Designing Deep-Learning models for surface and atmospheric retrievals from the IASI infrared sounder

Eulalie Boucher

▶ **To cite this version:**

Eulalie Boucher. Designing Deep-Learning models for surface and atmospheric retrievals from the IASI infrared sounder. Meteorology. Sorbonne Université, 2024. English. NNT : 2024SORUS145 . tel-04701253

HAL Id: tel-04701253

https://theses.hal.science/tel-04701253v1

Submitted on 18 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DESIGNING DEEP-LEARNING MODELS FOR SURFACE AND ATMOSPHERIC RETRIEVALS FROM THE IASI INFRARED SOUNDER

CONCEPTION DE MODÈLES D'APPRENTISSAGE PROFOND POUR LES INVERSIONS DE SURFACE ET ATMOSPHÉRIQUES À PARTIR DU SONDEUR INFRAROUGE IASI

Une thèse présentée pour l'obtention du grade de

DOCTEUR de SORBONNE UNIVERSITÉ

ÉCOLE DOCTORALE DES SCIENCES DE L'ENVIRONNEMENT D'ÎLE-DE-FRANCE (N°129)

Présentée et soutenue le 2 juillet 2024 par :

EULALIE BOUCHER

Sous la direction de :

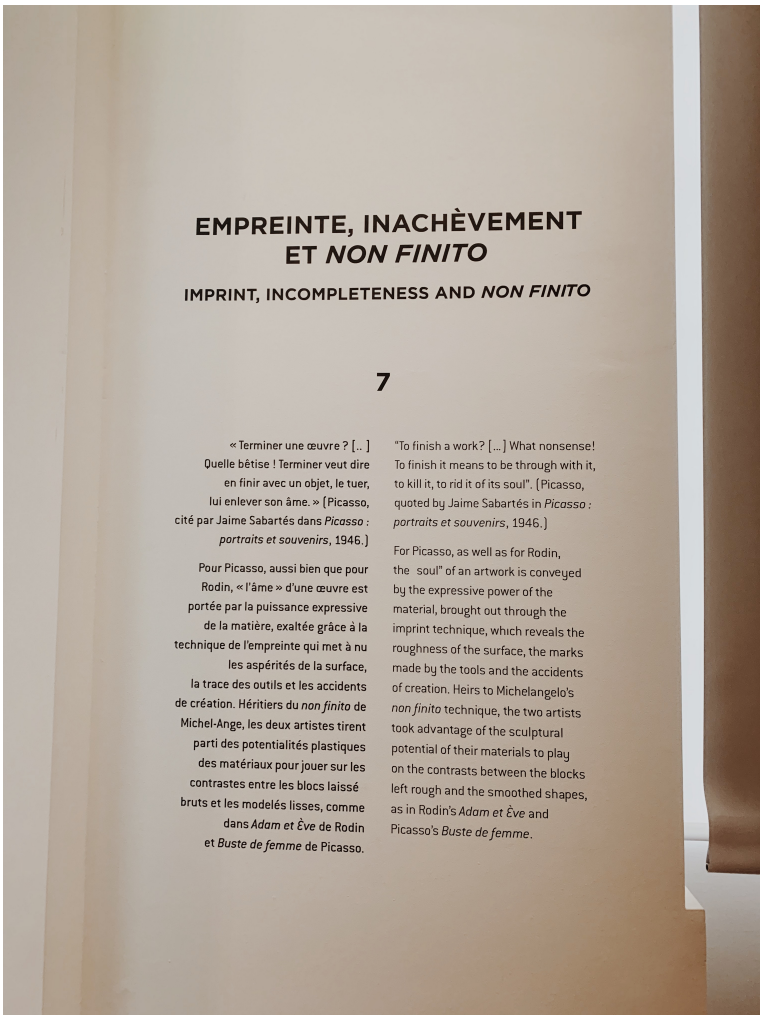
Filipe AIRES LERMA, Observatoire de Paris Directeur de thèse

Devant le jury composé de :

Filipe AIRES	LERMA	Directeur de thèse
Cathy CLERBAUX	LATMOS	Présidente du Jury
Venkatramani BALAJI	Schmidt Sciences	Rapporteur
Sid Ahmed BOUKABARA	NASA	Rapporteur
Raymond ARMANTE	LMD	Examinateur
Isabel TRIGO	IPMA	Examinatrice
Jörg SCHULZ	EUMETSAT	Examinateur
Laure CHAUMAT	Thales Services Numériques	Invitée

Eulalie Boucher: *Designing Deep-Learning models for surface and atmospheric retrievals from the IASI infrared sounder ,* © 2024

Terminer une oeuvre ?
Quelle bêtise !
Terminer veut dire en finir avec un object, le tuer, lui enlever son âme.
— Picasso, 1946



EMPREINTE, INACHÈVEMENT ET *NON FINITO*

IMPRINT, INCOMPLETENESS AND *NON FINITO*

7

« Terminer une œuvre ? [..] Quelle bêtise ! Terminer veut dire en finir avec un objet, le tuer, lui enlever son âme. » (Picasso, cité par Jaime Sabartés dans *Picasso : portraits et souvenirs*, 1946.)

Pour Picasso, aussi bien que pour Rodin, « l'âme » d'une œuvre est portée par la puissance expressive de la matière, exaltée grâce à la technique de l'empreinte qui met à nu les aspérités de la surface, la trace des outils et les accidents de création. Héritiers du *non finito* de Michel-Ange, les deux artistes tirent parti des potentialités plastiques des matériaux pour jouer sur les contrastes entre les blocs laissé bruts et les modelés lisses, comme dans *Adam et Ève* de Rodin et *Buste de femme* de Picasso.

"To finish a work? [...] What nonsense! To finish it means to be through with it, to kill it, to rid it of its soul". (Picasso, quoted by Jaime Sabartés in *Picasso : portraits et souvenirs*, 1946.)

For Picasso, as well as for Rodin, the soul" of an artwork is conveyed by the expressive power of the material, brought out through the imprint technique, which reveals the roughness of the surface, the marks made by the tools and the accidents of creation. Heirs to Michelangelo's *non finito* technique, the two artists took advantage of the sculptural potential of their materials to play on the contrasts between the blocks left rough and the smoothed shapes, as in Rodin's *Adam et Ève* and Picasso's *Buste de femme*.

RÉSUMÉ

L'observation de la Terre est essentielle pour comprendre et surveiller le comportement complexe de notre planète. Les satellites, équipés d'un certain nombre de capteurs sophistiqués, constituent une plateforme clé à cet égard, offrant une opportunité d'observer la Terre à l'échelle globale et de manière continue. Les techniques d'apprentissage automatique (ML) sont utilisées depuis plusieurs décennies, dans la communauté de la télédétection, pour traiter la grande quantité de données générées quotidiennement par les systèmes d'observation de la Terre. La révolution apportée par les nouvelles techniques de Deep Learning (DL) a toutefois ouvert de nouvelles possibilités pour l'exploitation des observations satellitaires. Cette thèse vise à montrer que des techniques de traitement d'images telles que les réseaux neuronaux convolutifs (CNN), à condition qu'elles soient bien maîtrisées, ont le potentiel d'améliorer l'estimation des paramètres atmosphériques et de surface de la Terre. En considérant les observations à l'échelle de l'image plutôt qu'à l'échelle du pixel, les dépendances spatiales peuvent être prises en compte. De telles techniques sont utilisées dans cette thèse pour l'estimation des températures de surface et atmosphériques, ainsi que pour la détection et la classification des nuages à partir des observations de l'Interféromètre Atmosphérique de Sondage dans l'Infrarouge (IASI). IASI, qui est placé à bord des satellites en orbite polaire Metop, est un sondeur hyperspectral collectant des données sur une large gamme de longueurs d'onde dans l'infrarouge. Chacune est adaptée à l'identification des constituants atmosphériques à différents niveaux de l'atmosphère, ou de paramètres de surface. En plus d'améliorer la qualité des restitutions, de telles méthodes d'Intelligence Artificielle (IA) sont capables de traiter des images contenant des données manquantes, de mieux estimer les événements extrêmes (souvent négligés par les techniques statistiques traditionnelles) et d'estimer les incertitudes des restitutions. Cette thèse montre pourquoi les méthodes d'IA, et en particulier les CNN avec convolutions partielles, devraient constituer l'approche privilégiée pour l'exploitation des observations provenant de nouvelles missions satellitaires telles que IASI-NG ou MTG-S IRS.

ABSTRACT

Observing the Earth is vital to comprehend and monitor the complex behaviour of our planet. Satellites, equipped with a number of sophisticated sensors, serve as a key platform for this, offering an opportunity to observe the Earth globally and continuously. Machine Learning (ML) techniques have been used in the remote sensing community for several decades to deal with the vast amount of data generated daily by Earth observation systems. The revolution brought about by novel Deep Learning (DL) techniques has however opened up new possibilities for the exploitation of satellite observations. This research aims to show that image-processing techniques such as Convolutional Neural Networks (CNNs), provided that they are well mastered, have the potential to improve the estimation of the Earth's atmospheric and surface parameters. By looking at the observations at the image scale rather than at the pixel scale, spatial dependencies can be taken into account. Such techniques will be used for the retrieval of surface and atmospheric temperatures, as well as cloud detection and classification from the Infrared Atmospheric Sounding Interferometer (IASI) observations. IASI, onboard the polar orbiting satellites Metop, is a hyperspectral sounder gathering data across a broad range of infrared wavelengths that are suitable to identify atmospheric constituents for a range of atmospheric vertical levels, as well as surface parameters. In addition to improving the quality of the retrievals, such Artificial Intelligence (AI) methods are capable of dealing with images that contain missing data, better estimating extreme events (often overlooked by traditional ML techniques) and estimating retrieval uncertainties. This thesis shows why AI methods should be the preferred approach for the exploitation of observations coming from new satellite missions such as IASI-NG or MTG-S IRS.

PUBLICATIONS

Aires, Filipe, Eulalie Boucher, and Victor Pellet (2021). « Convolutional neural networks for satellite remote sensing at coarse resolution. Application for the SST retrieval using IASI. » In: *Remote Sensing of Environment* 263, p. 112553. DOI: 10.1016/j.rse.2021.112553.

Boucher, Eulalie and Filipe Aires (2023a). « Improving remote sensing of extreme events with machine learning: land surface temperature retrievals from IASI observations. » In: *Environmental Research Letters* 18.2, p. 024025. DOI: 10.1088/1748-9326/acb3e3.

Boucher, Eulalie and Filipe Aires (2023b). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part II – Assessment. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1593–1611. DOI: 10.1002/qj.4472.

Boucher, Eulalie, Filipe Aires, and Marie Doutriaux-Boucher (2024). « Introducing a new partial Convolutional Neural Network for IASI cloud classification [Manuscript submitted for publication]. » In: *Submitted to IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Boucher, Eulalie, Filipe Aires, and Victor Pellet (2023). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part I – Methodology. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1180–1196. DOI: 10.1002/qj.4447.

CONFERENCES

ESA-ECMWF WORKSHOP 2021 - Oral presentation - Deep learning implementations to facilitate the assimilation of satellite observations. A case study for LST and SST using IASI observations.

IASI CONFERENCE 2021 - Oral presentation - Deep learning strategies for LST and SST retrieval using IASI observations.

EUMETSAT CONFERENCE 2022 - Oral presentation - Deep learning strategies for temperature retrievals using IASI observations.

EUMETSAT CONFERENCE 2023 - Poster presentation - Improving remote sensing of extreme events with machine learning: land surface temperature retrievals from IASI observations.

EUMETSAT CONFERENCE 2023 - Poster presentation - Convolutional neural networks for cloud classification using IASI observations.

REMERCIEMENTS

Je voudrais tout d'abord remercier V. Balaji et Sid Boukabara d'avoir pris sur eux le rôle de rapporteur de ma thèse, ainsi que tous les autres membres du jury, à savoir Raymond Armante, Cathy Clerbaux, Jörg Schulz et Isabel Trigo.

Cette thèse fut financée conjointement par le CNES et Thalès Services Numériques. Un grand merci à Clémence Pierangelo et Clément Luitot au CNES, ainsi qu'à Laure Chaumat d'avoir cru au sujet de recherche.

Un immense merci aussi à mon directeur de thèse Filipe, qui m'a toujours fait confiance, en me laissant beaucoup de liberté dans ma recherche, tout en étant toujours présent pour me conseiller et me guider. J'ai beaucoup appris pendant ces années, et je suis très reconnaissante d'avoir eu un directeur de thèse avec qui, non seulement il y avait une entente scientifique, mais également personnelle. Si tu n'avais pas accepté de me prendre en stage malgré la visite impromptue du Covid-19 en juin 2020, cette thèse n'aurait pas eu lieu, et ma carrière aurait sans doute été bien différente. Enfin, bien sûr, merci de m'avoir permis de terminer cette thèse de façon un peu accélérée.

J'en profite par la même occasion pour remercier Patricia, qui m'a donné le contact de Filipe mais aussi de précieuses informations quand il s'agissait des données de l'ECMWF. J'ai également une petite pensée pour mon professeur de physique à l'École Européenne, M. Bauer, qui serait content de voir que finalement, je retrouve un peu la physique après mes études de mathématiques.

Ces années au LERMA n'auraient pas été les mêmes sans mes collègues, que j'ai croisé plus ou moins longtemps au cours de ces dernières années : Victor (merci pour nos discussions toujours très rassurantes), Juliette (avec qui j'ai commencé cette thèse, et avec qui j'ai partagé les galères et bons moments de celle-ci), Lan Anh, Hang, Matthew, Clément, Samuel, Carlos, Megumi, Goulven, Lise, Catherine... sans oublier bien sûr l'équipe administrative qui a toujours été très efficace, mais aussi avec qui j'ai passé d'agréables moments aux pots, autour d'un cappuccino ou à la cantine. Merci à Elise, Murielle, Woihiba, Pélagie et Emi.

Je remercie particulièrement mes copines, Emilie (sans qui j'aurais probablement été en Maths-Eco et non Maths-Info), Marie, Valentine, Léna, Maureen, Maryline, Anael, Emma, Héméra... avec qui j'ai passé de nombreux merveilleux moments avant, pendant et, je n'en doute pas, après cette thèse. Merci à Emilie, de nous avoir accueillies pour des journées télétravail (avec pause brasserie le midi) et pour tous ces moments à trois avec Marie après le travail : verres de vin, pizzas Picard, apéros, fabrication de bracelets, fondue, sushis et j'en passe.

Je voudrais remercier ma famille, en particulier mes parents qui, sans vraiment le faire exprès, m'ont montré le chemin vers les sciences de l'environnement, et qui m'ont beaucoup aidée pendant ma thèse. Merci Maman pour l'accès privilégié aux données et aussi pour ta fonction plot_map qui m'aura servie à faire toutes les cartes dans cette thèse, et merci Papa pour toutes tes explications et la relecture du manuscrit aussi. Merci à ma sœur Léa, qui, n'habitant pas très loin de l'Observatoire, m'a souvent accueillie pour le goûter. Merci aussi à ma belle famille qui, eux aussi, m'ont soutenue tout au long de la thèse, en particulier Cyrielle avec qui j'ai partagé l'aventure qu'est une thèse.

Enfin, je remercie Luc-Henri (ou Boudou), celui avec qui je partage ma vie depuis bien plus longtemps que la durée de cette thèse, et qui a toujours cru en moi et soutenu dans mes choix. Durant ces trois ans de thèse, il n'y a pas seulement eu l'écriture d'un manuscrit, mais également, en parallèle, de bien d'autres projets de vie ensemble.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AI4EO	Artificial Intelligence for Earth Observation
AIFS	Artificial Intelligence/Integrated Forecasting System
AI	Artificial Intelligence
AMSU	Advanced Microwave Sounding Unit
ARSA	Analysed Radio Soundings Archive
AVHRR	Advanced Very High Resolution Radiometer
BERT	Bidirectional Encoder Representations from Transformers
BNN	Bayesian Neural Networks
BT	Brightness Temperature
CDR	Climate Data Record
CF	Cloud Fraction
CNES	Centre National d'Etudes Spatiales
CNN	Convolutional Neural Network
CO	Carbon Monoxide
DL	Deep Learning
ECMWF	European Centre for Medium-Range Weather Forecasts
ELDO	European Launcher Development Organisation
EOF	Empirical Orthogonal Function
EPS	EUMETSAT Polar System
ERA5	Fifth generation of ECMWF Atmospheric Reanalysis
ESA	European Space Agency
ESRO	European Space Research Organisation
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
FOV	Field Of View
GPT	Generative Pre-trained Transformer
IASI-NG	Infrared Atmospheric Sounding Interferometer New Generation
IASI	Infrared Atmospheric Sounding Interferometer
IFS	Integrated Forecasting System
IRS	InfraRed Sounder
LLM	Large Language Model
LMD	Laboratoire de Météorologie Dynamique
LSTM	Long Short-Term Memory
LST	Land Surface Temperature

MHS	Microwave Humidity Sounder
MLP	Multi-Layer Perceptrons
ML	Machine Learning
MSE	Mean Squared Error
MTG	Meteosat Third Generation
NEDT	Noise Equivalent Temperature Difference
NLP	Natural Language Processing
NN	Neural Network
NWP	Numerical Weather Prediction
OCA	Optimal Cloud Analysis
PCA	Principal Component Analysis
PC	Principal Component
PWLR3	Piece Wise Linear Regression
QRNN	Quantile Regression Neural Network
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RT	Radiative Transfer
ReLU	Rectified Linear Unit
SEVIRI	Spinning Enhanced Visible and Infrared Imager
SST	Sea Surface Temperature
STD	STandard Deviation
ST	Surface Temperature
TELSEI	Tool to Estimate Land Surface Emissivity in the Infrared
TIROS-1	Television Infrared Observation Satellite 1
ViT	Vision Transformer

1

AN INTRODUCTION TO ATMOSPHERIC REMOTE SENSING

Contents

This Chapter aims to briefly introduce the way radiative transfer in the atmosphere works and how it can be exploited for remote sensing. Paul (2013), Boucher (2015) and Chepfer et al. (2023) were used as a source of inspiration for writing this chapter.

