. The elementary components applied within the layers of a CNN

Features can be extracted from images by multiplying parts of the image with a filter or kernel matrix in order to create an output matrix. By applying the purple 3x3 filter on the example image (blue) in Figure 4.5a, an output image which is a feature map containing information about vertical edges in the original input image is created. The values in the kernel define the type of feature which is being extracted, in this case vertical edges. The filter, or kernel, is iterated over the entire input image to produce a feature map of the original image. This process is better known as edge detection and is widely applied, also in remote sensing, for example for agricultural field demarcation in visual satellite imagery. [48]
The filters in a CNN work similarly, to produce an output image from an input image, but the weights of those are not predetermined but learned (Figure 4.5b), or optimized, during the training phase.



(a) Filter processing with pre-defnied filter. [45]



(b) A filter of which the weights are not predetermined, but learned based on how to best an objective when multiplied with the input. [45]

Figure 4.5: Elementary filter operations visually shown. [45] An example of an Image (blue) in the context of methane plume is shown in Figure 4.4B

Figure 4.5a shows that the application of a filter reduces the dimensions of the input image. Moreover, from Figure 4.5a it stands out that the values on the edge of the input image have less influence on the output image than values which are not on an edge. This means not all information of the image is used to the full extent. By adding a band of zeroes around the image, this effect can be counteracted. This procedure is called padding p, more then one band of zeroes can be added, depending on what is required by the filtersize.
Figure 4.5a shows how the kernels applied to the Input image (green, yellow, red squares) result in a single value in the output Image. The 5×5 Image is turned into a 4×4 image, this happens because the filtersize is 3 and the stride s is 1. For larger images it is possible to use a stride larger than 1, resulting in an even smaller output image, but for small images it is better to use a stride of 1 to capture as much information from the image as possible.

the convolution product

Figure 4.5a shows the procedure for a 2D matrix, here the output values are the sum of the elementwise product of the input image and the filter (either predefined or learned). A Convolutional Neural Network (CNN) can consist of more than one channel, which makes the problem 3 dimensional, like in Figure 4.6.

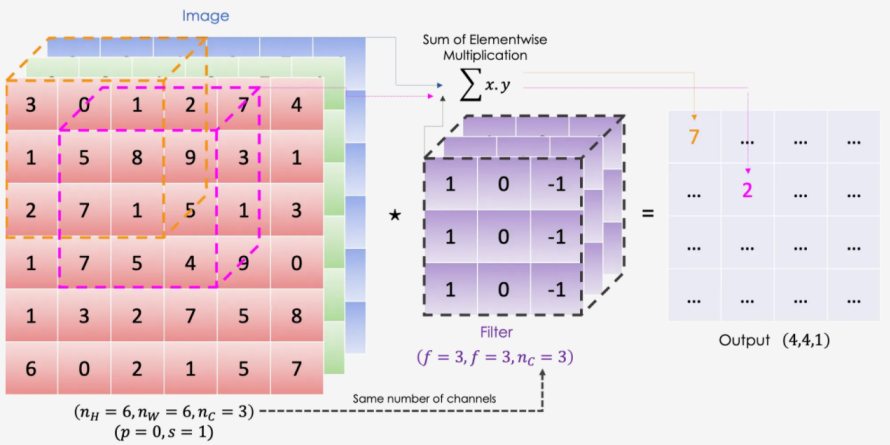


Figure 4.6: The concolutional product operation, the core component of the CNN. Do note the weights of the Filter or Kernel are not pre-defined as shown in this image, but these are learned, like in Figure 4.5b. [45]

The dimensions of the image are defined as:

$$\dim(\text{ image }, \text{I}) = (n_H, n_W, n_C) \tag{4.1}$$

where n_H is the size of the height of the image, n_W is the size of the width of the image and n_C is the number of channels. For the most common application of NN, photographs or other images visually interpretable by humans, the number of channels is 3, namely red, green and blue. The number of channels does not necessarily have to be 3, any positive integer number of channels is possible ($\mathbb{N}_{>0}$). Independent of the number of channels, the problem remains three-dimensional.
In order for the elementwise multiplication to be valid the 3D filter or kernel K must consist of the same number of channels n_C, which results in the following dimensions of the kernel K:

$$\dim(\text{ kernel }, \text{K}) = \left(f, f, n_C\right) \tag{4.2}$$

where f is the width/height of the kernel, or filtersize, n_C is the number of channels and the kernel K is assumed to be square and the dimension to always be an odd number $\{1,3,5,..\}$ in order for the kernel to consider all elements around a pixel from the image I.

Given these dimensions the convolutional product between an image I and kernel K results in a 2D output matrix. The elements in this matrix are the sum of the elementwise multiplication of the kernel and a sub-region of the input image as illustrated in Figure 4.6. The convolutional product (Equation 4.3), and its dimensions (4.4), can be mathematically described as [45]:

$$\text{conv}(I,K)_{x,y} = \sum_{i=1}^{n_H}\sum_{j=1}^{n_W}\sum_{k=1}^{n_C} K_{i,j,k} I_{x+i-1,y+j-1,k} \tag{4.3}$$

$$\begin{aligned}\dim(\text{conv}(I,K)) &= \left(\left\lfloor \frac{n_H+2p-f}{s}+1 \right\rfloor, \left\lfloor \frac{n_W+2p-f}{s}+1 \right\rfloor\right); s>0 \\ &= \left(n_H+2p-f, n_W+2p-f\right); s=0\end{aligned} \tag{4.4}$$

where the output image dimensions are floored when not an integer. In the example in Figure 4.6, no zero-padding and a stride of 1 is applied, resulting in an output (feature map) of shape $(4,4,1)$, relative to the input image of shape $(n_H=6, n_W=6, n_C=3)$ (Equation 4.4).
In order to get the same dimensions for the output image as the input image I, the condition $p=\frac{f-1}{2}$ has to be satisfied. With a filtersize f of 3, one band of zeros ($p=1$) has to be added to the input image I before convolution.

the pooling operation

Another tool which gives control over the sizes of subsequent layers is pooling. The features of an image can be pooled by applying a function on the feature maps. The most commonly applied pooling functions are average pooling and max pooling. With pooling, an empty filter is applied to the image, usually with the filtersize f and stride s equal, in order to have no overlap, the larger s and f the bigger the reduction in size. At the end of the (empty) convolution, the pooling function ϕ is applied on the selected element, resulting in an output layer of which the dimensions are defined in Equation 4.5 and where the output image dimensions are floored when not an integer.

$$\begin{aligned}\dim(\text{ pooling }(\text{ image })) &= \left(\left\lfloor \frac{n_H+2p-f}{s}+1 \right\rfloor, \left\lfloor \frac{n_W+2p-f}{s}+1 \right\rfloor, \mathbf{n_C}\right); s>0 \\ &= \left(n_H+2p-f, n_W+2p-f, \mathbf{n_C}\right); s=0\end{aligned} \tag{4.5}$$

activation function

The activation function $\psi^{[l]}$ is a function which the output of a layer is passed through before it is added to the output layer. An activation function makes sure the output is within a certain range (0 - 1 or -1 - 1). A kernel can potentially produce some very high outcomes, the activation function compresses the output to the desired range of values in order to avoid the values exploding over several layers. Figure 4.7 shows this principle, no matter the input x, the output of the activation function $\hat{y}$ is in the range (e.g. [0...1] or [−1...1]). Which of the activation functions works best depends on the specific situation. Sigmoid is most often used in Fully-connected NNs and ReLu is most often used in convolutional layers.

The softmax activation function is used for multi-class labels. The softmax function is useful because it converts the scores to a normalized probability distribution, which is more insightful.

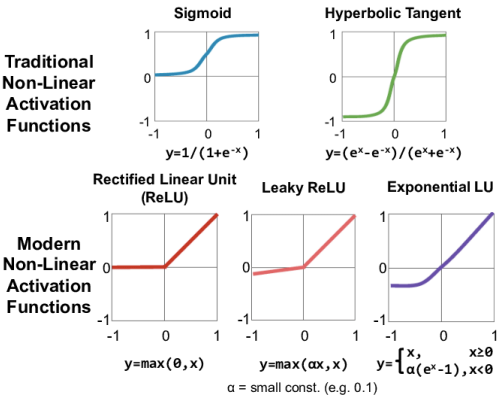


Figure 4.7: Some of the most commonly applied activation functions. From: [47]

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{4.6}$$

where $\vec{z}$ is the vector of scores corresponding to different classes, σ is the softmax output and C is the number of classes. For each of the scores corresponding to a class the normalized probability is computed.

4.3.2. The layers of a CNN

With the elementary components of the previous sections as building blocks, we will now move on to the layers of which a CNN consists. Three types of layers exist within a convolutional neural network. A CNN can consist of any number of layers, therefore in this section an arbitrary layer $a^{[l]}$ will be discussed, which takes as input image layer $a^{[l-1]}$ (= I in previous section).

Convolutional layer

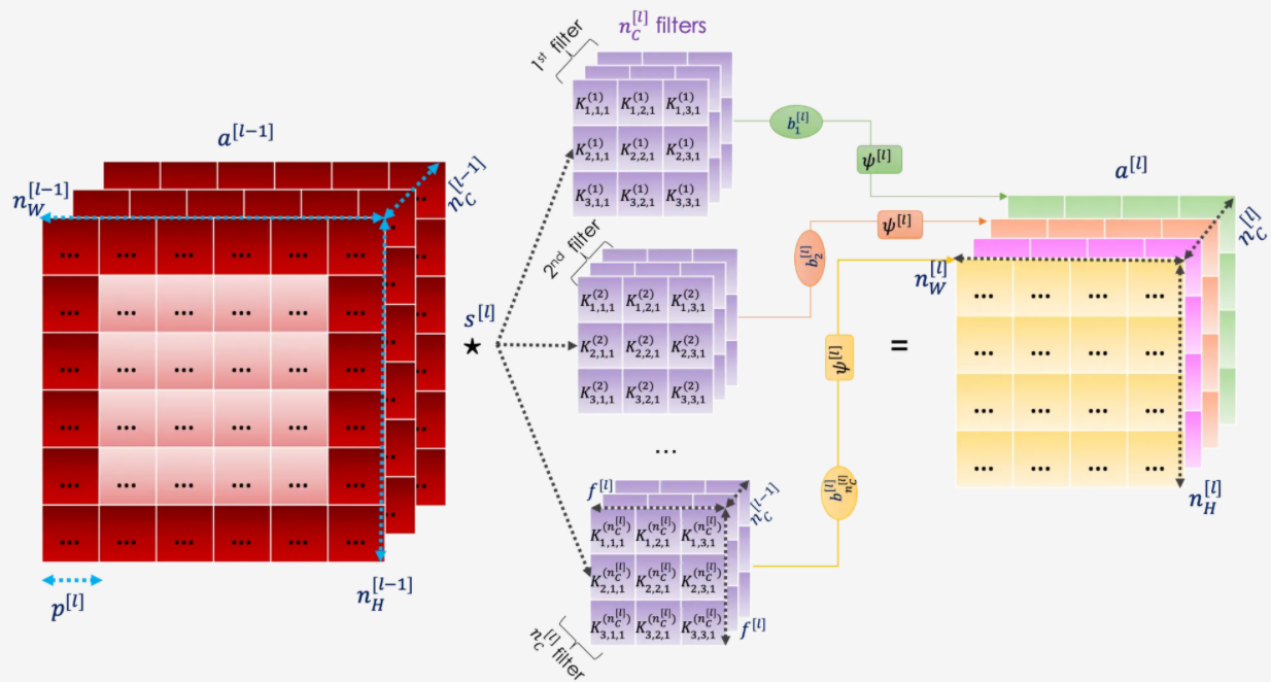


Figure 4.8: The convolutional layer visually shown for a scenario with three channels and three kernels. [45]

Figure 4.8 shows an overview of a convolutional layer [42]. Image $a^{[l-1]}$ is the input, with dimensions = $\left(n_H^{[l-1]}, n_W^{[l-1]}, n_C^{[l-1]}\right)$ (Equation 4.1). The kernel $K^{(n)}$ has the dimension = $\left(f^{[l]}, f^{[l]}, n_C^{[l-1]}\right)$ (Equation 4.2). The number of channels in the input image $a^{[l-1]}$ defines the number of channels in the kernel. n is the number of convolutions, and thus determines the number of kernels which determines the number of channels in the output image $a^{[l]}$.
The bias of convolution n is denoted as $b_n^{[l]}$ and the activation function used in this layer is denoted as $\psi^{[l]}$. The padding and stride at layer l are denoted as $p^{[l]}$ and $s^{[l]}$. The output size $\dim(a^{[l]})$ is = $\left(n_H^{[l]}, n_W^{[l]}, n_C^{[l]}\right)$.

Mathematically a convolutional layer (expanding on Equation 4.3) is expressed as, for each convolution n:

$$\forall n \in \left[1, 2, \ldots, n_C^{[l]}\right]:$$

$$\text{conv}\left(a^{[l-1]}, K^{(n)}\right)_{x,y} = \psi^{[l]}\left(\sum_{i=1}^{n_H^{[l-1]}} \sum_{j=1}^{n_W^{[l-1]}} \sum_{k=1}^{n_C^{[l-1]}} K_{i,j,k}^{(n)} a_{x+i-1,y+j-1,k}^{[l-1]} + b_n^{[l]}\right) \tag{4.7}$$

$$\dim\left(\text{conv}\left(a^{[l-1]}, K^{(n)}\right)\right) = \left(n_H^{[l]}, n_W^{[l]}\right) \tag{4.8}$$

which can be elaborated to the output layer $a^{[l]}$ as, when split for each n, applying Equation 4.3:

$$a^{[l]} = \left[\psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{(1)}\right)\right), \psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{(2)}\right)\right), \ldots, \psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{\left(n_C^{[l]}\right)}\right)\right)\right] \tag{4.9}$$

which results in an output shape $dim(a^{[l]})$ of:

$$\dim\left(a^{[l]}\right) = \left(n_H^{[l]}, n_W^{[l]}, n_C^{[l]}\right) \tag{4.10}$$

as the number of convolutions was defined to equal to $n_C^{[l]}$.

The height and width of the output image result from these equations (image is assumed to be square) as:

$$\begin{aligned} n_{H/W}^{[l]} &= \left\lfloor \frac{n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1 \right\rfloor \;;\; s > 0 \\ &= n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]} \;;\; s = 0 \end{aligned} \tag{4.11}$$

Equation 4.11 shows the size of the output is the size of the input image plus two times the padding parameter p, minus the filtersize f. See Figure 4.8.

The parameters which can be learned, or optimized during training, are the filters/kernels and the biases. There are $n_C^{[l]}$ kernels, which all have the shape $dimK(f^{[l]}, f^{[l]}, n_C^{[l-1]})$. Resulting in $(f^{[l]} \times f^{[l]} \times n_C^{[l-1]}) \times n_C^{[l]}$ parameters to be learned in the kernels of this convolutional layer. Each kernel also has a bias value which can be learned which results in an additional $(1 \times 1 \times 1) \times n_C^{[l]}$ parameters to be learned.

bias

Equation 4.7 (and Figure 4.8) contains a bias value, denoted as $b_n^{[l]}$. The bias value is a learnable parameter which is not part of the kernel itself. This value is used to provide a means to linearly offset or shift the output of a kernel, conceptually similar to the 0^{th} order term in linear regression (the b in $y = a \cdot x + b$).

Pooling layer

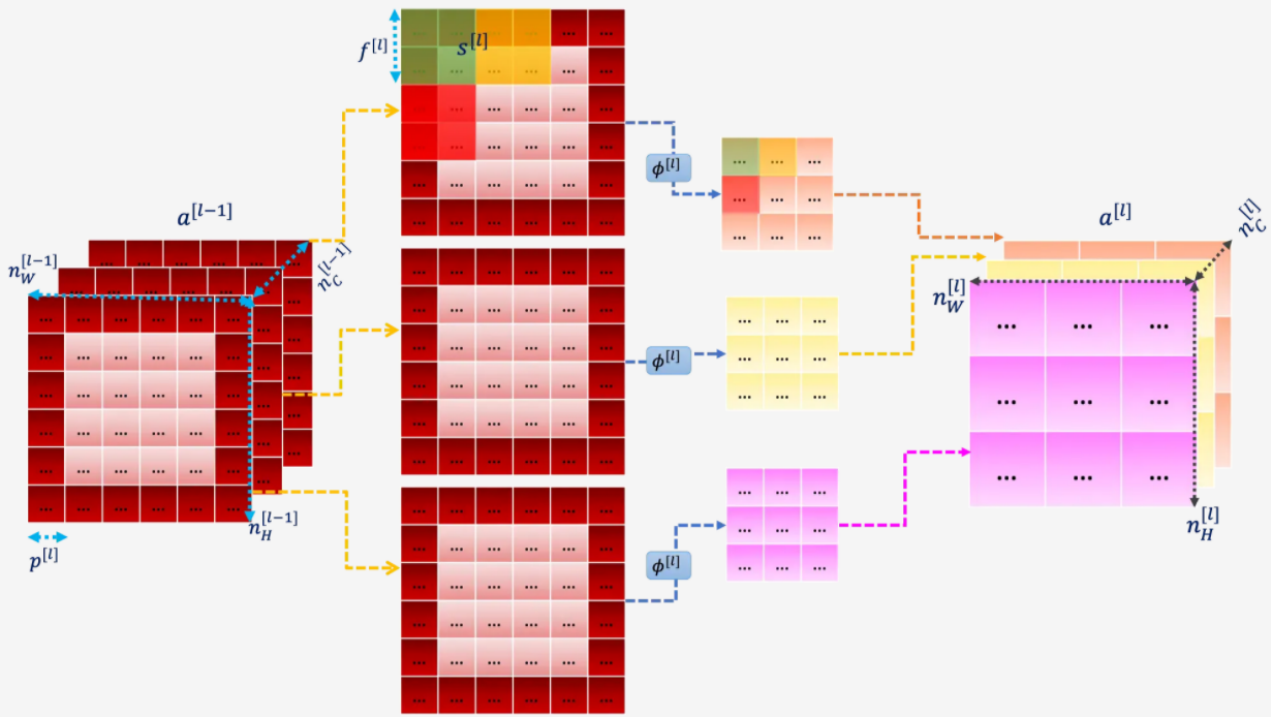


Figure 4.9: The pooling layer, shown for a scenario with three channel, each channel is separately pooled (dashed lines). [45]

As discussed above, the pooling layer is meant to downsample the input. The number of channels n_C is not affected, therefore $n_C^{[l]} = n_C^{[l-1]}$. As described above, a pooling layer uses an empty filter, with size $f^{[l]}$ and a pooling function $\phi^{[l]}$. The most common pooling functions are max pooling and average pooling, which result in either the maximum or average value of the subset being forward to the output layer as shown in Figure 4.9.

A pooling operation is applied on a single channel, which makes the mathematical formulation simpler. For each channel z, the pooling function is iterated in the x and y directions (width and height).

$$\begin{aligned} \text{pool}\left(a^{[l-1]}\right)_{x,y,z} &= \phi^{[l]}\left(\left(a^{[l-1]}_{x+i-1,y+j-1,z}\right)_{(i,j)\in[1,2,\ldots,f^{[l]}]^2}\right) \\ \dim\left(a^{[l]}\right) &= \left(n_H^{[l]}, n_W^{[l]}, n_C^{[l]}\right) \end{aligned} \tag{4.12}$$

The dimensions of the output layer are given by the expression below, once again those depend on the dimensions of the input layer, the padding p, stride s and filtersize f.

$$\begin{aligned} n_{H/W}^{[l]} &= \left\lfloor \frac{n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1 \right\rfloor \;;\; s > 0. \\ &= n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]} \;;\; s = 0 \\ n_C^{[l]} &= n_C^{[l-1]} \end{aligned} \tag{4.13}$$

The pooling layer has no learnable parameters, the only use is to downsample the input layer, but this can be very important if the network is faced with a large input image or many channels (either in the input, or due to convolutions).

Fully-connected layer

The last type of layer in a CNN is the fully-connected layer. This type of layer is what a regular neural network consists of exclusively. In a CNN, this type of layer is usually applied at the end, after all of the convolutional and pooling layers. A fully-connected neural network takes a vector as input and also has a vector as output layer. Each entry, or node, of both vectors is connected to all nodes from the other layer. These nodes have trainable weights, which can be optimised like the parameters of the kernel K in a convolution product. The value of a node in the output layer, layer $a^{[i]}$, is computed by summing the product of all nodes in layer $a^{[i-1]}$ multiplied with the weight w of the particular connection and multiplying that sum with the activation function $\psi^{[i]}$. This activation function can be different from, or equal to, the activation function used at a particular convolutional layer. (do note that where before l was used to indicate the layer, now i is used consistent with the convention of Figure 4.10, l in this image indicates which node of the input layer is regarded.)

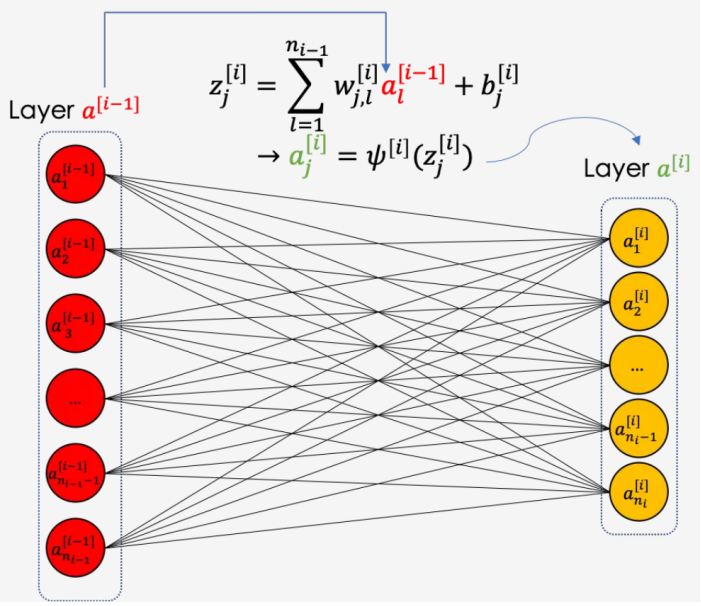


Figure 4.10: The fully-connected layer, linking every node along the input vector with every node in the output layer with a specific weight. [45]

In mathematical terms this is described as:

$$\begin{aligned} z_j^{[i]} = \sum_{l=1}^{n_{i-1}} w_{j,l}^{[i]} a_l^{[i-1]} + b_j^{[i]} \;\rightarrow \\ a_j^{[i]} = \psi^{[i]}\left(z_j^{[i]}\right) \end{aligned} \tag{4.14}$$

When considering the fully-connected layers in the context of final layers in a CNN, those outputs of earlier layers are layer $i-1$, therefore the dimensions of $a^{[i-1]}$ are $= \left(n_H^{[i-1]}, n_W^{[i-1]}, n_C^{[i-1]}\right)$. As discussed earlier, a FcNN takes a 1D vector as input, thus the input images need to be flattened, resulting in an input shape of $= \left(n_H^{[i-1]} \times n_W^{[i-1]} \times n_C^{[i-1]}, 1\right)$. Multiple FcNNs can be stacked after one another.

The number of nodes in input layer $a^{[i-1]}$ can be formulated as $n_H^{[i-1]} \times n_W^{[i-1]} \times n_C^{[i-1]}$. The number of learned parameters are the weights plus the biases. Since both sets of nodes are fully-connected, the number of weights is $n_{i-1} \times n_i$. Each node at the output layer also has a bias which is also a parameter, resulting in a total number of parameters of:

$$\begin{aligned} n_{i-1} \times n_i + n_i \;\; = \\ (n_{i-1} + 1) \times n_i \end{aligned} \tag{4.15}$$

This formulation of learnable parameters makes it clear that applying a fully-connected NN at large images, or in deep fully-connected networks is infeasible, since the number of nodes increases proportionally to the numbers of nodes in both layers. This proof underlines the statement at the end of Section 4.1: CNN are efficient due to the sharing of the learned parameters in the kernels, which is then applied throughout the entire image, as opposed to weights in a FcNN where each input and output are connected with each other.

4.3.3. Overview of the CNN structure

Figure 4.11 shows the general structure of a CNN where N, M, & F represent the number of convolution layers within a block before it is pooled at the end, the number of such blocks in sequence and the number of hidden layers in the fully-connected NN at the end of the network.

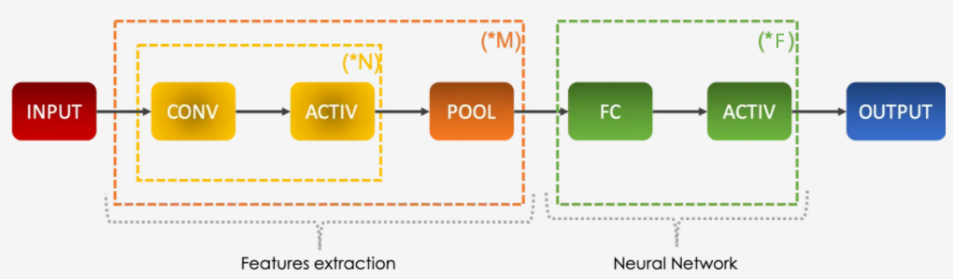


Figure 4.11: A schematic overview of a CNN, with the sections of the network which are commonly repeated together indicated. [45]

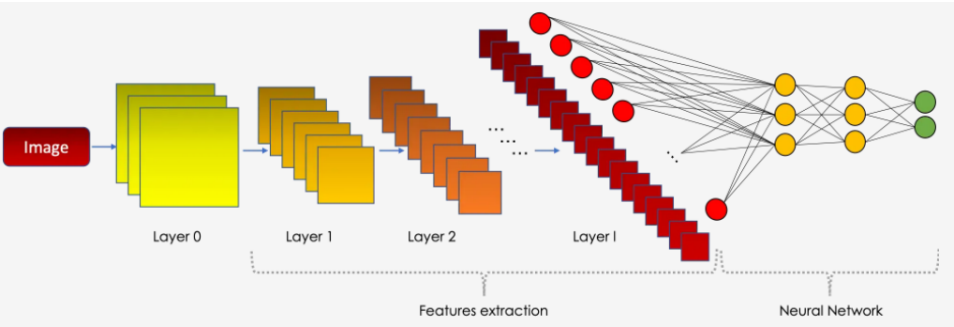


Figure 4.12: A schematic overview of a CNN, separated in the feature extraction part and the fully-connected neural network. This figure provides a more realistic view on what the different layers look like. [45]

The first part is called the feature extraction part, here abstract features are extracted from the (multi-channel) image. Based on these abstract features, higher level features are represented in subsequent feature maps. In the context of a photograph this could be e.g. first edges and contrasts, then shapes, then structures, then objects. Those hierarchical features are best fed through the network if the feature maps decrease in size (n_H, n_W), but increase in number n_C.

The feature maps and the fully connected NN are conceptually shown in relation to one another in Figure 4.12. This figure shows a similar architecture to the CNN in Figure 4.3 in Section 4.1 and to the images showing the individual layers in more details, Figures 4.8, 4.9 and 4.10.

4.3.4. training algorithm

This sections will elaborate how the trainable parameters of the CNN are optimized. The model has to be fitted to optimally differentiate between the different classes which are present in the dataset.
Neural Networks learn from existing examples, from training data, and so do CNN. The available data is split into two categories, the training set and the test set. The training set is where the majority of the data is, this data is used by the model to learn. This is done by splitting the data into a training subset and a validation subset. The validation subset is also used during the training phase, but this data is not actively used for training, but used for comparison (this is explained in more detail later in this subsection). The test set is not used during training, but is used after the model has been trained in order to test whether the model is generalized well. The data in the test set is 'new' for the model, therefore this can be used to infer how well the model performs when applied to new data.
All of the data samples are assigned an integer representing the class (plume versus no plume) it belongs to. This can be two separate classes (0 & 1), or multiple. During training the parameters of the model are adjusted in such a way that for an input sample x_i which is processed by the model the resulting prediction $\hat{y}_i$ is as close as possible to the true value y_i, the integer representing the class. For the entire dataset this can, in the simplest form be expressed as $J = \sum(y_i - model(x_i))$ or $\sum(y_i - \hat{y}_i)$, which is to be minimized. J is called the loss function.

The model is optimized by applying a combination of forward propagation, where the data is propagated through the network and the loss J is computed, and backpropagation where the parameters are improved using the gradients of the cost function. This process can either be done for the entire training set, or batches of data (discussed later in this section).

This process is repeated for N epochs in which the datasamples (or the batch) are fed through the network in a random order. Before starting the training phase, the model parameters (kernel values, weights of the fully-connected layers and the biases) are initiated with a random value.

In mathematical terms this process is expressed as:

$$\forall\ i\ \in [1,2,\ldots,N]:$$

$$\hat{y}_i^{\theta} = model^{\theta}(x_i) \tag{4.16}$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \mathscr{L}\left(\hat{y}_i^{\theta}, y_i\right) \tag{4.17}$$

where J is the loss function, $\mathscr{L}$ is the cost function, m is the size of the training set and θ is model parameters. This way the loss J for an entire epoch is computed. Different kinds of cost functions $\mathscr{L}$ exist, like the root mean-square error (RMSE) or (binary/categorical) cross-entropy. [46]

At the end of the epoch, the parameters of the model θ are updated and replaced with backpropagation:

$$\theta = G(\theta) \tag{4.18}$$

where G is a function which computes the gradients in which direction the models parameters can best be adjusted in order to perform better on the same dataset in the next epoch. An example of an optimization method is the stochastic gradient descend algorithm or ADAM.

This process is repeated until the model is sufficiently trained. Once the validation loss, which is computed at the end of each epoch starts increasing while the training loss keeps on decreasing, the model starts to overfit and worse performance will be obtained from application on the test set if training continues further. This effect is called overfitting and is analogous to polynomial fitting with a too high order.

batch normalization

When the learning algorithm is applied using batches, it is possible to apply batch normalization [46] (original paper [49]) before the forward propagation part. x is replaced by $\hat{x}$ which is the normalized version of x. For batch β, $\hat{x}$ can be computed as follows:

$$\hat{x}_i = \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \varepsilon}} \tag{4.19}$$

where ε is a small positive value preventing division by 0, μ_β (the mean value of the batch) and σ_β (the standard deviation of the batch) are computed as:

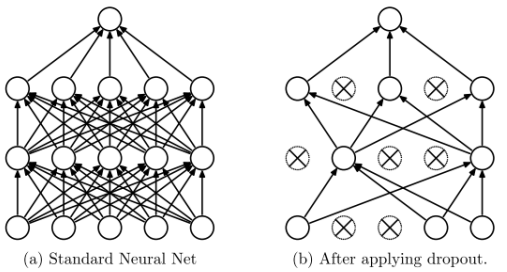
$$\mu_\beta = \frac{1}{M}\sum_{i=1}^{m} x_i \qquad \sigma_\beta^2 = \frac{1}{m}\sum_{i=1}^{m}\left(x_i - \mu_\beta\right)^2 \tag{4.20}$$

where m is the size of the batch, and M is the size of the total dataset. Batch normalization speeds up training, because it stabilizes the training and therefore greatly reduces the number of epochs required for training. Batch normalization is also shown to result in a lower final loss and a more stable loss curve, less bumps during training, as one might expect if the batches β have highly different means and standard deviations.

dropout

Pixels can be missing due to the cloud filter in the TROPOMI L1B data pipeline, or because the quality (qa_value) of the retrieval is not good enough to converge to a CH_4 mixing ratio in the retrieval algorithm (see Table 2.6).
Neural Networks and other Machine Learning models have difficulty with handling missing input data, therefore this is a crucial challenge to overcome for whatever type of ML model is to be applied within the TROPOMI-GHGSat project.

Dropout can be used to prevent overfitting in the fully-connected layers. [47] [46]. In the original paper introducing the dropout method [50] dropout is discussed to also be possible to be applied on the input layer. This is in some ways the same as passing inputs



(a) Standard Neural Net (b) After applying dropout.

Figure 4.13: Dropout applied to a simple NN, visualised. The crossed units have been dropped out in this example. From: [50]

with missing (zero) values in the input image grid (which will be the case for TROPOMI data). Dropping out a unit means temporarily removing it from the network, along with all its incoming and outgoing connections [50], this is schematically shown in Figure 4.13. Training a network on a dataset with a wide enough variety of plumes in TROPOMI data might render the network to capture the relevant features, as if it was trained using dropout on the input layer. This is not exactly the same, as with dropout the nodes which are dropped out circulate for different iterations, which is not the case with input data with missing pixels.

4.4. Performance evaluation metrics

Several different ways to assess the performance of a trained model exist, and those can be applied to different subsets of the dataset which is used for training. In Section 4.3.4 the training procedure was discussed as well as how the data used for training is split into the three categories. The fourth category is different, this is 'in-situ' data, which is not part of the initial dataset, which the trained model will be applied on.