1.1 PRINCIPLES OF ATMOSPHERIC RADIATIVE TRANSFER

1.1.1 *The Electromagnetic Spectrum*

Electromagnetic radiation is composed of oscillating waves of the electric and magnetic fields. The electromagnetic spectrum covers a wide range of wavelengths relevant for the atmosphere: gamma rays, X-rays, ultraviolet, visible, infrared radiation, microwaves and radio waves. Electromagnetic radiation can be characterized by its wavelength λ (usually expressed in m or μm), its frequency f (expressed in Hz) or its wavenumber ν (usually expressed in cm^{-1}). In the Earth's atmosphere, there is a natural separation between solar radiation that peaks in the visible wavelengths and terrestrial radiation that peaks in the infrared wavelengths, with some overlap in the near-infrared part of the spectrum.

1.1.2 *Interactions of Electromagnetic Radiation with Matter*

Matter interacts with electromagnetic radiation in three different ways: by scattering radiation, by absorbing radiation, or by emitting its own radiation. Molecules and particles in the atmosphere, as well as the Earth's surface, interact with radiation in one or several of these three ways. It is these interactions between the surface

and the atmosphere and electromagnetic radiation that allow us to use satellite observations to characterise the surface and the atmosphere.

(1) Scattering: Scattering is the process by which radiation is deviated in different directions by particles. If this scattering is equally distributed in all directions, the scattering is said to be isotropic. In contrast, if the radiation is deflected in a particular direction or set of directions, this is known as anisotropic or directional scattering. The part of radiation that is scattered in the opposite direction to the incident direction is said to be backscattered. Two types of scattering can be defined: elastic scattering where there is no loss of energy (this means the wavelength of the scattered radiation is the same as that of the incident radiation) and inelastic scattering which involve a change of energy (and thus wavelength of the radiation).

Rayleigh scattering corresponds to the scattering of a wave by a particle much smaller in size than the wavelength of the incident radiation. Molecular scattering predominantly follows the Rayleigh scattering phenomenon, which is elastic and symmetrical between the forward and backward directions (although not quite isotropic). The amount of scattering is inversely proportional to the wavelength to the power four ($1/\lambda^4$). For example, a blue radiation will be more scattered than a red radiation since $\lambda_{blue} < \lambda_{red}$, which is why the sky is blue. Indeed, the Sun emits radiation in the entire visible domain and the blue corresponds to the most diffuse radiation by the atmosphere.

Mie scattering occurs when the scattering particles are of similar size to the wavelength of the incident radiation. The intensity of the scattered radiation depends on the refractive index of the particles, the wavelength, the size and the shape of the particles (assumed to be spherical here). The larger the particles are, the more directional and forward-looking the scattering is. Scattering by aerosols and liquid cloud particles can largely be explained by Mie scattering. Scattering of solar radiation varies relatively little with the wavelength of the radiation, which explains that clouds appear white or grey (a mixture of all the different colours of radiation from the Sun).

(2) Absorption: Molecules can exist in different electronic states, quantified by their amount of energy. A molecule can therefore move away from its fundamental (stable) state if it receives the right amount of energy, this is called a transition. In practice, vibrational and rotational transitions are associated to electronic transitions. Therefore, a molecule can reach a number of excitation states corresponding to different electronic, rotational and vibrational energy levels. As the energy levels are discrete, a precise amount of energy is required for the molecule to move from one state to another, as illustrated in Figure 1. In this figure, the fundamental state, where the molecule has less energy, is represented at the bottom and the exited states are above. These states require energy to be reached. The energy levels required to reach intermediary levels are unique to each molecule as the vibrational and rotational states depend on its structure. The energy contained in electromagnetic radiation can cause an excitation, or in the extreme case, an ionisation or dis-

sociation of the molecule it interacts with. This exchange of energy occurs through the absorption of photons whose energy levels depend on the wavelength of the radiation. The energy E of a photon propagating at frequency f is worth $E = h \times f$, where $h = 6.62617.10^{-34}$ J.s is the Planck constant. In the microwave, this energy is thus relatively little, and can thus only generate rotations of the molecule. In the infrared, the energy is larger and vibrations can be generated in addition to rotations. In the visible and ultraviolet, the energy is even larger and this can generate even more changes in the electronic state of molecules, for instance dissociation or ionisation.

The combination of scattering and absorption is called extinction.

(3) Emission: Emission can be understood through the concept of a blackbody. By definition, a blackbody is a body that absorbs the entirety of the incident radiation, and emits its own radiation, in order to balance its energy budget. This emission is subject to Planck's law, and therefore follows the following spectral distribution:

$$B_\lambda(T) = \frac{2\,h\,c^2}{\lambda^5}.\frac{1}{e^{(\frac{h\,c}{\lambda\,k\,T})} - 1} \tag{1}$$

where λ is the wavelength of the radiation (in m), $h = 6.62617.10^{-34}$ J.s is the Planck constant, $k = 1.38066.10^{-23}$ J.k^{-1} is the Boltzmann constant, $c = 299792458$ m.s^{-1} is the speed of light, and T the absolute temperature (in kelvin or K) of the body. For instance, the Sun can be considered, as a first approximation, as a blackbody with a 5800 K temperature, to which absorption rays are added to represent the presence of absorbing chemical elements in its atmosphere. Its radiation ranges, for the large part, from the ultraviolet to the near infrared, and most of the energy is in the visible range. In contrast, terrestrial radiation is typical of that emitted by a blackbody with a temperature of 288 K, corresponding to the average temperature of the Earth's surface. It is strongest in the thermal infrared. As it is much hotter, the Sun emission is much larger than that of the Earth (approximately 10^{14} for the Sun vs. 10^7 W.m^{-2}.sr^{-1}.m^{-1} for the Earth).

Each state of the molecule (vibration, rotation etc.) corresponds to a certain level of energy, and the photon needs to have exactly the energy between two states for the molecule to absorb it and change states. Thus, only the few photons that have the correct level of energy are absorbed, which explains the fact that each molecule absorbs radiation in different spectral bands. Similarly, a molecule in an excited state can then emit radiation and return to a more stable state. The temperature of the atmosphere is such that most of the molecules are in their stable state. Only molecules at the top of the atmosphere are in excited states, because they are exposed to much higher solar radiation (and due to the low density of the atmosphere, which implies there are less energy exchanges through collisions). The transition of these molecules from excited states to the fundamental state leads to emission of photons in the ultraviolet and the visible. In the remaining layers of the atmosphere, the temperature and the larger number of molecule shocks leads to them being in

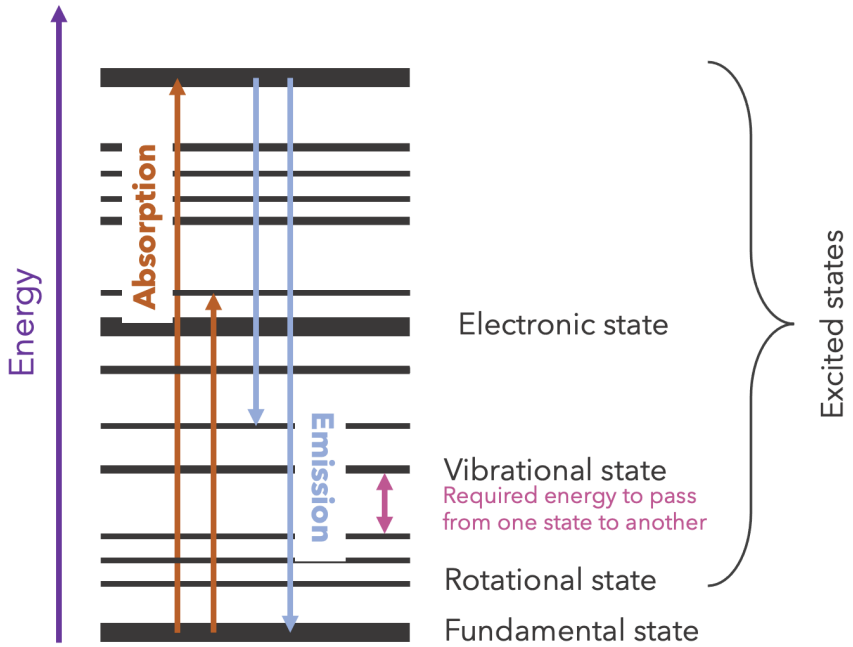


Figure 1 – Different energy levels for a given molecule. The electronic states are shown with a thicker line. The number of vibration and rotation levels are for illustration purposes and are not exact. The red and blue arrows show possible excitation and de-excitation of the molecule through absorption and emission of a photon. See Text.

their fundamental state, and they transition preferably from vibrational and/or rotational states back to the fundamental state. The radiation emission is thus low in energy (i. e. most of it is in the infrared). To balance out the absorbed energy, the atmosphere emits radiation. The absorption and emission coefficients are equal when local thermodynamic equilibrium is reached according to Kirchhoff's law. The radiation absorbed by the molecules is entirely re-emitted. Emission will therefore take place in the same spectral bands as absorption. However, atmospheric molecules do not emit in the same direction as the incident radiation.

1.1.3 *The Radiative Transfer Equation*

Radiation can be characterized by a physical quantity called radiance, and usually noted L, that depends on location (x, y, z), direction of propagation $\vec{s}$, wavelength λ and time. The Radiative Transfer (RT) Equation describes how radiation propagates in a medium. In its Lagrangian form along a direction $\vec{s}$ and a coordinate l, the RT Equation can be written as:

$$\frac{\mathrm{d}L_\lambda(l,\vec{s})}{\mathrm{d}l} = -\sigma_\lambda^{\mathrm{ext}}\, L_\lambda(l,\vec{s}) + \sigma_\lambda^{\mathrm{abs}}\, B_\lambda(T) + \frac{\sigma_\lambda^{\mathrm{sca}}}{4\pi}\left\{\int\int_{4\pi} \mathbf{p}_\lambda(\vec{s},\vec{s}_i)\, L_\lambda(l,\vec{s}_i)\, \mathrm{d}\omega_i\right\} \quad (2)$$

where $\sigma_\lambda^{\text{ext}}$, $\sigma_\lambda^{\text{abs}}$ and $\sigma_\lambda^{\text{sca}}$ are the extinction, absorption and scattering coefficients, $\mathbf{p}_\lambda(\vec{s}, \vec{s}_i)$ is the phase function (describing the scattering of radiation in direction $\vec{s}_i$ into the direction $\vec{s}$) and $\mathrm{d}\omega_i$ is the elemental solid angle for the integral over all incident directions. The RT Equation is thus an integral-differential equation whereby three terms contribute to the change in radiance: a loss term due to the extinction of radiation, a source term due to emission of radiation, and a source term due to scattering coming from other directions.

Various simplifications of the RT Equation exists for the atmosphere. For instance, there is no emission term in the visible and the scattering term is sometimes neglected in the thermal infrared. The atmosphere is often considered to be plane parallel, i. e. it is considered to be homogeneous in the horizontal. The RT Equation can be solved through different techniques depending on the quantity and accuracy that are required.

Solving the RT Equation is a “direct problem” in that it consists of calculating the radiances, knowing all the characteristics of the atmosphere and surface. We can for instance simulate the radiance coming out of a particular point in a particular direction as measured from a satellite instrument.

1.2 SATELLITE REMOTE SENSING

1.2.1 *Concept and Principles*

Several techniques exist to observe the atmosphere, either from above using satellites, within the atmosphere itself using airborne instruments onboard planes or balloons, but also directly from the surface.

There are two main categories of sounding techniques illustrated in Figure 2:

1. Passive sounding consists of measuring the spectral radiance received on a given point from natural radiation. The source of radiation thus comes from the Sun, the Earth’s surface or the various components of the atmosphere.
2. Active sounding consists of emitting radiation and measuring the spectral radiance received in return. In this case, the source of radiation is artificial and imposed by the sounder. Other sources of radiation (i. e. due to atmospheric molecules or particles) are assumed to be negligible in comparison to the artificial radiation. Examples of active sounding include lidar and radar systems operating in the visible or infrared and in the microwave spectrum, respectively.

The most frequent sounding onboard satellites is passive because the technology is more mature. This thesis therefore focuses on passive satellite remote sensing. As previously mentioned, satellite passive sounding consists of measuring outgoing spectral radiance at the top of the atmosphere. This spectral radiance is the result of interactions between the Earth’s atmosphere/surface and electromagnetic radiation. The different constituents of the atmosphere (i. e. water vapour,

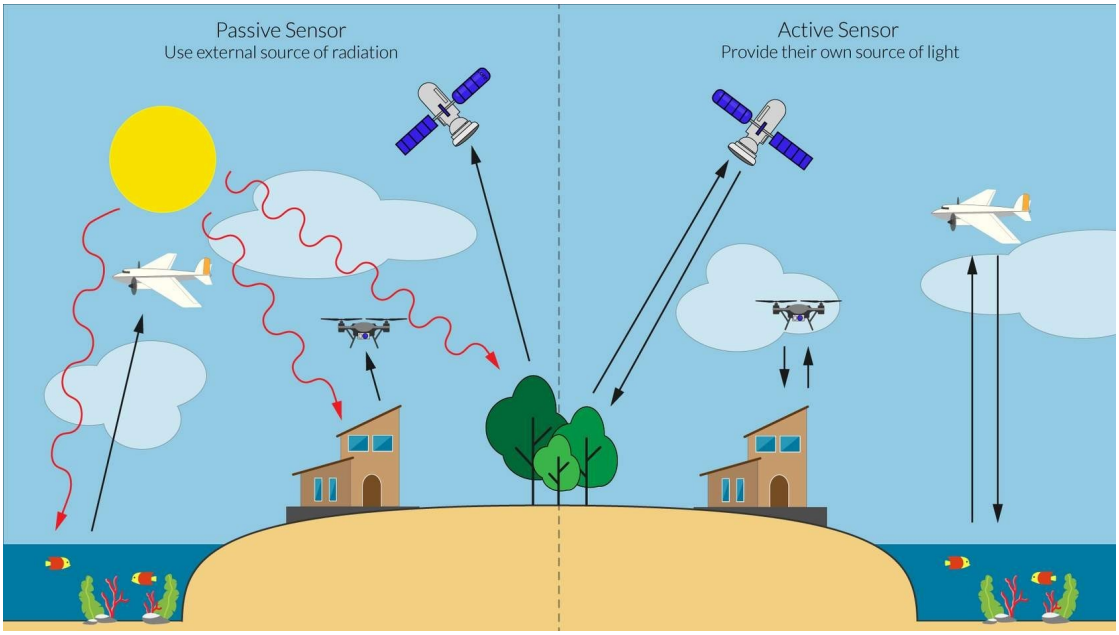


Figure 2 – Difference between active (right) and passive (left) sounding techniques, using different types of instruments (plane, drone, satellite). Source: The Nature Conservancy

carbon dioxide, ozone etc.) interact with the electromagnetic radiation at different wavelengths. By measuring the spectral radiation at several wavelengths it is possible to measure these interactions and characterize the different constituents of the atmosphere. The radiances (i. e. the observations) are then transformed into geophysical variables describing the state of the atmosphere. The conversion between the observations and the geophysical quantities is achieved through complex algorithms that aim to inverse the RT Equation. This is an “inverse problem”, which consists in measuring radiance and deduce from this measure the desired geophysical quantity by finding the inverse of the RT Equation in some form. The unknown variables here are the geophysical quantities.

Two main approaches exist to inverse the RT Equation, and both are often combined:

1. Physical methods that try to find the inverse of the RT Equation mathematically. This is a very complex and undetermined problem since radiances depend on a large number or geophysical variables, whose values are often unknown and vary a lot in time and space. The inverse problem is therefore weakly constrained because one set of radiances can be associated with a number of geophysical quantities. To reduce the number of possible solutions, constraints can be applied during the inversion. An approach often used relies on *Look-up Tables*. This consists in calculating, with the RT Equation, the radiances that would be observed by the satellite in a large but finite number of atmospheric situations. The *Look-up Tables* store the link between the theoretically calculated radiances and the geophysical quanti-

ties. The observations (i. e. the measured radiances) can then be compared to the precomputed radiances, and potential solutions can be extracted.

2. Statistical methods where the geophysical estimates are inferred statistically based on an observation-based database. Indeed, the inversion of satellite observations has become a huge playing field for Artificial Intelligence (AI) and Machine Learning (ML) methods because of the complexity of the inverse problem and the large amount of observations collected over the years. AI techniques use past observations to parameterise statistical methods ranging from linear regressions to more advanced ML and Deep Learning (DL) techniques.

This thesis will be focused on using statistical methods, and more particularly ML techniques.

1.2.2 *Exploiting the Electromagnetic Spectrum*

Wavelength	Wave number (cm $^{-1}$)	Name	Uses
$\leq$ 5 pm	$\geq 2.10^9$	Gamma rays	Nuclear fallouts
5pm - 10nm	2.10^9 - 10^6	X-rays	Medical imagery, border control
10nm-400nm	10^6 - 25000	Ultraviolet rays	Tanning, sterilisation, light therapy
400nm - 700nm	25000 - 14000	Visible light	Human vision
700nm - 1mm	14000 - 10	Infrared	Thermal cameras, heaters, remote controls
1 mm - 1m	10 - 10^{-2}	Microwave	Wi-Fi, Bluetooth, microwave oven, meteorological radars, phone
$\geq$ 1m	$\leq 10^{-2}$	Radio waves	Radio

Table 1 – The different uses of electromagnetic radiation.

Table 1 details the electromagnetic spectrum and the practical uses of each range. Figure 3 illustrates this, along with the atmospheric absorption rates and the correspondence between units. A large part of the electromagnetic spectrum is strongly absorbed by the atmosphere. It is not possible to sound the atmosphere in such part of the spectrum because, as explained above, it is the radiation's ability to penetrate and interact with the atmosphere that gives us the opportunity to access its characteristics. Where the atmosphere absorbs 100% of the radiation, a satellite can only see the top layers of the atmosphere (all radiation emitted from lower layers will be entirely absorbed by the higher layers). This is why, as it can be seen on the figure, atmospheric sounding instruments perform measurements in what is called "atmospheric window" regions of the spectrum. These regions (with a low absorption rate) are somewhat "transparent" to the atmosphere. The spectral variation of the absorption rate and the atmospheric transmission is what allows us to observe atmospheric properties at different layers of the atmosphere.

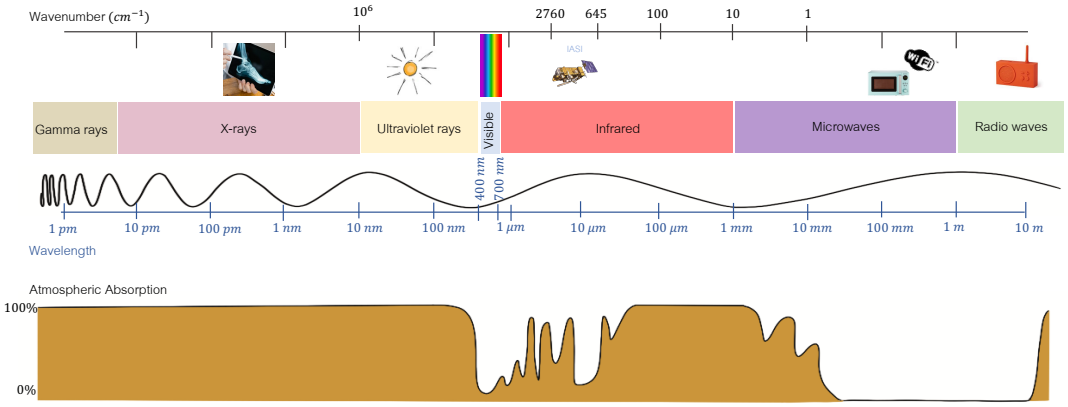


Figure 3 – Representation of the electromagnetic spectrum. The most common uses of the different wavelength ranges are shown.

1.2.3 *Infrared Atmospheric Sounding*

Principle absorbents in the infrared spectrum are water vapour (H_2O), carbon dioxide (CO_2) and ozone (O_3). Numerous other trace gases (less present in the atmosphere), such as methane (CH_4) or carbon monoxide (CO) are also active in the infrared. Since diatomic particles (that represent a majority of the molecules in the atmosphere) are almost inactive in the infrared, we can measure the absorption by the minor components. Precisely measuring the radiation emitted by the atmosphere and the surface at different wavelengths allows measuring the absorption bands of each atmospheric constituent and infer their mixing ratios. This is the case with new generation hyper-spectral infrared sounders such as Infrared Atmospheric Sounding Interferometer (IASI) (see Section 2.1).

Measuring atmospheric temperature and water vapour is another possibility when sounding the atmosphere in the infrared. For instance, absorption bands of water vapour gives direct access to its concentrations, whilst the CO_2 absorption bands help to determine the atmospheric temperature. However, this is only possible in clear sky situations. Hydrometers (e. g.., liquid droplet and ice crystal particles) are very absorbing in the infrared spectrum, which makes clouds very opaque in this part of the electromagnetic spectrum. It is therefore not possible to sound the atmosphere within and below a cloud in the infrared from space. This is an important characteristic of infrared sounding because it has a large influence on how inversion algorithms are chosen. Indeed, clouds cover approximately 70% of the Earth's surface at any time (Rossow and Schiffer, 1999). Infrared measurements from space are therefore bound to have a large amount of impractical data, that we will denote, in this thesis, as "missing". However it should be noted that sounding in those cloudy areas is still possible using, for instance, cloud-clearing techniques.

2

SATELLITES, INSTRUMENTS, DATABASES AND PRE-PROCESSING TECHNIQUES

Contents

The observation of the Earth from space is a fundamental discipline that, over the years, has brought a notable advantage to the human understanding of the Earth system. The history of Earth observation dates back to the 1960s where the first satellites were launched. Artificial satellites are human-made objects, sent into space with a rocket and orbiting around the Earth (or another planet) similarly to a natural satellite (like the Moon) orbits around the Earth. The energy transmitted to the satellite by the rocket to place it on its orbit which can then be maintained for a long time, typically for several years.

In 1957 the Union of Soviet Socialist Republics (USSR) launched the first satellite, called Spoutnik 1. Since then, almost 16,000 objects were launched into space, 11,000 of which are still in orbit today, and 7,000 of which are actually operational in 2023. Nowadays, satellites play an important role for all aspects of society, and serve many economic sectors (telecommunications for example) but also military, as well as scientific research of course.

Before this, the observation of the Earth relied on *in situ* measurements, performed from surface stations, radiosondes, planes, buoys, etc. : but Earth observation satellites have revolutionized our knowledge of the atmosphere and its functioning. It has notably considerably changed the way meteorology, environment and climate are studied.

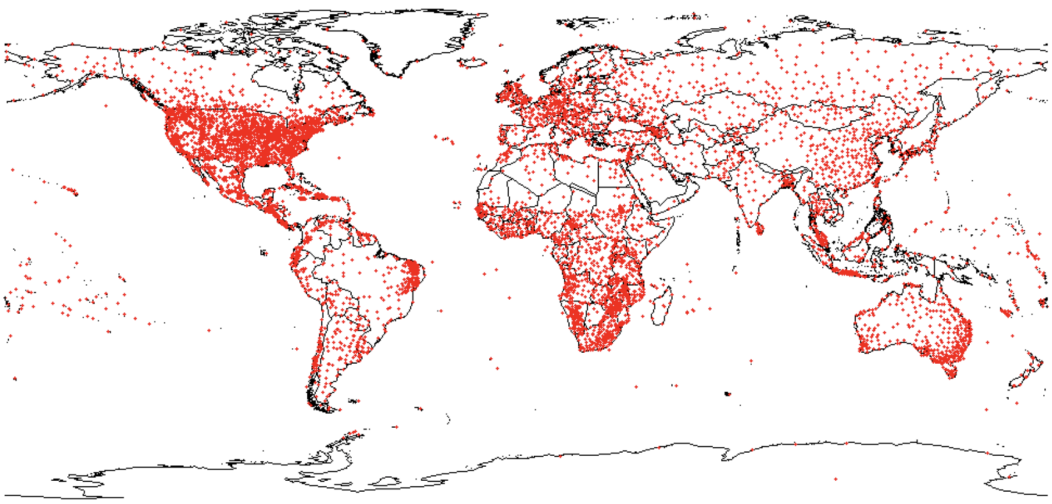


Figure 4 – Locations of about 6,000 temperature stations contributing to version 1 of the Global Historical Climatology Network (GHCN) dataset. Source: https://daac.ornl.gov/CLIMATE/guides/CDIAC_NDP41.html.

Meteorological satellites aim to collect the data necessary for weather forecasting and the study of the Earth's climate. They are the only tool allowing a global coverage of the Earth's surface with sufficient temporal repetitiveness. This is especially useful for uninhabited areas such as the polar regions, oceans, deserts, as well as developing countries, that are more difficult to observe *in situ*. Figure 4 shows the density of weather stations around the World, and shows the inequality of *in situ* monitoring between areas.

Global satellite products thus emerged in the 1960s with the first Television Infrared Observation Satellite 1 (TIROS-1) (Phulpin et al., 2024) and this was the start of European and International collaborations to accommodate spatial missions meeting the needs of national space agencies. For instance, the European Space Agency (ESA) was founded in 1975, when the European Space Research Organisation (ESRO) and European Launcher Development Organisation (ELDO) merged together. ESA's mission is to foster cooperation among European States in space research and technology and space applications. It establishes a cohesive space and develops the associated industrial strategy, making recommendations to member states regarding space goals, and positioning national initiatives, such as satellite development, within the broader European framework to the greatest extent feasible. Similarly, the European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT) is also an intergovernmental organization created through an international convention currently signed by 30 European Member States (and that entered into force in 1986). EUMETSAT has the mission to establish, maintain and exploit European operational meteorological satellites. It is responsible for the launch of the satellites to the analysis and delivery of the data.

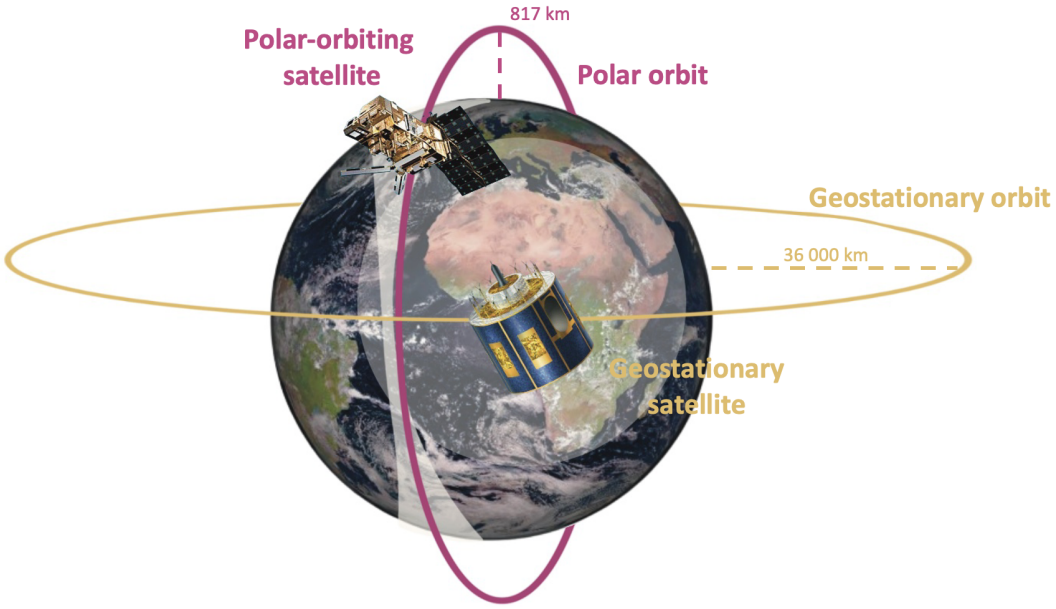


Figure 5 – Illustration of geostationary (yellow) and polar (pink) orbits. The picture of the Earth was distributed by EUMETSAT.

This chapter will present the various satellites, instruments and datasets of major interest for this thesis. The main instrument we consider is the IASI instrument, placed onboard the Metop satellites. We will first describe IASI and its observations, before moving on to the derived Level 2 IASI products. Reanalysis data from Numerical Weather Prediction (NWP) models that we employ consistently throughout the thesis, will then be presented and described. During the thesis, the Spinning Enhanced Visible and Infrared Imager (SEVIRI) instrument was needed at several occasions; this chapter will also describe the instrument. Other datasets of less importance to the thesis will be described in the chapters in which they come in useful.

2.1 THE IASI INSTRUMENT

2.1.1 *The Metop Satellite*

The Metop satellites consist of a family of 3 (Metop-A, -B and -C) polar-orbiting meteorological satellites. They were developed and launched by ESA and are operated by EUMETSAT. The first Metop-A satellite was launched in 2006 and was the first polar orbiting satellite used for meteorological purposes. Its two companions, Metop-B and -C were launched in 2012 and 2018 respectively. The three satellites are identical, which allows for the continuity of data acquisition. In 2021, Metop-A reached its end-of-life in orbit and has since then been switched-off.

The Metop orbit was chosen to complement geostationary satellites; Indeed, a geostationary satellite (shown in yellow in Figure 5) is positioned 36 000 km above the Equator and rotates around the centre of the Earth at the same speed as

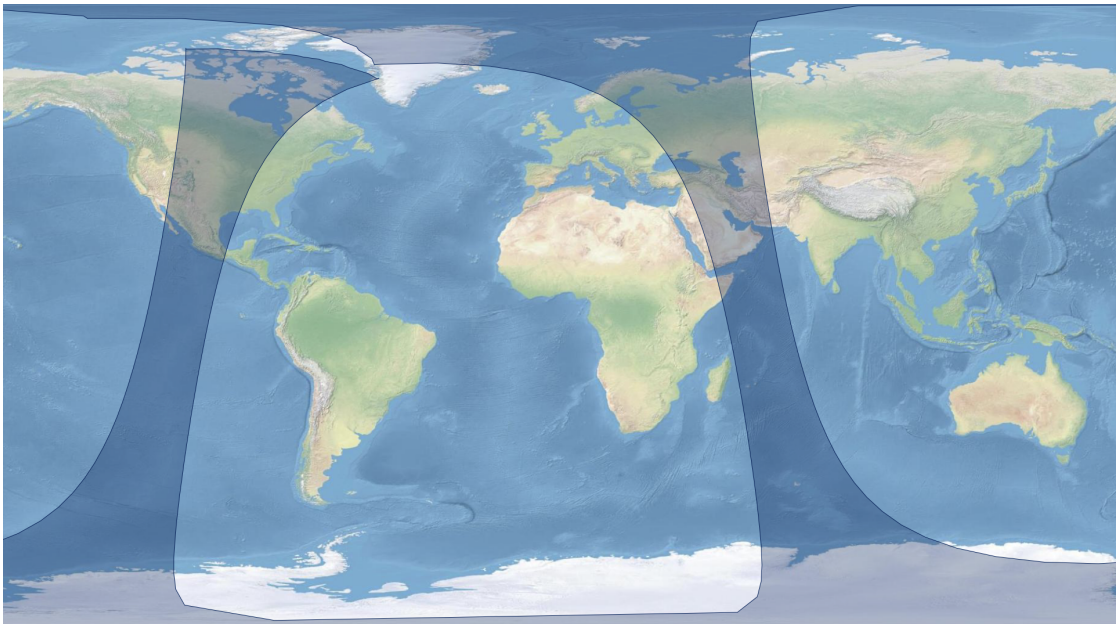


Figure 6 – An example of a IASI orbit footprint. Source: EUMETSAT Data Store https://data.eumetsat.int

the Earth itself. This means that the satellite is always positioned over the same part of the Earth, that is monitored continuously. Consequently, several geostationary satellites are needed to monitor the entire globe, which raises a number of inter-calibration issues. Furthermore, geostationary satellites are placed above the Equator and therefore cannot see the poles, where important weather events occur with influences on mid-latitudes.

The Metop satellites orbit at 817 km from the Earth surface, with a sun-synchronous polar orbit (shown in pink in Figure 5 and in grey in Figure 6), meaning that it passes over both poles at each orbit and crosses the Equator at the same local time. Metop satellites fly over each part of the globe twice daily. It crosses the equator a first time at 9:30 local time and a second time at 21:30 local time. Metop circles the Earth (corresponding to one orbit) in approximately 101 minutes, resulting in 14 daily orbits. Each orbit can be split into a descending (from the North Pole to the South Pole) and an ascending part (from the South Pole to the North Pole). Descending orbits fly over the equator in the daytime (9:30 local time), whilst ascending orbits fly over the Earth at nighttime (21:30 local time). Being quite close to the Earth, at an altitude of 817 km, the satellite does not take snapshots of the globe, but acts rather like a scanner that, as can be seen in white on Figure 5, only sees a strip of the Earth's surfaces.