- Labeled dataset
  - Training set
  - Validation set
  - Test set
- In-situ data (not part of the dataset used for the training procedure)

All of the data which is used during the training phase (the first three datasets, eventhough the test set is not used for training, but for evaluation of the trained model) were labeled before the training procedure, where a 0 indicates "no plume" (or a negative) and a 1 indicates a plume (or a positive). The model (either with randomly initiated weights at the beginning of training, or after the weights were updated several times) outputs a floating number $\hat{y} \in [0, \dots, 1]$ for each input sample. During the training process, subsets from the training and validation datasets are randomly selected and the model is optimized on this data by minimizing the loss function J (Equation 4.17) by subsequent forward and backward propagation (Section 4.3.4).
Once the training phase is finished the trained model is used to perform predictions on the test set, which is a random subset of the entire labeled dataset which is used for the training procedure, and therefore is assumed to be representative of the entire labeled dataset, and of new 'in-situ' data. The test subset was not used during the training phase, and therefore inferring the performance on the test set provides insight in how well the model is expected to perform on new data.

The trained model is then used to make a prediction based on each of the samples of the test set resulting in a prediction score (a floating number $\in [0, \dots, 1]$). The threshold, or classification threshold, value determines where the demarcation is drawn, 0.5 being the logical initial value for the threshold.

Each of the samples of the test set is part of one of four categories:

- (TP) True Positive
- (TN) True Negative
- (FP) False Positive
- (FN) False Negative

Where a True Positive means that the network estimates the sample to be a positive (or a plume) and that is actually the case. With a False Negative, the models estimates the sample to be a negative (no plume) which is false, because the sample actually does contain a plume, and so on.
Those results are commonly summarized in a confusion matrix [48], Figure 4.14, where the total number of the occurrences is shown in the table. Based on these definitions some metrics can be defined, which indicate the performance of the model.

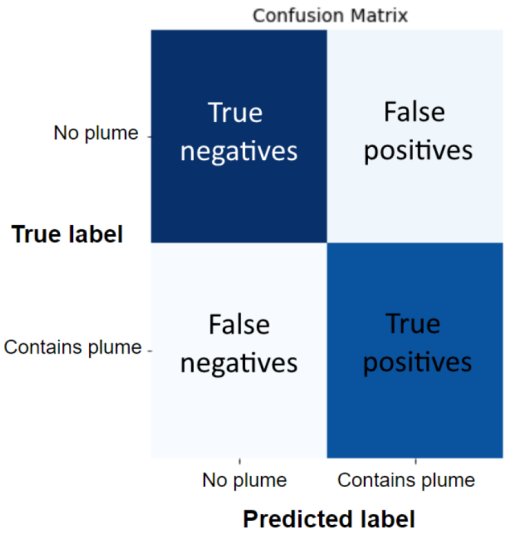


Figure 4.14: The confusion matrix which provides a visual summary of the performance of a model on the testset. This figure clearly shows which four groups a prediction can belong to.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \qquad \text{Precision} = \frac{TP}{TP+FP} \qquad \text{Recall} = \frac{TP}{TP+FN} \tag{4.21}$$

Another metric which, unlike accuracy, compensates for class imbalance is Cohen's Kappa (κ). [48] This metric is useful when dealing with imbalanced classes, which is the case in this context as scenes without plumes are much more frequent than scenes without plumes.

$$\kappa = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e} \tag{4.22}$$

where p_o is the relative observed agreement among raters (truth & model estimate) and p_e is the hypothetical probability of change agreement. κ values above 0.8 are considered to indicate very good agreement.
The F1-score [48] is a metric which combines precision and recall into the harmonic mean of the two, when both precision and recall are of importance, the F1-score is useful.

$$F_1 = 2 \cdot \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4.23}$$

An insightful visualization is the precision-recall graph for a varying classification threshold, this graph visualizes how costly it is in terms of additional FP (lower precision) to have less false negatives (higher recall). This is particularly interesting in problems where the detection of a rare class (plumes) is the objective, but one also wants as little false positives as possible in order to reduce the amount of manual effort required in verifying the model outcomes.

5
Data Preparation

TROPOMI has been in orbit since November 2017, which means three years of historical data is available at the time of writing. A large dataset of TROPOMI observations is therefore available, this vast amount of trainingdata allows for the application of a Machine Learning approach in order to detect plumes in the TROPOMI CH_4 data.
In order to be able to apply Deep Learning algorithms to the TROPOMI CH_4 data, this data has be to remapped into a suitable format first (as was conceptually shown in Figure 4.4). This chapter describes how the TROPOMI Level 2 CH_4 dataproduct is turned into a suitable format, how this data is filtered, and how a set of training data was created.

5.1. The TROPOMI Level 2 Methane dataproduct

SRON is the institute responsible for developing the TROPOMI Level 2 CH_4 dataproduct. This is the dataproduct which is used during this study.

The dry atmospheric mixing ratio is the unit in which the methane measurements are commonly expressed (Section 2.4.1). These quantities are concentrations, influenced by both local emissions and background levels, therefore emissions can never immediately be retrieved. This is illustrated in Figure 2.9a.

The retrieved CH_4 data is provided by SRON aggregated by orbitnumber, since observations are only performed during the sun-lit section of the orbit (as is shown in Figure 5.1) and therefore the data can be split in distinct sets numbered by the orbitnumber.
Alongside methane, several other atmospheric parameters are retrieved. A priori information is added to the Level 2 CH_4 dataproduct as well. A full overview of the data used from the Level 2 CH_4 dataproduct is stated in Table A.1.
Figure 5.2 shows a visualization of the most commonly used parameters where the data is plotted in a Mercator projection. This figure shows a clear plume. Alongside xch4 also the surface albedo measured in the SWIR, the aerosol optical thickness also measured in the SWIR and the surface pressure are displayed. In the top-center image the local windfieldfrom GEOS-FP [51] is shown as well.

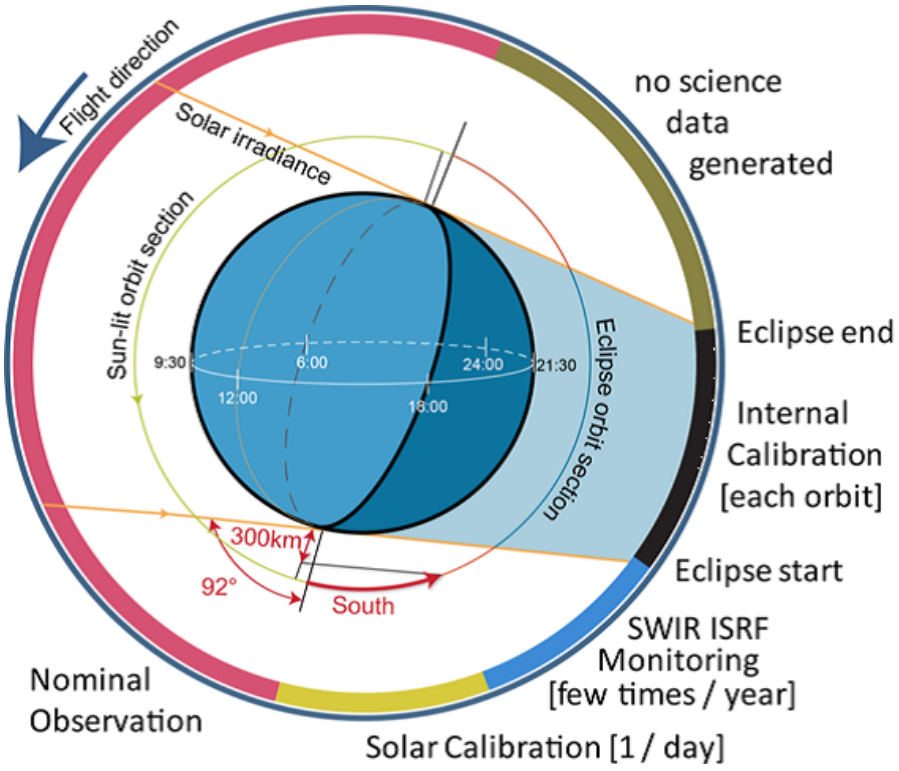


Figure 5.1: A schematisation of during which part of the orbit measurements are being performed.[1] In the final design the flight direction is reversed, this image is from the preliminary design phase.

[1] https://sentinel.esa.int/web/sentinel/missions/sentinel-5/instrument-payload last access: 24-07-2020

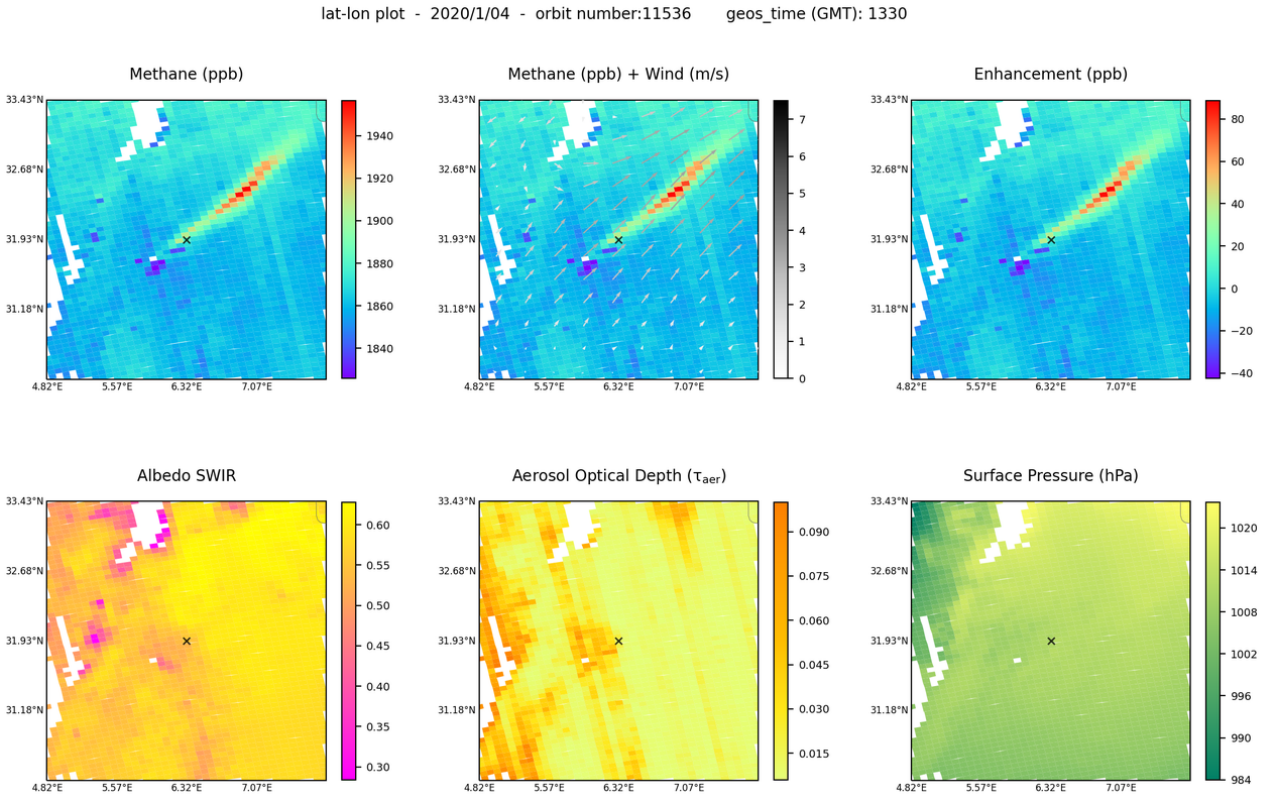


Figure 5.2: A latitude-longitude (Mercator) representation of the CH_4 TROPOMI data showing a clear plume. The pixels are defined as shown in Figure 5.3.

5.2. Re-mapping the TROPOMI pixel format

In order to be able to apply a Neural Network (NN) or Deep Learning (DL) method on TROPOMI CH_4 data, the data format has to be altered. The most commonly applied Mercator projection with the standard TROPOMI pixel shape can not be used as input for a NN. The data has to be in the shape of a 2/3D array, or more generally, a tensor. Figure 5.3 shows how the TROPOMI pixel is defined with four corner coordinates.

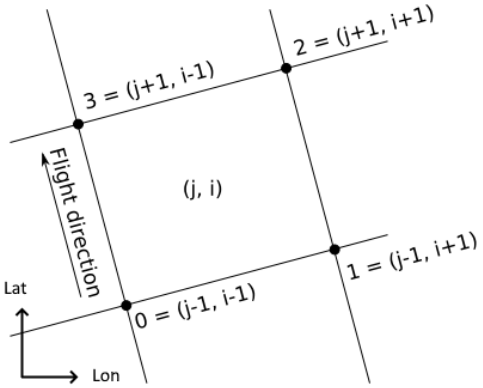


Figure 5.3: A schematisation of the groundpixels of the measurements performed by the TROPOMI instrument as defined in the Sentinel 5P Level 2 data product. The orbital inclination of 98.7°results in this flightpath and pixel geometry relative to Meridians and Parallels of the Mercator projection (with coordinates in latitude and longitude). From [32]

Two ways to turn the TROPOMI pixel format into a suitable format were investigated.
The first option is to oversample the data in the Mercator projection with a, say, 0.1° × 0.1° pixel size. This option would yield an orthogonal dataset. However this option is suitable when using all orbits of a day, which results in global coverage, or for longer term averages. When using single orbit data, this option has multiple downsides. The edges are not straight, but are saw-tooth shaped. Also given that the size of TROPOMI pixels in the along-orbit direction are more or less constant (5.5 km since September 2019, before this update 7km) but not in the across-orbit direction, where the pixel size varies between 7km in nadir to about 35km at the edges of the swath, this would yield strange patterns in the data. For a high resolution multiple adjacent pixels used for the NN input would be exactly the same value since they fall entirely within one TROPOMI pixel, and the datasize increases since the number of values to store is greatly increased. When selecting a sufficiently low resolution to counter those effects, a lot of information is lost. Another problem with the Mercator projection is the distortion at high latitudes (resulting in Greenland appearing to be as large as Africa with a Mercator project while the surface area is actually 15 times smaller).

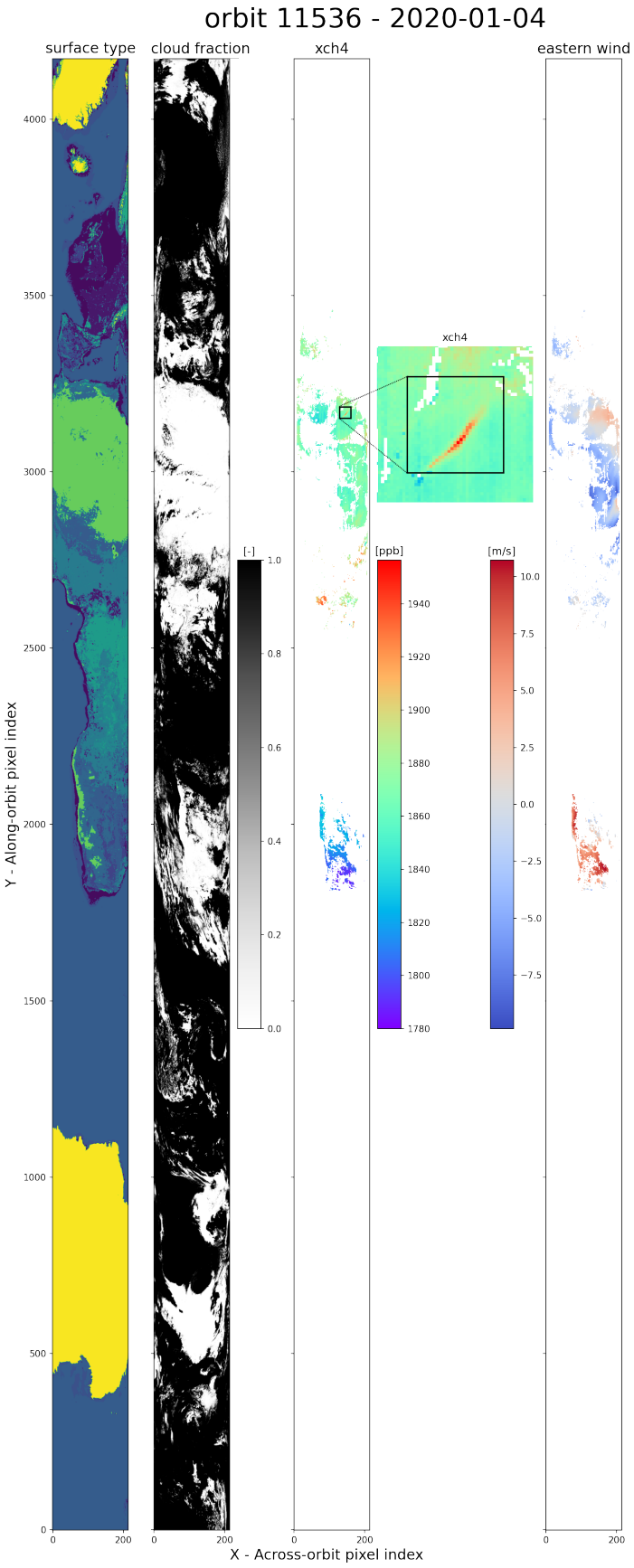


Figure 5.4: The ground-track based reference frame. Showing the shapes of the continents of this orbit together with several parameters and a zoomed-in plume. More details are provided in the text.

The other option is to define a reference frame which is not earth-centered, and not satellite-centered but ground-track centered, with the x-axis parallel to the line 0-1 and the y-axis parallel to the line 0-3 in Figure 5.3. This reference frame solves the issue of the zigzag sawtooth pattern along the edges of the single orbit data. Also in this reference frame the original TROPOMI pixels within an orbit are perpendicular (see Figure 5.3). More importantly, this also increases the compatibility between different orbits, since the observation geometry is constant (the effect of orbital decay is small).

This approach does however introduce the issue that the pixel area is not constant, but the alternative approach also struggles with this issue as is described above (This is an important issue though, and how it was dealt with regarding the training of the network is discussed in Section 5.5). Another advantage is the fact that the entire orbitfile is not resampled (which is a computationally expensive operation), but only remapped. This speeds up the detection process, which can be vital once more steps towards a near-real time approach are made.

Figure 5.4 shows what this ground-track based reference frame looks like for an entire orbit (~4200 × ~230 pixels). The surface type parameter of the Level 2 CH_4 dataproduct is plotted at the left in order to show the landmasses which are observed during this particular orbit, from bottom to top: Antarctica, Africa, south-western Europe, the United Kingdom, Iceland and at the top Greenland. The next image shows the cloud fraction product obtained by VIIRS (Section 2.5.1) where black denotes thick clouds and white indicates clear skies. The xch4 plot shows cloud-free observations over land and shows a clear difference in CH_4 concentrations between the northern and southern hemisphere. When zooming in a plume signature can be observed. In the very right plot the eastern-western windspeed at the time of observation at that particular location on Earth is shown. This illustrates how this external data is matched to the TROPOMI data, this wind-data will be discussed in more detail later in this section. An animation showing the orbital geometry of a full orbit of observations is available on the official TROPOMI website[2].

The windfield shown in Figure 5.4 is from GEOS-FP, an external source mapped to the TROPOMI pixels, it is not available within the TROPOMI

[2] http://www.tropomi.eu/gallery/bringing-air-pollution-focus last access: 18-01-2020

Level 2 CH_4 dataproduct. In the figure, only the eastern winds (western winds are denoted as negative values) are shown, but the northern wind product is also added to the TROPOMI pixels.

The GEOS-FP windfiles[3] distributed by National Aeronautics and Space Administration (NASA)'s Global Modeling and Assimilation Office (GMAO) are used in order to be able to deduce where a plume visible in the TROPOMI data originates from. These windfiles are a reanalysis version of a full-physical atmospheric model. The wind data is available every hour (XX:30) with a resolution of about 25×25 km^2. The lowest grid-points in the model are 60m above the surface, but an estimate of the windspeed 10m above the surface is provided as well. These are used in the project, since this is the most relevant wind regime for plume emissions. [24]

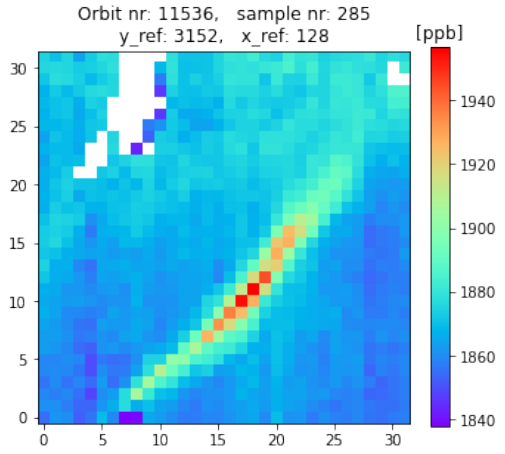


Figure 5.5: A zoomed-in version of the plume visible in Figure 5.4. The colorscale here stretches only the local min and max, rendering a much clearer view of the plume compared to Figure 5.4.

When combining the eastern and northen windfiles, these can be plotted as quiver plot, which gives insight in the direction and velocity of the winds over a certain area of interest. Plumes move downwind from point-sources, therefore these windfiles can be used to investigate where a point-source is likely to be located based on an area showing high methane concentrations in the atmosphere together with the direction of the wind.
The GEOS-FP data is added to the TROPOMI dataproduct in an automated way by matching the center of a certain TROPOMI pixel (see Figure 5.3 to the spatially nearest GEOS-FP pixel.

Next, the full-orbit data is split into images of a suitable size as input for a Neural Network. Given the morphological scale at which the phenomena of CH_4 plume emissions ranges, an image size of 32×32 pixels ($n_H = n_W = 32$) is chosen. Figure 5.5 shows the same plume which was highlighted in Figure 5.4, as a single sample, which shows more contrast in color. The coordinates in Figure 5.5 are a local reference frame, the vector [y_ref, x_ref] indicates the origin of this local reference frame in the 'global' reference frame of the entire orbit.

The full orbit data is split into data samples of size $n_H = n_W = 32$ by applying the "moving window" approach with a stride, or overlap, of 50%. This approach allows isolates a potential plume with its close surroundings which is most suitable as input for the CNN. Because it is not known beforehand whether the sample does contain a plume and where in the sample the plume is located, the 50% overlap ensures that if a plume is cut in half by the edge of the sample, it will be in the middle of the next sample. This approach was earlier applied by Kumar et al. [52]. Figure 5.6 shows the steps involved, subfigure A shows the Mercator projection of a full orbit of data, with the nadir groundtrack in red. This is then converted to the format shown in Figure 5.4 in subfigure B. Subfigure C shows how the moving window approach is applied to the full orbit into inidividual samples like Figure 5.5. Subfigure D shows the front view of the ground-track, whereas A, B and C show the 'top-view' or satellite viewpoint, subfigure D also illustrates why the pixels further away from nadir cover more area on the surface of the Earth, this is a result of the observation geometry.

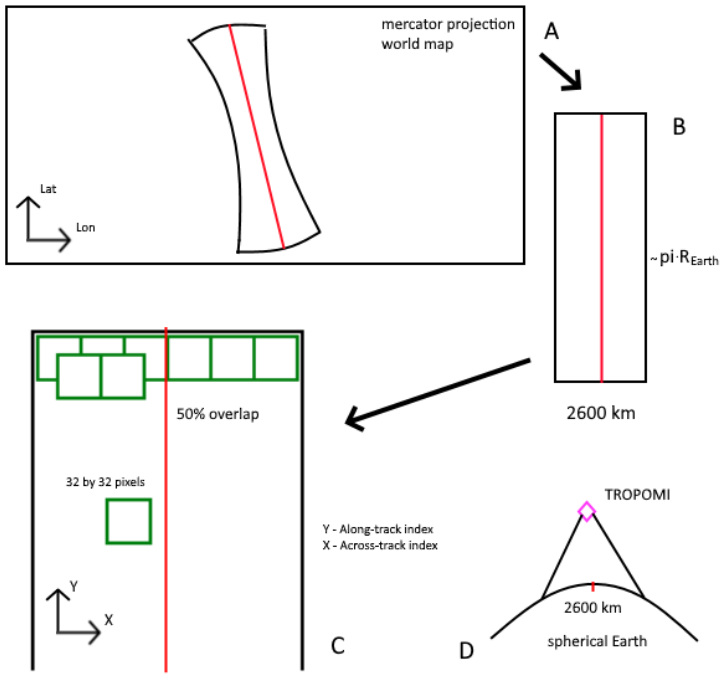


Figure 5.6: A schematisation of how a single orbit file like the one shown in figure 5.4 is processed to the standardized sample format shown in Figure 5.8.

[3] https://portal.nccs.nasa.gov/datashare/gmao/geos-fp/das/ last access: 11-06-2020

5.3. A posteriori filtering, albedo correction and de-striping

The TROPOMI Level 2 CH_4 dataproduct contains both a "raw" CH_4 dataproduct, and a bias-corrected dataproduct. Within the scope of this project, additional filtering was applied in order to improve un-supervised performance. Also the Arctic, Antarctic (latitudes > 75° and < −60°), Greenland and Iceland were always filtered out, as the xch4 dataproduct is less accurate in snowy scenes (see the red area in Figure 5.9).
Since nan-values are a problem for the Neural Network based approach, it is preferable not to use the advised quality settings (qa = 1, Table 2.6) but instead use a lower quality filter (qa = 0.4) and manually filter out only a select part of the data. An overview of these filters is provided in Table 5.1.

parameter	filtervalue	data filtered out
xch4 precision	10	>
aerosol optical thickness (SWIR)	0.10	>
latitude	-60, 75	<, >
cloud fraction (SWIR)	0.10	>
aerosol optical thickness (NIR)	0.30	>
Solar zenith angle (SZA)	70°	>
Viewing zenith angle (VZA)	60°	>
albedo (SWIR)	0.02	<
albedo (SWIR & NIR combined)	0.95	>

Table 5.1: A posteriori filtering, the filtersettings.
A description on the different parameters can be found in Table A.1 and [32].

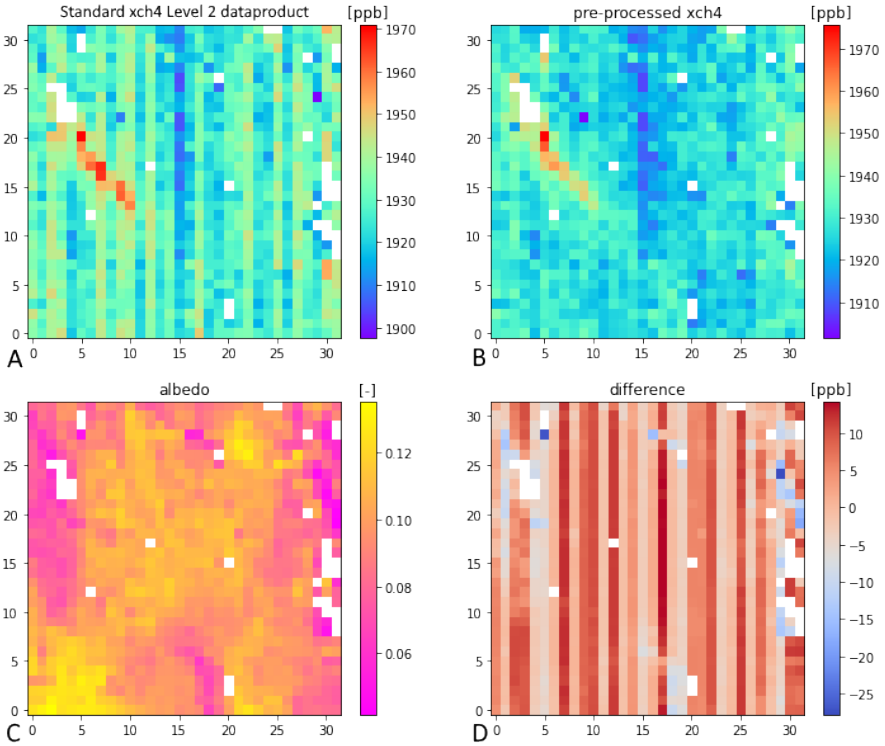


Figure 5.7: The a posteriori processing is applied on subfigure A (before), resulting in subfigure B (after). Subfigure C shows the corresponding albedo field, subfigure D shows the differences per pixel resulting from the operation.

On top of this two other operations were applied. The xch4 data was corrected with regards to biases in the albedo field as described by Lorente et al. [53]. The xch4 field was also de-striped, with TROPOMI data, one of the known issues is a striping effect in the along orbit direction. This effect is present in the dataproduct of several of the retrieved atmospheric trace gas concentrations. The stripe-correction approach which was applied in a study about Carbon monoxide (CO) pollution on city-level scale by Borsdorff et al. [54], this has since been modified for CH_4 and is applied a posteriori on the xch4 data[4]. This procedure is applied on an entire orbit file at once, the result of the different filtering and modifications is shown in Figure 5.7 for a single sample, where the effect of the de-striping algorithm and the effect of the albedo correction can clearly be distinguished.
The standard Level 2 bias-corrected xch4 dataproduct is shown in Figure 5.7A. Figure 5.7B is the custom a posteriori pre-processed xch4 sample (after filtering, albedo-correction and de-striping). The difference between the two is illustrated in Figure 5.7D. It is clear that in Figure 5.7B the background is much smoother and also the effects of the striping on the plume (clearly visible in the standard dataproduct in Figure 5.7A) are smoothed out, revealing the plume morphology. The effects of the albedo correction are also clearly visible.

[4]The de-striping was modified for CH_4 by J.D. Maasakkers based on the CO de-striping by Borsdorff et al. [54], the improved albedo correction was developed by Lorente et al. [53]. The total set of filters and a posteriori modifications to the standard xch4 dataproduct was designed by J.D. Maasakkers and applied by all students under his supervision for consistency.
I have included the steps taken here during preprocessing for reproducability, but the design is not my own work.

5.4. The different parameters included in an "input sample"

Table A.3 in the appendix shows which parameters are included in each of the datasamples. A visual representation of which parameter (or layers or channels) are part of a single datasample is shown in Figure 5.8. The scene shown in Figure 5.8, is the same one as in Figures 5.5, 5.4 and 5.2.

Most parameter are self-explanatory or were described before. Most parameters are part of the TROPOMI Level 2 CH_4 dataproduct, except for the GEOS winddata which is mapped to the TROPOMI pixels from an external source as described in Section 5.2. The albedo (SWIR), aerosol optical thickness (SWIR) and surface pressure are the most relevant atmospheric parameters to distinguish artefacts from real plumes.
The surface area covered by a pixel is computed using the haversine function to compute the distance along the surface of the WGS-84 globe model between two pairs of coordinates. The source mask field is used to visualize where the source is located for samples with a known source (the plumes in the training data, discussed in the next section). The timestamp, latitude and longitude values of each pixel are used to pinpoint the date and time and the location on Earth of a plume.
The enhancement is the xch4 minus the a priori estimated CH_4 atmospheric mixing ratio based on simulations by Copernicus Atmosphere Monitoring Service (CAMS), but this remains unused because of sometimes strange values of the a priori xch4. The quality value gives an indication about the quality of an observation, the values were described in Table 2.6.
The cloud fraction (SWIR), the orthometric (relative to the sea level) and surface type are three a priori parameters added to the TROPOMI Level 2 CH_4 dataproduct which are useful for identifying why certain pixels are filtered out. The cloud fraction denotes cloudy scenes, too cloudy pixels are filtered out, but slightly cloudy pixels can still be useful, but might have an influence on measured xch4 levels. The orthometric surface altitude can show the topography of the surface, revealing mountain ranges and valleys, which can explain accumulation effects of methane (do note that accumulation is different from the difference in column height due to surface elevation, which was discussed in Section 2.4.1 and Figure 2.9a). The land surface type is a bitmask layer which differentiates in about forty types of surfaces, however this is simplified to either water or land in order to be able to visually identify coastal scenes or lakes.
These last three parameters are particularly useful in manual inspection, since those are not filtered out together with missing xch4 values, as they are a priori values. They can provide insight in to why a certain pixel is most likely missing, and provide context about the surface and atmospheric conditions under which the measurement was performed, also for pixels with missing xch4 values.

The parameters of a sample, visualized in Figure 5.8, could all potentially be used as channels in a CNN by matching the number of channels, n_C, to the number of parameters used. Some parameters are unfit to serve as inputs to the NN approach, since it is preferable to keep the samples indifferent of locations and conditions to prevent overfitting on this information, but those parameters can still be useful for other purposes as will be described in Chapters 7 and 9.
A full overview of all parameters included in an input sample (some unused) is given in Table A.3. The dimensions of a set of samples are $[m, n_H, n_W, n_C] = [m, 32, 32, n_C]$, where m is the number of samples.

The plume in Figure 5.8 is actually one of the easiest to detect, since it is such a large plume with high enhancements (potentially the clearest plume of 2020). Two other such overviews are shown in the appendix in Figures A.4 & A.6. Especially Figure A.6 gives some insight in what more difficult to detect plumes look like as this image contains cloudiness close to the plume, high enhancements elsewhere in the sample which are not part of the plume, large differences in surface altitude and a non-uniform albedo field.