The Metop satellites carry twelve instruments that all help to accomplish their primary missions that are to acquire atmospheric temperature profiles and observe clouds, winds and the Earth's surface. An important purpose is to "introduce" the data into weather forecasting models. Some additional instruments are on-board for humanitarian purposes. The list of the instruments is given below:

1. IASI, an interferometer probing upward radiation with wave numbers between 645 and 2760 cm^{-1} which allows the measurement of various atmospheric profiles and surface characteristics;
2. AMSU-A1 and -A2, microwave sensors with 15 channels [1] between 23 and 90 GHz for measuring atmospheric temperature, precipitation and surface characteristics;
3. MHS, a microwave sounder with five channels between 89 and 190 GHz to measure atmospheric humidity;
4. A-DCS, a UHF signal sensor used to collect environmental data;
5. ASCAT: C-band scatterometer used mainly for wind measurements on the ocean surface;
6. AVHRR, an imaging radiometer with 6 channels in the visible, near infrared and infrared spectrum, that can image land, oceans and clouds;
7. GOME-2, a visible and ultraviolet spectrometer for ozone measurements;
8. GRAS, a radio receiver (measuring GPS signal attenuation) for measuring temperature and humidity profiles;
9. HIRS, an infrared sounder with 19 channels between 666 and 2631 cm^{-1} and one channel in the visible range. This “historic" instrument measures atmospheric temperature profiles as well as cloud height;
10. S&R, a VHF and UHF receiver, transmitter and signal processor that enables the reception and processing of emergency signals from ships and aircraft;
11. SEM-2, a multichannel spectrometer for measuring different ionized particle flows.

This thesis will focus mainly on the IASI instrument. However, Advanced Very High Resolution Radiometer (AVHRR) will be mentioned here and there, as they are complementary.

2.1.2 *The IASI Instrument*

The IASI (Blumstein et al., 2004) instrument was designed by the Centre National d’Etudes Spatiales (CNES). During the satellite’s advance, IASI scans the Earth’s surface underneath. It uses a cross-track scan, whereby the instrument’s swath is perpendicular to the satellite track (or trace), as illustrated by Figure 7. The orbit advance is shown in white. The black ellipses on the Earth’s surface show the zone of measurement. It can be seen that the angle at which it observes the surface (shown in orange on the figure) varies, ranging from 0° to 48.3° on either side of the track. The angle between the vertical below the satellite and the radiation received by the sensor is called the viewing zenith angle. Surveying at zero viewing angle is called nadir. The viewing angle has an impact on the measured zone on the surface,

1. A channel can be defined from the notion of frequency at which the instrument measures the radiance.

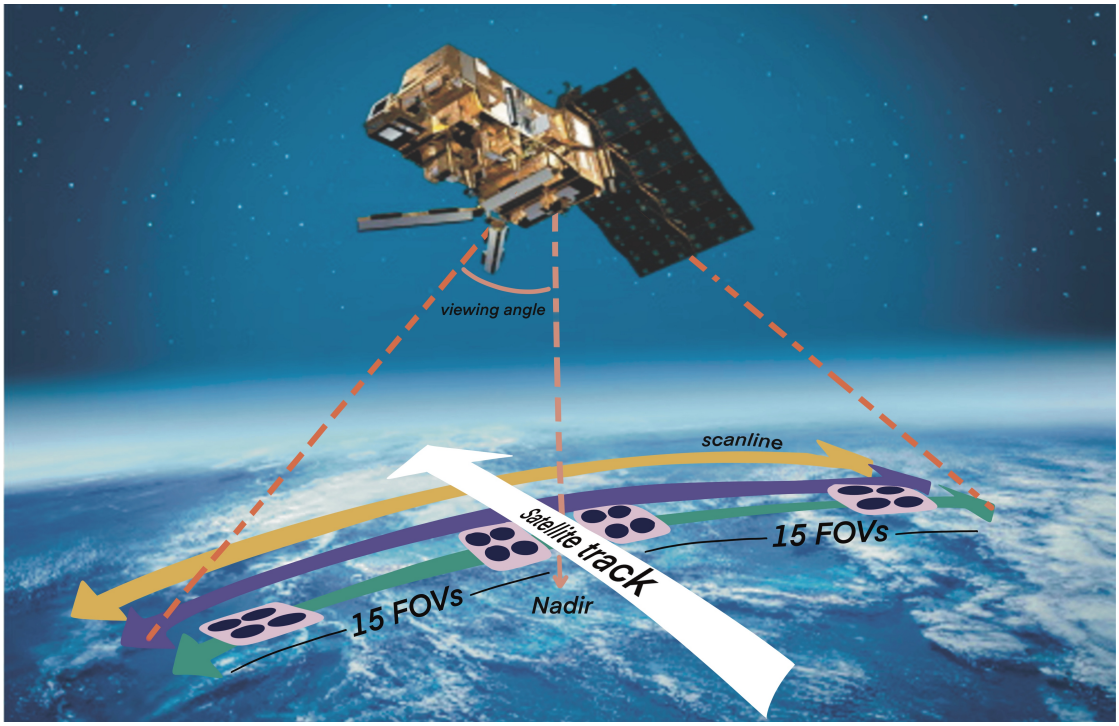


Figure 7 – Schematic diagram of the IASI observing system. The photograph of the Metop satellite was extracted from the ESA website.

but also on the atmosphere thickness it passes through. The larger the viewing angle (i. e. the more the instrument aims to the side), the thicker the atmosphere that the surface radiation has to pass through before reaching the instrument. In addition, the size and shape of the pixels on the ground also varies with the viewing angle: smaller (approximately 12 km diameter) and circular near the track, and larger (approximately 40 km in diameter) and elliptical at the edges of the swath. This cross-track viewing allows the IASI instrument to see every point on the globe in just two days, whereas it takes Metop five full days to fly over every point on the Earth.

The IASI instrument acquires its measurements with 30 cells (or Field Of View (FOV)) across each scan line (15 on each side of the satellite track). Each FOV is composed of a 2 $\times$ 2 pixel grid. These pixels (the smaller measurement units) are what we count as a measurement point of the IASI instrument. This can be seen both in Figures 7 and 8. As previously mentioned, the viewing angle has an impact on the shape and size of the observed pixels. This is evident in Figure 8. Approximately 850 scan lines make up a polar orbit.

2.1.3 *Measurement Principle*

The IASI instrument is a Michelson interferometer that measures the infrared radiation coming out of the atmosphere at a large number of wavelengths. The Fourier transform spectrometer of IASI decomposes the infrared radiation emitted by the

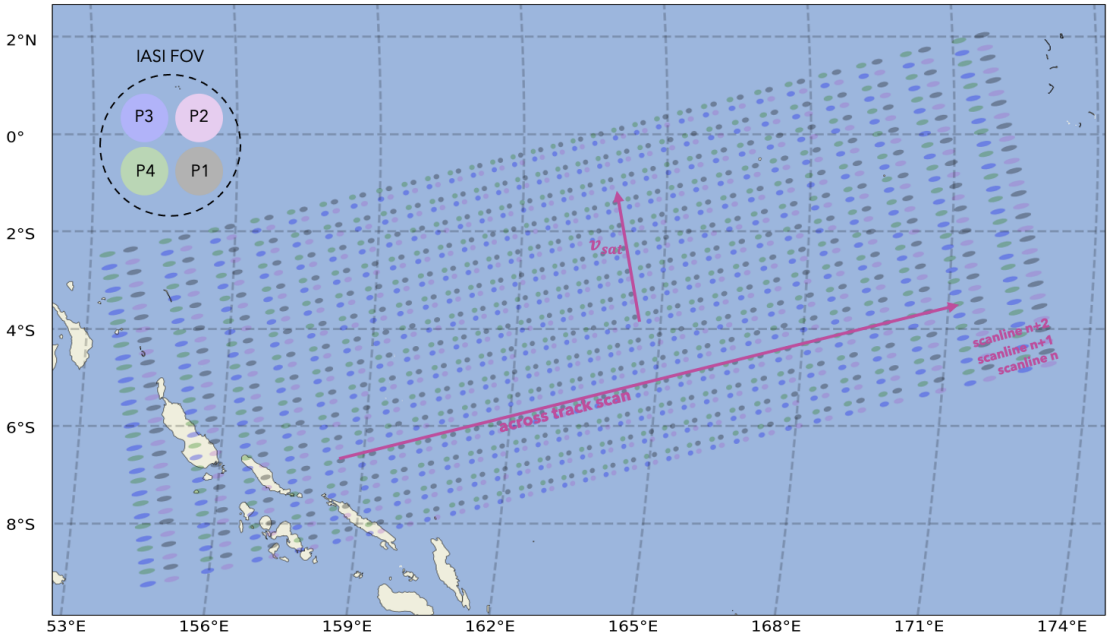


Figure 8 – Example of the acquisition geometry of IASI pixels for an ascending orbit. The vertical arrows show the motion direction of the satellite orbit, whilst the horizontal arrows indicate the direction of the across track scan. Each IASI FOV is made up of four pixels: P1 (grey), P2 (pink), P3 (purple) and P4 (green)

atmosphere and the Earth's surface. The 8461 channels provide a high resolution infrared spectrum between 645 and 2760 cm^{-1} (equivalent to between 3.6 and 15.5 μm wavelengths). Each measurement point being made up of 8461 values, the amount of data measured by IASI is huge, and represents almost half of the daily data transferred by the Metop satellite. Its fine spectral resolution (0.25 cm^{-1} un-apodized) enables the instrument to retrieve skin temperature, temperature and water-vapour profiles in the troposphere and the lower stratosphere, as well as other atmospheric constituents. IASI measurements are split into three bands:

— Band 1 B1 from 645 cm^{-1} to 1210 cm^{-1}
— Band 2 B2 from 1210 cm^{-1} to 2000 cm^{-1}
— Band 3 B3 from 2000 cm^{-1} to 2760 cm^{-1}

As explained in the previous Chapter (see Section 1.2) the different parts of the spectrum can be used to measure a particular constituent or property active in it. For IASI, this can be seen in Table 3, and in Figure 10(a), where an example of a IASI spectrum is plotted, and its absorption bands highlighted. In other words, each part of the spectrum is sensitive to different characteristics and layers of the atmosphere. Different channels are used simultaneously to obtain information on one of the atmospheric variables (temperature, water vapour, etc.). Figure 9 shows the weighting functions for selected channels. The weighting functions are the derivative of the brightness temperature (for a given IASI channels as a function of the atmospheric temperature on a given pressure level (unit K/K) and thus represent

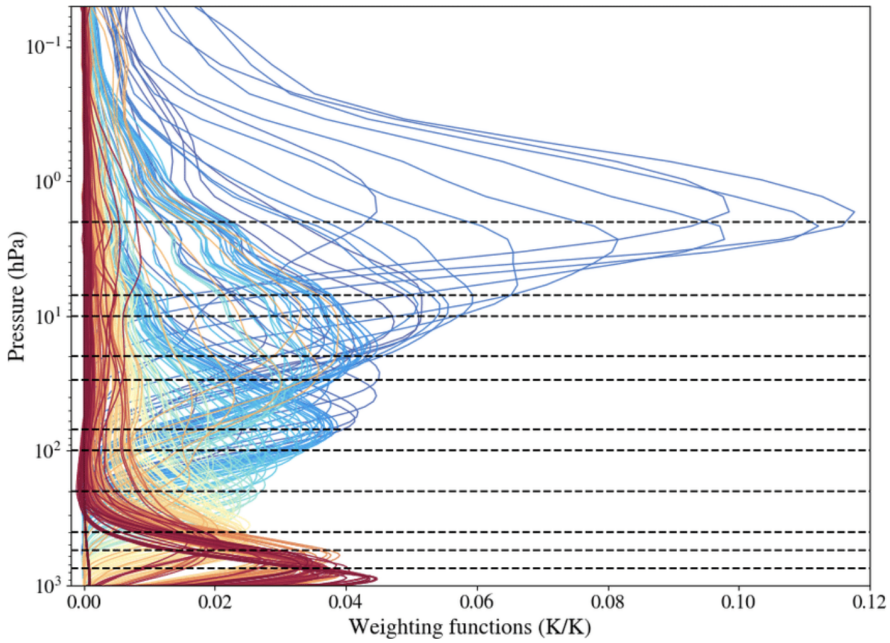


Figure 9 – Weighting functions for 231 selected IASI channels (channel selection presented in Pellet and Aires (2018)). The blue lines represent channels in Band 1 and the red lines are for Band 2. Source: Bouillon et al. (2022), Creative Commons license.

the sensitivity of the Brightness Temperature (BT)s at the top of the atmosphere to atmospheric temperatures.

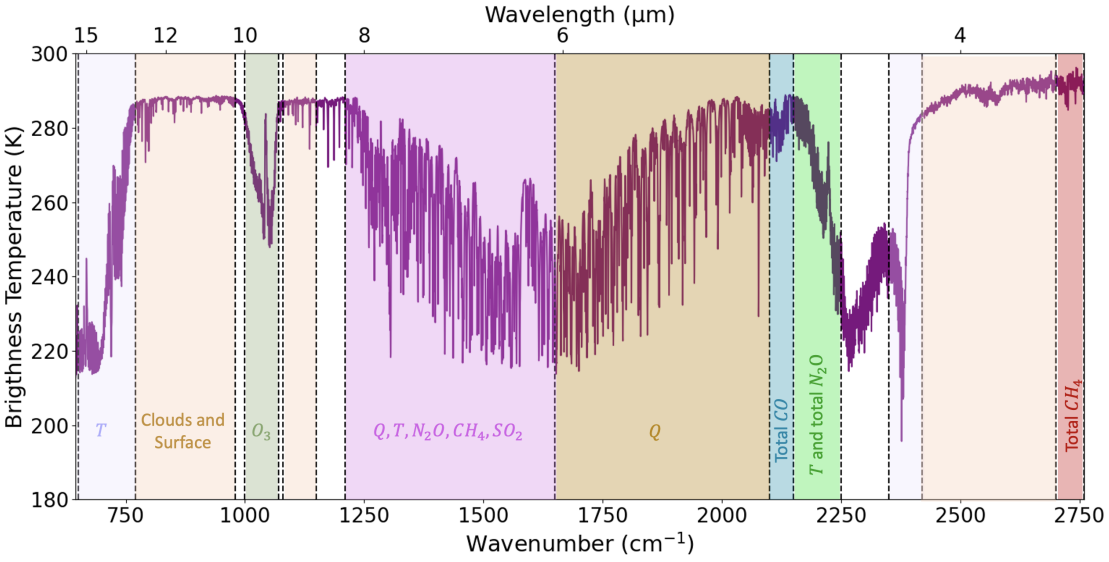
The information contained in each channel is coupled with the others to better deduce the wanted variables. The first two bands are the most used, since band B3 has a very low signal-to-noise ratio and is therefore difficult to exploit.

The IASI instrumental noise (i. e. its radiometric resolution) Noise Equivalent Temperature Difference (NEDT) is the standard noise at 280 K. Its specifications are provided by the CNES in Cayla and Javelle (1995) and shown in Figure 11(a) in orange. The detailed values of the specified noise are given in Table 2. From this figure, it is possible to understand why band 3 measurements are often left out of retrieval algorithms.

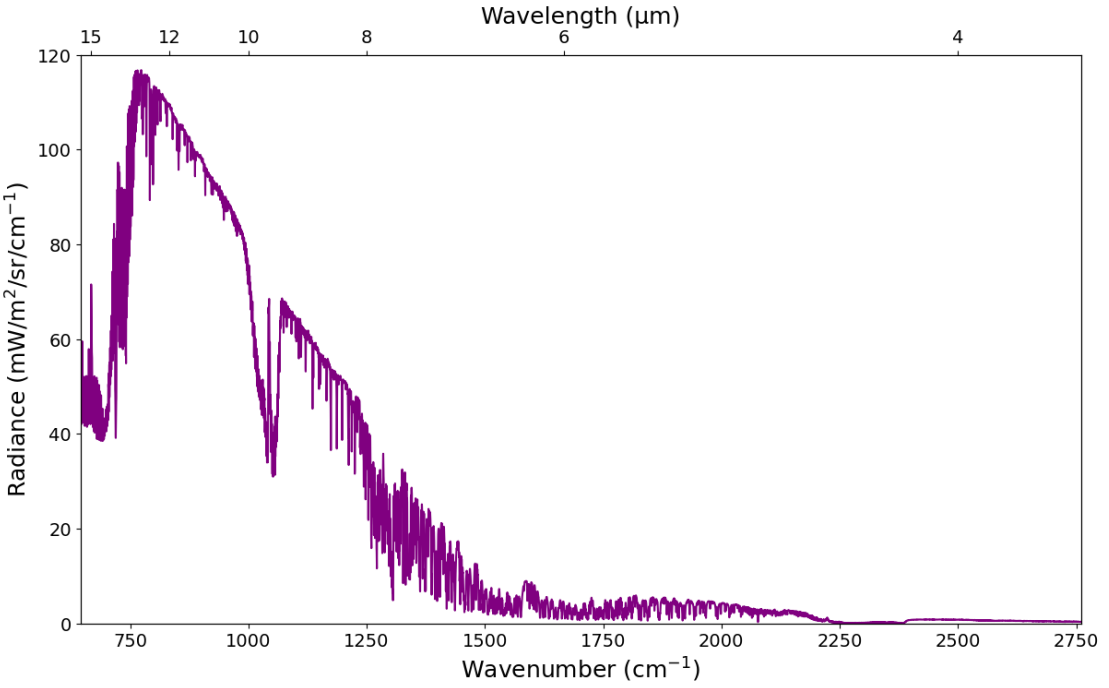
The measured NEDT at 280 K is also plotted in this figure (in red), and the detailed values are given in Appendix A.1. The $NEDT_\lambda$ at 280 K is the noise at 280 K for a given wavelength λ. At a different temperature T', the $NEDT_\lambda$ is given by the following formula:

$$NEDT_\lambda(T = T') = \frac{\frac{\partial B_\lambda(T=280\text{K})}{\partial T}}{\frac{\partial B_\lambda(T=T')}{\partial T}} \dot{N}EDT_\lambda(T = 280\ K),$$

where $B_\lambda(T)$ is Planck's law, defined in Equation 1.