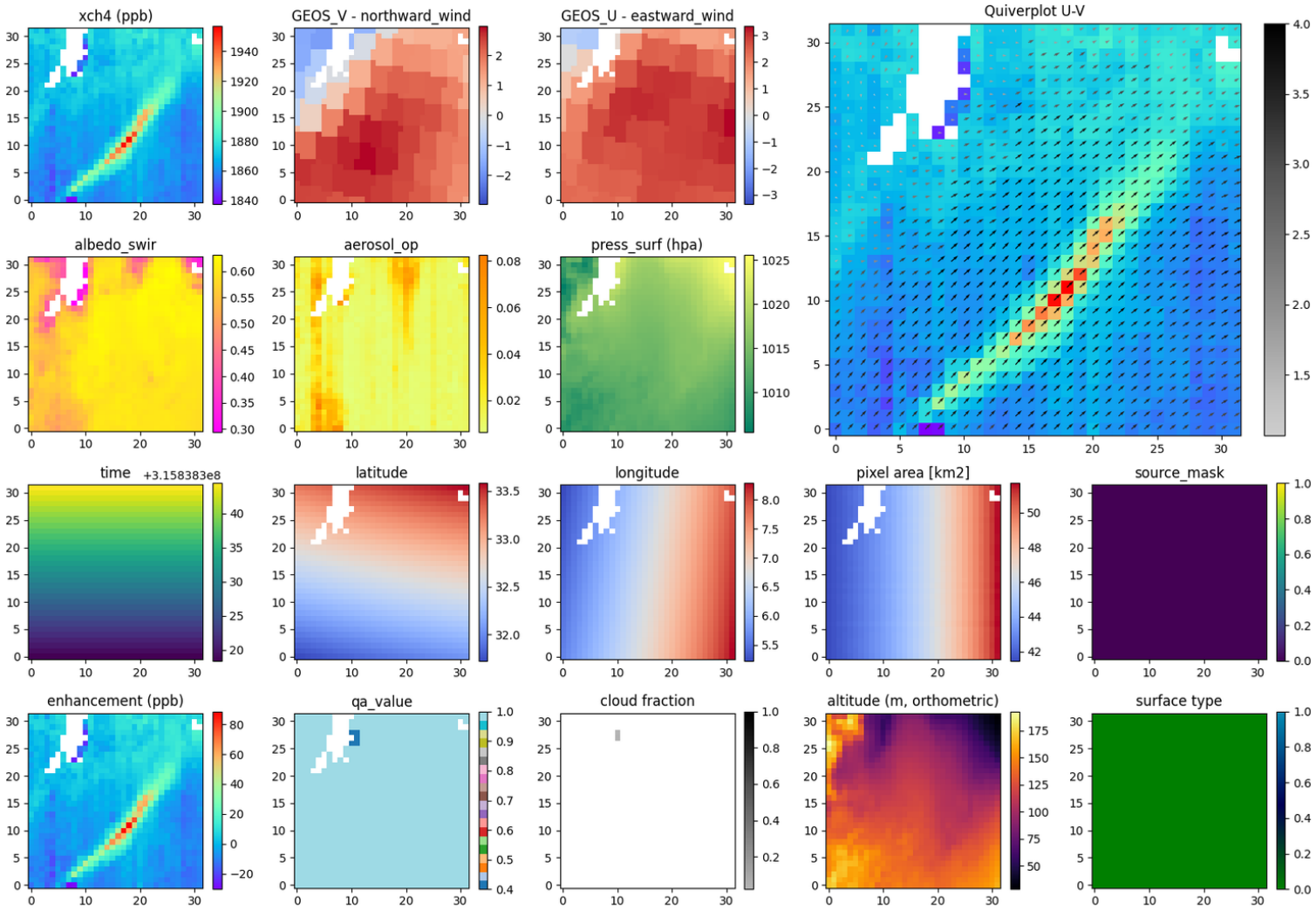


Figure 5.8: An overview of the most relevant parameters included in a sample. The different parameters share their height and width coordinates. A full overview of all parameters included in the standard sample format is given in Table A.3

5.5. Generating training samples

The approach of generating standardized images from the full-single orbit datasets results in compatible historic and future datasets. The image which will later be used as input in a CNN is a local, regional representation of the XCH_4 field and other atmospheric parameters. Given this standardized way of preprocessing, the network is indifferent to the date and location of an observation.
Observations from the past can be used to train a NN which will be able to detect plumes similar to those it has been trained on in future observations, using the features (feature maps) learned the CNN. At the time of writing, TROPOMI has been operational for over three years, which means a vast library of TROPOMI CH_4 data is available. This earlier data is used in order to create a trainingset. Knowledge gained during the TROPOMI-GHGSat project about locations with consistent CH_4 emissions before the start of this project is used in order to construct a dataset of images containing a wide variety of methane plumes with different morphologies.

In Section 5.2 the issue of different surface areas per pixel depending on where in the swath (viewing angle) the observation is taken was mentioned. In order to counter this effect, it is important to include plumes which were measured all over the swath into the training set. Therefore care is taken to also include numerous off-nadir detections. This way the network will learn plume morphologies both for low and high viewing angles.

Figure 5.9: An overview of the spatial distribution of the locations which were used for the trainingdata, chosen to be as diverse as possible. The red areas indicate which area of the globe is filtered out by default in new single orbit data.

5.5.1. The initial training set

27 different locations which were earlier identified as locations with frequent large methane emissions were used to generate a dataset of plumes. A limited number of locations within the same country was used, and a diverse mix of categories of sources (Figure 5.10, Oil&Gas, Urban/Landfill or Coal (Section 2.3.3)) was used. Figure 5.9 shows the locations which were selected.
Visualisations for days with enough coverage and enhancements near the known source location were made. This information was used to generate samples in the correct format, resulting in a dataset of 407 plumes.

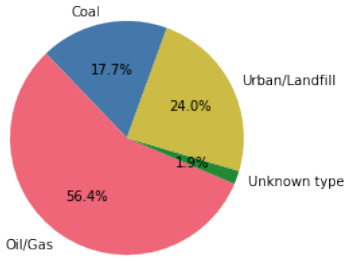


Figure 5.10: The types of sources of the plumes of which the positives dataset consists. (also see Section 2.3.3)

Negative samples, samples which do not contain a plume, were generated by applying the data preparation workflow on a total of eight orbits, and manually inspecting each of the resulting samples. Any plumes which were present in those orbits were masked out, resulting in 2135 negative samples in this dataset.
Each of those samples was manually verified after it had been turned into the correct format.

Dataset A consists of relatively many plumes, when compared to a regular single orbit file (like in Figure 5.4, which only contains one plume in all samples resulting from this single orbit file). This is done on purpose in order to allow the network to learn what the features of a plume look like.

Name	nr of samples	nr of positives	nr of negatives
Dataset A	2542	407	2135

Table 5.2: An overview of which kinds of samples Dataset A consists of. The plumes in this dataset are obtained from the locations shown in Figure 5.9.

6
Model Architecture development and Training

With the training dataset in place, the CNN model can now be constructed and trained.

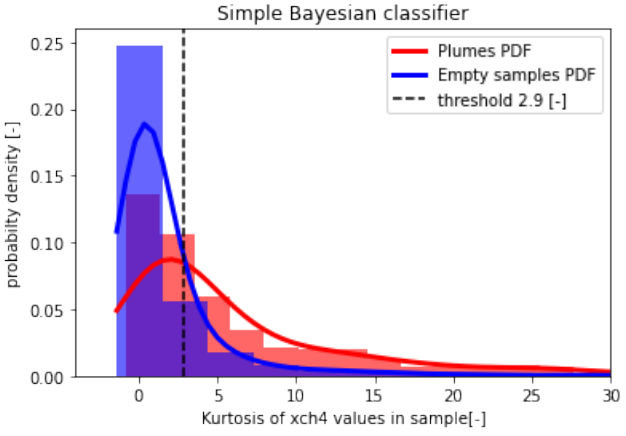
6.1. Simple baseline model, Bayesian classifier based on Kurtosis

Before moving on to developing a CNN model, first a simple Bayesian classfier algorithm is designed in order to infer how well the dataset can be classified using a baseline model. CH_4 plumes are characterised by enhancements relative to the surrounding area, therefore it makes sense to use the kurtosis of the pixels within a single sample of 32×32 pixels ($n_H = n_W = 32$). The kurtosis statistical parameter gives insight in how 'heavy-tailed' a probability distribution is, something which we would expect the distribution of mixing ratio values in a sample containing a plume to be (e.g. see Figure 5.5). The kurtosis is defined as:

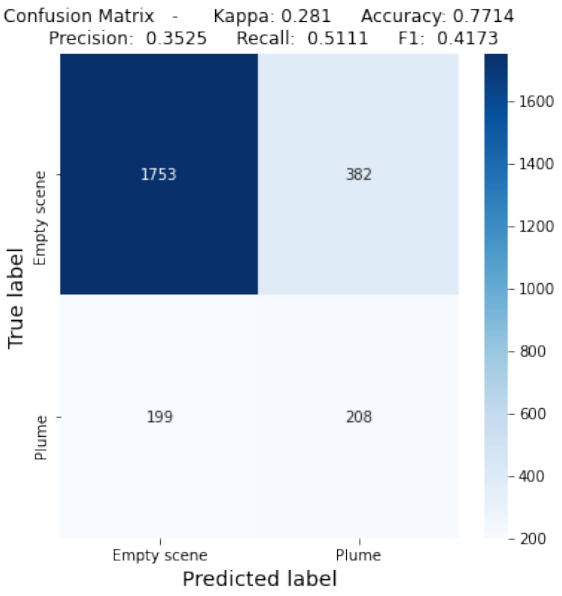
$$\text{Kurt}[X] = \text{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] \tag{6.1}$$

where μ is the mean of the distribution, σ is the standard deviation and X are the values of the distribution (xch4 values in this case). The kurtosis Kurt$[X]$ is the expected value of the right part of the equation, this equation shows that values far in the tail are weighted heavily which makes the kurtosis a suitable parameter for this purpose.

Figure 6.1a shows the probability density functions (gaussian kernel density estimate) of the subsets of the trainingdata containing plumes and no plumes. Left of the threshold the normalized probability of a random sample to belong to the set of empty scenes is higher than the probability of the sample being a plume, and the other way around for kurtosis values above the threshold. This threshold is used as a very simple classifier and the results are displayed in the confusion matrix shown in Figure 6.1b.



(a) The distributions of the kurtosis values for the subset of plumes and the subset of empty scenes.



(b) Confusion matrix showing the results of the simple Bayesian classifier based on kurtosis.

Figure 6.1: The results of the modified Bayesian classifier on the kurtosis values of training dataset A.

The metrics in the confusion matrix show an example of how accuracy can be misleading as a metric, especially for imbalanced datasets (here containing many more negatives than positives). This basic classifier illustrates how the variation of concentrations in a single sample can be distilled into a single value (the kurtosis) which is still somewhat representative of the original image (this approaches is continued upon in Chapter 7 about Feature Engineering).

This example, however, also illustrates that is not sufficient to only infer whether high concentration enhancements are present in the image (which is represented in the kurtosis value here) it shown it is also important to take the spatial distribution of these high values and the morphology into account. The Convolutional Neural Network is very well suited for this approach.

6.2. Data pre-processing (normalization & augmentation)

Data normalization

Since the background concentration is not equal for the different samples, the samples have to be normalized before they can be compared and used in the CNN. First of all there is the yearly trend in the atmospheric CH_4 mixing ratio, described in Section 2.3.1 and illustrated in Figure 2.5. Since the trainingdata is obtained from observations ranging from early 2018 to August 2020, this effect is not negligible. In addition there are also the seasonal variations and spatial effects both on large scale (dependent on hemisphere) and smaller scale effects (due to weather/atmospheric transport effects and the interaction between sources and sinks (Section 2.3.1)).
Since we are interested in detecting plume signatures, the enhancement in mixing-ratio relative to the the local background is of interest, not the exact xch4 value. Therefore the xch4 channel of the samples is normalized before using it as input to the network.
A logical first attempt was to subtract the prior based on atmospheric simulations by Tracer Model 5, The global chemistry transport model (TM5) [55], which is used in the TROPOMI CH_4 retrieval algorithm (described in Section 2.5.2, original paper [31]) to compute the enhancement and use this as input. However, as the prior is a simulated value computed long beforehand, it is not always representative of the in-situ background level, therefore the enhancement is not always correct and shows strange artefacts.
A better option turned out to be to use the local background of the sample itself, as the 32×32 pixels ($n_H = n_W = 32$) dimensions of each sample usually yield sufficient pixels to compute a local background level by masking out the plume (this is discussed later in Section 7.1.1) and computing the mean of the background (pixels which are not part of the plume mask).

Another problem to be tackled in the input format is missing pixels (Table 2.6, Figure 4.4), represented as NaN-values in the TROPOMI Level 2 CH_4 product. Missing values are known to pose a problem for NN in general, the most common way to deal with missing values is to replace them with the mean of the distribution of the parameter, though this does not make sense in a physical way in the case of atmospheric mixing ratio. Because xch4 samples contain a background level, noise and potentially a plume the distribution is skewed with only one tail, which is the part of interest. Replacing missing value with the mean of the xch4 in the sample would be to assume there are enhanced concentrations at that point which is not clear. This would be correct if the missing values are part of the plume, but this is cannot be known and therefore an interpolation would be unfounded.
The Dropout mechanism (Section 4.3.4) is commonly applied near the deepest layers on a NN. It disables several parameters (pixels in this case) in order to make the network less dependent on the connections of a particular node (Figure 4.10) or Kernel parameter in a CNN. When missing values in the input samples are substituted with zeroes this principle can (to some extend) be applied but then on the input layer, since those particular pixels will pass on no information to the next layer when they are multiplied by the kernel of a convolutional layer.
Substituting missing values with zeroes also makes the most sense from a physical point of view, since the concentration is unknown in those pixels it is best to assume the background concentration. Another option is to substitute it by a (large) negative value, this is also commonly applied, but can cause problems when a large fraction of the input consists of missing values. Both approaches, with several different alterations were tested, an overview of all tested normalization schemes is shown in Table A.2. The normalization scheme which was selected is shown in Table 6.1.

	lowest input	highest input	NaN value after	min value after	max value after
Normalization scheme	mean - std	mean + 100 ppb	0	0	1

Table 6.1: The selected normalization scheme to pre-processing the xch4 channel of the samples. An overview of all normalization schemes is given in Table A.2, normalization scheme N5 is shown in this table.

The normalization scheme listed in Table 6.1 was found to perform best as it preserves information about the absolute enhancement in ppb to some extent, as opposed to the normalization schemes which normalize the maximum value to 1. This normalization scheme therefore maintains the information about high enhancement, next to the relative morphology of the plume. Figure 6.2 shows what the xch4 sample, normalized or pre-processed with the selected normalization scheme looks like.

Data augmentation

In order to artificially increase the amount of trainingdata the samples in the training dataset were rotated by 90°, 180° and 270° [56]. The inputdata format is indifferent to the location on Earth because of the standardized inputformat. As long as the parameter-channels with latitude and longitude are not included and all other parameters are rotated along there is no issue in rotating the samples and applying it as additional trainingdata. In the first version of the model, the shallow CNN, only the samples containing a plume were rotated in order to reach a more balanced dataset A_s (see Table 6.2). In later models the entire dataset was rotated, because a way was found to properly correct for the imbalance of the dataset by using class weights during training.

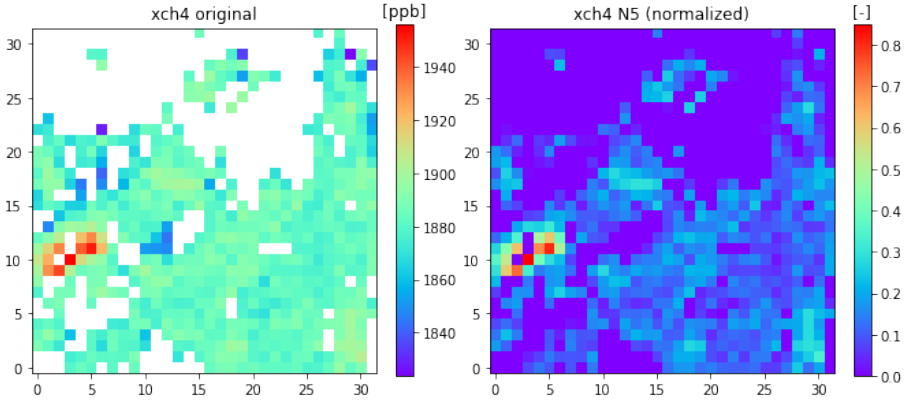


Figure 6.2: Normalization scheme N5 applied to the original xch4 sample. NaN values (missing pixels) are transparent in the left image.

6.3. Design and training of the (shallow) Convolutional Neural Network

Next the CNN can be designed. One of the challenging aspects is the low resolution of the data, some plumes are only 3-5 pixels large. When working with high-resolution images an object of interest within the image, like a bus in a photograph, could easily be made up of hundreds or thousands of pixels. Such a scenario is what CNNs were initially designed for. Koziarski and Cyganek [44] performed a study on the effects of low-resolution imagery in image recognition problems using DL models, they found decreased performance for lower resolution versions of the same images, which indicates low resolution imagery is more challenging to correctly classify.

Name/subset	split	# samples	# positives	# negatives
Original Dataset A (Table 5.2)	-	2542	407	2135
Augmented Dataset A	1.0	10168	1628	8540
- Training set A	0.8	8134	1331	6803
— Training subset	0.8 × 0.8	-	-	-
— Validation subset	0.8 × 0.2	-	-	-
- Test set A	0.2	2034	297	1737
Augmented Dataset A_s	1.0	3763	1628	2135
- Training set A_s	0.8	3039	1331	1708
— Training subset	0.8 × 0.8	-	-	-
— Validation subset	0.8 × 0.2	-	-	-
- Test set A_s	0.2	724	297	427

Table 6.2: Dataset A, split into a Training, Validation and Test set. Dataset A_s indicates the dataset used to train the shallow CNN where only the positive samples were augmented.

With the trainingdata in place and pre-processed, the CNN can be designed.
First Dataset A_s is split into a training, validation and test set. The number of samples in each subset is shown in Table 6.2. In this first CNN only the plumes are augmented, not the negatives, resulting in a ratio of 1628:2135 (Table 6.2). Only the xch4 channel is being taken into account, therefore the shape of the training set A_s is:

$$[m, n_H, n_W, n_C] \longrightarrow [3039, 32, 32, 1] \tag{6.2}$$

where m is the number of samples, and the image size is $n_H = n_W = 32$ pixels with one channel. The channel which is used is xch4 with the normalization scheme which was discussed in the previous section.

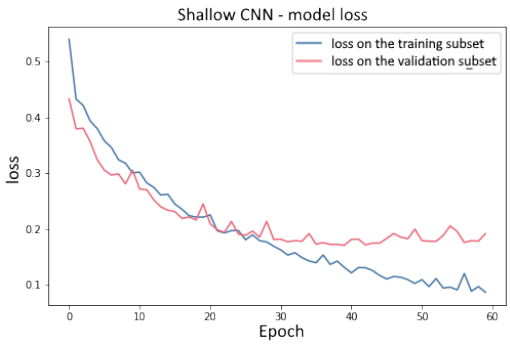
A shallow CNN is designed based on the format of the training data. It consists of two blocks of a single convolutional layer followed by a max-pooling layer (Figure 4.11). No padding is used $p = 0$, the kernel-size K of the convolutional layers is $[3 \times 3]$ (Figure 4.8) and the activation function which is used is the Rectified Linear Unit (ReLU) (Figure 4.7), this is the same for both convolutional layers. The kernel has a suitable size to detect the plume morphology. The first convolutional layer consists of 32 kernels K, and the second convolutional layer has 64 kernels. The $[3 \times 3]$ kernels of the convolutional layers are able to capture the plume-like morphology which takes place at this scale.
The max-pooling layers use a stride s of 2 and a kernel-size (without parameters) of $[2 \times 2]$, resulting in an output image with half the size of the input image ($n_W^{[l]} = n_H^{[l]} = \frac{1}{2} n_W^{[l-1]} = \frac{1}{2} n_H^{[l-1]}$), (Figure 4.9). The output of the second max-pooling layer is flattened to a vector and connected to a single output neuron via a Dense FcNN (Figure 4.10). The activation function used at the output node is the sigmoid activation function (Figure 4.7). During training 50% of the 2305 nodes are temporarily disabled in order to prevent too heavy dependence on a few neuron connections, this is done using the dropout layer (Figure 4.13). The output of the last node is a value between 0 and 1, where 0 indicates a confident prediction of a scene without a plume, and a 1 indicates a confident prediction that the scene does contain plume-like features.

Layername (type)	Output Shape	# Parameters
input_1 (InputLayer)	[(None, 32, 32, 1)]	0
conv2d (Conv2D)	(None, 30, 30, 32)	320
max_pooling2d (MaxPooling2D)	(None, 15, 15, 32)	0
conv2d_1 (Conv2D)	(None, 13, 13, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 6, 6, 64)	0
flatten (Flatten)	(None, 2304)	0
dropout (Dropout)	(None, 2304)	0
dense (Dense)	(None, 1)	2305
Total parameters:		21,121
Trainable parameters:		21,121
Non-trainable parameters:		0

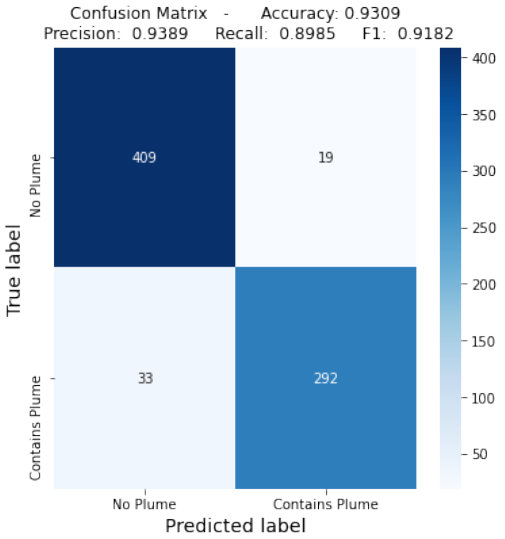
Table 6.3: The architecture of the shallow CNN.

The model is trained on the Dataset A_s. In order to balance the dataset for optimal learning, the augmented version of the postives (each sample 4x rotated) is used combined with the un-augmented version of the negatives. This resulted in a dataset of 1628+2135=3763 samples. At the start the weights of the connections between the nodes and the parameters of the convolutional kernels θ (trainable parameters in Table 6.3) are randomly initiated. Then the model is trained on batches of trainingdata. This is done through optimization of the loss function J (Equation 4.17) by use of forward propagation and back-propagation (Equation 4.18). The metric which is optimized is the loss on the validation set. The trainingset part of Dataset A_s is randomly split into a training subset (80%) and a validation subset (20%) each epoch (Table 6.2). The model is trained for a maximum number of 100 epochs, but halts the training process after no improvement has been observed for 20 consecutive epochs after which the weights are restored to those corresponding to the epoch with the lowest loss J on the validation set. The loss function which is used is binary crossentropy, which is the most suitable option for binary classification problems.

Figure 6.3a illustrates how the model performance improves for subsequent epochs during training. The figure shows how the performance is initially quite poor (both losses are high), due to the randomly initiated model parameters θ, Gradually over the course of several epochs the performance on the training and validation subsets improves because the model parameters are improved. Around epoch 30-40 however the loss on the training subset keeps improving, but the loss on the validation subset starts to increase. This is a sign that the



(a) The training loss and validation loss shown to improve both during the training of the model.



(b) The confusion matrix of the shallow CNN applied to testset A.

Figure 6.3: The results of the training of the shallow CNN on dataset A.

model is overfitting on the training subset, further learning leads to too much specialization on the training subset, which lowers the performance on a general dataset. This difference is clearly visible near epoch 60, which is why the training process is halted and the weights of epoch 38 are restored, since the performance on the validation subset with the model parameters θ of that epoch was optimal.

Next the trained model is applied to the test subset. The test subset of A_s was not used during the training process and therefore is completely new for the model, it is used as an indication of how well the model would perform on new data, which is the objective. The results on the testset are shown in the confusion matrix in Figure 6.3b. This confusion matrix shows that the performance of the shallow CNN on the test set is quite good, much better than the simple Bayesian classifier, as is shown in the comparison in Table 6.4. The vast majority of the samples is correctly classified and an accuracy of 93% is obtained.

model	dataset (size)	TP	TN	FP	FN	Acc.	Prec.	Recall	F1
Bayesian Classifier	A (2034)	208	1753	382	199	0.7714	0.3523	0.5111	0.4173
Shallow CNN	A (753)	292	409	19	33	0.9309	0.9389	0.8985	0.9182

Table 6.4: A comparison of the performance of the simple Bayesian classifier and the shallow CNN.

Figure 6.4 shows another view of the performance of the shallow CNN on the test set, with on the x-axis the predictions score that each sample was assigned by the CNN and on the y-axis the number of occurrences as a histogram. The performance would be perfect if all plumes would have been assigned a score > 0.5, and all empty scenes < 0.5. The false positives and false negatives can be clearly seen in the zoomed-in version of the histogram.

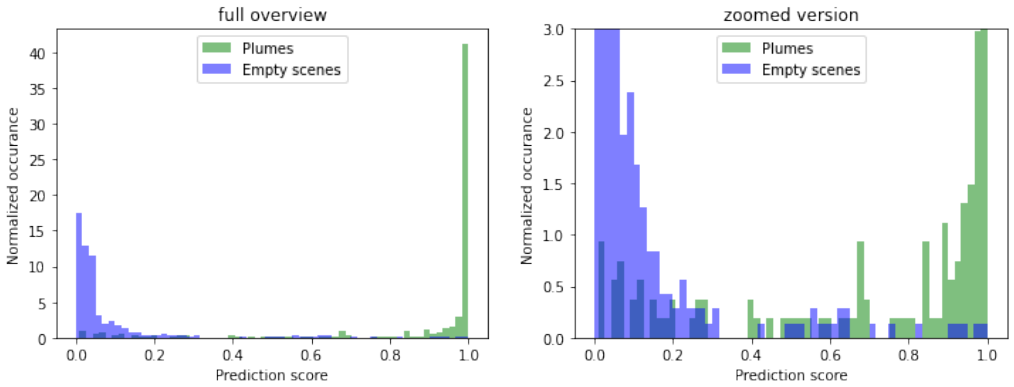


Figure 6.4: Performance of the shallow CNN on testset A, the predictions scores of both classes are shown in a histogram, showing the false positives and false negatives.

It can be concluded that the shallow CNN approach functions well to correctly classify most samples in the test set. The shallow CNN is able to correctly capture the morphology of the plumes and is able to identify the relevant regions (plumes) within the samples. The performance on the test set is good, but the performance should also be investigated on new 'in-situ' data.

Dataset A_s has been created in such a way that the model is able to learn from it how to recognize the morphology of plumes and to extract those features and use them for classification. However the ratio between plumes and scenes without a plume is not realistic for a single orbit of TROPOMI CH_4 data or a full week of measurements. Therefore the performance on a single orbit, which is known in advance to contain just one plume, and on a full week of data is assessed in the following subsection.

6.3.1. Application of the trained shallow CNN to a known single orbit and full week of data

First the trained shallow CNN was applied to a single orbit which was known beforehand to only include one plume. Figure 6.5 shows the four samples with the highest predictions out of this orbit.
This figure shows that the same plume feature is present in four different samples due to the moving-window approach (Section 5.2).
This result shows that the moving-window approach works as intended, because if one of the samples would have been filtered out due to a lack of suitable pixels, or if the plume would have been cut in half by the edge of the sample, the other samples would still be included showing the plume. These results also illustrate that the CNN is indifferent to where exactly in the image the plume is present, it detects it in all four samples with a prediction score of > 99%.
This translation-invariance is one of the most important reasons why a CNN was selected (Section 4.1), since due to the automated sample pre-processing from a full-orbit datafile, it is not known beforehand where in a sample a plume might be present. This translational-invariance arisis due to the same kernels being applied to all of the sample (Section 4.3.1 & 4.3.2). This is not the case for a FcNN consisting solely of fully-connected (dense) layers (Section 4.3.2).

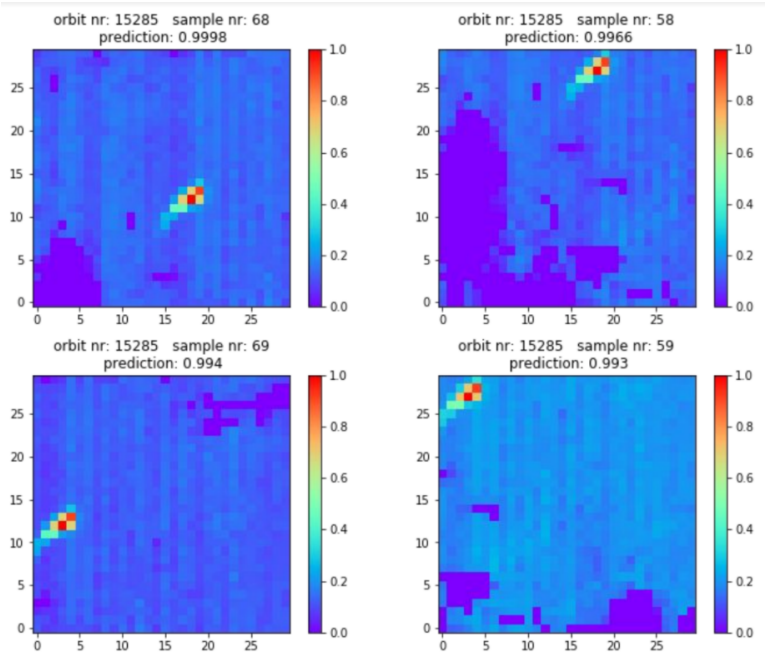


Figure 6.5: The four samples with the highest prediction scores in a single orbit file, showing the moving-window and translational invariance principles.

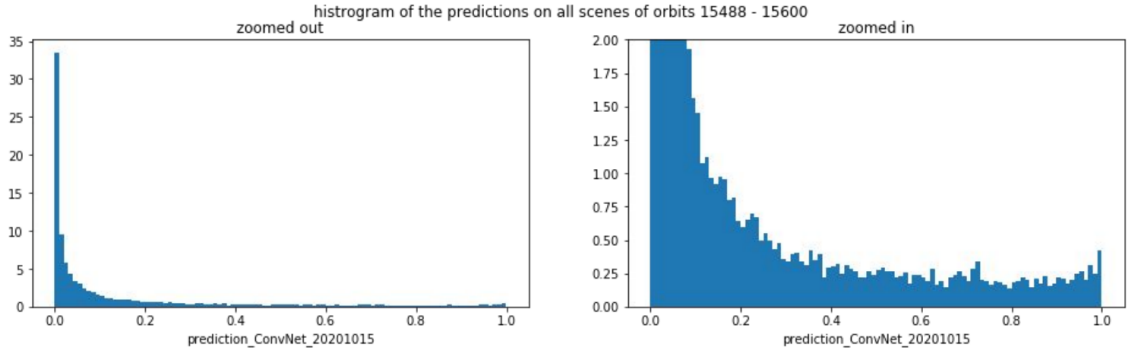


Figure 6.6: A histogram of the prediction scores by the shallow CNN on all samples from a full week of observations. The figure on the right is a zoomed-in version of the histrogram in the left figure.

Next, the shallow CNN was applied to a full week of data (the ratio of plumes versus empty scenes is different for a full week of data then it was for dataset A(or A_s), which makes it important to verify the performance [56]). Figure 6.6 shows the prediction scores on all the samples resulting from those 113 orbits (1 week of measurements).
The 'banana-shape' in this histogram clearly shows what one would expect, most samples being predicted to be negatives, some on which the network is doubtful on in the middle and an increase in occurrence again closer to 1 where the network is confident about the predictions of the samples.
(This analysis was performed again for a later version of the model, which lead to much better performance. This is shown in Figure 6.11.)

6.4. Class Activation Map

In order to get a better insight into which parts of the input image leads to a high prediction by the network the Class Activation Map (CAM) algorithm was applied [57]. A CAM is a lower resolution representation of which parts of the image trigger activations for a certain class. In this case there is only one class (1, plume). The resolution of the deepest convolution or pooling layer (Table 6.3), which is 6×6 in the case of the shallow CNN, is the resolution of the CAM.
A similar approach was also used by Sheng et al. [58] who applied a version of DenseNet & ResNet in order to detect oil&gas facilities in high resolution aircraft remote sensing imagery, also making use of a CAM to identify which parts of an image caused their network to be triggered.
Figure 6.7 shows the CAM for a certain input image. Figure 6.7A shows a xch4 field to which the CAM in Figure 6.7B corresponds. Figure 6.7C is an upsampled version of Figure 6.7B to match the original resolution of the input image. The upsampling has no real meaning and the exact outcome depends on the upsampling algorithm which is applied. Figure 6.7C is added only for more convenient visual matching, it contains values > 1 and < 0 due to the bi-cubic upsampling algorithm.

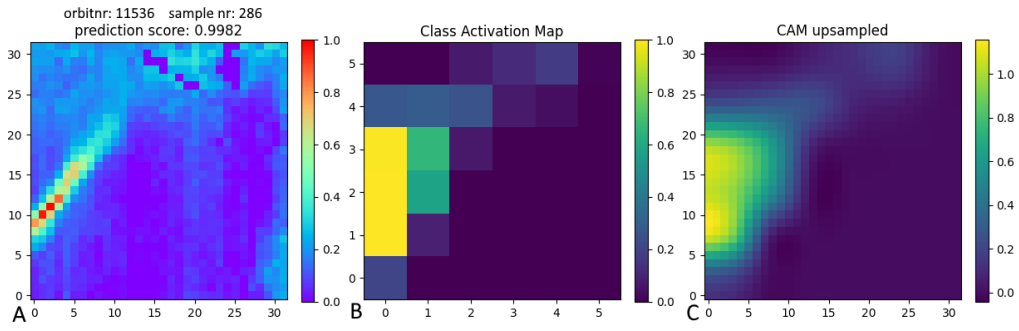


Figure 6.7: The Class Activation Map (CAM) together with the xch4 image it was based on.

The CAM is a useful tool to gain more insight into how a prediction has come into place. A common complaint about CNN and NN & DL in general, is that the predictions are a 'black-box'. This is true to some extent when compared to a hard-coded algorithm, but the CAM provides useful insights into how the CNN came to a certain prediction.
The CAM is also useful when one wants to infer how a false-positive came to be, for example when a sample is predicted to be a plume, but actually is an artefacts. The next section will discuss artefacts in further detail.

Figures A.4 and A.6 show two other examples of what the CAM plot looks like for a specific xch4 field.
The CAM in Figure A.6 illustrates that the model is not triggered by all high values, but only by plume-like morphology in the xch4 field. The enhancements at the left do not trigger the network, only the plume triggered the network. Possibly there might even be a second smaller plume in the right part of the sample, this is best observed in the top-right image. The CAM triggers also on that plume, but not as much as on the main plume.

6.5. Reducing False Positives & False Negatives (artefacts & difficult negatives mining)

The shallow CNN was shown to be functioning properly in the previous sections, performing much better compared to the reference model (Bayesian classifier, Section 6.1) and obtaining an accuracy of over 93% on the test Dataset A_s. But the performance can be further improved.

From the third week of October 2020 onwards, the shallow CNN has been utilised to detect plumes in recent TROPOMI CH_4 data in the context of the TROPOMI-GHGSat project (Section 2.6 & 9.1). These continuous detections were on the one hand useful within the TROPOMI-GHGSat project, but were also used to identify which kinds of plumes the shallow CNN struggled with to properly classify.
The performance on Dataset A can easily be inferred, as all the samples in this (test) dataset were manually labeled beforehand. When the model is applied to 'new' data no labels are available since the samples have not yet been manually inspected and classified. The CNN was developed in order to perform these classifications, but there is no other way to double-check whether the predictions of the CNN are actually correct, other than to visually verify the predictions, which is very time consuming. All that is available is the prediction score assigned to the sample by the shallow CNN (like in Figure 6.6).

Although visual confirmation is quite common in problems like this [58], it is desireable to minimize the need for it. Improving the classification performance comes down to either decreasing the number of false negatives or de-

creasing the number of false positives, since a decrease in either of the two by definition leads to an increase in true positives or true negatives and therefore this consequently improves all of the performance metrics.

False negatives are the most difficult to identify since the vast majority of the samples of a single orbit, or a week of data, are classified as negative (Figure 6.6). For most of those samples, this classification is correct, but for some it might be incorrect, meaning those samples actually do contain a plume. A week of data usually consists of 10,000-15,000 samples after pre-processing. Due to the vast number of samples which are predicted by the shallow CNN to be a negative, it is very difficult to identify the false negatives out of the vast majority of true negatives. In Chapter 7 the algorithms which were developed in order to compute metrics and generate visualizations which could be used to detect false negatives from the set of samples which were predicted to be negatives are discussed.

False positives are easier to identify as only a relatively small fraction of samples from a week of data is predicted to be a positive (right part of Figure 6.6). Therefore it is feasible (though still time consuming) to manually inspect all samples which were classified as positives in order to separate the false positives from the true positives.

Two categories of false positives can be distinguished. One is referred to as difficult negatives because they do not contain a plume-like feature in the xch4 field, but somehow ended up being classified as a positive which means they are difficult for the network to classify correctly. The other is artefacts, these look like a plume based on just the xch4 field, but are actually not a plume. The difficult negatives could in theory be filtered out by an even better model, artefacts can only be identified based on external features, outside of the xch4 field, such as other atmospheric parameters like the aerosol optical thickness or surface albedo [34]. Adding those parameters to the input as additional channels is discussed in Section 6.6.3.

The most predominant types of artefacts are:

- albedo artefacts
- cloud-boundary artefacts
- coastal artefacts
- accumulation artefacts (not actually artefacts, but no emission plumes either)

Two examples of artefacts are shown in Figures 7.4 & 7.6.
Some of the feature engineering algorithms which were mentioned before and are elaborated in Chapter 7 can also help to separate true positives from false positives (difficult negatives and artefacts).

6.5.1. An additional training dataset, focused on separating artefacts from real plumes

In order to be able to better train the model on these difficult negatives and artefacts, an approach similar to difficult negatives mining is used. The most difficult samples out of the entire set are isolated, and the model is trained further with the focus on this subset of samples.

Because separation of true positives (real plumes), difficult negatives and artefacts is feasible by visual inspection, this was done for all TROPOMI CH_4 measurements performed between December 1st, 2020 and January 22nd, 2021. Only the samples which were assigned a predictions score of > 80% and passed filters (discussed in Chapter 8) based on the metrics of the feature extraction algorithms which will be discussed in Chapter 7 were manually inspected. These samples are only the highest scoring subset of all samples from that period of time, just 1267 out of 86,127.
This approach loosely follows the procedure of 'hard negative mining' applied before by Kellenberger et al. [56] who faced similar difficulties with an imbalanced dataset in the context of detecting wildlife in drone footage.

This resulted in an additional dataset which could be used for further training, as well as in the detection of recent plumes which could be used within the TROPOMI-GHGSat project.

Table 6.5 shows the statistics of the samples in this dataset. It is important to realize that this dataset does only contain the plumes (true positives) and the most difficult negatives together with artefacts.
Using the obtained knowledge from applying the shallow CNN to 'new, in-situ' data from the third week of October, 2020 onwards combined with Dataset A and Dataset B, the original model was improved. The next section describes which steps were taken in attempts to improve the model and whether or not this lead to an improvement of the classification performance.

Name/subset	split	# samples	# true positives	# difficult negatives	# artefacts	# doubts
All samples 01-12-20/22-01-21	-	86,217	-	-	-	-
Dataset B total (incl doubts)	-	1267	273	584	216	194
Dataset B total	1.0	1073	273	584	216	-
- Training set B	0.8	858	209	473	176	-
— Training subset	0.8×0.8	-	-	-	-	-
— Validation subset	0.8×0.2	-	-	-	-	-
- Test set B	0.2	215	64	111	40	-

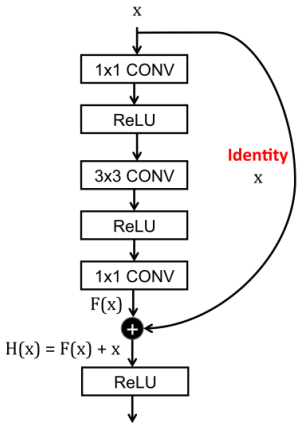
Table 6.5: Dataset B, split into a Training, Validation and Test set.

6.6. Attempts to improve the CNN model

Next, using both Dataset A and Dataset B (which is focused on the most difficult to identify samples of seven weeks of detections) an attempt is made to create a model with improved classification performance. First a complex standard model with many more parameters is tested. Then a custom network is developed. Additional parameters in addition to xch4 are added and at the end of this section the performance of the different developed models is compared.

6.6.1. ResNet

The ResNet50V2 model [59], or in general Residual (convolutional neural) Network (ResNet), with over 23.5 million parameters is one of the very deep DL models [47]. This model is 50 layers deep and is optimized to still function well at this depth using shortcut connections. Deep CNN models and other DL models can be troubled by the vanishing gradient problem, which simply put is that backpropagation does not properly work anymore because the gradients become too small, because of the depth of the model. The main strength of the ResNet model is the shortcut connections which propagates some unmodified information from a previous block to the next, to decrease the effect of this common issue.



(b) With bottleneck

Figure 6.8: The shortcut module of the ResNet or Residual Neural Network shown in a schematic block diagram. [47]

After various tests it was concluded that ResNet is not suitable for this particular problem. The performance on the training set was good, but performance on new data was worse.
However it was not possible to point to a reason for this with certainty, it is speculated that the format of the inputdata (32×32) and the size of the trainingset is just too small for such a deep network (issues with deep networks and low resolution imagery were discussed in Section 6.3 [44]). This lead to overfitting and worse generalization compared to the shallower CNN. ResNet was developed for the ImageNet challenge of 2015, comprising of 1 million images and 1000 well defined and separate classes [47], which is quite different from this problem with just one class which actually is not even binary. For this problem it makes sense a less deep CNN will have better performance as it is better suited to learning the more abstract morphology of plumes in these relatively low resolution images.[44]

6.6.2. Deep CNN

With the knowledge gained from the design of the shallow CNN discussed in Section 6.3 and from the tests with a very deep CNN 'ResNet', a CNN of medium depth is designed. This CNN is referred to as the 'deep CNN' as opposed to the earlier shallower model discussed in Section 6.3.

It is kept in mind that the model should not be too deep in order to maintain the favorable 'generalizability' of the shallow CNN, this is, it should be able to use the features learned from one plume and apply these to detect different, but in some ways similarl plumes without overfitting on the specific details of the plumes of the trainingset.

Layername (type)	Output Shape	# Parameters
input_1 (InputLayer)	[(None, 32, 32, n_C)]	0
conv2d (Conv2D)	(None, 32, 32, 64)	640
conv2d_1 (Conv2D)	(None, 32, 32, 64)	36,928
max_pooling2d (MaxPooling2D)	(None, 16, 16, 64)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64)	36,928
conv2d_3 (Conv2D)	(None, 16, 16, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 8, 8, 64)	0
flatten (Flatten)	(None, 4096)	0
dropout (Dropout)	(None, 4096)	0
dense (Dense)	(None, 64)	262,208
dense_1 (Dense)	(None, 1)	65
Total parameters:		373,697
Trainable parameters:		373,697
Non-trainable parameters:		0

Table 6.6: The architecture of the deep CNN.

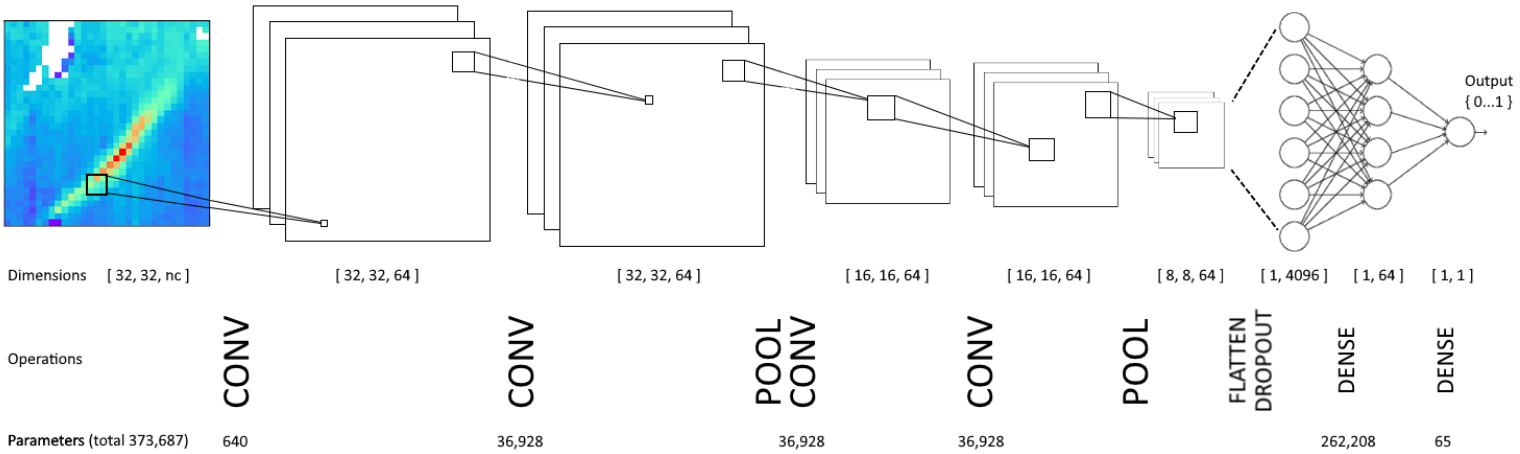


Figure 6.9: A visual representation of the developed "deep" CNN. The different layers are shown as operations with their outputs shown above. The dimensions of the outputs are shown in brackets and the number of parameters (or weights) in the operations are displayed below.

The architecture of the developed model is shown in Table 6.6 and Figure 6.9. The model consists of more convolutional layers and an extra fully-connected layer is added at the end. This allows for more detailed information to be learned by the model, compored to the shallow CNN (section 6.3). Furthermore a zero-padding on $p = 1$ (Section 4.3.1) is introduced in order to maintain the same dimensions in adjacent convolutional layers and to maintain powers of 2 as the image size when pooling. When propagating further down the network, this principle is applied consistently, this was not the case for the shallow CNN. The number of parameters is increased from 21,121 to 373,697, but is still much lower than the 23.5 million parameters of ResNet. With the amount of trainingdata available, this depth of the model is found to be optimal.

The shallow CNN was trained on Dataset A_s where the samples with a plumes were augmented [56], but the samples without a plume were not, in order to create a well-balanced dataset.

Here all augmented samples are used, which leads to an imbalanced dataset, but this is corrected for by applying a class weight [56]. This means the samples belonging to the underrepresented class, plumes, are weighted heavier during the backpropagation

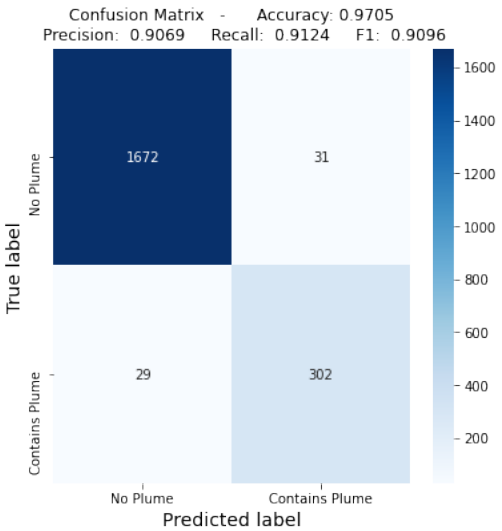


Figure 6.10: Confusion matrix of the performance of the deep CNN trained on normalized xch4.

part of the training phase to compensate for their lower relative abundance.

The model is trained on xch4 samples (n_C=1) with the described normalization scheme. Figure 6.10 shows the performance of the model on test-set A, after training on trainingset A (Table 6.2). An accuracy of 97% was obtained. Figure 6.11 shows the prediction scores on eights days of data in early February 2020. When comparing this to Figure 6.6, the improved performance is clearly visible in the smaller number of samples with a mediocre predictions score.

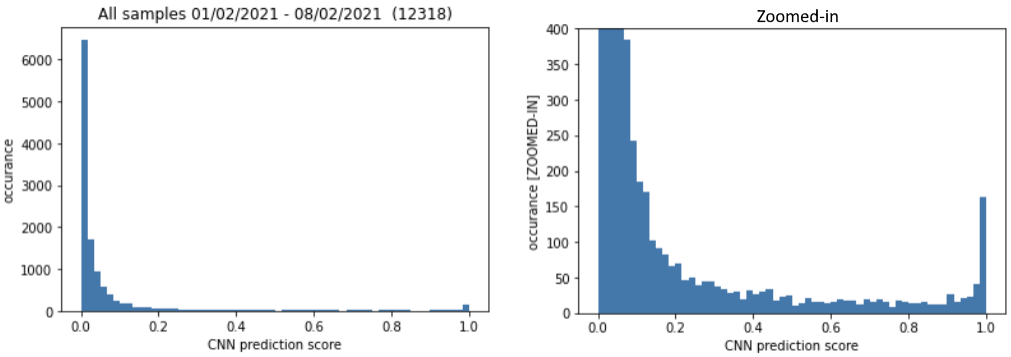


Figure 6.11: A histogram of the prediction scores by the deep CNN on 12318 samples resulting from 8 days of observations in the February 2021.

Later, an attempt was also made to perform a training procedure with similarities to the hard negative mining approach proposed in [56]. First the model was trained on the original dataset A, then for a few epochs the model was exposed to the difficult samples of dataset B. The rationale behind this is that the model first learns to correctly identify the plume morphology as before, and then 'finetunes' itself by learning the subtle differences between plumes and artefacts & difficult negatives of dataset B (Section 6.5.1). Several attempts with adaptations to the approach were made to perform this procedure, but model performance decreased every time. Therefore this approach was abandoned and different ways to correctly classify the most difficult samples were explored.

6.6.3. Utilising additional parameters next to xch4 for training

As discussed in Section 6.5, artefacts can be distinguished from plumes by using additional atmospheric parameters. Artefacts are sometimes remnants of small local inaccuracies in the retrieval algorithm which lead to enhancements in xch4 without actual enhanced concentrations of methane being present.
In order to separate artefacts from real plumes it would make sense to use these additional parameters (see Figure 5.8) as input channels next to (pre-processed) xch4 in order for the network to learn the difference. As this was recognized early in the study, all relevant parameters present in the TROPOMI CH_4 Level 2 dataproduct were incorporated in the standardized sample format.

Several different combinations were tested, of which xch4, surface albedo and aerosol optical thickness is the most promising one. However since the different channels used are multiplied pixelwise at the first convolutional layer (Figure 4.8), as was discussed in the section about the mathematics of the Convolutional Neural Network in Section 4.3, the information in the different channels is not properly utilized to the fullest extent by the model.

Another problem with this approach is that e.g. the entire albedo field itself is not really of interest here, only if it caused an artefact at the location where the model is convinced a plume might be present based on the xch4 field. Therefore using the information of the additional parameters in this way does not work. Another approach where several convolutional parts of CNNs are trained on different channels in parallel and then are concatenated before the fully-connected part of the CNN would face similar challenges.
This leads to the assumption that another approach, dedicated to extracting as much relevant information from a sample as possible and using this in the classification process in order to determine whether a sample does contain a plume or an artefact, separate from the CNN, would be better. This approach is explored in Chapters 7 & 8.
The performance of the different combinations of channels which were tested are shown in the next section in Table 6.7.

6.7. Performance comparison on Datasets A & B

Table 6.7 shows the performance of different models with different combinations of channels used as input. The metrics shown in the table are the same as in a confusion matrix (Figures 6.10)
All models were trained on the training subset of dataset A (Table 6.2) and were then tested on the testset of dataset A, and on dataset B as a whole (since the training part of dataset B was not used for training it can be used for testing as well). The only exception on this is the shallow CNN which was trained on trainingset A_s with only the plume samples augmented, not the negatives (Section 6.6.2). The random split between training and validation data and the random initiation of model weights was seeded in order to allow for comparison between the models developed without differences in the trainingdata which might bias this experimental setup.
Dataset B represents the most difficult to correctly classify samples, this is a subset of all the samples of the seven weeks of observations these samples were taken from. Performance on a dataset containing all samples from that period would be much representative, as all easy to classify samples were omitted now. This could not be done due to the practical limitations of labeling all those samples.

Based on the results in Table 6.7 the deep CNN trained on the channel with xch4 pre-processed with the normalization scheme appears to perform best, but it is difficult to confidently conclude this only based on the classification performance metrics in this table. It stands out though that adding additional channels does not improve performance, as could be expected with this limited number of training samples. Another method will have to be developed to further classify the scenes which were identified to contain plume-like features in the xch4 field. Chapters 7 and 8 will dive into this topic, but first the differences in performance between the shallow CNN, the deep CNN and ResNet are investigated in more detail in the next section.

model	channels	testset (size)	TP	TN	FP	FN	Acc.	Kappa	Prec.	Recall	F1
Bayesian Clsfr.	xch4 (kurtosis)	A_{test} (2034)	208	1753	382	199	0.7714	x	0.3523	0.5111	0.4173
Shallow CNN	xch4 (normlzd)	$A_{S\text{-}test}$ (753)	292	409	19	33	0.9309	x	0.9389	0.8985	0.9182
		B_{total}	273	0	800	0	0.2544	x	0.2544	1	-
Deep CNN	xch4 (normlzd)	A_{test} (2034)	302	1672	31	29	0.9705	0.8920	0.9069	0.9124	0.9096
		B_{total}	577	1148	1024	71	0.6117	0.2764	0.3604	0.8904	0.5131
Deep CNN	xch4 (normlzd N6)	A_{test} (2034)	305	1664	39	26	0.9680	0.8846	0.8866	0.9215	0.9037
		B_{total}	562	983	1189	86	0.5479	0.2003	0.3210	0.8672	0.4685
Deep CNN	xch4, albedo, AOT	A_{test} (2034)	300	1663	40	31	0.9651	0.8733	0.8824	0.9063	0.8942
		B_{total}	559	1040	1132	89	0.5670	0.2182	0.3306	0.8627	0.4780
Deep CNN	xch4, albedo, AOT	A_{test} (2034)	278	1676	27	53	0.9607	0.8509	0.9115	0.8399	0.8742
	wind E, wind N	B_{total}	440	1254	918	208	0.6007	0.1852	0.3240	0.6790	0.4387
ResNet	xch4 (normlzd)	A_{test} (2034)	257	1686	17	74	0.9553	0.8236	0.9380	0.7764	0.8496
		B_{total}	471	1443	729	177	0.6787	0.3012	0.3925	0.7269	0.5097

Table 6.7: A comparison of the performance of the simple Bayesian classifier and the shallow CNN and the deep CNN for different combinations of channels as input.

6.8. More detailed comparison between best performing model architectures

It is difficult to conclude from Table 6.7 which CNN performs best. It did stand out though that a CNN trained only on the xch4 field performs better compared to additional parameters.
It looks like the Deep CNN performs best, but differences in performance on the test set of Dataset A & B are quite small. Therefore a more detailed comparison is required to be able to confidently select the optimal CNN.

All samples resulting from TROPOMI data in a timespan of two weeks, 2020-12-25/2021-01-07, (which is a subset of Dataset B) were analysed and classified using the shallow CNN, deep CNN and ResNet. This timespan was selected for convenience as it had been mostly manually labeled already in the context of the TROPOMI-GHGSat project. The samples which were flagged as plume by any of the networks was manually inspected and labeled as either "plume", "artefact" or "scene without plume / empty". Some samples were identified as a plume by each of the CNNs, others only by one or two. This allowed for constructing a Venn diagram, which clearly shows the overlap between the predictions of the different CNN models. The Venn diagram is shown in Figure 6.12. With this approach it is not certain that none of the 21,803 samples which were not classified as a plume by any of the networks actually does contain a

plume in fact, but for comparison between these models this does not matter. The performance metric investigated here is Recall.

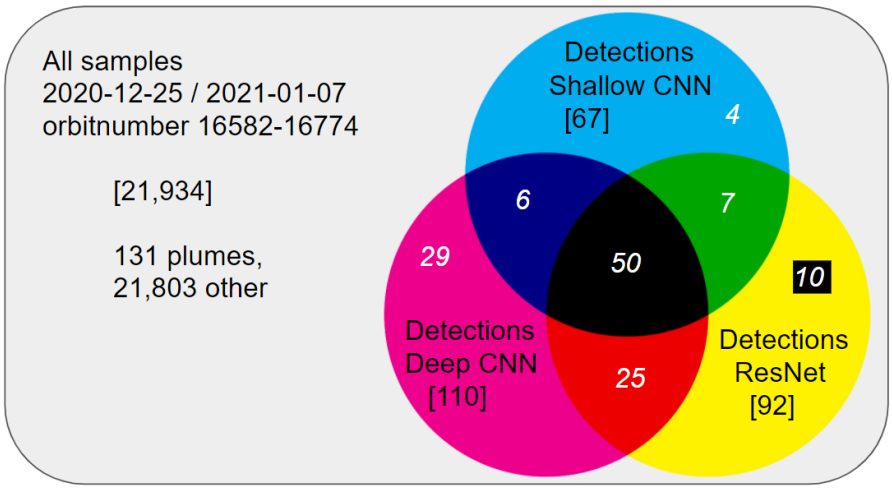


Figure 6.12: A Venn diagram displaying the number of plumes detected by each of the three models, and the overlap between the models. The number in black between brackets is the total number for that model (color), the number in white is the number of plumes detected exclusively in that part of the Venn diagram. The shapes are not to scale.

It is also relevant how many false positives result from each model, as this number should be as small as possible (high precision). This is more clearly illustrated in a separate confusion matrix for each model as shown in Figure 6.13. In the confusion matrix the samples which were labeled as empty or artefact were also included which allows for the precision to be computed. The accuracy is also computed but is meaningless since there are 21,485 samples (assumed to be all true negatives) which are not being taken into account here when computing the accuracy. When taking these into account (and if they truly are negatives) the accuracy is >99% for each model. Therefore those are omitted to highlight the subtle differences. The precision is unaffected by the omission of these samples, as these missing samples were not classified as plumes by the model. In the Venn diagram only the samples in the second row, which were labeled as plume by any of the models, are included. The column with artefact predictions is empty, as the CNNs only make a distinction between "plume" and "no plume / empty". In computing the metrics the "empty" and "artefact" cells are grouped, as the classes need to be binary.

Both Figure 6.12 and 6.13 show that the performance of the Deep CNN is best. Both the precision and recall of the Deep CNN are better compared to those of the other models. These results also show that however the Deep CNN is quite good at detecting plume-like features, there are still quite a lot of false positives. The next two chapters aim to address this issue. Since the CNN only utilises the xch4 channel, a lot of additional information can be extracted from each sample from the additional parameters.
The fact that the recall is much better than the precision is a good thing, as false positives can still be filtered out later in the pipeline, but false negatives are lost. This is part of the design.

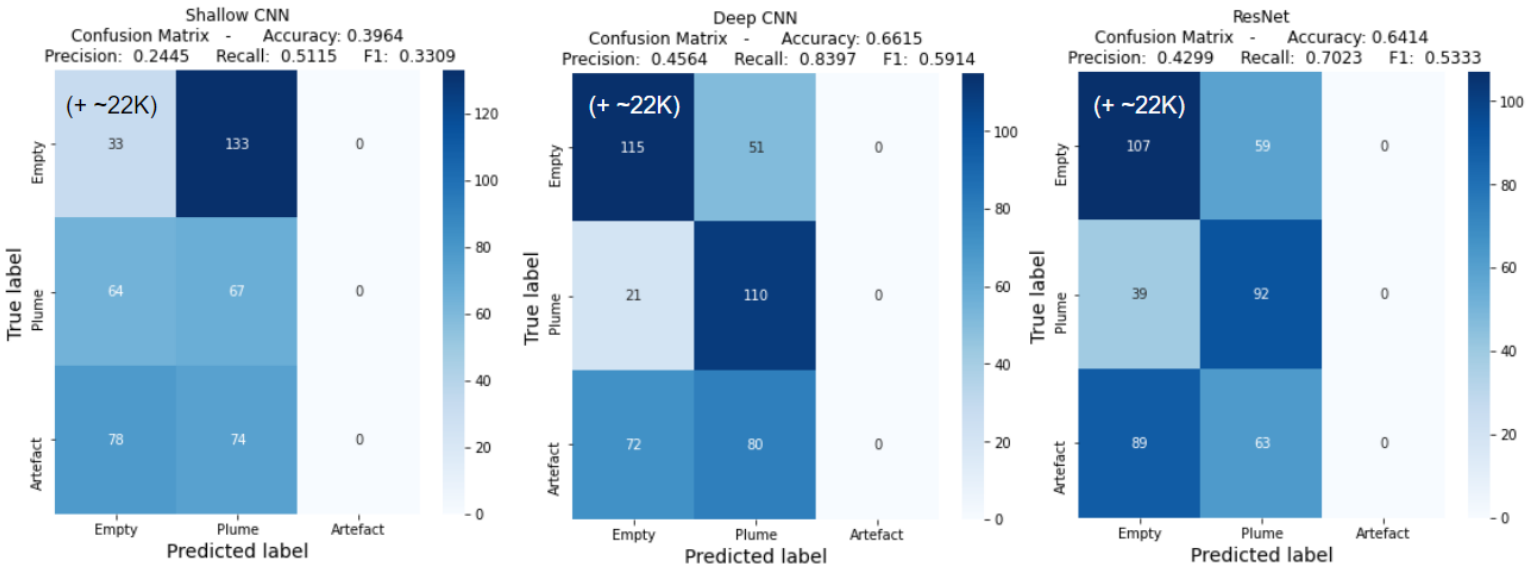


Figure 6.13: The confusion matrices for the three CNNs applied to all samples for which any of the models has predicted it to be a plume. The true negative cell actually contains a lot more samples.

7
Feature Engineering

Samples which are flagged by the CNN to contain a plume can be manually double checked, but it is preferable to automate this process as much as possible since this is infeasible to perform manually for large datasets.
The standardized image size of 32×32 pixels ($n_H = n_W = 32$) allows for the application of automated algorithms which compute useful additional information and metrics (feature engineering) based on a single sample.
The feature engineering algorithms were developed to:

- Identify the plume from the background using a plume mask
- Estimate the pixel most likely to be the source of emissions based on the plume mask and windfield
- Make an initial rough estimate for the quantification of emissions
- Infer correlations between xch4 and the albedo, aerosol optical thickness & surface pressure fields in order to detect possible artefacts
- Infer how influenced a plume could be by adjacent cloudy pixels, as this is the predominant type of artefact
- Derive the angle of the principal eigenvector of the plume (elongated direction) and the mean windvector

7.1. Emission quantification estimate

7.1.1. Plume mask

The goal of the plume mask is to identify the pixels which are part of the plume from the background. This is both useful for the other algorithms as well as for visual inspection and it could potentially be used in the future as the mask for a pixel-based NN approach.

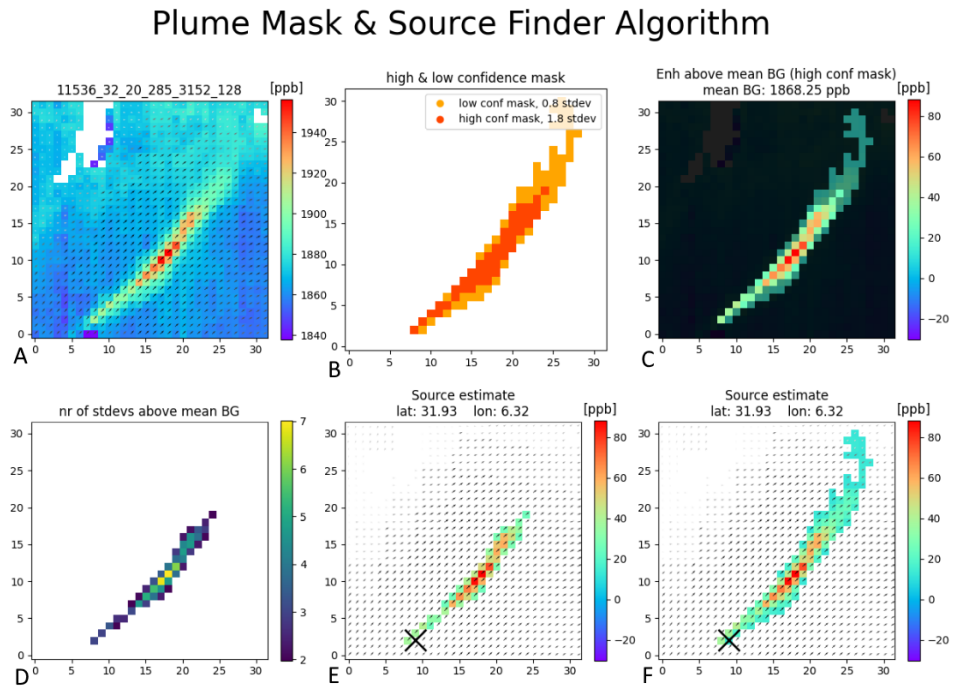


Figure 7.1: The output of the plume mask and source finder algorithms.