(a)



(b)

Figure 10 – Example of the IASI (a) BTs, (b) radiances spectrum made up of 8 641 channels, shown both in wavenumbers in cm^{-1} (bottom x-axis) and wavelengths in μm (top x-axis). This spectrum was extracted from Metop-A data on 2016/03/25.

Using this formula, it is possible to plot the measured IASI noise for a particular spectrum. This is done for the spectrum presented in Figure 10, and results in the IASI noise spectrum shown in Figure 11(b).

Wavenumber (cm^{-1})	$NEDT_\lambda(280)$ (K)	Wavenumber (cm^{-1})	$NEDT_\lambda(280)$ (K)
650	0.28	1660	0.34
770	0.28	2090	0.5
790	0.34	2100	0.5
980	0.34	2420	0.5
1000	0.28	2430	0.6
1070	0.28	2500	0.77
1080	0.34	2600	1.1
1200	0.34	2700	1.58
1210	0.28	2760	1.97
1650	0.28		

Table 2 – Overview of the IASI spectral domain and associated geophysical variables.

2.1.4 *Level 1 IASI Data*

The interferograms measured by IASI are processed onboard the Metop satellite to calibrate and correct the measurements. The data is also geolocated and dated. The processed data is then sent to the EUMETSAT Polar System (EPS) receiving stations, which dispatches the data to the Level 1 ground segment where the raw (Level 0) IASI data are transformed into Level 1a, 1b and 1c data.

- Level 1a consists of calibrated spectra;
- Level 1b is obtained by resampling the Level 1a data to a nominal wavenumber scale;
- Level 1c data builds on Level 1b data to obtain a spectral response to a predefined function. In the end, the Level 1c products consist of radiances (electromagnetic radiation received on a surface in a given direction at a given wavenumber). They are expressed in watt per m^2 per steradian per cm^{-1}, but they can also be expressed in terms of BT in kelvin (K) via the black body law.

It is the Level 1c product that will be used in this study, the specificities of which will be detailed in the different relevant chapters. Figure 10 shows the IASI Level 1c (a) BTs, and (b) radiances, for one measurement point (i. e. one IASI pixel extracted from one orbit of March 25th, 2016). This figure shows the radiance values of the 8461 IASI channels for this point.

2.1.5 *The Spectral Principal Components Analysis*

As said in the previous section, each IASI measurement point is made up of 8461 channels and thus, 8461 values. Each Level 1c orbit file represents 2.3 gigabytes

Wavenumber (cm^{-1})	Geophysical variable
650-770	Temperature (T)
770-980	Cloud and surface properties
1000-1070	Ozone (O_3)
1080-1150	Cloud and surface properties
1210-1650	Humidity (Q), T, N_2O, CH_4, SO_2
1650-2100	Q
2100-2150	Total Carbon Monoxide (CO)
2150-2250	T and total N_2O
2350-2420	T
2420-2700	Cloud and surface properties
2700-2760	Total CH_4

Table 3 – Overview of the IASI spectral domain and associated geophysical variables

of data, or 12 terabytes for a year's worth of one IASI instrument data. In addition to the computational limitations this can bring, it is obvious that there is redundancy of information between some of the 8461 channels.

In order to ensure a viable usage of IASI data, it is obvious that developing dimensionality reduction techniques is necessary. For this, the Principal Component Analysis (PCA) method is very effective. PCA (García-Sobrino, Serra-Sagristà, and Bartrina-Rapesta, 2017; Jolliffe, 1986) has been used for the last 25 years to compress the information contained in a large number of channels into a small number of uncorrelated (and orthogonal) components suitable for the retrieval of a certain variable (Aires et al., 2002a,b).

More generally, PCA is a dimensionality reduction algorithm whose objective is to transform a high-dimensional dataset into a lower-dimensional representation while retaining as much of the original variance as possible. PCA achieves this by identifying and capturing the principal components of the data. The Principal Component (PC) are the new orthogonal "axes" in the data space that capture the maximum variance. The first PC explains the most variance, the second the second-most variance, and so on. Each PC is a linear combination of the original features. In total, there are as many PCs as there are dimensions in the original dataset. Keeping all the PCs would retain 100% of the variance, but the idea is to keep a smaller number of PCs in order to reduce dimensionality, whilst keeping as much of the variance as possible.

Given a dataset X with n data points in an m-dimensional space (each data point has m features, for example, this would be the 8461 IASI channels). The first step is

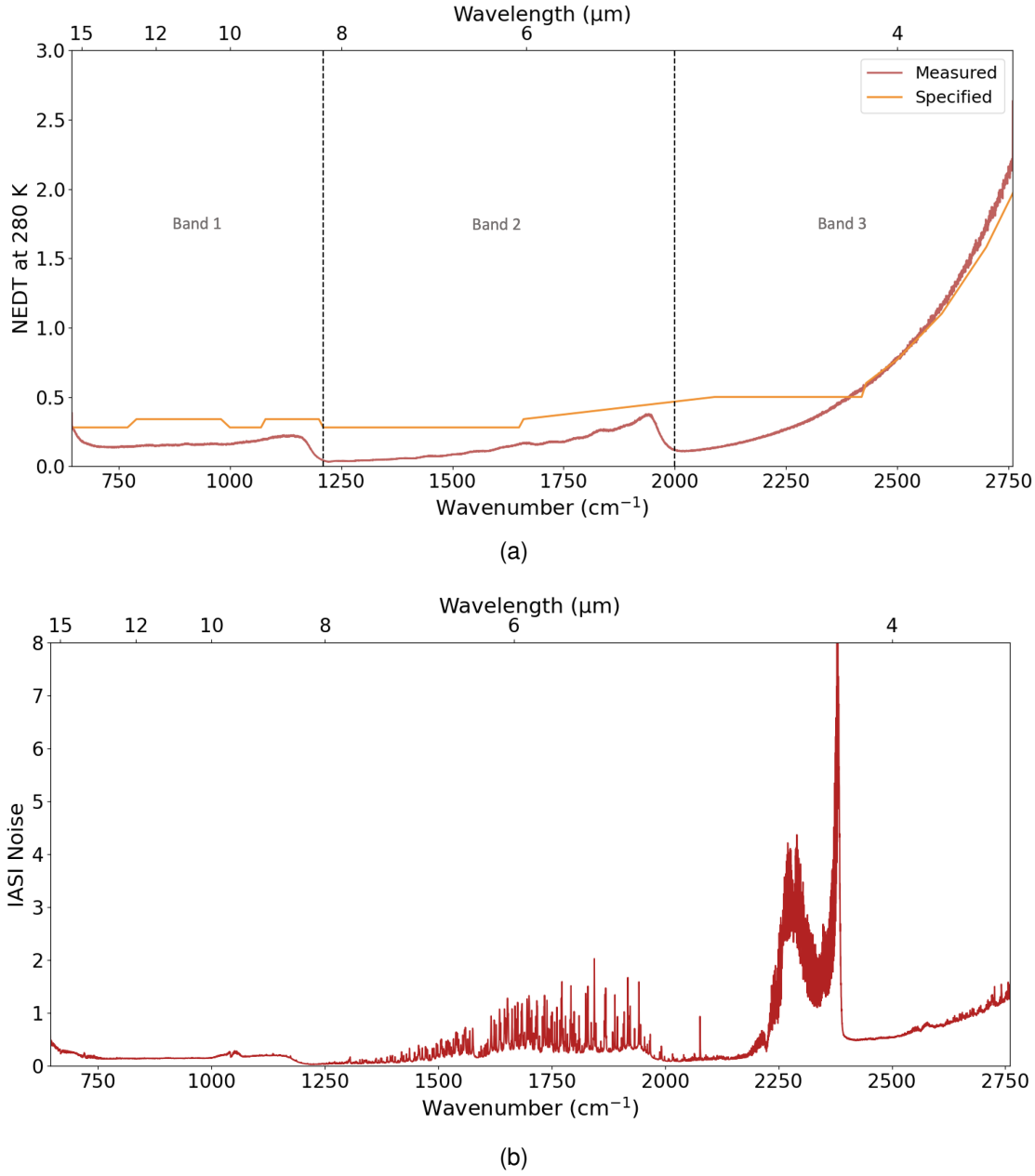


Figure 11 – (a) The measured (in red) IASI NEDT noise compared to the CNES specifications (in orange) at 280 K. (b) The derived measured noise across the spectrum extracted from Metop-A data on 2016/03/25 (shown in Figure 10).

to compute the covariance matrix σ. The covariance matrix represents the pairwise covariances between the different features, and is calculated as follows:

$$V = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T,$$

where x_i is a data point vector, $\bar{x}$ is the mean of all data points, and T denotes the transpose.

Then, the eigenvalues λ of the covariance matrix have to be found. This is typically done by solving the characteristic equation $\det(V - \lambda I) = 0$, with I being the identity matrix of the same shape as V ($m \times m$). The corresponding eigenvectors v can be found by solving the equation $VX = \lambda X$. The eigenvalues represent the amount of variance explained by each principal component, and the corresponding eigenvectors represent the direction of each principal component. We can thus sort the eigenvalues and keep the first k, depending on how much variance we want to conserve. This highly depends on the application, and there is no predefined way of determining how many PCs should be kept.

The PCs are obtained by projecting the dataset X onto the k dimensions defined by the eigenvectors v by computing $P = X.v$. P is a matrix of shape $n \times k$. This matrix P serves as the new dataset, with reduced dimensionality.

The PCA algorithm can be applied to the entire IASI spectrum, or only a subset of channels, depending on the utility of the desired new dataset. More details on which subset of channels this processing was applied will be given in the different chapters of the thesis.

2.1.6 *The IASI Principal Component Scores Data Record*

EUMETSAT has recently released the IASI Principal Component Scores (PCS) Fundamental Data Record for Metop-A and -B (EUMETSAT, 2022a). This dataset is obtained with the PCA algorithm presented in Section 2.1.5. EUMETSAT applies the PCA algorithm individually to each of the three IASI spectral bands. Using this dataset, the full spectrum can be reconstructed using the eigenvectors, but it is also possible to directly use the PCS.

It could also be considered that the PCA algorithm is performed onboard the satellite, and only the PCs be sent to Earth. This is in fact something that will be given thought in future missions such as InfraRed Sounder (IRS) and Infrared Atmospheric Sounding Interferometer New Generation (IASI-NG) that will generate even larger amounts of data.

2.1.7 *IASI Level 2 Data*

The challenge of remote sensing is to convert what is measured by the satellite into physical quantities that characterize the Earth's atmosphere and surface. This is an inverse problem that requires solving the radiative transfer equation in the atmosphere. A number of methods exist for this, some of which were described in Chapter 1. These inversions result in so-called Level 2 products.

Several Level 2 IASI products are distributed via EUMETCast (EUMETSAT's platform for automatic, near-real-time dissemination of surface and satellite data) (Murphy, 2013). These are the operational Level 2 Products. Near-real time products

undergo changes in the inversion algorithms through time and also suffer from occasional failure.

This is the reason why alongside the near-real time dissemination, EUMETSAT also provides IASI Climate Data Record (CDR)'s. CDRs are reprocessed (long) time-series of a variable. The reprocessing ensures that the time series is of sufficient length, consistency and quality. For instance, the IASI CDR is derived from the (partly operational) "all-sky IASI Piece Wise Linear Regression (PWLR3) retrieval"[2]. The PWLR3 is a machine learning algorithm (see EUMETSAT (2022) and EUMETSAT (2020) for more details) trained with real satellite observations. It nominally exploits IASI measurements in synergy with collocated microwave data from the Advanced Microwave Sounding Unit (AMSU) and Microwave Humidity Sounder (MHS) companion instruments onboard Metop. This makes the retrieval possible in all-sky conditions. The training is performed with about 100 million of IASI, MHS and AMSU observations, collocated Fifth generation of ECMWF Atmospheric Reanalysis (ERA5) atmospheric variables.

The CDR files include four surface parameters (surface pressure, surface air temperature, surface air dew point temperature, and surface skin temperature) as well as the atmospheric profiles of temperature and humidity. In addition, the CDR also provides a cloud indicator and an uncertainty estimate for each product.

2.2 NUMERICAL WEATHER PREDICTION MODEL DATA

In complement to observations of the Earth provided by satellites and other instruments, Earth system models aim to simulate all Earth processes and interactions. Forecast models also introduce observations into the model through the process of data assimilation. Of these models are derived long-term databases describing the state of the atmosphere. Such databases will be of great interest for the works presented in this thesis, and thus a brief description of the models is made in this section.

2.2.1 *NWP Models and Data Assimilation*

NWP or forecast models, are based on complex mathematical equations that describe the behaviour of the atmosphere. It describes, on one hand, the dynamics of the atmosphere (i. e. fluid dynamics, mass conservation,etc.) and on the other hand the physics (i. e. wind-surface interactions, convection,etc.). These models aim to simulate the evolution of weather variables such as temperature, wind, pressure, humidity over time.

Data assimilation is the process of integrating observational data into these models to determine their initial conditions, called an analysis, and improve the accuracy of weather forecasts. The analysis is therefore the best possible estimate of the

2. DOI: 10.15770/EUM_SEC_CLM_0027

state of the atmosphere at a certain time. Data assimilation works by determining an initial state x that is as close as possible to x_b, the "estimate" of the atmosphere obtained with the latest forecast (called the background) and such that $H(x)$ is as close as possible to the observations y, where H is the operator that maps the model outputs to the observation space. The observations include satellite observations from several instruments (in particular, IASI data accounts for 40% of the data), but also radiosondes data or surface observations.

To do this, we minimize a cost function J, that is weighted according to the error covariance matrices associated to both the background B and the observations R:

$$J(x) = (x - x_b)^T B^{-1}(x - x_b) + (y - H(x))^T R^{-1}(y - H(x)).$$

Note that the observations y cannot be directly compared to the state x, because satellite observations measure radiances that are linked to the model output through the radiative transfer equation. It is therefore necessary to first map the model output x to the observation space, with the operator H.

Minimizing this cost function can be done, for example, with an iterative method, like done at European Centre for Medium-Range Weather Forecasts (ECMWF) in their 4D-Var method (Talagrand, 2014), that relies on gradient descent. However, it would be too complicated and too costly to calculate the true gradient by differentiating the system against each input variable. The "trick" is therefore to use what is called the adjoint model, a key component in the data assimilation process. It works by calculating sensitivity information about how changes in the model's initial conditions (for example, small changes in temperature or pressure at various locations and altitudes) affect the forecasted variables at a future time. This sensitivity information is critical for adjusting the model's initial conditions to better match the observed data. The adjoint model does this efficiently by working backward in time from the forecast period to the initial time. The adjoint model therefore outputs an approximation of the gradient, which can be used to update x. A few iterations are needed until convergence. The analysis is then used to launch the forecast. This process is done every 12 hours.

2.2.2 *Reanalysis: The ERA5 Dataset*

Every few years, when the model has improved significantly, the ECMWF uses the current forecast model and data assimilation systems to reprocess archived observations. This creates a global and hourly dataset that describes the atmosphere, land and oceans from 1959 onwards. This is called a reanalysis, and the ERA5 reanalysis dataset is the latest available production. The dataset uses a single model version and data assimilation method to ensure consistency over the entire reanalysis period.

The ERA5 dataset (Hersbach et al., 2020) includes all important variables that describe the Earth's atmosphere and climate system. Some key variables available are surface, 2 meter and atmospheric temperatures, wind components at different

levels, specific and relative humidity, surface pressure, snowfall, precipitation, etc. Many more derived meteorological variables are also available.