The main assumption of the automated plume mask algorithm is that the pixel with the highest enhancement is assumed to be part of the plume. The plume mask originates at the pixels with the highest xch4 value and dilates outwards, with a step of 1 pixel in each direction, only adding pixels with a value above the high confidence threshold. After a study on a limited amount of samples from the trainingset, a suitable value for the high confidence threshold was found to be 1.8× the standard deviation of all xch4 values in the sample. A suitable value for low confidence threshold was found to be 0.8× the standard deviation. This dilation is iterated until the high confidence mask no longer expands. In the next iteration the threshold is lowered to the low confidence threshold and the dilation continues until this converges as well. For almost all applications only the high confidence mask is used, but the lower confidence mask adds information about which adjacent pixels with slightly lower concentrations could also be part of the plume. The only exception is the source location estimate, which performs better results in general when the low confidence mask is utilesed.

The main advantage of this approach is that noise pixels, randomly high-valued pixels which are not part of the plume, are not included in the mask. The spatial proximity criterion makes sense, given the physical continuity processes involved in plume dispersion.
An example of the input of the algorithm in shown in Figure 7.1A, showing the atmospheric methane mixing ratio and the windfield as quivers indicating direction and magnitude. The high and low confidence masks which are the output of the algorithm are shown in Figure 7.1B and in C presented in a different way.

7.1.2. Source finder

The next algorithm is the Source Finder algorithm which estimates the most likely pixel to contain the source of the plume given the plume mask and the wind-field. The output of the source finder algorithm for a trivial windfield is shown in Figure 7.1. In this section the algorithm will be elaborated based on a scenario with a more challenging plume mask and wind-field.

Figure 7.2A & B show a scene with a complex windfield and the high & low confidence plume masks. Figure 7.2C shows the current estimate (second iteration) of the source as the black pixel in the middle of the plume and the pink pixel denotes the true source, which is known in this case since this scene is one of the training sample. The source estimate started at the pixel with the highest xch4 value.
Figure 7.2D shows how the algorithm iterates. The gridded along-orbit vs. across-orbit format is dropped and instead each pixel of the low confidence plume mask is plotted in latitude-longitude space. Originating from each pixel is the windvector at that pixel. The low confidence mask, instead of the high confidence mask was selected since the source pixel was often located outside of the high confidence plume mask, which resulted in better performance using the low-confidence mask. The problem is approached in the latitude-longitude space since this closely resembles the most realistic scenario of using a 2d domain where distances along the surface of the Earth would be measured in meters. In the regional scope of an image size of 32 × 32 pixels ($n_H = n_W = 32$), for latitudes [−60° < lat < 75°] the error caused by this approximation is limited.

In Figure 7.2 the current pixel is shown in red, and the eight adjacent pixels (if they are part of the plume mask) are shown in blue. As methane in a plume is transported downwind, the source is most-likely located upwind from the highest enhancements within the plume. Therefore the algorithm iterates upwind from the pixel with the highest xch4 by computing a line segment, which is directed 180° opposite from the wind direction. The distance from each of the eight pixels adjacent the current pixels to the line segment is computed. The line segment is given a slight offset upwind in order to favour upwind pixels. The pixel located closest to the line segment (except for the current pixel) is selected as the next pixel. This process is iterated until convergence.
This methodology allows for some limited sideways movement if no upwind pixels are available. This methodology performs well for most windfields and plume masks, but it is however prone to local minima. The latitude and longitude of the optimized source pixel estimate can be used as a startingpoint for further analysis.

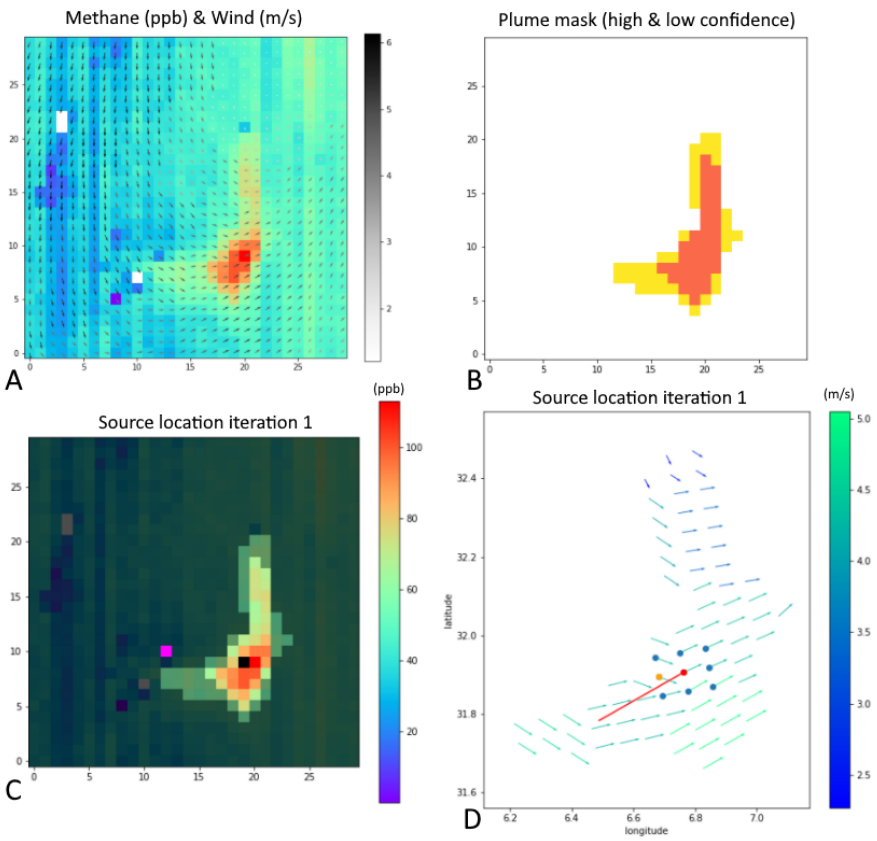


Figure 7.2: Figures illustrating the way the source finder algorithm works, both in along-orbit vs across-orbit space, and in latitude-longitude space in the background.

7.1.3. Integrated Mass Enhancement (IME) emission quantification estimate

The plume mask and source finder are useful on their own, but they can also be used as inputs for a third algorithm which estimates the CH_4 emissions in the scene. This is done using the Integrated Mass Enhancement (IME) method. This method was originally proposed within the field of CH_4 plume quantification by Frankenberg et al. in 2016 [60], and described in more detail by Varon et al. in 2018 [24].
Next to the IME method, other methods exist to estimate total emissions and source rates such as the Gaussian plume inversion method, the source pixel method and the cross-sectional flux method [20]. Those algorithms require custom tuning and/or are not possible to use with only the data present in a sample, therefore the IME method was selected.

The IME method relates the total enhancement of CH_4 mass in the plume, relative to the background, to the source rate based on the time methane spends within the plume. First the column mass enhancement is computed.

$$\Delta\Omega = \frac{M_{CH_4}}{M_{\text{air}}}\ \Omega_{\text{air}}\ \Delta X \tag{7.1}$$

where $\Delta\Omega$ denotes the column mass enhancement [kg m^{-2}], M_{CH_4} is the molecular mass of methane [kg mol^{-1}], M_{air} is the molecular mean mass of dry air [kg mol^{-1}], Ω_{air} is the column of dry air [kg m^{-2}] and ΔX [ppb] is the enhancement relative to the local background X_b [ppb]. ΔX is found by subtracting X_b from the column average dry molar mixing ratio xch4 [ppb], $\Delta X = \text{xch4} - X_b$. [20]

Equation 7.1 can be modified such that the parameters in the standard sample can be used:

$$\Delta\Omega = \frac{M_{CH_4}}{M_{\text{air}}} \cdot \Omega_{\text{air}} \cdot \Delta X = M_{CH_4} \cdot \Delta X \cdot \frac{\Omega_{\text{air}}}{M_{\text{air}}} = M_{CH_4} \cdot \Delta X \cdot N_{air} \tag{7.2}$$

where N_{air} is the number of moles of air in a column from the surface to the top of the atmosphere per m^2 [mol m^{-2}].

Next the equation for atmospheric surface pressure is used.

$$P = \frac{F}{A} = \frac{m \cdot g}{A} \longrightarrow m = \frac{P \cdot A}{g} \tag{7.3}$$

Where P is surface pressure [Pa = kg m^{-1} s^{-2}], F is the force exerted by the atmosphere on the surface [N], m is the mass of the atmosphere over the surface [kg], A is the area [m^2] and g is the gravitational acceleration 9.80665 [m s^{-2}]. If we set A to 1 m^2, the mass becomes the mass per unit of area.
Next we divide this mass per area by the molecular mean mass of air M_{air} in order to arrive at N_{air}. We first define $A = 1\text{m}^2$ to convert the equation to per unit of area.

$$N_{air} = \frac{m}{M_{air}} = \frac{P \cdot 1}{M_{air} \cdot g} = \frac{P}{M_{air} \cdot g} \tag{7.4}$$

Next we combine Equations 7.2 and 7.4 which results in:

$$\Delta\Omega = M_{CH_4} \cdot \Delta X \cdot N_{air} = M_{CH_4} \cdot \Delta X \cdot \frac{P}{M_{air} \cdot g} \tag{7.5}$$

In Equation 7.5, only P and ΔX are variables, the other parameters are constants. The surface pressure P can be directly obtained from the sample and the xch4 enhancement ΔX above the background concentration can be obtained from the xch4 field. This is done by computing the mean background concentration which can be found by applying the plume mask in order to determine which pixels are part of the background. Therefore this equation can be computed

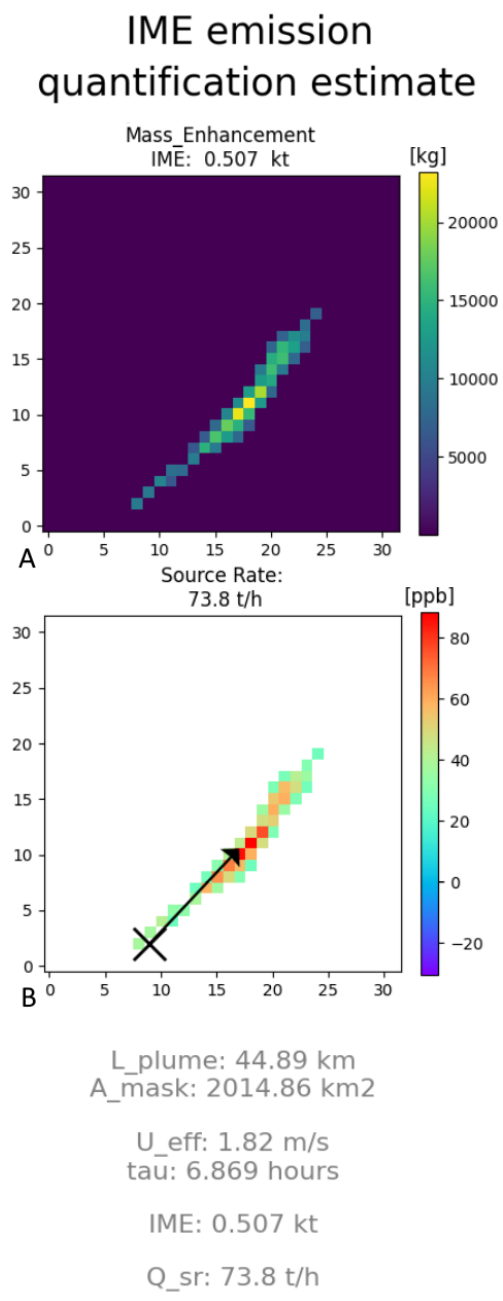


Figure 7.3: Output of the IME algorithm, showing a visual representation and the most important metrics such as the total mass of enhanced methane [kt] and the source rate [t/h].

per pixel of the plume mask in order to compute the CH_4 mass enhancement per m^2.

Next the Integrated Mass Enhancement (IME) for the entire plume is computed, taking into account the surface area per pixel by applying:

$$\text{IME} = \sum_{j=1}^{N} \Delta\Omega_j A_j \tag{7.6}$$

where N is the number of pixels which are part of the plume mask and $j \in [1,2,\ldots,N]$. The area per pixel which is also part of each sample is used here. The total CH_4 mass enhancement of the plume can be related to the source rate Q [kg s^{-1}] using the residence time τ [s] of CH_4 within the plume as Q =IME τ^{-1}. [20]

τ can be expressed as $\tau = L\, U_{eff}^{-1}$ where L is the plume size [m] and U_{eff} is the effective wind speed within the plume [m s^1]. Combining those expressions with Equation 7.6 results in:

$$Q = \frac{1}{\tau}\text{IME} = \frac{U_{\text{eff}}}{L}\text{IME} = \frac{U_{\text{eff}}}{L}\sum_{j=1}^{N} \Delta\Omega_j A_j \tag{7.7}$$

This equation assumes uniform transport of the methane in the plume to terminal distance (at the end of plume length L). The actual mechanism for plume dissipation is turbulent diffusion. This process takes place in all directions, not just in the downwind direction along L, however it is a reasonable assumption commonly applied in literature [20] [60].

The plume length L can be estimated by $L = \sqrt{A_M}$ where A_M [m^2] is the area of the total plume mask [20]. In the case 'cigar-shaped' plumes of the plume shown in Figure 7.3, this leads to an underestimation of L and thus an overestimation of Q.
U_{eff} is more difficult to estimate. In [20] a relationship between U_{10} [m s^{-1}] and U_{eff} [m s^{-1}] is found, where U_{10} is the windspeed at the source at an altitude of 10 meters above the surface, as this windspeed is most representative for plume transport through the atmosphere. In this study, the U_{10} parameter is obtained from the GEOS-FP dataset, which is available globally each hour, but in a relatively low resolution (see Section 5.2). In [20], high resolution synthetic data is used, whereas this study uses low resolution measured data. The relation found between U_{10} and U_{eff} by [20] is used which is:

$$U_{\text{eff}} = \alpha_1 \cdot \log U_{10} + \alpha_2 \tag{7.8}$$

where $\alpha_1 = 0.9$ [-] and α_2 is 0.6 [m s^{-1}]. These values were found using Large Eddy Simulations (LES), and are therefore also used in this study. Varon et al. use the windspeed at the source location, but since here we are dealing with much larger areas, a weighted average (by area) of the wind speeds in all pixel which are part of the plume is used instead, since this is more representative for large-scale transport phenomena.

These three algorithms provide a first estimate of the emissions in a sample. Since this is an automated method it is sensitive to errors in the plume mask and/or inaccuracies in the GEOS-FP windfield. A detailed study for a particular emission-event will result in a more accurate estimate. These results should therefore be interpreted as a first estimate indicating the order of magnitude of emissions. The clearer the plume signature in the sample, the better the estimates. Figure 7.3 shows the output of IME algorithm applied on the sample which was shown earlier in Figures 7.1.

In Section A.1 in the Appendix, the steps which were taken to verify the outcomes of the automated IME algorithm against a thoroughly studied case [21] are described.

7.2. Additional filtering

In this section we define additional filters to detect artefacts in the data that are incorrectly flagged as plumes by the CNN. We can perform more sophisticated filtering (not pixel based, like was described in Section 5.3), but sample-wide taking into account the meaning of different adjacent pixels and multiple parameters. This can be used in order to filter out artefacts which solely based on the xch4 field might look like real plumes. Three algorithms were developed to this end, which will be elaborated in the following sections.

7.2.1. Correlation metrics

In order to detect artefacts without visual inspection it can be useful to infer the correlation between the xch4 field and the most important retrieval parameters. In order to do this the Coefficient of determination, denoted as R^2, between xch4 and another parameter is computed for different subsets of pixels in the sample relative to the high confidence plume mask.

Figure 7.4 shows a sample, the xch4 field in subfigure C shows some enhancements near coordinates [10, 20], which based solely on the xch4 field looks like a plume. When compared to Figure 7.4D though, it turns out the area with enhancements coincides with an area of lower values in the albedo field. Subfigure B shows five categories of pixels, where red is the high confidence plume mask, green is the low confidence plume mask, orange is one dilation around the low confidence mask, brown is two dilations and grey is the background.

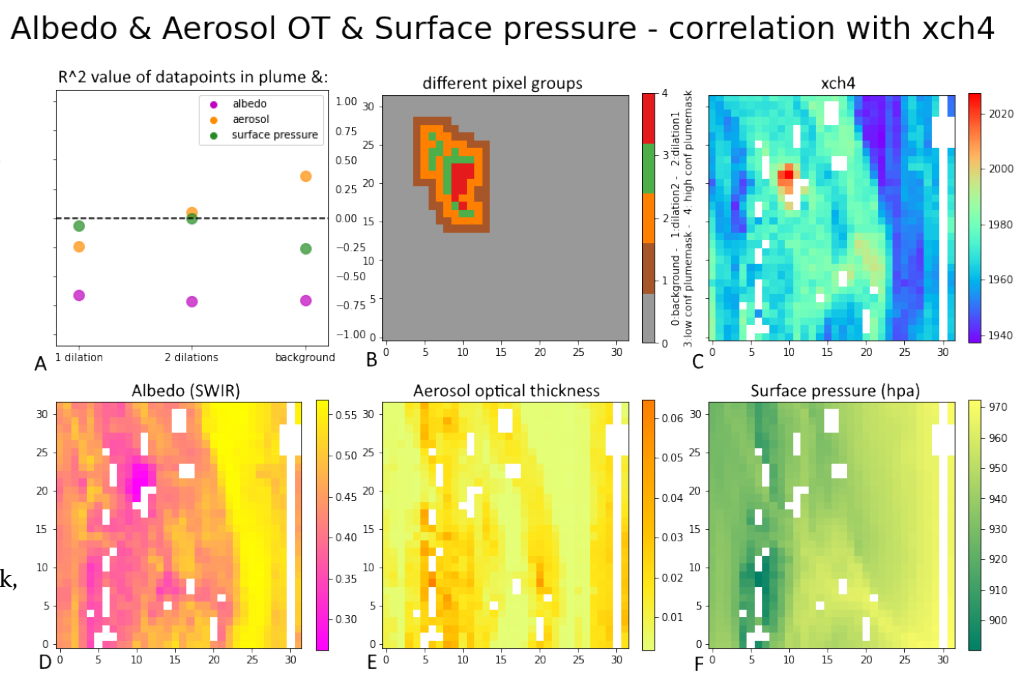


Figure 7.4: A visual summary of the correlation metrics between xch4 and the most important atmospheric parameters for artefacts.

Both the xch4 and albedo fields are separated into these categories and linear regressions are performed for the pixels in the categories:

- high + low confidence plume mask & 1 dilation
- high + low confidence plume mask & 2 dilations
- high + low confidence plume mask & the entire background

The results of these linear regressions between the xch4 and albedo subsets of the sample and their coefficient of determination are shown in Figure 7.5. The coefficient of determination is displayed in the title, and also shown in Figure 7.4A as the three pink dots. The same procedure is applied for the aerosol optical thickness and surface pressure parameters. Varying the part of the background with which the plume is compared can give insight in whether the correlation effect is local around the plume, or exists throughout the entire image. Here the values are similar for the three subsets, but this is a coincidence due to the low xch4 / high albedo band in the right part of the sample. Generally speaking a strong correlation around the plume, but a lower correlation throughout the entire sample signals an artefact.

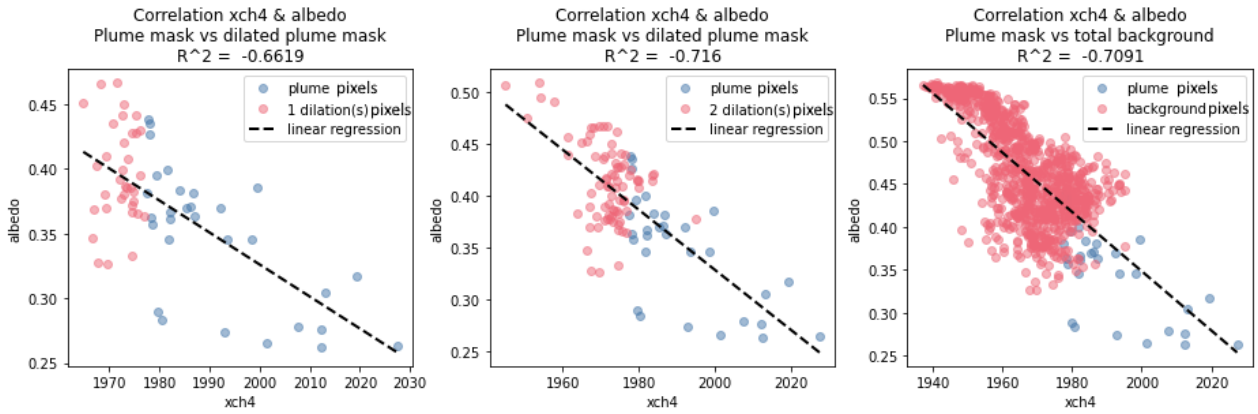


Figure 7.5: Scatter plots of the three different subgroups of the background, plotted together with the pixels which make up the plume. The plume subset (blue) is the same in all three figures, only the amount of background that is taken into account, together with the plume pixels, is varied.

7.2.2. Cloud boundary artefact metric

The dominant type of artefacts is the cloud boundary artefact, where high xch4 values are located close to a cloud (Figure 7.6A&B). These pixels are not filtered out since at the particular pixel the cloud fraction and aerosol optical thickness (both SWIR and Near infrared (NIR)) values in the TROPOMI CH_4 dataproduct do not indicate clouds. This is probably due to inaccuracies in the cloud product and this results in higher values in the xch4 field which are most likely artefacts.
Another reason these artefacts to not cause a high R^2 value is that pixels with an aerosol optical thickness of over 0.10 are filtered out. Computing the correlations has limited use to filter out cloud boundary artefacts as the pixels which would indicate the presence of clouds (high aerosol optical-thickness values) are filtered out. This calls for another approach.

Instead the cloud fraction parameter is used in combination with xch4 since this parameter does contain data on pixels where the xch4 is filtered out. This can be particularly useful to detect whether a missing pixel in the xch4 field next to an enhanced value was filtered out due to cloudiness or not. If this is the case, the enhancement in the xch4 field might be less trustworthy.

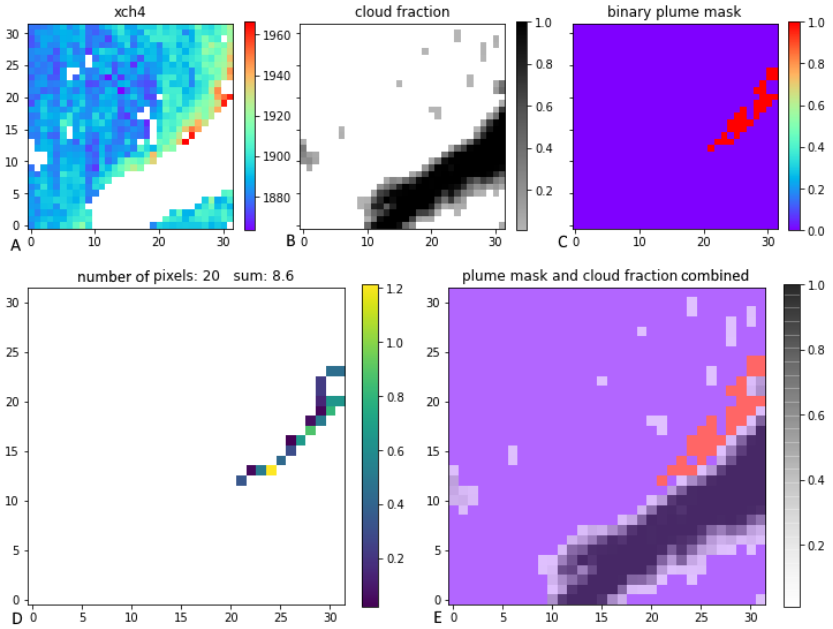


Figure 7.6: The inputs and outputs of the cloud boundary algorithm.

The developed procedure is visualized in Figure 7.6. Subfigures A and B are layers of the sample format, the binary high confidence plume mask is shown in subfigure C. First the cloud fraction image is zero-padded with $p = 1$, that is a band of zeroes is added to the edges of the image, making the dimensions ($n_H = n_W = 34$). Next for each pixel $pm_{i,j}$ of the binary plume mask ($n_H = n_W = 32$) the eight adjacent pixels, and the pixel itself, of the cloud fraction image are selected.
Next the sum of the product of the plume mask pixel with each of the nine cloud fraction pixels is computed. If the plume mask value is 0 (indicating "no plume"), the output will be 0. If the plume mask value is 1, but all 9 adjacent cloud fraction values are 0 (indicating "no cloud"), the output will also be 0. But if the plume mask value is 1 and not all adjacent cloud fraction values are 0, then the output will be > 0 indicating the pixel in the plume mask is adjacent to a cloudy pixel.
Since the cloud fraction values are a floating number between 0 and 1, the magnitude of the output also serves as

indication of how cloudy the scene is.
The output of the example is displayed in Figure 7.6D, which shows cloud adjacency in the correct pixels when visually verified with subfigure E. The number of pixels and the sum of values in the output (and thus indirectly also the average) are stored and can be used as a metric to describe how much a plumemask might be affected by a cloud.
However adjacency to a cloud does not always mean the enhancement is an artefact instead of a real plume, when moving to a more automated workflow, it can be useful to put limitations on how much adjacency to clouds there can be in order to reduce the number of false positives. In the scenario of fully automated detections a false positive might be worse than a false negative.

7.2.3. The angle between the mean wind direction & the principal eigenvector of the pixels in the plume mask

Plumes are generally elongated in the downwind direction, as this is where the Methane is transported from the source. Mathematically speaking, this can be inferred by computing the angle between the mean windvector and the principal eigenvector of the plume mask.

This principle is clearly illustrated in Figure 7.7. Figure 7.7A shows a xch4 sample with the background shaded. Figure 7.7B shows the pixels of the plume mask, plotting in latitude-longitude space together with the windvectors of each pixel. The eigenvector has to be computed in latitude-longitude space as the along-orbit & across-orbit space is not area equal and is therefore distorted, this effect is visible when comparing subfigures A and B.
The eigenvectors of the pixels of the plume mask in latitude-longitude space are computed using a Principal Component Analysis (PCA) which can be used to compute the mean and the eigenvectors of a dataset. In Figure 7.7C the results are plotted on top of the same pixels as shown in Figure 7.7B. The red line indicates the principal eigenvector, the blue line indicates the secondary eigenvector which is perpendicular to the principal eigenvector. The mean of the dataset is indicated with the orange X. The mean windvector of the pixels of the plume mask is shown as the orange line.
Next the angle between the principle eigenvector and the mean windvector can be computed. For real plumes a small angle is expected. Larger angles are expected to occur for cloud boundary artefacts (The artefact in Figure 7.6, has an angle of over 60°since the mask does is not coherent with the windfield) and accumulation artefacts which are shaped by topography and not the windfield.
As addition a weighted version of the PCA is computed as well, where the pixels in the plume mask are weighted by the enhancement in ppb above the background concentration. In most cases the difference is relatively small. The black line in the plot indicates the wind vector at the estimated source location which was described earlier, but this is for visualization purposes only.

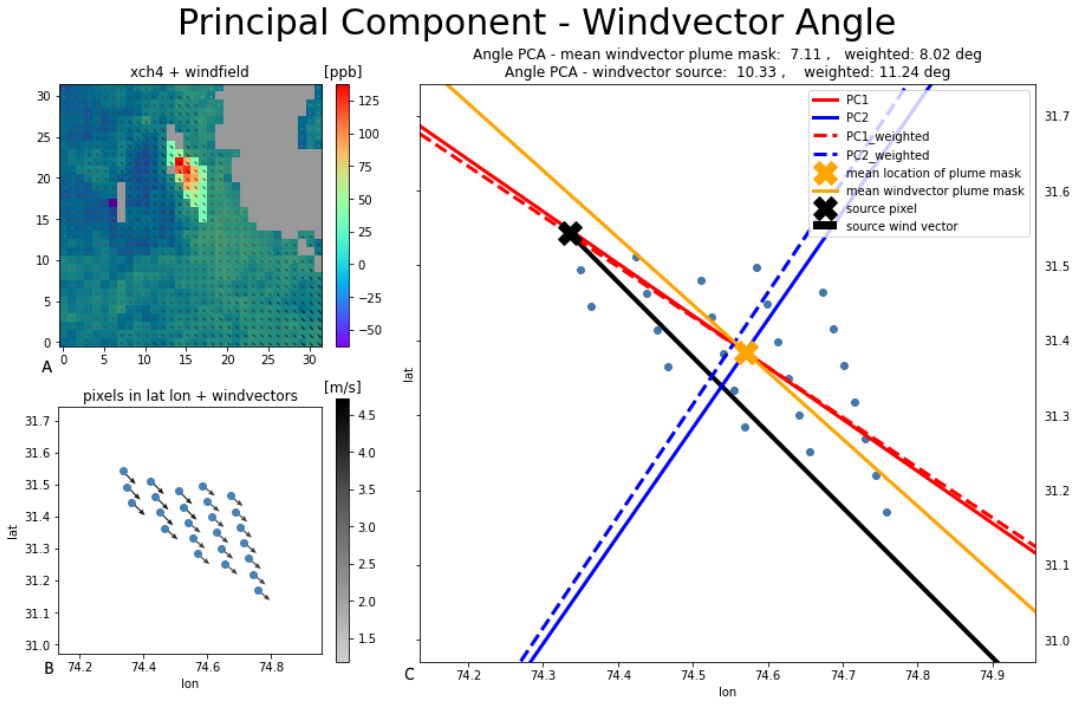


Figure 7.7: The output of the principal component windvector angle algorithm. Figure A shows the input sample, Figure B shows the pixels in latitude-longitude space and Figure C shows the mean windvector and the eigenvectors of the pixels in the plume mask.

8
Classification of plume-like samples based on metrics

8.1. Automating manual classification of promising samples using a fully-connected neural network

In Chapter 7 the various feature engineering algorithms which were developed were elaborated. Based on the classification by the CNN (ranging from 0 to 1, indicating how confident the network is that plume-like morophology is present in the xch4 field of the sample) and the figures from the feature engineering algorithms, it is possible to visually determine whether or not a sample does contain a plume or not. (Figure A.4 + A.5 & Figure A.6 + A.7).
The feature extraction algorithms are, next to the visualizations shown in Chapter 7, used to compute metrics for each sample. A few examples of those metrics are the number of pixels in the plume mask, the estimated source rate [t/h] and the correlation between the xch4 and surface albedo field near the plume mask. Some of these metrics can be used to confidently filter out some samples based on a single metric. The first and foremost is ofcourse the prediction score of the CNN. Others are a low source rate, little pixels in the plume mask or a very high correlation between albedo and xch4 near the plume mask.

These are some filters which can be applied based on a single parameter, but higher dimensional filters are also a possibility. This chapter describes how this last manual step where one has to visually inspect the samples is automated using a FcNN, which is trained on Dataset B (Table 6.5).
The metrics which are used are listed in Table 8.1. For each sample, this results in a vector containing 28 numerical values. Because Dataset B is used for training, all values for the category 'CNN prediction score' are > 0.8, as this is how Dataset B was created. One could argue this introduces a bias to this network, but the sole purpose of this network is to further classify samples which were earlier selected by the CNN.
Leaving out the samples labeled as doubtful, 1073 samples remain which are either a true positive, a difficult negative or an artefact.
The labeling of Dataset B was done by visually inspecting the sample itself as well as the figures which were generated, and metrics which were computed, by the post-processing algorithms. By training a FcNN on these same metrics, with the corresponding manually assigned labels as a target, the objective is to (partially) replace the manual inspecting by a human by using a second NN.

Because the values of the metrics in Table 8.1 differ wildly in magnitude (e.g. IME in [kt] vs CNN prediction score [0-1]), each metric is normalized by subtracting the mean of that category (e.g. the mean of

metric	Source
CNN prediction score	Section 6.3
# valid pixels	-
sample mean xch4 above bg	-
sample xch4 standard deviation	-
sample xch4 skewness	-
sample xch4 kurtosis	-
count low confidence plume mask	Section 7.1.1
sum low confidence plume mask	Section 7.1.1
count high confidence plume mask	Section 7.1.1
sum high confidence plume mask	Section 7.1.1
source windspeed north	Section 7.1.2
source windspeed east	Section 7.1.2
IME [kt]	Section 7.1.3
Q [t/h]	Section 7.1.3
L [m]	Section 7.1.3
A_{mask} [m^2]	Section 7.1.3
U_{eff} [m s^{-1}]	Section 7.1.3
tau [s]	Section 7.1.3
albedo R^2, 1 dilation	Section 7.2.1
albedo R^2, 2 dilations	Section 7.2.1
aerosol OT R^2, 1 dilation	Section 7.2.1
aerosol OT R^2, 2 dilations	Section 7.2.1
surface pressure R^2, 1 dilation	Section 7.2.1
surface pressure R^2, 2 dilations	Section 7.2.1
Cloud boundary pixel sum	Section 7.2.2
Cloud boundary pixel count	Section 7.2.2
Angle PCA - mean windvector	Section 7.2.3
Angle PCA - mean windvector weighted	Section 7.2.3

Table 8.1: The metrics (mostly resulting from the feature engineering algorithms) which are used as inputs by for the classification of potentially plume-like samples to plumes, artefacts and difficult negatives by the FcNN.

all the albedo R^2 (1 dilation) values) and divided by
the standard deviation of the trainingset. This results in a distribution for each parameter where the mean is 0 and the standard deviation is 1. This approach is similar to batch-normalization (Section 4.3.4), but then applied to the entire trainingset at once, before training.
The network learns which (normalized) parameters (or combinations of parameters) are important for the classification of the samples, and therefore increases the weight of those connections, while lowering those of less relevant parameters.

Layer (type)	Output Shape	# Parameters
Input	(None, 28)	-
dense (Dense)	(None, 128)	3840
dense_1 (Dense)	(None, 64)	8256
dense_2 (Dense)	(None, 64)	4160
dense_3 (Dense)	(None, 1)	65
Total params:		16,321
Trainable parameters:		16,321
Non-trainable parameters:		0

Table 8.2: The architecture of the fully-connected NN.

The network which was designed is a Fully-connected Neural Network (FcNN), this is different from the Convolutional Neural Network (CNN) which was discussed in Section 6.3. This network only consists of fully-connected layers (Section 4.3.2). This type of network is more suited for vector input, which is why it was selected for this purpose. The network architecture is shown in Table 8.2.

The network consists of three layers, starting from the input shape of a vector with 28 values and resulting in a value on a continuous scale between 0 and 1. During training a distinction is made between empty scenes and artefacts, but for the computation of the classification performance both are assigned a 0, as the main objective is to separate the plumes from non-plumes. This is done because the classification performance metrics (like recall etc.) require binary input.

The network is trained on the training set of Dataset B. Dataset B is the subset of all samples generated from the TROPOMI CH_4 measurements between December 1st, 2020 and January 22nd, 2021. Only the samples which were assigned a prediction score of > 0.8 by the CNN were added to Dataset B..

The training procedure is similar to that of the CNN. Again a training-test split of 80-20 is used. Categorical cross-entropy was used instead of binary cross-entropy as the loss function. Softmax is used as the activation function at the end of the model, this results in a percentage of chance the samples belongs to that particular class for each of the three classes.
The optimal weights of the model are obtained within 10 epochs. The results of the application of the trained FcNN model on the testset of Dataset B are shown in the confusion matrix in Figure 8.1.
Although manual labeling is more precise, as it will result in a accuracy, precision and recall of 1.0, this FcNN based approach can help speed up the process of making detections.
Every single sample in Dataset B has features in the xch4 field which look more or less like a plume (since the CNN assigned them with a score > 0.8). With this in mind, an accuracy of 86% on the hardest to distinguish samples is quite good and can therefore be used to speed up the (manual) classification of the samples which are highlighted by the CNN. However, this approach does not outperform manual inspection, but has as an advantage that it is fully automated.
If this approach is carried on for several months, an updated version of Dataset B with many more samples can be created, which is expected to result in improved performance of the FcNN. This is further discussed in Section 9.2.2.

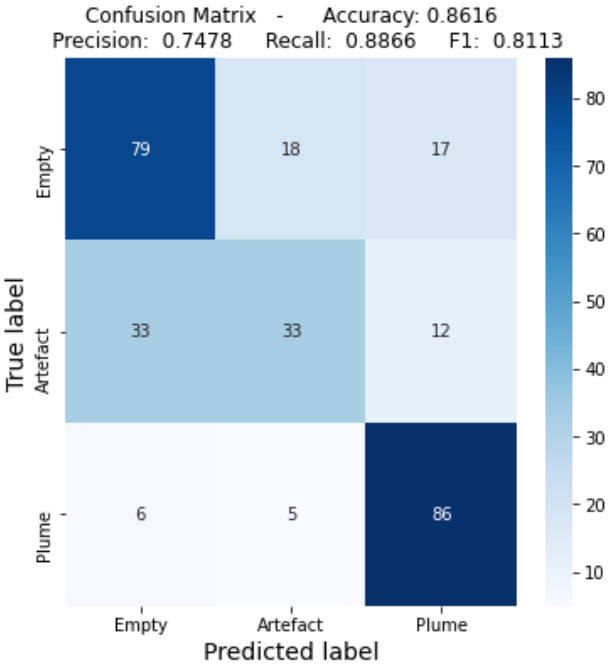


Figure 8.1: The confusion matrix of the fully-connected NN applied to the testset of Dataset B (Table 6.5)

9
Applications of the trained Model

9.1. Application within the TROPOMI-GHGSat project

Figure 9.1 shows how the different components discussed in this report fit together and how those are applied together as a workflow, amongst others in the TROPOMI-GHGSat project.

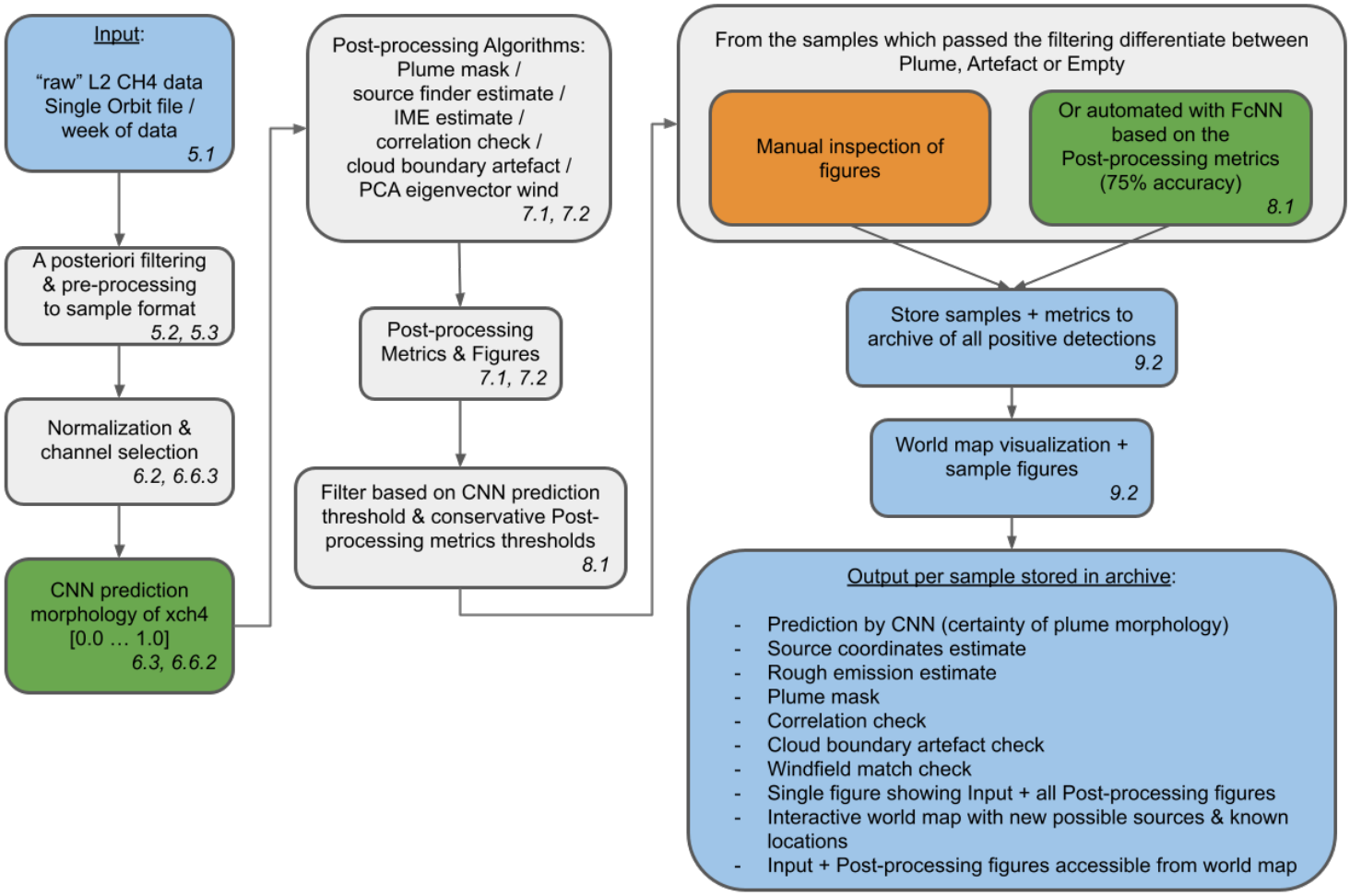


Figure 9.1: Flowchart showing the workflow as applied in the TROPOMI-GHGSat project based on the research described in this report. Blue boxes indicate input or output, green boxes indicate the application of one of the two NNs and the orange box indicates manual effort is required. The number at the bottom right of each box indicates in which section of the report this step is further elaborated. A summarized version of this figures was shown in the Introduction in Figure 1.2.

As soon as a new TROPOMI CH_4 Level 2 Single Orbit file becomes available it can be converted to the standardized sample format $[m, n_H = 32, n_W = 32, n_C]$. This can also be done for longer time periods at once, depending on the application. Then it is normalized and the Convolutional Neural Network (CNN) performs a prediction on whether the sample does contain plume-like morphology in the xch4-field, and where in the image it is located. Next the different feature engineering algorithms as discussed in Chapter 7 are applied to the sample, masking the potential plume, performing a first estimate for the coordinates of the source and performing a first rough estimate of the emissions based on the IME method. Also several metrics are computed which can help to indicate whether the enhancements are a plume or an artefact. This information about the sample, also including the date and the time and a unique identifier of the sample are stored together with the metrics.
Then the samples are filtered mostly based on the prediction score the CNN assigned to the sample, but also on

conservative thresholds resulting from the Feature Engineering algorithms.
Then there are two possible next steps, the manual step where the figures resulting from the feature engineering algorithms are inspected to infer whether a candidate sample is a plume, artefact or does not contain a plume (a difficult negative). The other option is to use the Fully-connected Neural Network (FcNN) which was trained on the metrics of the samples of dataset B. Using the FcNN to make this final classification is less accurate, but this turns the workflow in Figure 9.1 into a fully automated process.
Next the figures of the plumes are stored and the metrics are added to an archive and the information about the samples can be used for further training of the NNs in the future. The blue box at the bottom right of Figure 9.1 provides a list of the most noteworthy information and parameters stored for each sample which is flagged as a plume.
Processing a week of data to the suitable inputformat takes about 20 minutes (for 10-15,000 samples). Running the entire process starting from the suitable inputformat takes about 25 minutes (excluding the visualizations). For the visualizations other data has to be preprocessed to make the latitude-longitude plots, when this is available it takes about 15 seconds per sample to generate the figures. This is done entirely based on the metrics stored in the archive, all information about a sample is condensed here, no new metrics are computed for the visualizations.
This workflow with the two neural networks and the archive was constructed as a project which can expand over time, with the archive of detections growing and therefore providing a larger dataset which can be used for further training of both models.

9.1.1. An example of a successful detection by GHGSat based on a detection in TROPOMI data

The workflow described above has been applied within the context of the TROPOMI-GHGSat project (Sections 2.6.2 & 2.6). From the last week of October 2020 onwards preliminary versions of the CNN, and later also the FcNN were used to detect locations with potential plumes. These locations were then further analysed by scientists at SRON, also with TROPOMI observations over a long timespan at the location of interest included.
Figure 9.2 shows the first detection which lead to the recommendation of a location in Turkmenistan. This plume is not a textbook example, but the source at the location was found to be consistently emitting after further analysis.
GHGSat has performed a successful observation (Figure 9.3) at this location with their third satellite, GHGSat-C2 "Hugo", in early February about a week after launch.
Over the past few month several other locations were detected and recommended to GHGSat after further analysis at SRON. A selection of recent detections is shown in Figure A.3.

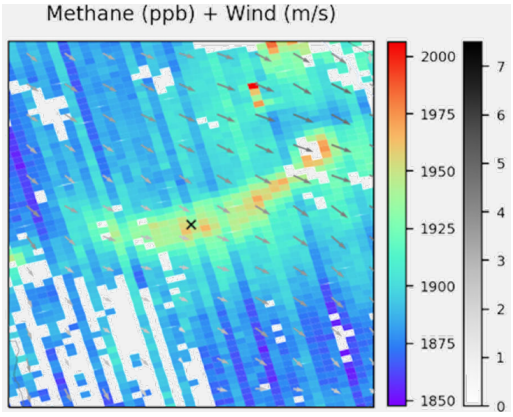


Figure 9.2: The first detection by a preliminary version of the CNN in November leading to the recommendation of this location to GHGSat, following further analysis at SRON. The windfield is questionable.

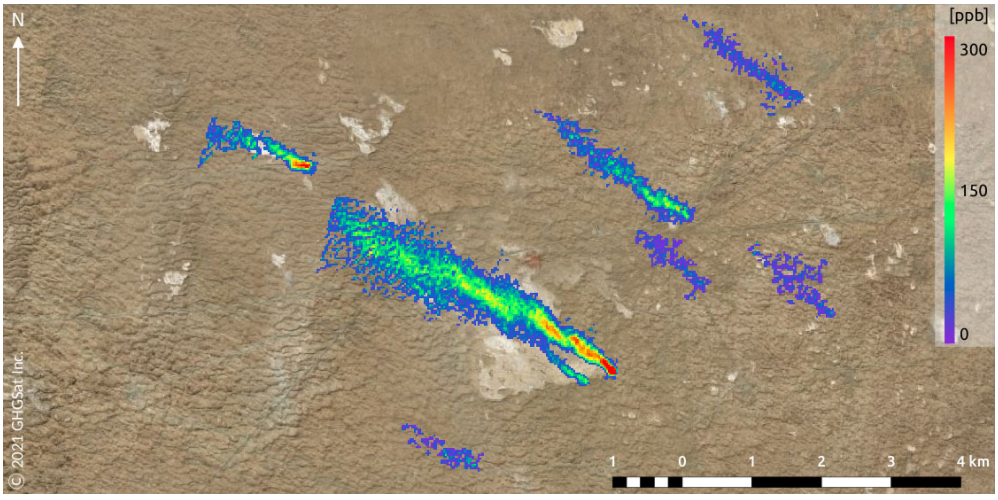


Figure 9.3: Multiple CH_4 plumes detected by GHGSat-C2 "Hugo" in Turkmenistan in the first week of February at a site following a detection with an early version of the CNN, and multiple more later on.[1]

[1] https://www.bloomberg.com/news/articles/2021-02-12/new-climate-satellite-spotted-giant-methane-leak-as-it-happened last access: 07-03-2021. Image credit: GHGSat Inc.

9.2. A point source emissions inventory of 2020 using TROPOMI data

A dataset with all samples resulting from the TROPOMI measurements of 2020 was made, ranging from orbitnumber 11487 to 16679. Pre-processing these 5192 single orbit files resulted in 648,547 samples. Figure 9.6A shows the number of samples of which the center coordinates are within a certain gridcell of 2° × 2°. High latitudes only show a limited number of samples since the pre-processing does contain a snow filter.
The automated workflow discussed in Section 9.1 and shown in Figure 9.1 was applied to the samples resulting from the pre-processing of all observations of 2020. Stricter filtersettings were used compared to when only analysing several weeks of data in the context of the TROPOMI-GHGSat project.

Out of the 648,547 samples, 2195 samples were classified to contain plume-like features by the CNN, then passed the filtering and finally were classified as plume by the FcNN. When inspecting the global distribution of detections some were found to be located at sites with known issues in TROPOMI retrievals. The samples at those locations were manually inspected and removed if need be. 2084 detections remained after this manual inspection, of which 1400 were unique combinations of date & location. The spatial distribution is shown in Figure 9.5.
These results are as expected, with lots of detections at known clusters such as the Permian Basin in the USA, the Hassi-Messaoud oilfield in Algeria, several coal mines in China and multiple different sources in Turkmenistan. There are also several detections at various locations along two major natural gas pipelines in western Russia.

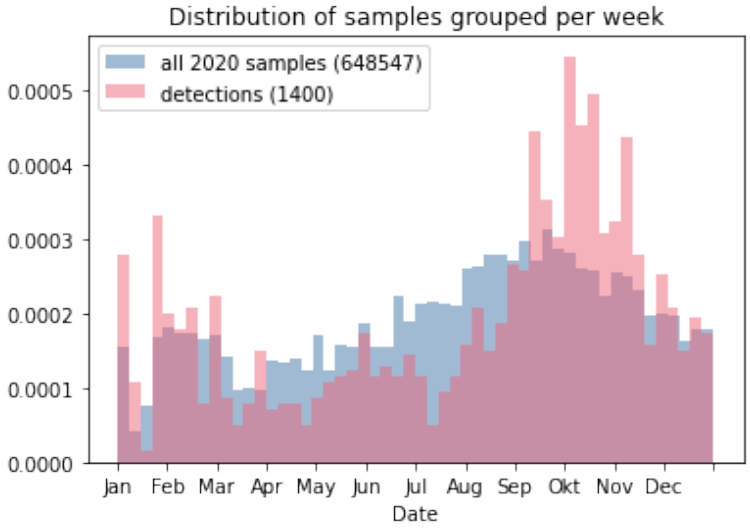


Figure 9.4: The temporal distribution of all samples of 2020 and the detections out of this larger dataset. Both distributions are shown normalized and are grouped by date per week.

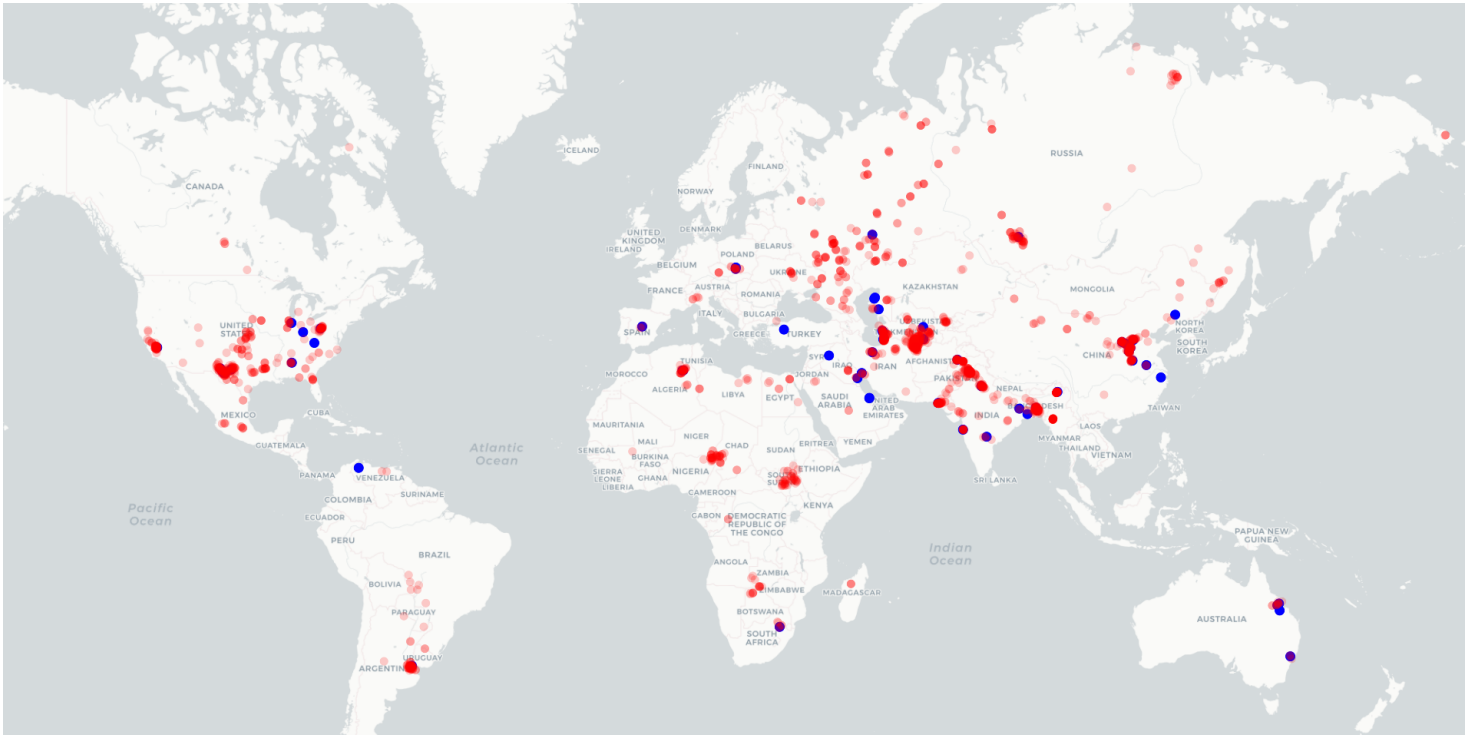


Figure 9.5: An overview of all detected plumes in 2020 shown in red (alpha=0.2). The persistent source which were earlier identified are shown in blue.

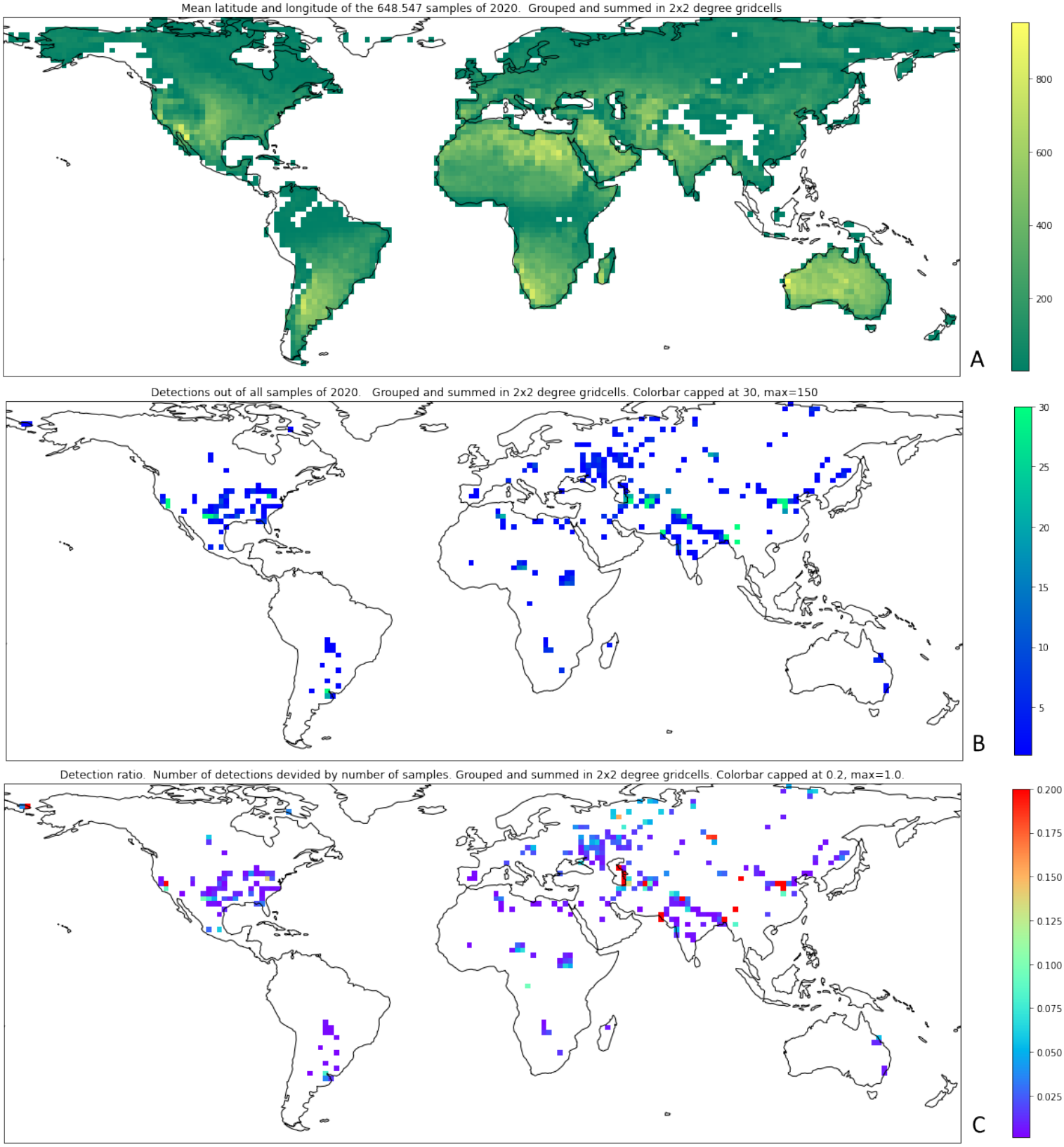


Figure 9.6: A gridded overview of A) all samples resulting from preprocessing of all 2020 data, B) all detections in the 2020 dat and C) the ratio of B/A which yields insight into the persistence of sources in different areas.

Figure 9.4 shows the difference in the normalized temporal distribution between all samples of 2020 and the samples which were classified to contain a plume. Both distributions show an above average fraction during the summer on the northern hemisphere, this signal is stronger in the detections distribution. Since most landmasses are located in the northern hemisphere this effect is to be expected due to the illumination of high latitudes during the summer on the northern hemisphere, during the winter these areas are not illuminated. This effect is most likely strengthened by the snow filter in the pre-processing pipeline.

Figure 9.6B shows the detections of Figure 9.5 resampled to 2° × 2° gridcells, matching the format of Figure 9.6A. The colorbar is capped at 30, but the maximum value is 150, this is done to allow some color contrast for the lower and medium values.
Figure 9.6C shows the ratio of detections (Figure 9.6B) over the available samples (Figure 9.6A). This metric provides insight into how persistent one or more sources in a given gridcell are. This metric would be more representative when compared for exact sources, but this is infeasible as the source finder algorithm is not always perfectly accurate. Therefore the detections were grouped in gridcells instead. This provides the same insight, but one a more regional scale.

9.2.1. A rough estimate of total CH_4 point source emissions in 2020

For each of the detections in the 2020 dataset, the source rate Q [t h^{-1}] was computed in an automated way (Section 7.1.3). The resulting distribution is shown in Figure 9.7.
A rough estimate of the total emissions resulting from the point source emission sources which were detected is made. Each of the detected point source emissions is assumed to be active during the 12 hours before and after the observation. Because of the daily coverage, the revisit time on each location on Earth is 24 hours (or less), which means a new datapoint from which the source rate can be computed is available every 24 hours (if the data is valid), hence this assumption.

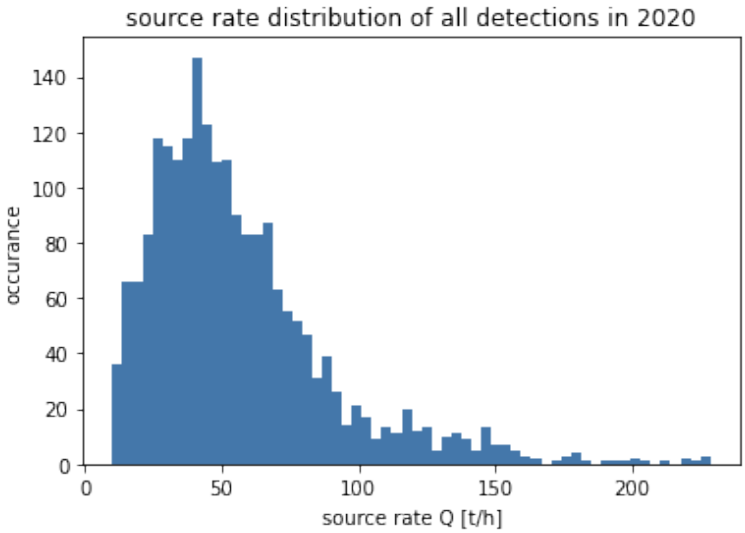


Figure 9.7: The distribution of source rates corresponding to the detections of 2020, which are shown in Figure 9.5.

The total emissions resulting from the detected point sources for all of 2020 are computed as follows:

$$Q_{total} = \sum (Q_{sample} \cdot 24\,\text{h}) = 1824404.53\ \text{t} = 1.82\ \text{Mt}\ (= \text{Tg}) \tag{9.1}$$

1.82 Tg is roughly 0.5% of global annual anthropogenic emissions (Table 2.2). These are the emissions from the detected plumes only, when assuming these are all active for 24 hours.

Using the ratio of detections over the valid samples per cell of 1° × 1°, it is possible to derive the ratio of how often plumes are present out of all TROPOMI observations which result in valid retrievals at a particular location on Earth. This principle was illustrated in Figure 9.6C (shown with a resolution of 2° × 2° in order to make sure the figure is still readable, the computation is done with gridcells of 1° × 1° though).

This ratio allows for a rough estimate of the yearly flux resulting from point source emissions from a particular gridcell. Equation 9.1 is applied again, but now per gridcell, which is elaborated in Equation 9.2. We assume that a source is active for 24 hours, then sum the source rates of each of the detected plumes in that gridcell and divide it by the number of detected plumes in that gridcell to arrive at an average mass of emitted CH_4 per day on which a plume was detected for that particular gridcell.
This average amount of emissions per day is multiplied by the ratio between detections and total samples, and is then multiplied by 365 days to arrive at the estimated total yearly emission flux resulting from point source emis-

sions within that gridcell. This is based on the assumption that the ratio between detected plumes and total number of samples is also representative for days when there is no coverage. Finally, this number is summed for all gridcells. The obtained result is 17.2 Tg a^{-1}, which is equal to 5.2% of total annual anthropogenic CH_4 emissions (Table 2.2). This is roughly equal to the annual CH_4 emissions by the European Union in 2018, which was 17.8 Tg.[2]

$$
\begin{aligned}
Q_{gridcell\_i\_j\_avg} &= \textstyle\sum_1^N \ (Q_{sample} \cdot 24\,\text{h}) \ / \ \text{number of detections} \quad \text{t}\,\text{d}^{-1} \\
AF_{gridcell\_i\_j} \ &= \ Q_{gridcell\_i\_j\_avg} \cdot \tfrac{\text{detections}}{\text{all samples}} \cdot 365 \cdot 10^{-6} \ \text{Mt}\,\text{a}^{-1} \ (= \text{Tg}\,\text{a}^{-1}) \\
AF_{global} \ &= \ \textstyle\sum_{i=-180}^{180} \ \sum_{j=-90}^{90} \ AF_{gridcell\_i\_j} \ \text{Mt}\,\text{a}^{-1} \ (= \text{Tg}\,\text{a}^{-1})
\end{aligned}
\tag{9.2}
$$

Where $AF_{gridcell\_i\_j}$ is the annual CH_4 flux per gridcell, N is the number of detections in the given gridcell, i denotes the longitude index, j denotes the latitude index and AF_{global} is the estimated global annual CH_4 flux resulting from point sources.

This same principle is applied to a smaller, well studied area, instead of the entire globe. The area selected was the Permian Basin, USA. Zhang et al. [16] published a study in early 2020 where the annual CH_4 flux of the 4° × 4° area around the Permian Basin was computed. Using Bayesian atmospheric inverse modeling, the authors showed that emissions were higher than was previously thought based on the bottom-up Environmental Protection Agency (EPA) inventory (2.9 instead of 1.2 Tg a^{-1}, Figure 9.8).

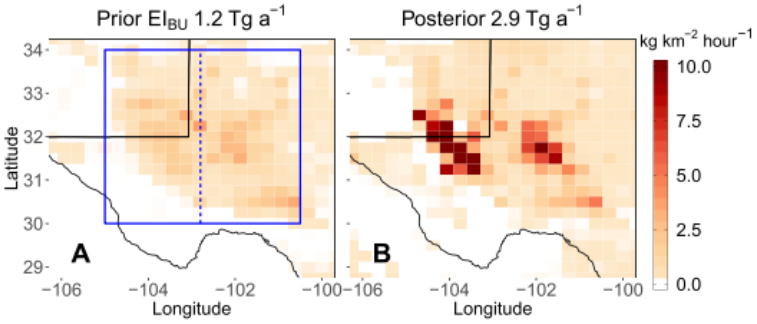


Figure 9.8: The area of study in the analysis by Zhang et al. [16]. source: [16]

The same area of study (30° – 34° N, 101° – 105° W) was selected and using the approach outlined above, a rough estimate of annual CH_4 emissions resulting from point sources in the area which can be automatically detected using the developed NNs was made. In Figure 9.9 an enlarged version of the relevant parts of the maps of Figure 9.6 are shown. The right subfigure of Figure 9.9 shows the estimates of the annual CH_4 flux per gridcell.

The total rough estimate for the emissions by point sources for the area studied by Zhang et al. [16] is 0.33 Tg a^{-1}, which is 11.4% of the value found by the authors of the paper who used Bayesian atmospheric inverse modeling. A difference of this magnitude was expected because this study only includes emissions in the form of point sources, opposed to the other study. The methodology used by Zhang et al. [16] included all emissions, including area sources, and is therefore is actually much better for this purpose. The Permian basin is known to mostly be an area source, with sporadic point source emissions as a result of venting or fracking.
However, this analysis gives insight into which fraction of total emissions is emitted as a point source and is detectable in an automated way using the developed NNs and automated source rate estimation (based on IME, Section 7.1.3).