2.3 THE METEOSAT SATELLITE AND SEVIRI INSTRUMENT

Similarly, other satellite observations will be recurrently used throughout the thesis, notably to validate IASI-based retrievals. This is mostly done using the SEVIRI instrument. This section briefly describes the instrument and it's host satellite.

The Meteosat series of weather satellites is a group of geostationary satellites operated by EUMETSAT. These satellites are positioned in geostationary orbits, which means they orbit the Earth in count with the Earth's rotation, allowing them to remain fixed over specific regions. The geostationary position centred at 0°, shown in Figure 12, is particularly advantageous for monitoring weather and climate because it provides continuous and uninterrupted observations of a specific area, in this case, Africa, Europe and nearby regions.

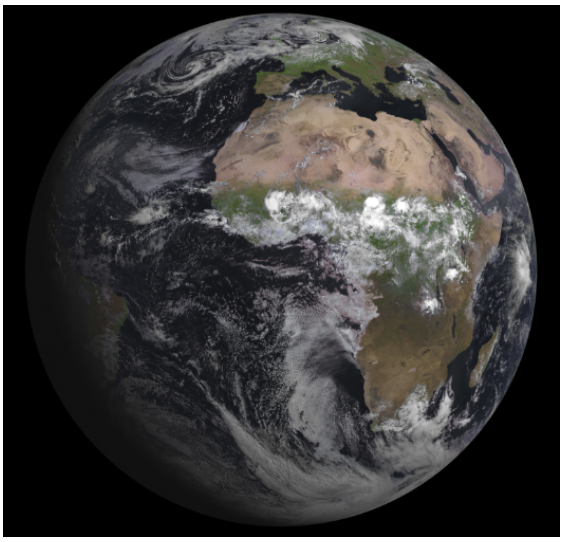


Figure 12 – Area covered by the Meteosat geostationary satellite. Source: EUMETSAT.

Among the advanced instruments aboard these satellites, the Spinning Enhanced Visible and Infrared Imager (SEVIRI) stands out as a key component. SEVIRI captures high-resolution images of the Earth's surface and atmosphere in multiple spectral channels, including visible, infrared, and water vapour bands, every 15 minutes. This versatile instrument provides valuable data for weather forecasting, tracking severe weather events, monitoring cloud cover, and analysing atmospheric properties. Many meteorological products can be extracted from the Meteosat satellites and its instruments, the ones that were used in this thesis will be detailed in due course in the relevant chapters.

3

A GUIDE TO NEURAL NETWORKS AND THEIR USE IN REMOTE SENSING

Contents

AI is defined as the ability of a system or machine to reason, act and adapt similarly to a human. ML is a class of AI methods enable a machine to extract knowledge from data. Artificial neural networks are a subset of methods to do this, and is part of the supervised learning family of ML algorithms [1].

3.1 ARTIFICIAL NEURON AND PERCEPTRON

Some AI methods refer to neurons by analogy to the brain. The biological model of a neuron is made up of a dendrite, the “input” signal to the neuron; a main cell body (the soma) that processes the information; an axon that corresponds to the output of the neuron; and synapses that are the connections with other neurons. In a simplified view, a neuron in a human brain takes an input, processes it and outputs a signal. In the early work of Mcculloch and Pitts (1943), a simple artificial neuron was introduced as a copy of this simplified biological model. This work was continued in Rosenblatt (1958), that introduced the concept of Perceptrons.

1. see Section **??** for the definition of supervised learning.

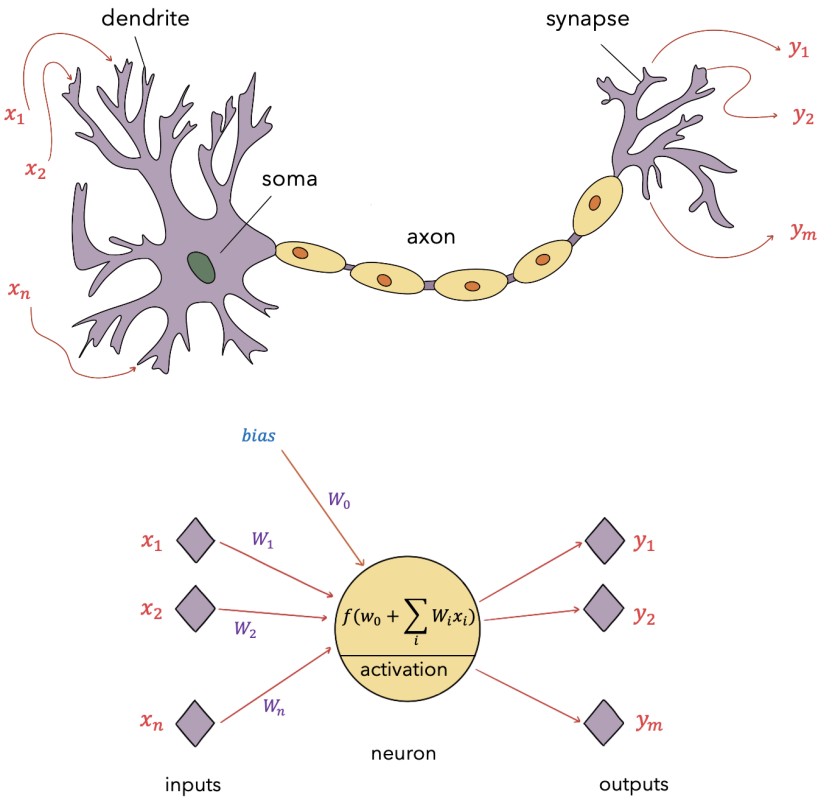


Figure 13 – Schematic view of a biological neuron (top), and its interpretation into an artificial neuron (bottom).

The artificial neuron, depicted in Figure 13, is therefore characterized by: (1) the inputs $x_1, x_2, ...x_n$ where n is the number of inputs (this corresponds to the dendrite), (2) a weight vector W, (3) an activation function f, and (4) one or more outputs $y_1, y_2, ..., y_m$. The figure also helps to link each part of the artificial neuron to its biological neuron analogue.

The activation function f is applied to a linear combination of the inputs, the coefficients of which are the weight vector $W = \{w_1, ..., w_n\}$. To this weight vector is added the bias term of the neuron. These weights are optimized during the learning process.

Many activation functions exist such as the linear function, the logistic function ($f(x) = \frac{1}{1+e^{-x}}$), the threshold function, the hyperbolic function ($f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$), and more recently the Rectified Linear Unit (ReLU) function ($f(x) = \max(0, x)$). Generally, a sigmoid type function is used to introduce non-linearity into the network. However, the most important constraint is that the function needs to be continuous so that a training algorithm can minimise a loss function. The reason behind this is that the function needs to be differentiable with respect to W for an algorithm such as gradient descent to be applied.

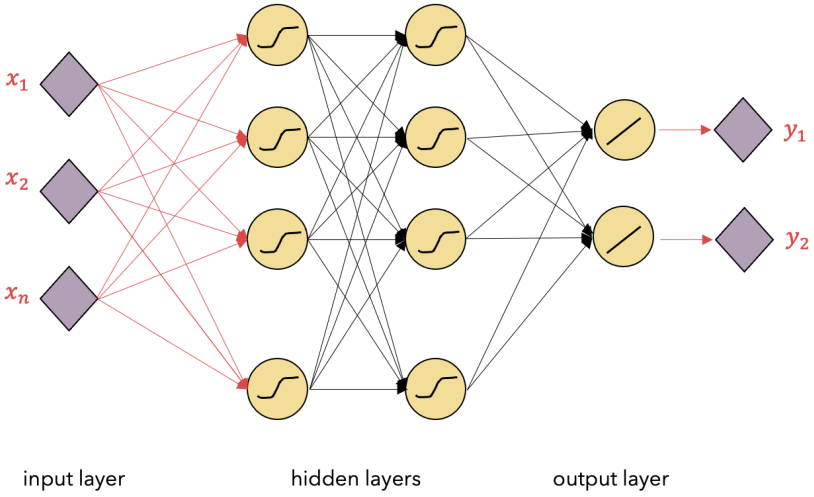


Figure 14 – Schematic view of an MLP model with an input layer, two hidden layers and an output layer.

3.2 MULTI-LAYER PERCEPTRONS

The human brain can be seen as a huge network of 100 billion interconnected neurons processing everything we see, feel, hear, etc. Different neurons are responsible for different tasks, and are arranged in a hierarchical manner (i. e. in layers). By analogy, feed-forward networks or Multi-Layer Perceptrons (MLP) combine several interconnected layers of artificial neurons (perceptrons), and is an attempt to get closer to what is happening inside the human brain.

MLP models are thus non-linear parametric models that compute a potentially multivariate output from a multivariate input. Neurons are organized in successive layers of independent neurons. Traditionally, the input layer is not counted in the number of layers in the network. All layers, except the input and output layers, are called fully connected "hidden" layers. In a layer, each neuron is connected to all the neurons of the previous layer by a "synaptic link" with an associated synaptic weight $w_{i,j}$. $w_{i,j}$ is therefore the weight of the neuron j with respect to its input, coming from the neuron i. Figure 14 shows a simple example of a MLP model with an input layer, two hidden layers, and an output layer.

As previously stated, each neuron j in each hidden layer (1) computes the linear combination s_j of all its inputs x_i (either coming from the input layer for the first hidden layer, or the previous hidden layer for all other layers) and bias b_j:

$$s_j = b_j + \sum_{i=1}^{p} w_{ij} x_i$$

and (2) applies the activation function f to this weighted sum, producing the output y_j of the neuron:

$$y_j = f(s_j).$$

Generally, the neurons in the hidden layers apply a sigmoid activation function, whilst the neurons of the output layer apply the linear activation function.

A MLP thus transforms the inputs x into $y = \Phi(x_1, ..., x_m; W, b)$ where W stands for all weights w_{ijl} where l is the number of layers, j the number of neurons in the layer, and i the number of inputs to each neuron. The weights and biases are the parameters that are adjusted during the training of the network. The network thus "learns" to reproduce the behaviour described by a dataset of input-output samples. This is done following the delta rule, that is, comparing the target output to the predicted output. The training algorithms are described in the following section.

3.3 TRAINING ALGORITHMS

Historically, the MLP model was trained with what is called a gradient descent. At that time, this technique was not very successful because it was computationally expensive. The development of the gradient back-propagation algorithm gave a new impetus to neural networks that could begin to be used for real-world applications. During the gradient back-propagation, the output values are compared with the target value in order to calculate the output error. This error is then fed back into the network so that the algorithm adjusts the weights given to each connection to reduce the error, at a low computational cost.

3.3.1 *Gradient Descent*

During the training procedure, the parameters (i. e. the weights and the biases) are adjusted to minimize the error made by the model on the training database. This error term is the loss function. It is a scalar function of m variables (i. e. the number of output variables), which we denote F. For example, the loss function is often taken to be the binary cross entropy loss for models that require outputting a probability between 0 and 1, or the Mean Squared Error (MSE) loss for regression models that estimate a quantity (i. e. continuous real numbers). The gradient of this loss function, denoted ∇F is a vector of dimension m of the partial derivatives of F with respect to the m variables. By definition, the derivative of a function shows its rate of change with respect to a variable. Thus $\nabla F(x)$ indicates the direction in which F is increasing around the point x, and its norm (value) shows how fast it is increasing. Therefore, to minimize F, we must move in the opposite direction to the gradient.

In summary, gradient descent is an iterative algorithm used to numerically find $\overline{x}$ such that:

$$F(\overline{x}) = \min_{x \in \mathbb{R}^m} F(x).$$

We thus start with an initial solution x_0 and, for each iteration, we move closer to $\overline{x}$ by moving in the direction of $-\nabla F(x)$. More specifically, the algorithm is defined as follows:

- Initialization
  - — choose x_0
  - — choose the learning rate τ
  - — choose a threshold ϵ and a maximum number of iterations
- Iterations (on k)
  - — compute $\nabla F(x_k)$
  - — update $x_{k+1} = x_k - \tau \nabla F(x_k)$
  - — check for convergence: $|F(x_{k+1}) - F(x_k)| < \epsilon$
- End of Algorithm when the check for convergence is verified or the maximum number of iterations is reached

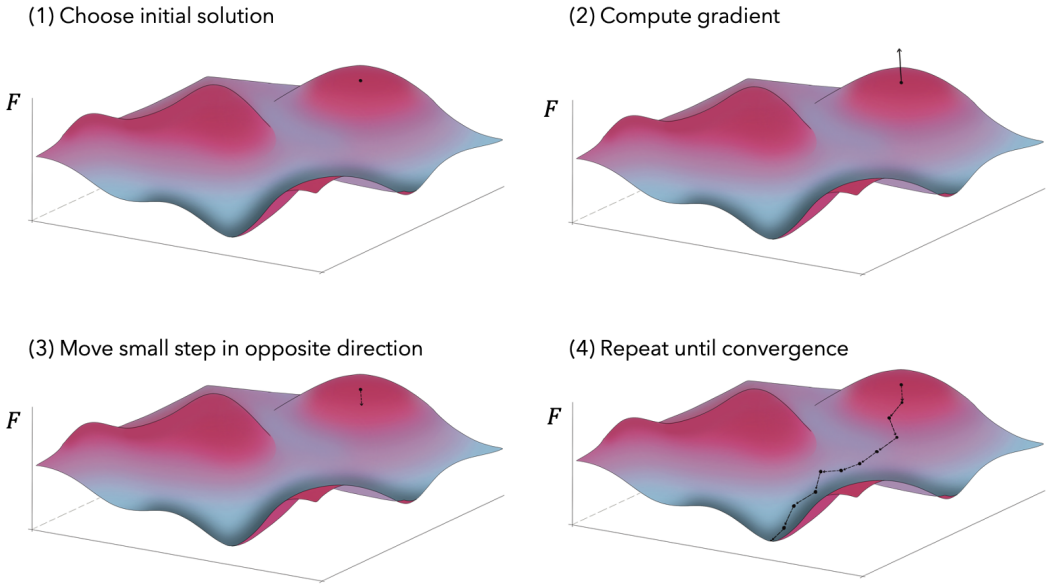


Figure 15 – Illustration of gradient descent on a function $F(x_1, x_2)$.

3.3.2 *Backpropagation*

Backpropagation (Rumelhart, Hinton, and Williams, 1986), short for "backward propagation of errors" is a specific algorithm of gradient descent, used to compute the gradients of the loss function with respect to the weights and biases by applying the chain rule. During the forward pass of training, the input data is passed through the neural network to make predictions, and then during the backward pass, the gradients of the loss function with respect to each parameter are calculated layer by layer and used to update the model parameters using gradient descent or a variant of it.

In the following, we will use the notations described below, inspired from Haykin (1998) and Mathieu (2018):

- The indices i, j, k symbolize different neurons. Considering the signal is propagated from left to right, the neuron j belongs to a layer to the right of neuron i and neuron k belongs to a layer to the right of the one containing neuron j. Neuron j is thus necessarily in a hidden layer.
- At the n^{th} iteration, the n^{th} sample is fed to the network.
- f_j is the activation function of neuron j.
- $e_j(n)$ is the error of the output of neuron j at the end of iteration n.
- $\mathcal{E}(n)$ is the error of the network (i. e. the sum of the squared errors across all output neurons) at the n^{th} iteration.
- $E(n)$ is the loss function at the n^{th} iteration.
- For a neuron k in the output layer, $y_k(n)$ is the target output. This is used to compute the difference between the output of the neuron and its target value (i. e. $e_k(n)$) after the n^{th} iteration.
- $v_j(n)$ is the weighted sum of the inputs to neuron j
- $\widehat{y}_j(n)$ is the output of neuron j, after applying the activation function, at the n^{th} iteration: $\widehat{y}_j(n) = f_j(v_j(n))$
- $w_{ij}(n)$ is the synaptic weight between the output of neuron i and the input of neuron j at the n^{th} iteration. The update on this weight is noted $\Delta w_{ij}(n)$
- m_l is the number of neurons in layer l, with $l = 0, 1, ..., L$. m_0 is therefore the size of the input layer (in this case it is equivalent to the number of inputs), m_1 of the first hidden layer, and m_L of the output layer.

Let us first consider a neuron k belonging to the output layer:

At the n^{th} iteration, the output error of neuron k is defined by:

$$e_k(n) = y_k(n) - \widehat{y}_k(n) \tag{3}$$

and the total error of the output layer is defined by the sum of $e_k(n)$ for all neuron k in the output layer:

$$\mathcal{E}(n) = \sum_{k=1}^{m_L} \frac{1}{2} e_k(n)^2. \tag{4}$$

The loss function

$$E(n) = \frac{1}{N} \sum_{n=1}^{N} \mathcal{E}(n) \tag{5}$$

is the average error over all samples in the training dataset. This is the function we want to minimize during training. Both $\mathcal{E}(n)$ and $E(n)$ depend on the network parameters (i. e. the weights and biases).