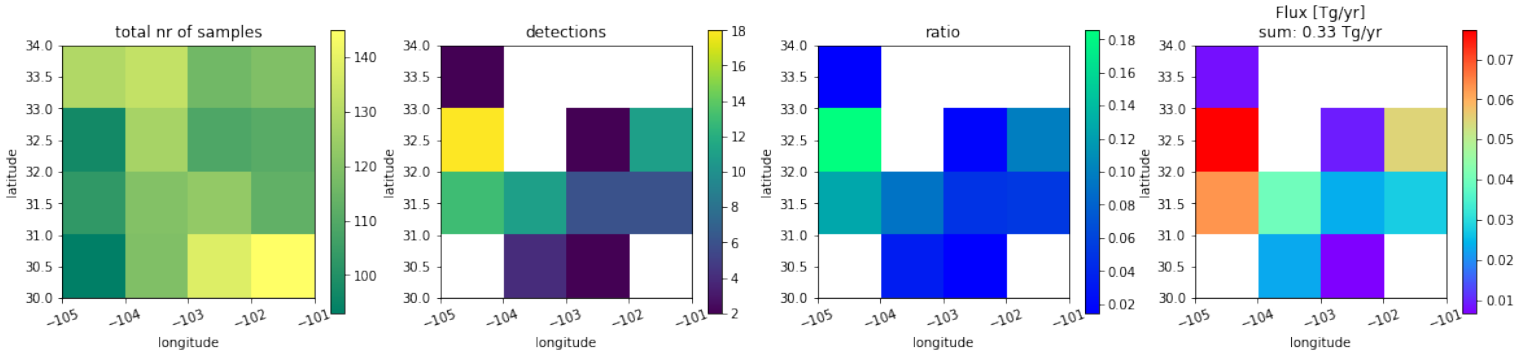


Figure 9.9: The same area of study as analysed by Zhang et al. [16]. The results shown here are from the rough estimate of annual CH_4 flux emissions resulting from point source emissions which were detecting using the approach discussed in Section 9.1.

[2] https://di.unfccc.int/detailed_data_by_party last access: 11-03-2021

9.2.2. Detected plumes archive

As stated before, a 'plume archive' was created which contains all metrics which resulted from the feature engineering algorithms, together with additional information on the location and date of detection for all detected plumes. The corresponding Figures 5.8, 5.2, 6.7, 7.1, 7.3, 7.4, 7.6 and 7.7 are grouped together for each individual sample and stored for later reference. Two examples of the figures are shown in the Appendix, (Figure A.4 + A.5) and Figure A.6 + A.7).

This plume archive contains all plumes from datasets A & B, the detections in the 2020 dataset and additional detections which were performed in the context of the TROPOMI-GHGSat project from January 22nd, 2021 onwards.
The metrics are stored in such a way that the local coordinate system [0...32, 0...32] can be projected back to the entire orbit in along-orbit vs across-orbit coordinates (e.g. from Figure 5.5 back to Figure 5.4). This does allow for plume masks per full orbit file to be constructed, this could be useful for future research on TROPOMI CH_4 plumes, and allows for the plume masks generated here to be remapped to any other desired projection (like an oversampled latitude-longitude grid for example).

This archive also allows for further statistical analysis of the properties of CH_4 plumes in TROPOMI data and it can be used to generate larger training datasets on which future models could be trained.
This archive will be extended with new detections made with the developed model in the context of the TROPOMI-GHGSat project as discussed in Section 9.1.

9.2.3. Interactive map showing detections and supporting figures

The archive discussed in the previous section can be accessed through an interactive map.
Hovering over a detection on the map displays the most relevant information about the detection, such as the coordinates of the estimated source location, the date of the observation and an estimate of the source rate. The samples are uniquely identified by the orbit_number + sample_number combination. This provides an interface in which all of the figures and information of all of the detections made can be conveniently accessed for further analysis. Two examples of the figures are shown in the Appendix, one which is a very clear plume (Figure A.4 + A.5) and one which is more challenging to interpret (Figure A.6 + A.7).
The source finder algorithm is the one most prone to errors, because it is difficult to backtrace the plume in some windfields in an automated way. Therefore, the location shown on the map might be a few pixel (tens of kilometers) off from the most likely source location a human reviewing the figures would select.
Figure 9.5 is a screenshot of this interface displaying all 2020 detections.

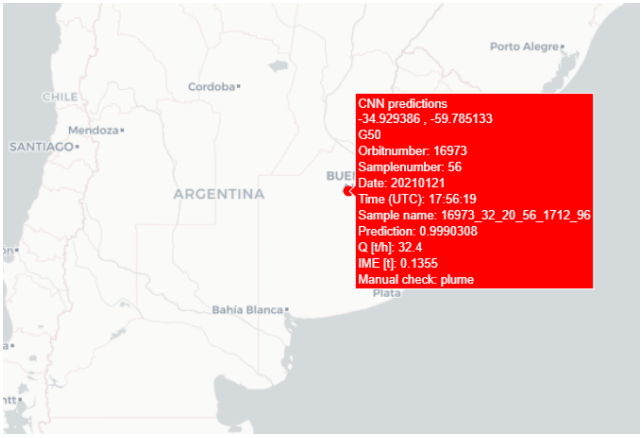


Figure 9.10: A screenshot of the world map showing CH_4 plume detections as dots on the map with information about the detection as label.

10
Conclusions & Future work

Conclusions

The main research question of this thesis, as outlined in Section 3.4, is:

> **"What is the most suitable machine learning algorithm to automatically detect plumes in TROPOMI atmospheric methane data, and how can this model be optimized, implemented and utilized?"**

It can be concluded from this research that the objective of this study has been met. It was shown to be feasible to design a machine learning model which can detect CH_4 plumes in TROPOMI data in a mostly autonomous workflow with sufficient accuracy. The sub-research questions are answered below.

A literature study was performed on the available types of machine learning models and which of those were commonly applied in remote sensing problems. The Convolutional Neural Network (CNN) was selected as the most promising type of model because of the 2D input format which means it takes into account the spatial relations and the ability to detect features anywhere in the input image due to the shared kernels. Also any number of channels could be used and a CNN can be designed to be either shallow or very deep. The CNN outputs a value between 0 and 1, for which a manual threshold can be set to arrive at a binary classification. However, CH_4 emission plumes are not a binary phenomenon, plumes vary in source rate, with low source rates being harder to distinguishable from the measurement noise. Binary classification is the most suitable approach, but it was shown to lead to doubtful predictions for scenes which are difficult to classify.

The TROPOMI CH_4 Level 2 dataproduct had to be turned into a suitable format for a machine learning approach, preferably in a standardized, automated way. Given that a CNN requires gridded input, that is in a matrix or 2D/3D tensor, two feasible options were identified. The option which is used, utilizes the groundtrack as a local reference frame with local along-orbit by across-orbit coordinates. The downside of this approach is that off-nadir pixels cover a larger surface area, but this format is more suitable for single-orbit data and is more suitable for a standardized approach. A downside is that it is TROPOMI-specific.

The along-orbit by across-orbit reference frame can be split into $[32 \times 32]$ pixel images using the moving-window approach with a stride of 50% and a threshold of 20% pixel coverage. This approach ensures that if a plume is cut off, then it is in the center of the next image. This approach results in a standardized format, to which any number of parameters present in the TROPOMI CH_4 Level 2 dataproduct can be added as additional channels. Furthermore the xch4 data is filtered a-posteriori, de-striped and corrected for the albedo biases. Before being fed to the network the xch4 field is normalized relative to the local background in order to make all samples consistent.
As trainingdata, scenes with plumes were extracted from locations which were earlier identified within the TROPOMI-GHGSat project, those were manually identified, pre-processed to the correct format and double-checked. For the negatives in the trainingset eight orbits were used, of which all plumes were filtered out resulting in only empty scenes. The samples in the trainingset were augmented by applying rotations.

Next a CNN of limited complexity (shallow) was designed. This CNN is trained on the prepared dataset achieving a correct classification accuracy of 93%. By extracting a Class Activation Map (CAM) from the network it could be verified that the network correctly identifies plume-like features to base its predictions on and is not triggered on every high enhancement.
Several attempts were made to further improve the performance of the network. A very deep CNN ResNet with over 23.5 million parameters was tested, but it was concluded that such a network is too deep for a binary classification problem with input images of shape 32×32 and a relatively small trainingset.
Next an attempt was made to improve performance by finetuning the network on some of the most difficult to classify samples, which are referred to as difficult negatives and artefacts. Artefacts are enhancements in xch4 field not due to actual emissions but due to correlations with other atmospheric parameters, such as the surface albedo. Next to xch4, several other parameters of the sample were added, but this did not lead to improved performance. The decrease in performance is expected to occur because the elementwise multiplication in the convolutional operation

for multi-dimensional input (Figures 4.6 & 4.8) does not properly capture the information contained in the layers with additional parameters, this suggests the information contained in those additional layers is better extracted separte from the CNN.
Therefore a custom CNN of medium depth was designed and trained only on the xch4 field as input. This network attained an accuracy of 97%. After a detailed comparison, it was concluded the deep CNN with only the xch4 field as input performed best.

Instead of filtering the artefacts in the CNN, another method had to be developed to filter out artefacts, this is why several automated feature engineering algorithms were developed. The outputs of those algorithms are a plume mask, an estimate for the source location, a rough estimate of the amount of emitted CH_4 and source rate, correlation coefficients between xch4 and the atmospheric parameters most commonly causing artefacts, a check whether an enhancement is on the boundary of a cloud and comparison of the principal eigenvector of the plume with the windvector. Using metrics based on those algorithms a second Neural Network, a Fully-connected Neural Network was trained to classify samples which were classified by the CNN to contain plume-like morphology in the xch4 field as either true plumes, artefacts or difficult negative. This network attained an accuracy of 86% on a subset of the most difficult to classify ~ 1000 samples out of ~ 80,000 from seven weeks of observations. Depending on the application it might be favourable to manually inspect the samples classified by the CNN or to rely on this FcNN.
This approach shows that the additional information from the other included retrieved atmospheric parameters in the TROPOMI dataproduct are best handled outside of the first CNN. Using specialized algorithms, much more information can be extracted from a single sample.

This workflow was incorporated in the TROPOMI-GHGSat project. From the third week of October onwards the preliminary versions of the CNN were used to make detections in recent observations. From December 1st onwards the metrics of the feature engineering algorithms were stored to an archive, which is since then being expanded with new plume detections. This archive can be used to improve both the CNN and FcNN later on, when more observations have been made.
The workflow which has been developed can now be applied to perform detections of plumes in the new recent TROPOMI CH_4 data. The model which was developed can therefore be used for global monitoring.

Using the developed automated workflow an overview of all detectable point sources in the year 2020 could be made, without this automated data pipeline this would have been unfeasible. Pre-processing resulted in 648,547 samples. and the detection process led to 2195 detected plumes. Manual inspection of detections at locations known to be challenging for TROPOMI retrievals, lead to identification of several coastal artefacts, and noisy scenes over rainforests. After this manual inspection 2084 samples remained, of which 1400 were unique combinations of date & location.
These detections were mainly clustered around a small number hotspots for pointsource emissions. Most of these locations had been already indentified using preliminary versions of the model since October, or before this project was started. A number of new potential hotspots were identified as well and are further investigated at the time of writing.
A relative increase in detections during summer on the northern hemisphere is observed, which is speculated to be mainly caused by Russia being illuminated and snow free. Using the source rate estimate which was computed for each of the pointsource emissions an estimate of the total emissions resulting from the detected pointsources could be computed.

When each of the pointsources is assumed to be active for 24 hours (the time between two TROPOMI observations at a constant source rate, the estimated total emissions resulting form these pointsources is 1.82 Tg. This is roughly 0.5% of global annual anthropogenic emissions of all categories (Table 2.2).
When the ratio of detections over all valid samples is computed per gridcell of $1° \times 1°$ gridcells, a rough estimate can be made about annual global emissions resulting from point sources. With the assumption that this ratio, and the average source rate of the days where there is a detection is representative for the days without data, an estimate of the annual CH_4 flux is computed. This was found to be 17.2 Tg a^{-1}, which is 5.2% of annual antrhopogenic CH_4 emissions, and is roughly equal to all CH_4 emissions by the European Union in the year 2018.
The same analysis was performed on the Permian Basin, a $4° \times 4°$ area studied in detail by Zhang et al. [16]. An estimate of 0.33 Tg a^{-1} is found, which is 11.4% of the value the authors found using Bayesian atmospheric inverse modeling, which also includes area sources. This result gives insight into which fraction of emissions is emitted as a point source and is detectable in an automated way using the developed NNs and automated source rate estimation.

Future work

Several opportunities for improvements or follow-up studies were identified during this research:

- The most convenient opportunity for future work is to train the same two Neural Networks again on a larger dataset which was obtained from the implementation of the workflow within the TROPOMI-GHGSat project and the 2020 year overview. The limiting factor for further improvement was most likely the lack of additional trainingdata. These detections which can be used as trainingdata are stored in a convenient format.

- The approach outlined in this research was applied to all data of 2020. Since TROPOMI was launched in October 2017 , it could also be applied on the observations of the years 2018 and 2019. However for this application it should be noted that the TROPOMI pixel size changed in September 2019 from 7 km to 5.5 km in the along-orbit direction.

- The filtersettings used in this study could be reviewed and improved where possible.

- The plumes detected in the archive can also be used for other follow-up studies than continuation of this study. The plume masks are stored in such a way that the information can be extracted and remapped to any desired format.

- Several other machine learning approaches which might be feasible for this problem can be attempted, taking advantage of the created archive of plumes for training. Image segmentation networks are an approach where samples are not labeled binary, but where a segmentation mask is produced. This segmentation mask indicates which part of the image corresponds to the target class. This approach does require segmentation masks as input during training, therefore it was not a viable option at the start of this project. Now, the plume masks generated for each plume could be used for this purpose. An image segmentation approach might however encounter difficulties in a problem setup with only two classes, and the question is whether this would yield better results than the plume mask algorithm which was developed. Therefor this approach was not pursued in this research.

- Several of the feature engineering algorithms could be improved and additional feature engineering algorithms could be added.
  The source-finder algorithm can be improved as the estimate of the source location can be off by a few pixels, which quickly adds up to tens of kilometers given the surface area covered by a single TROPOMI pixel.
  The IME algorithm could also be improved by computing a more detailed plume length.
  The source finder and PCA/windvector angle analysis operate in latitude-longitude space. It would be more accurate to project the datapoints to a flattened 2D Earth surface with dimensions in km × km. The windvectors of the GEOS-FP dataproduct are given in m s^{-1}, which is on a local scale not that different from latitude-longitude space, but this projection to km × km space would improve the accuracy.
  If new feature engineering algorithms are developed, the metrics resulting from those can also be utilized in order to further improve the performance of the FcNN.

- The approach which is described in this research could potentially also be applied in other contexts (with some modifications) with other (future) instruments measuring CH_4 (Sentinel 5, CO2M, MethaneSAT) or to other trace gasses, like CO or NO_2 measured by TROPOMI.

- The models and workflow developed in this research could potentially be applied to measurements from Sentinel 5 once operational. If the specifications of the instrument, and the resulting pixel properties are similar, this workflow should be able to be implemented with only very limited modifications. Potentially the NNs trained on TROPOMI CH_4 plumes could be used to detect plumes in Sentinel 5 observations shortly after launch already, thus without the need to first build up a training dataset.

Data Availability

TROPOMI CH_4 data are available at ftp://ftp.sron.nl/open-access-data-2/TROPOMI/tropomi/ch4/10_9/.
GEOS-FP data are available at https://portal.nccs.nasa.gov/cgi-lats4d/opendap.cgi?&path=.

The developed models and model output is available upon request via email:

B.J.Schuit@sron.nl / berendjschuit@gmail.com (author) or
J.D.Maasakkers@sron.nl (supervisor)

Bibliography

[1] D Shindell, F-m Bréon, W Collins, J Fuglestvedt, J Huang, D Koch, J-f Lamarque, D Lee, B Mendoza, T Nakajima, A Robock, G Stephens, T Takemura, H Zhang, D Qin, G-k Plattner, M Tignor, SK Allen, J Boschung, A Nauels, Y Xia, V Bex, and PM Midgley. AR5 Climate Change 2013: The Physical Science Basis, Chapter 8. Anthropogenic and Natural Radiative Forcing. Contribution of Working Group I. Technical report, 2013.

[2] Daniel J. Jacob. Introduction to Atmospheric Chemistry. Princeton University Press, illustrate edition, 1999. ISBN 9780691001852. URL http://acmg.seas.harvard.edu/publications/jacobbook/index.html.

[3] Hartmut Bösch K. Fletcher Editing, Heinrich Bovensmann. Report for Mission Selection: CarbonSat (ESA SP-1330/1, June 2015). 2015. ISBN 9789292214289. URL http://esamultimedia.esa.int/docs/EarthObservation/SP1330-1_CarbonSat.pdf.

[4] Philippe Ciais, Christopher Sabine, G Bala, Laurent Bopp, Victor Brovkin, J. Canadell, A Chhabra, R DeFries, J. Galloway, Martin Heimann, C Jones, C. Le Quéré, R.B. Myneni, S Piao, and P Thornton. AR5 Climate Change 2013: The Physical Science Basis, Chapter 2. Contribution of Working Group I. Technical report, 2013. URL http://www.ipcc.ch/report/ar5/wg1/docs/review/WG1AR5_SOD_Ch06_All_Final.pdf%5Cnhttp://ebooks.cambridge.org/ref/id/CBO9781107415324A023.

[5] NOAA, NASA. U.S. Standard Atmosphere 1976. U. S. Government Printing Office,Washington, DC, 1976.

[6] Daniel J Jacob, Alexander J Turner, Joannes D Maasakkers, Jianxiong Sheng, Kang Sun, Xiong Liu, Kelly Chance, Ilse Aben, Jason McKeever, and Christian Frankenberg. Satellite observations of atmospheric methane and their value for quantifying methane emissions. Atmos. Chem. Phys, 16:14371–14396, 2016. doi: 10.5194/acp-16-14371-2016. URL www.atmos-chem-phys.net/16/14371/2016/.

[7] Eystein Jansen, Jonathan Overpeck, Keith R Briffa, Jean-Claude Duplessy, Fortunat Joos, Valérie Masson-Delmotte, Daniel Olago, Bette Otto-Bliesner, W Richard Peltier, Stefan Rahmstorf, Rengaswamy Ramesh, J-m Barnola, E Bauer, E Brady, Jean Jouzel, John Mitchell, J Overpeck, KR Briffa, J-c Duplessy, F Joos, V Masson-Delmotte, D Olago, B Otto-Bliesner, WR Peltier, S Rahmstorf, R Ramesh, D Raynaud, D Rind, O Solomina, R Villalba, D Zhang, D Qin, M Manning, Z Chen, M Marquis, KB Averyt, M Tignor, and HL Miller. AR4 Climate Change 2007: The Physical Science Basis, Chapter 6. Contribution of Working Group I. Technical report, 2007.

[8] Venkatachalam Ramaswamy, Piers Forster, Paulo Artaxo, Terje Berntsen, Richard Betts, David W. Fahey, James Haywood, Judith Lean, David C. Lowe, Gunnar Myhre, John Nganga, Ronald Prinn, Graciela Raga, Michael Schulz, Robert Van Dorland, Graciela Raga, Teruyuki Nakajima, Veerabhadran Ramanathan, P Artaxo, T Berntsen, R Betts, DW Fahey, J Haywood, J Lean, DC Lowe, G Myhre, J Nganga, R Prinn, G Raga, M Schulz, R Van, Change Solomon, D Qin, M Manning, Z Chen, M Marquis, and KB Averyt. AR4 Climate Change 2007: The Physical Science Basis, Chapter 2. Contribution of Working Group I. Technical report, 2007.

[9] Simon K Allen, Nathaniel L Bindoff, François-marie Bréon France, Ulrich Cubasch, Myles R Allen Uk, Olivier Boucher France, Jens Hesselbjerg, Christensen Denmark, Philippe Ciais France, Matthew Collins Uk, Viviane Vasconcellos, and Richard A Feely. AR5 Climate Change 2013: Technical Summary. Climate Change 2013 - The Physical Science Basis, pages 31–116, 2014. doi: 10.1017/cbo9781107415324.005.

[10] Philippe Ciais, Christopher Sabine, G Bala, Laurent Bopp, Victor Brovkin, J. Canadell, A Chhabra, R DeFries, J. Galloway, Martin Heimann, C Jones, C. Le Quéré, R.B. Myneni, S Piao, and P Thornton. AR5 Climate Change 2013: The physical science basis, Chapter 6. Contribution of working group I. Change, IPCC Climate, pages 465–570, 2013. ISSN 1098-6596. doi: 10.1017/CBO9781107415324.015. URL http://www.ipcc.ch/report/ar5/wg1/docs/review/WG1AR5_SOD_Ch06_All_Final.pdf%5Cnhttp://ebooks.cambridge.org/ref/id/CBO9781107415324A023.

[11] V. Ramaswamy, W. Collins, J. Haywood, J. Lean, N. Mahowald, G. Myhre, V. Naik, K. P. Shine, B. Soden, G. Stenchikov, and T. Storelvmo. Radiative Forcing of Climate: The Historical Evolution of the Radiative Forcing Concept, the Forcing Agents and their Quantification, and Applications. Meteorological Monographs, 59:1–14, 2019. ISSN 0065-9401. doi: 10.1175/amsmonographs-d-19-0001.1.

[12] Robert L Kleinberg. Kleinberg GWP & Climate Policy 200901 The Global Warming Potential Misrepresents the Physics of Global Warming Thereby Misleading Policy Makers. (September), 2020. doi: 10.13140/RG.2.2.20069.22249.

[13] Stefanie Kirschke, Philippe Bousquet, Philippe Ciais, Marielle Saunois, Josep G Canadell, Edward J Dlugokencky, Peter Bergamaschi, Daniel Bergmann, Donald R Blake, Lori Bruhwiler, Philip Cameron-Smith, Simona Castaldi, Frédéric Chevallier, Liang Feng, Annemarie Fraser, Martin Heimann, Elke L Hodson, Sander Houweling, Béatrice Josse, Paul J Fraser, Paul B Krummel, Jean-François Lamarque, Ray L Langenfelds, Corinne Le Quéré, Vaishali Naik, Paul I Palmer, Isabelle Pison, David Plummer, Benjamin Poulter, Ronald G Prinn, Matt Rigby, Bruno Ringeval, Monia Santini, Martina Schmidt, Drew T Shindell, Isobel J Simpson, Renato Spahni, L Paul Steele, Sarah A Strode, Kengo Sudo, Sophie Szopa, Guido R van der Werf, Apostolos Voulgarakis, Michiel van Weele, Ray F Weiss, Jason E Williams, and Guang Zeng. Three decades of global methane sources and sinks. 2013. doi: 10.1038/NGEO1955. URL www.nature.com/naturegeoscience.

[14] Tia Scarpelli, Daniel Jacob, Joannes Maasakkers, Melissa Sulprizio, Jian-Xiong Sheng, Kelly Rose, Lucy Romeo, John Worden, and Greet Janssens-Maenhout. A global gridded (0.1° × 0.1°) inventory of methane emissions from oil, gas, and coal exploitation based on national reports to the United Nations Framework Convention on Climate Change. Earth System Science Data Discussions, pages 1–21, 2019. ISSN 1866-3508. doi: 10.5194/essd-2019-127.

[15] Haili Hu, Jochen Landgraf, Rob Detmers, Tobias Borsdorff, Joost Aan de Brugh, Ilse Aben, André Butz, and Otto Hasekamp. Toward Global Mapping of Methane With TROPOMI: First Results and Intersatellite Comparison to GOSAT. Geophysical Research Letters, 45(8):3682–3689, 4 2018. ISSN 00948276. doi: 10.1002/2018GL077259. URL http://doi.wiley.com/10.1002/2018GL077259.

[16] Yuzhong Zhang, Ritesh Gautam, Sudhanshu Pandey, Mark Omara, Joannes D. Maasakkers, Pankaj Sadavarte, David Lyon, Hannah Nesser, Melissa P. Sulprizio, Daniel J. Varon, Ruixiong Zhang, Sander Houweling, Daniel Zavala-Araiza, Ramon A. Alvarez, Alba Lorente, Steven P. Hamburg, Ilse Aben, and Daniel J. Jacob. Quantifying methane emissions from the largest oil-producing basin in the United States from space. Science Advances, 6 (17):1–10, 2020. ISSN 23752548. doi: 10.1126/sciadv.aaz5120.

[17] Daniel Zavala-Araiza, David Lyon, Ramon Ramon, A Alvarez, Virginia Palacios, Robert Harriss, Xin Lan, Robert Talbot, and Steven P Hamburg. Toward a Functional Definition of Methane Super-Emitters: Application to Natural Gas Production Sites. Environ. Sci. Technol, 49:8167–8174, 2015. doi: 10.1021/acs.est.5b00133. URL https://pubs.acs.org/sharingguidelines.

[18] European Commission. an EU strategy to reduce methane emissions. Technical report, European Commission, 2020. URL https://ec.europa.eu/commission/presscorner/detail/en/IP_20_1833.

[19] Sudhanshu Pandey, Ritesh Gautam, Sander Houweling, Hugo Denier Van Der Gon, Pankaj Sadavarte, Tobias Borsdorff, Otto Hasekamp, Jochen Landgraf, Paul Tol, Tim Van Kempen, Ruud Hoogeveen, Richard Van Hees, Steven P Hamburg, Joannes D Maasakkers, and Ilse Aben. Satellite observations reveal extreme methane leakage from a natural gas well blowout. 2019. doi: 10.1073/pnas.1908712116/-/DCSupplemental. URL https://youtu.be/D0F450ESHP8.

[20] D. J. Varon, J. McKeever, D. Jervis, J. D. Maasakkers, S. Pandey, S. Houweling, I. Aben, T. Scarpelli, and D. J. Jacob. Satellite Discovery of Anomalously Large Methane Point Sources From Oil/Gas Production. Geophysical Research Letters, 46(22):13507–13516, 11 2019. ISSN 0094-8276. doi: 10.1029/2019GL083798. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2019GL083798.

[21] J.D. Maasakkers et al. Lousiana blowout paper. In Review.

[22] S. Conley, G. Franco, I. Faloona, D. R. Blake, J. Peischl, and T. B. Ryerson. Methane emissions from the 2015 Aliso Canyon blowout in Los Angeles, CA. Science, 351(6279):1317–1320, 2016. ISSN 10959203. doi: 10.1126/science.aaf2348.

[23] D. R. Thompson, A. K. Thorpe, C. Frankenberg, R. O. Green, R. Duren, L. Guanter, A. 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. ISSN 19448007. doi: 10.1002/2016GL069079.

[24] Daniel J Varon, Daniel J Jacob, Jason Mckeever, Dylan Jervis, Berke O A Durak, Yan Xia, and Yi Huang. Quantifying methane point sources from fine-scale satellite observations of atmospheric methane plumes. Atmos. Meas. Tech, 11:5673–5686, 2018. doi: 10.5194/amt-11-5673-2018. URL https://doi.org/10.5194/amt-11-5673-2018.

[25] J. P. Veefkind, I. Aben, K. McMullan, H. Förster, J. de Vries, G. Otter, J. Claas, H. J. Eskes, J. F. de Haan, Q. Kleipool, M. van Weele, O. Hasekamp, R. Hoogeveen, J. Landgraf, R. Snel, P. Tol, P. Ingmann, R. Voors, B. Kruizinga, R. Vink, H. Visser, and P. F. Levelt. TROPOMI on the ESA Sentinel-5 Precursor: A GMES mission for global observations

of the atmospheric composition for climate, air quality and ozone layer applications. Remote Sensing of Environment, 120:70–83, 5 2012. ISSN 00344257. doi: 10.1016/j.rse.2011.09.027.

[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, (September), 2020. doi: https://doi.org/10.5194/amt-2020-301.

[27] Yasjka Meijer, H. Boesch, A. Bombelli, D. Brunner, M. Buchwitz, P. Ciais, D. Crisp, R. Engelen, K. Holmlund, S. Houweling, G. Janssens-Maenhout, J. Marshall, M. Nakajima, B. Pinty, M. Scholze, J.-L. Bezy, M.R. Drinkwater, T. Fehr, V. Fernandez, A. Loescher, H. Nett, and B. Sierk. Copernicus CO 2 Monitoring Mission Requirements Document. Technical report, ESA, 2020. URL https://esamultimedia.esa.int/docs/EarthObservation/CO2M_MRD_v3.0_20201001_Issued.pdf.

[28] ESA. MIssion database: Copernicus, Sentinel-5P. https://directory.eoportal.org/web/eoportal/satellite-missions/c-missions/copernicus-sentinel-5p, 2020. (Accessed on 28/05/2020).

[29] Maarten Sneep. Sentinel 5 precursor / TROPOMI KNMI and SRON level 2 Input Output Data Definition. Technical report, 2019.

[30] Otto Hasekamp, Alba Lorente, Haili Hu, Andre Butz, Joost aan de Brugh, and Jochen Landgraf. Algorithm Theoretical Baseline Document for Sentinel-5 Precursor Methane Retrieval. Technical Report v1.10, SRON Netherlands Institute for Space Research, 2019.

[31] Haili Hu, Otto Hasekamp, André Butz, André Galli, Jochen Landgraf, Joost Aan De Brugh, Tobias Borsdorff, Remco Scheepmaker, and Ilse Aben. The operational methane retrieval algorithm for TROPOMI. Atmospheric Measurement Techniques, 9(11):5423–5440, 2016. ISSN 18678548. doi: 10.5194/amt-9-5423-2016.

[32] Arnoud Apituley, Mattia Pedergnana, Maarten Sneep, J. Pepijn Veefkind, Diego Loyola, and Otto Hasekamp. Sentinel-5 precursor / TROPOMI Level 2 Product User Manual Methane. Technical Report 0.11.6, SRON & KNMI, 2017.

[33] A. Butz, O. P. Hasekamp, C. Frankenberg, J. Vidot, and I. Aben. CH4 retrievals from space-based solar backscatter measurements: Performance evaluation against simulated aerosol and cirrus loaded scenes. Journal of Geophysical Research Atmospheres, 115(24):1–15, 2010. ISSN 01480227. doi: 10.1029/2010JD014514.

[34] Alana K. Ayasse, Andrew K. Thorpe, Dar A. Roberts, Christopher C. Funk, Philip E. Dennison, Christian Frankenberg, Andrea Steffke, and Andrew D. Aubrey. Evaluating the effects of surface properties on methane retrievals using a synthetic airborne visible/infrared imaging spectrometer next generation (AVIRIS-NG) image. Remote Sensing of Environment, 215(December 2017):386–397, 2018. ISSN 00344257. doi: 10.1016/j.rse.2018.06.018.

[35] Jochen Landgraf, Alba Lorente, Bavo Langerock, Mahesh Kumar Sha, J. C. Lambert, D Loyola, and J.P. Veefkind. S5P Mission Performance Centre Methane Readme. Technical report, 2020.

[36] Yuzhong Zhang, Ritesh Gautam, Sudhanshu Pandey, Mark Omara, Joannes D. Maasakkers, Pankaj Sadavarte, David Lyon, Hannah Nesser, Melissa P. Sulprizio, Daniel J. Varon, Ruixiong Zhang, Sander Houweling, Daniel Zavala-Araiza, Ramon A. Alvarez, Alba Lorente, Steven P. Hamburg, Ilse Aben, and Daniel J. Jacob. Supplementary Materials for Quantifying methane emissions from the largest oil-producing basin in the United States from space. Science Advances, 6(17), 2020. ISSN 23752548. doi: 10.1126/sciadv.aaz5120.

[37] Daniel J. Varon, Daniel J. Jacob, Dylan Jervis, and Jason Mckeever. Quantifying Time-Averaged Methane Emissions from Individual Coal Mine Vents with GHGSat-D Satellite Observations. Environmental Science and Technology, 54(16):10246–10253, 2020. ISSN 15205851. doi: 10.1021/acs.est.0c01213.

[38] J.D. Maasakkers. Using TROPOMI to guide fine-resolution satellite instruments towards methane point sources. Number June. IWGGMS, 2020.

[39] Claus Zehner. Sentinel-5 Precursor Mission Status and Results on Methane measurements IWGGMS-16 Sentinel-5 Precursor. Number June. IWGGMS, 2020.

[40] Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofei Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. ISPRS Journal of Photogrammetry and Remote Sensing, 152 (November 2018):166–177, 2019. ISSN 09242716. doi: 10.1016/j.isprsjprs.2019.04.015. URL https://doi.org/10.1016/j.isprsjprs.2019.04.015.

[41] John E. Ball, Derek T. Anderson, and Chee Seng Chan. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. Journal of Applied Remote Sensing, 11(04):1, 2017. ISSN 1931-3195. doi: 10.1117/1.jrs.11.042609.

[42] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems 25 (NIPS 2012), 2012. doi: 10.1201/9781420010749.