The information that reaches the activation function of neuron k in the output layer is:

$$v_k(n) = \sum_{j=0}^{m_{L-1}} w_{jk}(n) \widehat{y}_j(n) \tag{6}$$

with m_{L-1} the number of neurons in the previous (hidden) layer, and w_{0k} corresponds to $\widehat{y_0} = 1$ which stands for the bias. Therefore, the signal output by a neuron k is:

$$\widehat{y_k} = f_k(v_k(n)). \tag{7}$$

Backpropagation aims to update the weights w_{jk} by adding $\Delta w_{jk}(n)$, proportionate to the partial derivative $\frac{\partial \mathcal{E}(n)}{\partial w_{jk}(n)}$. Using the chain rule (that expresses the derivative of the composition of two differentiable functions f and g in terms of the derivatives of f and g), we obtain

$$\frac{\partial \mathcal{E}(n)}{\partial w_{jk}(n)} = \frac{\partial \mathcal{E}(n)}{\partial e_k(n)} \frac{\partial e_k(n)}{\partial \widehat{y_k}(n)} \frac{\partial \widehat{y_k}(n)}{\partial v_k(n)} \frac{\partial v_k(n)}{\partial w_{jk}(n)} \tag{8}$$

because $\mathcal{E}(n)$ is a function of $e_k(n)$ that is a function of $\widehat{y_k}(n)$, that itself is a function of $v_k(n)$ that is a function of $w_{jk}(n)$.

More precisely, we obtain

- $\frac{\partial \mathcal{E}(n)}{\partial e_k(n)} = e_k(n)$, obtained from Equation (4)
- $\frac{\partial e_k(n)}{\partial \widehat{y_k}(n)} = -1$, obtained from equation (3)
- $\frac{\partial \widehat{y_k}(n)}{\partial v_k(n)} = f_k'(v_k(n))$, obtained from Equation (7)
- $\frac{\partial v_k(n)}{\partial w_{jk}(n)} = \widehat{y_j}(n)$, obtained from Equation (6)

Therefore, we get:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{jk}(n)} = -e_k(n) f_k'(v_k(n)) \widehat{y_j}(n), \tag{9}$$

and we apply a small change to $w_{jk}(n)$, defined by $\Delta w_{jk}(n) = -\tau \frac{\partial \mathcal{E}(n)}{\partial w_{jk}(n)}$, where τ is the learning rate.

Consequently:

$$\Delta w_{jk}(n) = \tau \delta_k(n) \widehat{y_j}(n), \tag{10}$$

where

$$\delta_k(n) = -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} = -\frac{\partial \mathcal{E}(n)}{\partial e_k(n)} \frac{\partial e_k(n)}{\partial \widehat{y_k}}(n) \frac{\partial \widehat{y_k}(n)}{\partial v_k(n)} = e_k(n) f_k'(v_k(n)). \tag{11}$$

We thus have $\delta_k(n)$, and consequently $\Delta w_{jk}(n)$ that is proportionate to, on one side, the difference between the actual output of the neuron and the target output (if this difference is small, the weight will not be corrected much), and on the other hand, the derivative of the activation function (the steeper the activation function, the more we will correct the weights). We update the weights $w_{jk}(n+1) = w_{jk}(n) + \Delta w_{jk}(n)$.

Let us now consider a neuron j belonging to a hidden layer:

Although neurons in the hidden layer are not easily accessible, they are also responsible for the error term output by a neuron in the output layer. The neuron that is responsible for the most of the error needs to have its weight corrected the most. The difficulty here is that we do not have a target output value for these neurons in the hidden layers. This is the reason why it is necessary to determine the hidden errors recursively from the neurons to which the hidden neuron in question is connected to in the output layer.

Similarly to equation (11), we can define $\delta_j(n)$, the local gradient of neuron j, as

$$\delta_j(n) = -\frac{\partial \mathcal{E}(n)}{\partial \widehat{y}_j(n)} \frac{\partial \widehat{y}_j(n)}{\partial v_j(n)} = -\frac{\partial \mathcal{E}(n)}{\partial \widehat{y}_j(n)} f_j'(v_j(n))) \tag{12}$$

To compute the partial derivative $\frac{\partial \mathcal{E}(n)}{\partial \widehat{y}_j(n)}$, we derive equation (4) with respect to $y_j(n)$. This gives

$$\frac{\partial \mathcal{E}(n)}{\partial \widehat{y}_j(n)} = \sum_{k=1}^{m_L} e_k(n) \frac{\partial e_k(n)}{\partial \widehat{y}_j(n)} = \sum_{i=1}^{m_L} e_k(n) \frac{\partial e_k(n)}{\partial v_k(n)} \frac{\partial v_k(n)}{\partial \widehat{y}_j(n)}. \tag{13}$$

This is obtained using the chain rule that links $\widehat{y}_j(n)$ to $v_k(n)$ (the input to the output layer). Since $e_k(n) = y_k(n) - \widehat{y}_k(n) = y_k(n) - f_k(v_k(n))$, we get:

$$\frac{\partial e_k(n)}{\partial v_k(n)} = -f_k'(v_k(n)), \tag{14}$$

with $v_k(n) = \sum_{j=0}^{m_{L-1}} w_{jk}(n) \widehat{y_j(n)}$. Differentiating this with respect to $\widehat{y}_j(n)$, we get $\frac{\partial v_k(n)}{\partial y_j(n)} = w_{jk}(n)$. Injecting this into equation (13), and using equations (14) and (11), we have:

$$\frac{\partial \mathcal{E}(n)}{\partial \widehat{y}_j(n)} = -\sum_{k=1}^{m_L} e_k(n) f_k(v_k(n)) w_{jk}(n) = -\sum_{k=1}^{m_L} \delta_k(n) w_{jk}(n) \tag{15}$$

By grouping equations (11) and (15), and because $\widehat{y}_j(n) = f_j(v_j(n))$ we obtain:

$$\begin{aligned} \Delta w_{ij}(n) &= -\tau \frac{\partial \mathcal{E}(n)}{\partial w_{ij}(n)} \\ &= -\tau \frac{\partial \mathcal{E}(n)}{\partial \widehat{y_j(n)}} \frac{\partial \widehat{y_j(n)}}{\partial w_{ij}(n)} \\ &= -\tau \Big(-\sum_{k=1}^{m_L} \delta_k(n) w_{jk}(n) f_j'(v_j(n)) \Big) y_i(n) \\ &= \tau \delta_j(n) y_i(n), \end{aligned} \tag{16}$$

with $\delta_j(n) = f_j'(v_j(n)) y_i(n) \sum_{k=1}^{m_L} \delta_k(n) w_{jk}(n)$.

This formula is proportional to the derivative of the activation function, as well as the weighted sum of the errors of the neurons in the next layer to which the neuron

j is linked to. This means that the next layer's neuron most responsible for the error will have the most impact on the weight correction.

To conclude, considering $E(n)$ as the average error over all training samples ($E(n) = \frac{1}{N}\sum_{n=1}^{N}\mathcal{E}(n)$), the backpropagation algorithm is as follows:

- Initialization
  - draw weights w_{ijk} according to a uniform law on $[0, 1]$
  - normalize the training data to $[0, 1]$ range
  - choose $error_{max}$ and $iter_{max}$
- **while** $E(n) > error_{max}$ or $N_{iter} < iter_{max}$:
  - shuffle the training database
  - **for** each sample $n = 1...N$ in the training database:
    - calculate $e_k(n) = y_k(n) - \widehat{y}_k(n)$ with a forward pass
    - update the weights of the output layer with

$$\Delta w_{jk}(n) = \tau\delta_k(n)\widehat{y}_j(n),$$

      with $\delta_k(n) = e_k(n)f'_k(v_k(n))$
    - Propagate the error backwards to the previous layers:

$$\Delta w_{ij}(n) = \tau\delta_j(n)\widehat{y}_i(n),$$

      with $\delta_j(n) = f'_j(v_j(n))\widehat{y}_i(n)\sum_{k=1}^{m_L}\delta_k(n)w_{jk}(n)$

The algorithms presented in Sections 3.3.1 and 3.3.2 are in fact the stochastic versions of the two algorithms. In reality, the algorithms are trained on mini-batches rather than single training samples. This leads to more accurate estimations of the gradient and thus smoother convergence.

3.3.3 *Adam, a Method for Stochastic Optimization*

Some loss functions can be difficult to optimize. Optimization through gradient descent (and backpropagation) requires choosing a learning rate. If the learning rate is set too small, convergence will be slow, and it is likely that we get stuck in local minima. On the other hand, setting the learning rate too large can create instabilities that will prevent the algorithm from converging. To counterbalance this, it is common to design an adaptive learning rate that adjusts itself to the loss function. This way, the learning rate is no longer fixed but can be made larger or smaller depending on how large the gradient is, or how fast learning is happening, for example.

Adam (Kingma and Ba, 2014), short for Adaptive Moment Estimation, is an example of an algorithm that uses such a strategy. Nowadays, Adam is one of the most popular optimization algorithms to train deep neural models. It combines ideas from both the RMSprop and Momentum optimization algorithms to provide efficient and

adaptive learning rates for each parameter during the training: Momentum is an enhancement of the basic gradient descent. Instead of updating the parameters solely based on the current gradient, momentum introduces a moving average of past gradients. This helps to smooth out the updates and accelerate convergence, especially in the presence of noise or saddle points. RMSprop is another optimization algorithm that addresses the problem of setting a global learning rate for all parameters. It scales the learning rate for each parameter by dividing it by a running average of the squared gradients for that parameter. This helps to adapt the learning rates individually for different parameters.

Adam combines the two concepts by maintaining two types of moving averages:

— Exponential Moving Average of Gradients m_t: Similar to the Momentum algorithm, Adam keeps track of the moving average of past gradients. This helps smooth out the gradient updates and accelerate convergence.

— Exponential Moving Average of Squared Gradients (v_t): This moving average is similar to the concept of RMSprop. It keeps track of the moving average of the squared gradients, which helps adapt the learning rates for each parameter individually.

3.4 CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Network (CNN)'s are one of the most used techniques of ML and AI. They are inspired by the brain's visual cortex and have proven useful for automating tasks such as object recognition and detection, as well as image segmentation.

Indeed, between the 1950s and 1960s, when artificial neural networks were developed and became increasingly popular, research showed that a cat's visual cortex was made up of neurons that react to small sections of the visual field separately. A single neuron reacts to a region of the visual space that is called its receptive field. These receptive fields are comparable and overlap from one neighbouring neuron to the next. This inspired the development of the convolutional layer as we know and use it today. Convolutional layers are used as the main building block in CNNs, that take in input an image (as opposed to a single point measurement).

3.4.1 *The Convolutional Layer*

The convolutional layer in a neural network can be compared to the way a cat's visual cortex processes visual information. It functions as a specialized filter for images, just as the feline brain has evolved to identify crucial visual cues. Simply put, for every pixel within an image, the convolutional layer scrutinizes its immediate surroundings, amalgamating their information. This amalgamated signal is subsequently passed on to the subsequent layer for further processing.

The process is shown in Figure 16. In mathematical terms, a convolution is a fundamental operation involving the running of a filter K, typically represented as a small matrix, across the input image, pixel by pixel. At each step, the filter is subject to element-wise multiplication with the corresponding pixels in the input image. The results of these multiplications are then summed, ultimately yielding a singular output value. This value serves as a representation of the presence or absence of a particular feature at that precise location in the image. The convolution of an image X with the kernel K, of shape $L \times L$, gives another image matrix Z, where each element $z_{i,j}$ is the result of the convolution operator:

$$z_{i,j} = \sum_{k=0}^{L-1} \sum_{l=0}^{L-1} K_{k,l}\, x_{i+k,j+l}.$$

The dimensions of Z vary depending on the parameters chosen for the convolution. We detail the calculations further in the section. Figure 16 illustrates the functioning of a convolutional layer, showing as input a 2D image matrix and its corresponding output.

Significantly, it is these filter weights that play the main role in the convolutional layer. These filter weights $K_{k,l}$ can be considered as the neurons of the network. During the network's training procedure, these weights are fine-tuned through a process of optimization, allowing the network to recognize distinctive patterns and features within the data. Just as a cat's visual cortex refines its ability to discern objects and shapes through experience, the convolutional layer adapts its filter weights to become proficient in identifying meaningful features within the input data, making it an essential component of image recognition and processing in neural networks.

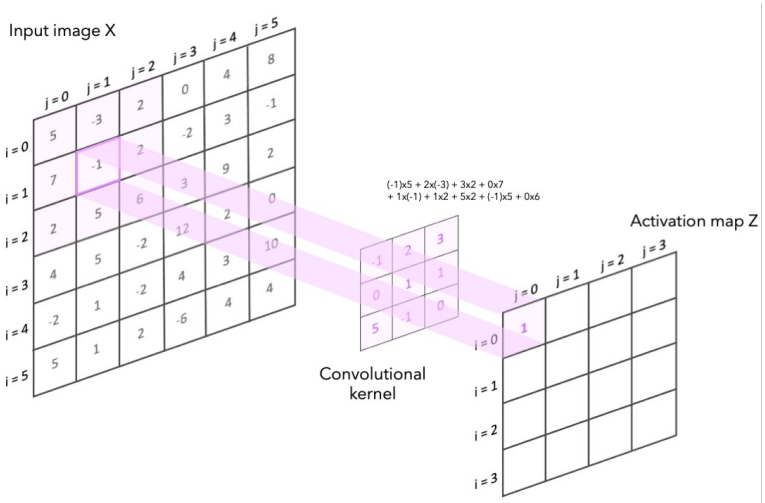


Figure 16 – Illustration of the convolutional layer, on a 2D image matrix X of dimension 6×6, using a filter of shape 3×3 resulting in an activation map Z of dimension 4×4. The padding used is $P = 0$ and the stride is $S = 1$.

Four hyperparameters control the size of the activation map:

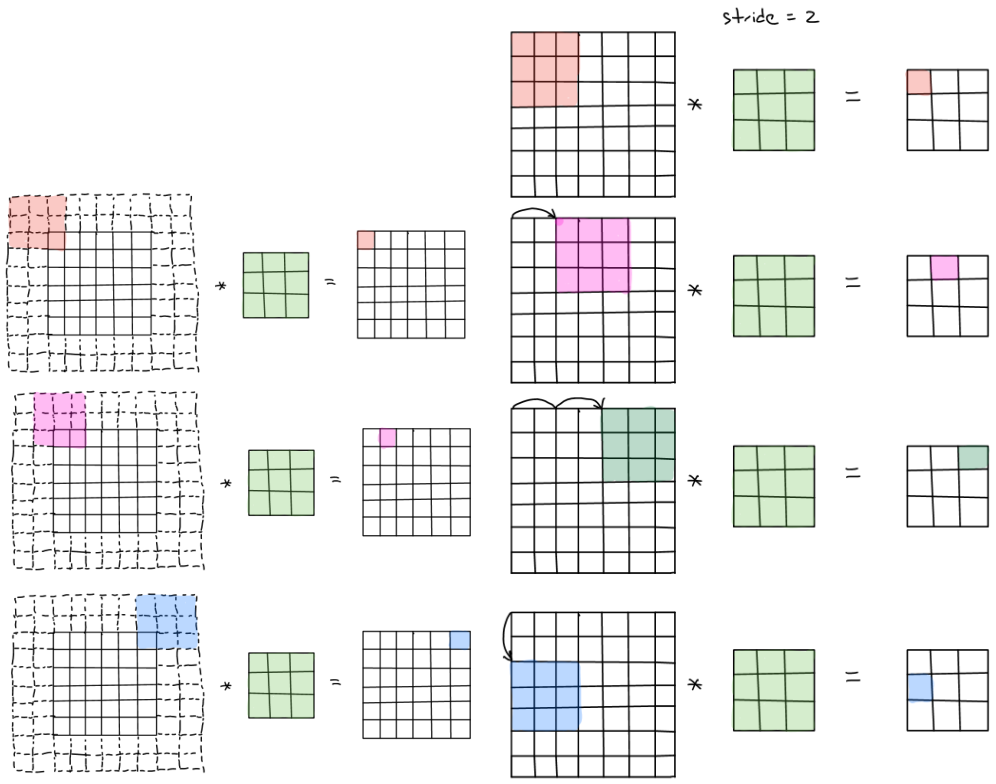
1. The kernel size L that controls the spatial extent of the convolution. It is a crucial hyperparameter that defines how many adjacent pixels the filter considers at each step of the convolution. The size chosen is typically 3×3 or 5×5 (generally an odd number is used because then the filter focuses on a central pixel), but the choice is highly dependent on the data. Even more importantly, the choice depends on the spatial coherency of the input data.
2. The depth of the kernel is the number of filters to be convolved. The number of filters specified in a convolutional layer influences the complexity and diversity of features the layer can extract. Increasing the number of filters allows the network to capture more intricate information.
3. The stride that dictates how the filter moves across the input image. A stride of 1 means the filter moves one pixel at a time, while a stride of 2, shown in Figure 17(a), would make the filter skip every other pixel. A larger stride reduces the spatial dimensions of the output feature map and can lead to a coarser representation of the input data. Smaller strides typically result in higher spatial resolutions.
4. The padding, illustrated in Figure 17(b), is applied to the input image to control the spatial dimensions of the output activation map. The most used types of padding are *valid*, meaning no padding (this therefore produces an output with smaller dimensions than the input), and *same*, which produces an output of same dimension than the input, by zeros around the input image. Figure 17 shows, in dotted lines, the padding (here, $P = 2$) necessary to keep an output size that is identical to the input. Padding is useful for ensuring that features at the edges of the input image are adequately processed and preserved in the output. The choice once again depends on the wanted output. For example, for pixel-to-pixel regression tasks, the *same* padding is generally applied to ensure that the CNN produces an output that is the same size as the input, whilst for an image classification task, *valid* padding is often selected.