[43] Grace Lindsay. Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future. Journal of Cognitive Neuroscience, pages 1–15, 2020. ISSN 0898-929X. doi: 10.1162/jocn{_}a{_}01544.

[44] Michał Koziarski and Bogusław Cyganek. Impact of low resolution on image recognition with deep neural networks: An experimental study. International Journal of Applied Mathematics and Computer Science, 28(4): 735–744, 2018. ISSN 20838492. doi: 10.2478/amcs-2018-0056.

[45] Ismail Mebsout. Convolutional neural networks - mathematical foundation, 2020. URL https://www.ismailmebsout.com/Convolutional%20Neural%20Network%20-%20Part%201/.

[46] Adrian Rosebrock. Deep Learning for Computer Vision with Python. 2017. ISBN 9788578110796. doi: 10.1017/CBO9781107415324.004.

[47] Vivienne Sze, Yu Hsin Chen, Tien Ju Yang, and Joel S. Emer. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. Proceedings of the IEEE, 105(12):2295–2329, 2017. ISSN 15582256. doi: 10.1109/JPROC.2017.2761740.

[48] S. Lhermitte. Remote Sensing & Big Data - Delft University of Technology course, 2020. URL https://studiegids.tudelft.nl/a101_displayCourse.do?course_id=56333.

[49] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 32nd International Conference on Machine Learning, ICML 2015, 1:448–456, 2015.

[50] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, 15:1929–1958, 2014. ISSN 03702693.

[51] Andrea Molod, Lawrence Takacs, Max Suarez, Julio Bacmeister, In-Sun Song, and Andrew Eichmann. The GEOS-5 Atmospheric General Circulation Model : Mean Climate and Development from MERRA to Fortuna, Technical Report Series on Global Modeling and Data Assimilation, volume 28. 2012. ISBN 2012104606.

[52] Satish Kumar, Carlos Torres, Oytun Ulutan, Alana Ayasse, Dar Roberts, and B. S. Manjunath. Deep remote sensing methods for methane detection in overhead hyperspectral imagery. Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020, (March):1765–1774, 2020. doi: 10.1109/WACV45572.2020.9093600.

[53] Alba Lorente, Tobias Borsdorff, Andre Butz, Otto Hasekamp, Joost Aan De Brugh, Andreas Schneider, Lianghai Wu, Frank Hase, Rigel Kivi, Debra Wunch, David F. Pollard, Kei Shiomi, Nicholas M. Deutscher, Voltaire A. Velazco, Coleen M. Roehl, Paul O. Wennberg, Thorsten Warneke, and Jochen Landgraf. Methane retrieved from TROPOMI: Improvement of the data product and validation of the first 2 years of measurements. Atmospheric Measurement Techniques, 14(1):665–684, 2021. ISSN 18678548. doi: 10.5194/amt-14-665-2021.

[54] Tobias Borsdorff, Joost Aan De Brugh, Haili Hu, Otto Hasekamp, Ralf Sussmann, Markus Rettinger, Frank Hase, Jochen Gross, Matthias Schneider, Omaira Garcia, Wolfgang Stremme, Michel Grutter, DIetrich G. Feist, Sabrina G. Arnold, Martine De Mazière, Mahesh Kumar Sha, David F. Pollard, Matthäus Kiel, Coleen Roehl, Paul O. Wennberg, Geoffrey C. Toon, and Jochen Landgraf. Mapping carbon monoxide pollution from space down to city scales with daily global coverage. Atmospheric Measurement Techniques, 11(10):5507–5518, 2018. ISSN 18678548. doi: 10.5194/amt-11-5507-2018.

[55] V. Huijnen, J. Williams, M. Van Weele, T. Van Noije, M. Krol, F. Dentener, A. Segers, S. Houweling, W. Peters, J. De Laat, F. Boersma, P. Bergamaschi, P. Van Velthoven, P. Le Sager, H. Eskes, F. Alkemade, R. Scheele, P. Nédélec, and H. W. Pätz. The global chemistry transport model TM5: Description and evaluation of the tropospheric chemistry version 3.0. Geoscientific Model Development, 3(2):445–473, 2010. ISSN 1991959X. doi: 10.5194/gmd-3-445-2010.

[56] Benjamin Kellenberger, Diego Marcos, and Devis Tuia. Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning. Remote Sensing of Environment, 216(April):139–153, 2018. ISSN 00344257. doi: 10.1016/j.rse.2018.06.028. URL https://doi.org/10.1016/j.rse.2018.06.028.

[57] Aditya Chattopadhyay, Anirban Sarkar, and Prantik Howlader. Grad-CAM ++ : Improved Visual Explanations for Deep Convolutional Networks. arXiv, 2018.

[58] Hao Sheng, Jeremy Irvin, Sasankh Munukutla, Shawn Zhang, Christopher Cross, Kyle Story, Rose Rustowicz, Cooper Elsworth, Zutao Yang, Mark Omara, Ritesh Gautam, Robert B Jackson, and Andrew Y Ng. OGNet : Towards a Global Oil and Gas Infrastructure Database using Deep Learning on Remotely Sensed Imagery. (NeurIPS):1–8, 2020.

[59] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem: 770–778, 2016. ISSN 10636919. doi: 10.1109/CVPR.2016.90.

[60] Christian Frankenberg, Andrew K. Thorpe, David R. Thompson, Glynn Hulley, Eric Adam Kort, Nick Vance, Jakob Borchardt, Thomas Krings, Konstantin Gerilowski, Colm Sweeney, Stephen Conley, Brian D. Bue, Andrew D. Aubrey, Simon Hook, and Robert O. Green. Airborne methane remote measurements reveal heavytail flux distribution in Four Corners region. Proceedings of the National Academy of Sciences of the United States of America, 113(35):9734–9739, 2016. ISSN 10916490. doi: 10.1073/pnas.1605617113.

A
Appendix

A.1. IME verification with thoroughly studied Louisiana blowout case

In order to verify the automated IME algorithm, it was compared to a thoroughly studied case of a gas wellpad blowout [21]. The emissions arising from this event were studied in detail and quantified accurately. As a comparison, for the four days with clear plumes the corresponding 32 × 32 pixel samples were selected and the automated plume mask, source finder and IME algorithms were applied with standard settings without any finetuning or tweaking. The results are shown in Figures A.2 and A.1. A reasonably well agreement was found, but the automated IME estimate does consistently underestimate emissions, a phenomena which is well known for such emission estimates. Furthermore, in the study of the Louisiana blowout, the TROPOMI retrieval were enhanced and filtersettings were loosened specifically for that particular scene on those days in order to obtain more data. Therefore the standard 32 × 32 pixel sample which was used by the IME contains less valid pixels.
This comparison, even-though only consisting of four samples does show that the automated IME algorithm is able to provide a useful first estimate for the source rate.

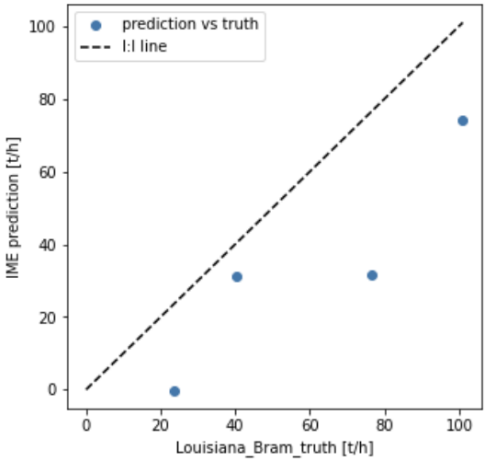


Figure A.1: comparison of source rate estimate versus thoroughly studied results in [21]

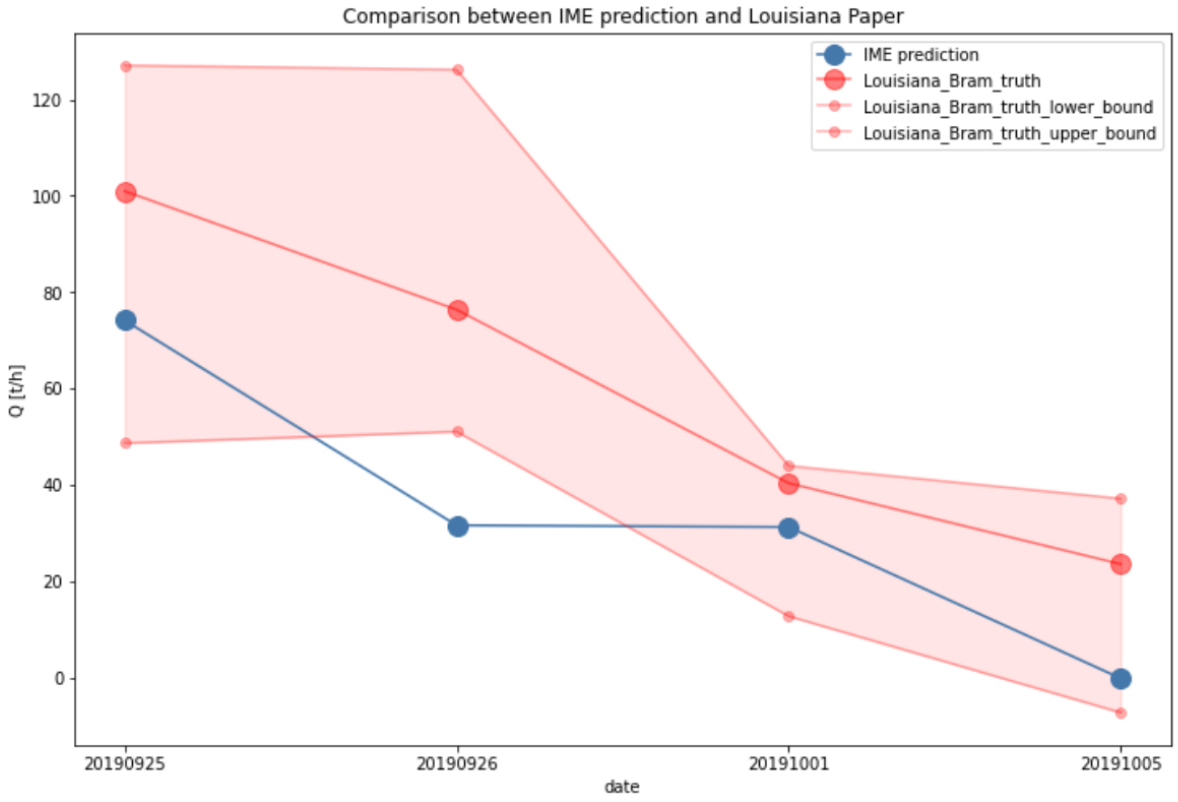


Figure A.2: Comparison of source rate estimate versus the thoroughly studied results (including uncertainty bounds) per day for the Louisiana Blowout event in 2019 discussed in [21]

A.2. Additional Tables and Figures

SRON column name	**CH_4 Level 2 dataproduct location**	**Unit**	**Typical value range**
lat_corn (lat0, lat1, lat2, lat3)	PRODUCT-SUPPORT_DATA-GEOLOCATIONS-latitude_bounds	°N	[-90, 90]
lat	PRODUCT-latitude	°N	[-90, 90]
lon_corn (lon0, lon1, lon2, lon3)	PRODUCT-SUPPORT_DATA-GEOLOCATIONS-longitude_bounds	°E	[-180, 180]
lon	PRODUCT-longitude	°E	[-180, 180]
xch4 (bias corrected)	PRODUCT-methane_mixing_ratio_bias_corrected	ppb (1e-9)	≈ 1850
xch4_uncor	PRODUCT-methane_mixing_ratio	ppb (1e-9)	≈ 1850
qa	PRODUCT-qa_value (Table 2.6)	-	[0, 0.4, 0.8, 1]
weekday	from date	-	[0, 6]
year	from date	-	[2017, 2020]
day	from date	-	[1, 31]
month	from date	-	[1, 12]
albedo_swir	PRODUCT-SUPPORT_DATA-DETAILED_RESULTS-surface_albedo_SWIR	-	[0, 1]
aerosol_op	PRODUCT-SUPPORT_DATA-DETAILED_RESULTS-aerosol_optical_thickness_SWIR	-	[0, 0.2]
aerosol_num	PRODUCT-SUPPORT_DATA-DETAILED_RESULTS-aerosol_number_column	m^{-2}	≈ [14, 17]
ch4col	(surfpres * xch4 * η / m_{dry_air} / g / 1e4) / 6.022141e19	mol m^{-2}	[0.50, 0.70]
surfpres	PRODUCT-SUPPORT_DATA-INPUT_DATA-surface_pressure	hPa	[970, 1010]
prior	(prior_col / surf_pres / η * m_{dry_air} * g * 1e4) * 6.022141e19	ppb (1e-9)	≈ 1850
prior_col	PRODUCT-SUPPORT_DATA-INPUT_DATA-methane_profile_apriori	mol m^{-2}	[0.50, 0.70]
orbitnr	specified in .nc filename	-	[0, 15000]
time	PRODUCT-time & PRODUCT-time_delta	s (ms accuracy)	seconds from TAI2010 reference

Table A.1: Description of the Level 2 data used. [32]

Normalization scheme	lowest input	highest input	NaN value after	min value after	max value after
N1	min(sample)	max(sample)	0	0	max(sample)
N2	min(sample)	max(sample)	0	0	1
N3	min(sample)	max(sample)	0	0.10	1
N4	min(sample)	max(sample)	-10	0.10	1
N5 (threshold)	mean - std	mean + 100	0	0	1
N6 (threshold)	mean - 2std	mean + 150	0	0.10	1

Table A.2: The normalization schemes used for pre-processing the xch4 samples.

index	normalization scheme	Plumes Dataset A	All other datasets
0		xch4	xch4
1		lat	lat
2		lon	lon
3		surf_pres	surf_pres
4		albedo	albedo
5		aerosol	aerosol
6		qa_val	qa_val
7		time	time
8		enh	enh
9		empty	surf_alt
10		empty	cloud_frac_VIIRS_SWIR
11		empty	surf_class
12		wind_north_V	wind_north_V
13		wind_east_U	wind_east_U
14		plume_mask	plume_mask
15		source_mask	source_mask
16		pixel_area	pixel_area
17		empty	albedo_nir
18		empty	aerosol_nir
19		empty	cloud_frac_VIIRS_NIR
20		empty	xch4_bias_cor
21		empty	xch4_uncor
22		empty	xch4_elevation_cor
23	N1	dataprep_normed_zero	dataprep_normed_zero
24	N2	dataprep_normed_zero_scaled	dataprep_normed_zero_scaled
25	N3	dataprep_min_10p	dataprep_min_10p
26	N4	dataprep_min_10p_nan_min10	dataprep_min_10p_nan_min10
27	N5	dataprep_norm_by_std_fixed	dataprep_norm_by_std_fixed
28	N6	dataprep_norm_by_std_fixed_min10	dataprep_norm_by_std_fixed_min10

Table A.3: The different layers in the standardized sample format. Mostly xch4 is used as the only channel ($n_C = 1$), but when more channels are selected the datasets containing the samples of single orbit / week of data / year of data have the dimensions: $[m, n_H, n_W, n_C] = [m, 32, 32, n_C]$.

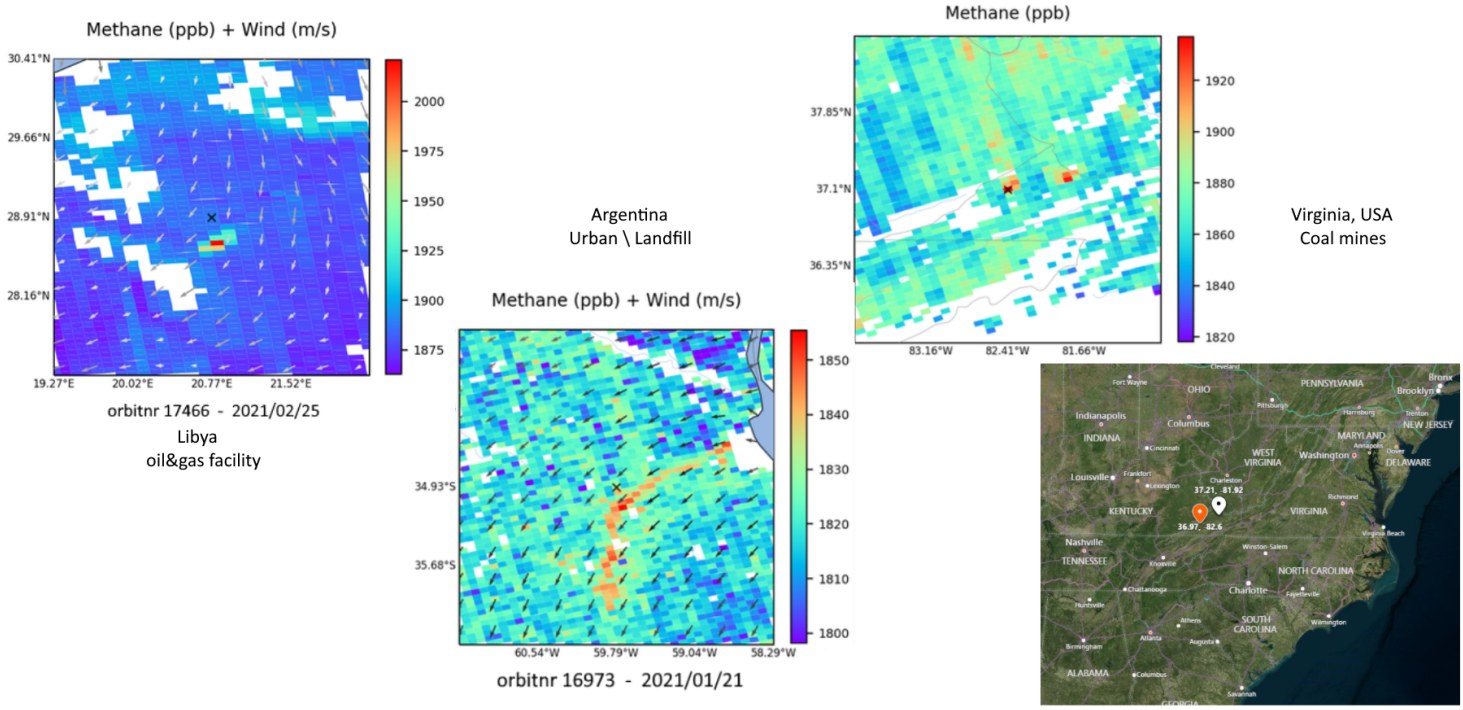


Figure A.3: Examples of detections in recent data using the fully-automated detection workflow.

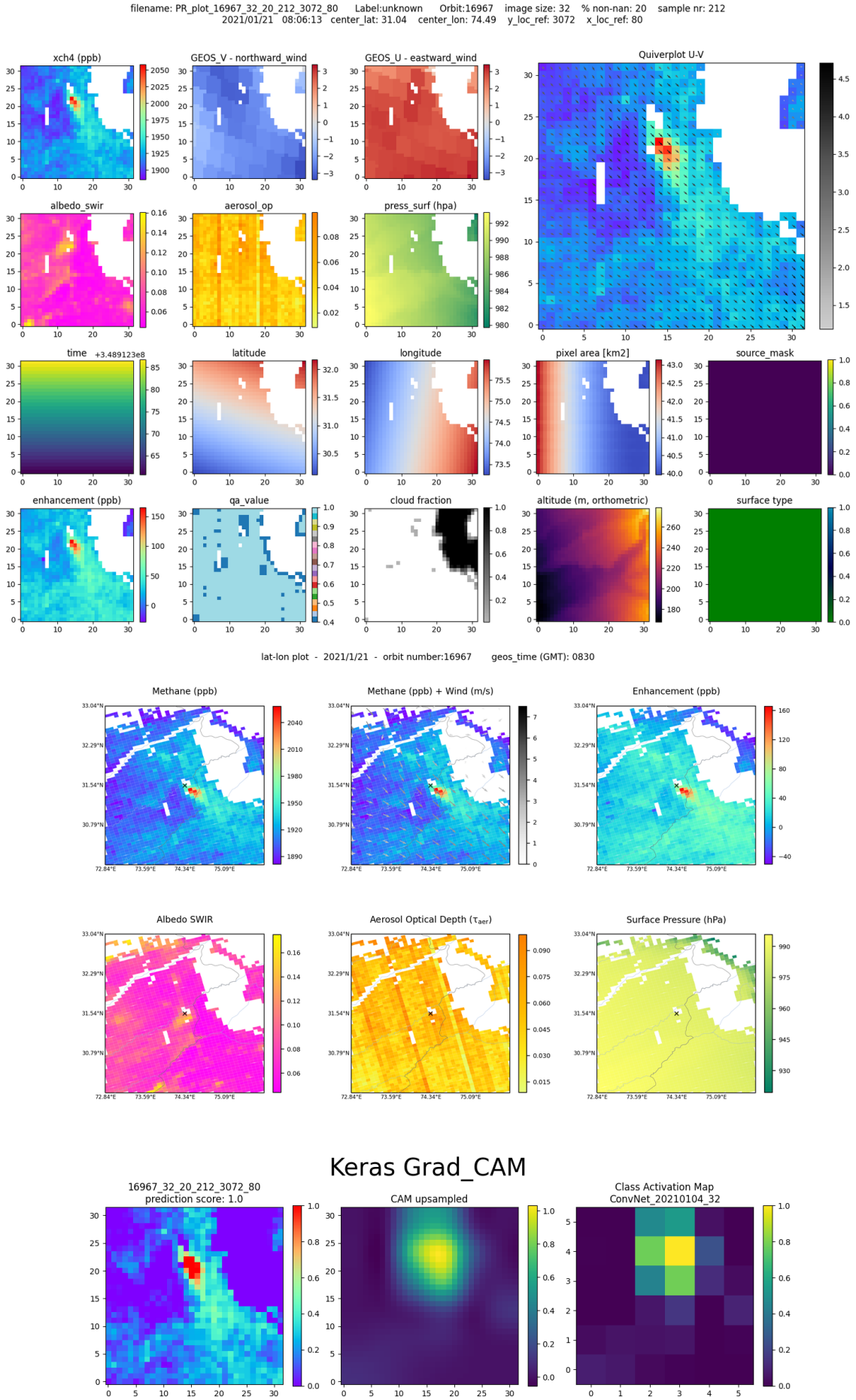


Figure A.4: Visualization of sample 16967_212, clear plume, first half

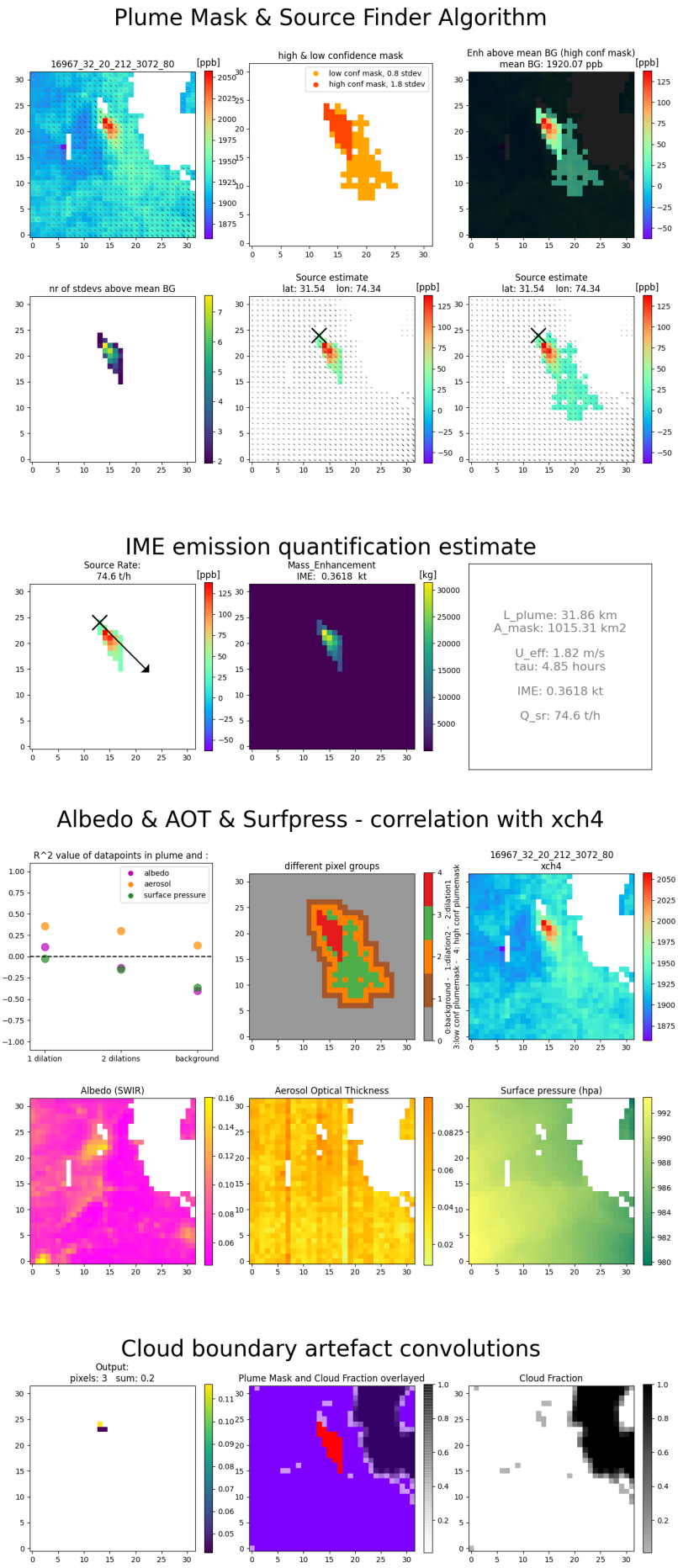


Figure A.5: Visualization of sample 16967_212, clear plume, second half

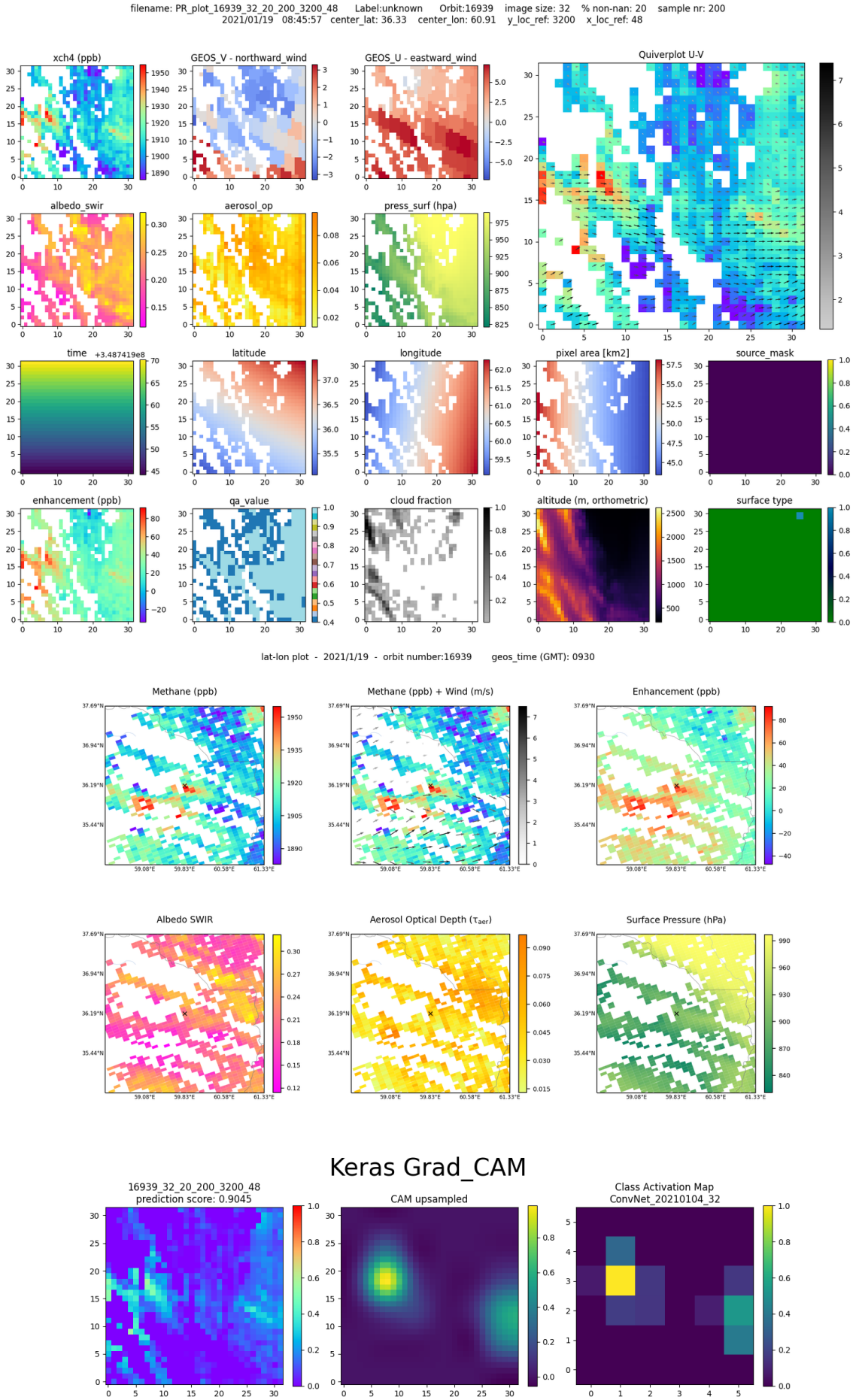


Figure A.6: Visualization of sample 16939_200, challenging plume, first half

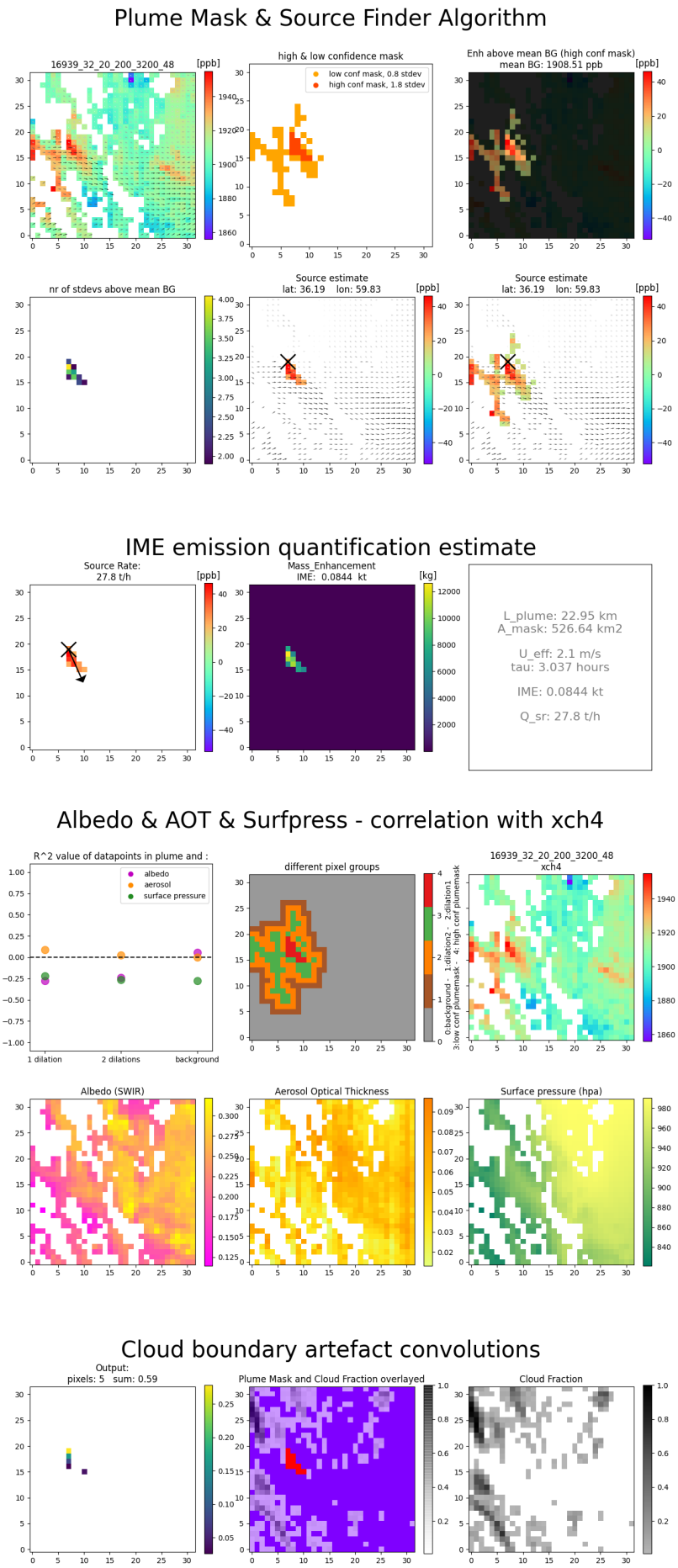


Figure A.7: Visualization of sample 16939_200, challenging plume, second half