The output size of a convolutional layer is thus

$$\left(\frac{N_{X_i} + 2P - L}{S} + 1, \frac{N_{X_j} + 2P - L}{S} + 1, D\right),$$

where N_{X_i} and N_{X_j} are respectively the size of X along the i and j axis, L is the size of the convolution filter, S the stride, and P the padding.

An activation function is applied at the end of a convolutional layer. This function is the same as the activation function after an artificial neuron in a MLP, but applied independently on each pixel of the activation map. In CNNs, it is most common to apply the ReLU function. Each convolutional filter also has a learnable bias, which



(a) Convolution with “same” padding. (b) Convolution, with stride of 2.

Figure 17 – Illustration of convolutions with (a) “same” padding, and (b) a stride $S = 2$. Source: https://guandi1995.github.io/Padding/

is commonly applied to all the pixels. The output of a convolutional layer is therefore:

$$z_{i,j} = f\left(\left(\sum_{l=0}^{L}\sum_{k=0}^{L} K_{l,k}x_{i+l,j+k}\right) + b_K\right). \tag{17}$$

3.4.2 *The Pooling Layer*

The typical architecture of a CNN is a succession of convolutional layers (including activation) and pooling layers. Pooling layers include Max Pooling, Min Pooling, and Average Pooling. These are simple operations that extract the most important feature in a sliding window, and used for reducing spatial dimensions of the activation maps. They are especially useful for performing image classification where the output of the CNN should be a scalar.

3.4.3 *The Locally Connected Layer (or Localized Convolution)*

The convolutional layer relies on weight sharing: This means that the same convolution filters will be used throughout the entire image. It is the stacking of a high

number of convolution filters that allows to approximate close to all spatial patterns present in the training database. Naturally, this feature is the reason why CNNs are able to complete tasks such as face detection; the CNN will learn to detect spatial features such as eyes, no matter where they are placed on the picture (i. e. whether the picture is a close up of a face or a full body picture, the CNN will detect the face). However, in some applications, the training database can contain pictures that are all taken in the same format (for example a database of passport photos). In this case, the eyes, for instance, will, ideally, always be placed at the same place in the picture. It is thus unnecessary that the CNN attempts to recognize eyes all over the image.

With this in mind, removing the weight sharing from a convolutional layer does just this. Instead of sliding the filters over the entire image, each pixel of the image has a dedicated set of filters. Figure 18 illustrates the difference between a traditional convolutional layer and a locally-connected layer, that we will from now on call a "localized convolutional layer". The figure shows a sample of a passport photo, and the corresponding collection of filters that would be learnt by the CNN. In the case of a traditional convolutional layer, the same filters will be convolved over the entire image. On the other hand, each filter found by a localized convolutional layer is specific for each part of the image.

Although this makes sense for some applications to limit the number of filters the CNN must learn, it also leads to a much larger number of learnable parameters and thus enlarges the computational needs.

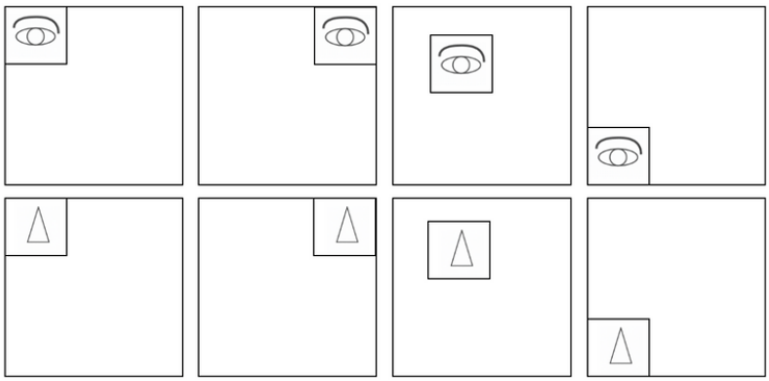
3.4.4 *The Partial Convolution Layer*

One of the limitations of traditional convolutional layers, is that they do not tolerate missing values in the input images, both in training and inference mode. Common practices consist in replacing missing values by a constant value (zero or the average value for example), or adding a pre-processing step to interpolate missing data (Boucher, Aires, and Pellet, 2023). However, this is only possible when there are a very few missing values in the training database. If too many constant values are inserted into the database, this defeats the purpose of using neighbouring pixels since the CNN will learn not to trust a pixel's surrounding.

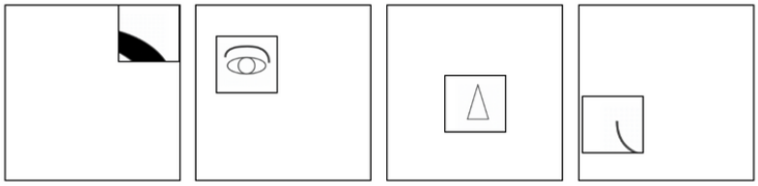
The "partial convolution layers", first introduced in (Harley, Derpanis, and Kokkinos, 2017; Liu et al., 2018), can replace the traditional convolutional layer. Such layers are exploited in this thesis for the first time as a tool to deal with missing data present in the input images. We expect this to become a very general and important tool for the satellite remote sensing community.

The partial convolution layer operates in just the same way as a classic convolution, except the convolution operation is performed only on valid pixels and is normalized by the number of valid pixels. This is made possible by adding a binary mask m

(a) Sample of an image in a database of passport photos.



(b) A collection of filters found by a traditional convolutional layer and how they are convolved over the input image.



(c) A collection of filters found by a localized convolutional layer and how they are convolved over the input image.

Figure 18 – Example highlighting the difference between traditional and localized convolutions.

to the layers' inputs, where $m_{i,j} = \begin{cases} 0 & \text{if missing} \\ 1 & \text{otherwise} \end{cases}$, and transforming the traditional convolution operation (17) into:

$$z_{i,j} = f\left(\frac{1}{\sum_{k=0}^{L-1}\sum_{l=0}^{L-1} m_{i+k,j+l}} \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} K_{k,l}(x_{i+k,j+l} \odot m_{i+k,j+l}) + b_K\right), \tag{18}$$

where $z_{i,j}$ is the activation map (the output of the convolution layer) at position (i,j), $x_{i,j}$ is the input signal at position (i,j), w is the weight of the filter with size $L \times L$, b is the bias term, m is the binary mask for missing data, and $\odot$ denotes element-wise multiplication.

3.5 RECURRENT NEURAL NETWORKS, TRANSFORMERS AND VISION TRANSFORMERS

For sequence modelling Natural Language Processing (NLP) tasks such as translation or text generation (i. e. chatbots), the more traditional approaches include Recurrent Neural Network (RNN) that are designed to handle sequential data. This is done by maintaining an internal memory. Sequential data can be time-series or, in the case of NLP tasks, sentences. The internal "memory" captures information from previous time steps in the sequence. At each time step, the network takes an input, as well as the memory (called the hidden state) from the previous time step, producing an output, alongside with the updated hidden state. This iterates over the whole sequence, so that the network captures dependencies and patterns over time. More advanced RNN architectures such as Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) were developed to overcome some limitations, like the difficulty in capturing long-range dependencies.

To further overcome these limitations, transformers were introduced in Vaswani et al. (2017). Transformers use a new mechanism called "self-attention" or "multi-head attention", which allows the model to focus on different parts of the input sequence when processing each time step. A transformer model consists of an encoder and a decoder, each composed of multiple layers of self-attention mechanisms and feed-forward Neural Network (NN)s. The encoder processes the input sequence, while the decoder generates the output sequence. Where traditional NNs such as RNNs process an input query sequentially (from the beginning of a sentence to the end), the transformer can parallelize this input, considerably reducing training times.

An example of the advantage of this architecture is given in Andler (2023); In the phrase "I have a brother, he is an architect", the words *brother* and *he* refers to the same person. This is quite easy to grasp because the sentence is extremely simple. In the sentence "When my brother got into a fight with his associate, I confessed to him that I had never liked him", the words *brother*, *him* are further apart, and it is not as obvious which *him* we are referring to. This is where the transformer's self attention mechanism that takes into account these long-range effects on the context come into play. It is this mechanism that enables a context to be taken into account when processing an input. Actually, as I am describing this example that comes from a book written in French, I translated the latter sentence

on one hand with a traditional translator, and on the other with a transformer-based translator. The transformer-based translator accurately captured the meaning of a complex sentence, while the traditional one struggled. This highlights its advantage in understanding language nuances. Thus, transformers have shown remarkable performance in NLP tasks and have become the backbone of many state-of-the-art Large Language Model (LLM) models like Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) and Generative Pre-trained Transformer (GPT) (Radford et al., 2018).

Vision Transformer (ViT) extend the transformer architecture from NLP tasks to computer vision tasks (i. e. image processing tasks). They were introduced in 2020 in Dosovitskiy et al. (2021). Although we do not use ViTs in this thesis, they are a modern image processing technique that deserve to be described, and could in fact be applied to most tasks presented in this thesis.

In traditional CNNs, hierarchical features are extracted from images through layers of convolutions and pooling operations. In contrast, ViTs directly apply self-attention mechanisms to image patches, treating the image as a sequence of patches (one patch is like a word in a sentence). This allows the model to capture global dependencies between patches and has shown promising results in various computer vision tasks such as image classification, object detection, and segmentation.

3.6 THE USE OF NEURAL NETWORKS AND IMAGE-PROCESSING IN REMOTE SENSING

The use of ML methods is rising, no matter the sector of application, and this ever since the first boom of AI in the 1960s. This is no different for Earth Observations and Remote Sensing applications. Earth Observation applications, by the enormity and complexity of its data, is an excellent field of application for ML and AI techniques. Indeed, over the last decades, both the atmospheric sensors and forecasting models have generated uncountable amounts of data, of increasing quality. This leads to a large playground for AI models that can help to assist most, if not all, aspects of environmental and climate sciences.

AI has found itself in the middle of numerous applications in remote sensing. An example includes canopy height estimation (Dalagnol et al., 2023; Schwartz et al., 2024). Since forests play an important role in the global carbon cycle, their conservation is an essential part of mitigation policies. DL techniques have allowed to combine observations from several instruments, creating more accurate high-resolution canopy height maps. Of course, many more examples could be shown; in fact, almost all aspects of remote sensing studies are now subject, in some way or another, to ML approaches, whether that be for pre-processing steps or for the actual inversion models. Many initiatives such as ESA's ϕ-lab and Artificial Intelligence for Earth Observation (AI4EO) have recently emerged to deploy these techniques.

Lastly, maybe one of the main advances over the recent years is the development of ViTs, briefly described in Section 3.5. ViTs have actually spurred the use of DL techniques in weather forecasting (Bi et al., 2022; Lam et al., 2023b) by revolutionizing how meteorological data is processed and analysed, so much so that the ECMWF decided to make creating the first operational DL weather forecasting system their goal, and have started to develop the Artificial Intelligence/Integrated Forecasting System (AIFS). The self-attention mechanisms can model long-range dependencies in weather data, enabling ViTs to capture interactions between distant regions and phenomena. This is particularly beneficial for weather forecasting, where large-scale circulation patterns can influence local weather conditions, for example.

3.6.1 *Machine Learning in IASI-based Remote Sensing*

The use of NNs for IASI-based remote sensing dates back to the end of 1990s when the IASI mission was in preparation. The first NN algorithm was introduced in Hadji-Lazaro, Clerbaux, and Thiria (1999) for the retrieval of CO, in preparation of the huge amount of data IASI will soon provide. In parallel, neural networks were developed in Aires et al. (2002a, 2001) for data compression and the retrieval of many other variables. Using NNs has since then become one of the most used inversion algorithms for retrieving geophysical parameters from IASI observations.

It is clear that thanks to the huge amount of data IASI provides, and the wide choice of target data (IASI L2 products, *in situ* data, model data etc.), NNs have the ability to create consistent time series, like it was done in Bouillon et al. (2022) and Bouillon (2021).

EUMETSAT, that produces the operational Level-2 IASI products have also turned towards the use of ML methods by developing the PWLR algorithm starting from November 2016 (becoming the operational algorithm since 2018 EUMETSAT (2016)).

Unlike for remote sensing in general, the literature is much poorer if a focus is made on ML algorithms that use an image-processing approach on IASI observations. This is mostly due to the difficulties inherent to the ill-posed nature of IASI observations, that we will mention in the next section.

The first paper to start the investigation on their uses for IASI data is Malmgren-Hansen et al. (2019). This work explores the possibility to view an orbit as an image, and manages to retrieve atmospheric temperature with increased accuracy over competing approximations, such as linear regression. These promising results were further investigated in Aires, Boucher, and Pellet (2021) in which several approaches were tested out to quantify the advantage of using image-processing techniques. More sophisticated CNN architectures have also been recently used to detect tropical cyclones from IASI observations (Lam et al., 2023a). Boucher and Aires (2023b) and Boucher, Aires, and Pellet (2023) and Boucher and Aires (2023a) were the result of the work presented in the thesis, and aim to set the grounds to extract the full potential of using image processing techniques on the IASI observations available.

Paving the way to the use of novel image-processing for infrared spectrometers in general is primordial for future missions that will perhaps be more prone to extracting the potential of the methods. For example, the new generation of the IASI instrument, IASI-NG, with augmented spatial (and spectral) resolution is expected to launch in 2025. The IASI-NG FOV will double, from a 2×2 pixel grid covering 50×50 km^2, to a 4×4 pixel grid covering 100×100 km^2. This, coupled with the radiometric noise being reduced by a factor 2, will increase spatial coherency between IASI pixels and will favour the use of image-processing taking into account neighbouring pixels. Another example is the IRS mission, planned to be launched in 2025 on the Meteosat Third Generation (MTG)-S. The IRS instrument will be the first operational hyper-spectral infrared sounder in geostationary orbit, based on IASI heritage. IRS will benefit from the best of both worlds: a high spectral resolution like IASI, and (much) higher spatial and temporal resolutions. Whilst some difficulties inherent to infrared sounding will remain, the geostationary orbit of IRS will open up countless possibilities to apply the methodologies presented in this thesis.

3.6.2 *Challenges in Image-Processing Using IASI Observations: Solutions to Consider*

The IASI instruments are placed onboard the polar orbiting satellites Metop. By definition, this comes along a particular acquisition geometry of the observed data, that unfortunately is not prone to an easy and straightforward use of NNs, even less so of image-processing based approaches. Several challenges arise when wanting to apply ML techniques to retrieve surface or atmospheric properties from IASI observations. This section will go through a non-exhaustive list of them.

Geometry of Acquisition

The most obvious obstacle to using image-processing techniques for IASI observations, and in general for any polar-orbiting sensor, is the lack of images, so-to-speak. This arises due to the way observations are acquired, as described in Section 2.1. After a full circle around the Earth, we are left with a long strip of data that represents the orbit. This strip of data cannot directly be considered as an image, because, as can be seen in Figures 7 and 8 the geometry of the pixels is not like a regular grid of pixels. A regular photo (or image) is a grid of 1000×1000 pixels for instance, where every pixel is connected to its neighbouring pixels in the same way, and with the same "distance". The strip of data extracted from the IASI sensor cannot directly be associated to the pixel grid of an image as several factors come into play and must be taken into account.

Firstly, the scanning angle must be taken into account. The angle at which the Earth is seen has a large impact on the thickness of the atmosphere separating the surface from the sounder, which itself has a large impact on the measured radiation, as well as the surface measured (12km at nadir vs. 30 or 40km on the scanline edges). Secondly, due to the Earth's curvature, the distance between each IASI pixel differs between the equator and the poles. Again, this impacts the measurements made by IASI. If a physics-based retrieval was considered (i. e. using the inverse of the RT equation), this would be taken into account. However, when considering ML methods, even those operating at the pixel scale, the models consider each observation pixel equal, disregarding the acquisition parameters. It is even more important to consider this when wanting to use image-processing techniques, that naturally consider an identical relationship between all pixels.

One objective of this thesis is to figure out the best way to use image-processing techniques, taking into account all the aforementioned constraints and staying close to the RT equation. The first part of the research work will consider working on a regular grid, whilst the second part focuses on remaining true to the geometry acquisition of the data.

Missing Data

A major challenge arising with IASI observations is the presence of missing data. Missing data is defined as observation points that cannot be used for convenient retrieval of atmospheric parameters. Several factors can lead to an observation point being unavailable. Instrumental failure can occur, and occasionally one or several pixels of the IASI FOV can be reported as missing. However, the main source of missing data when it comes to IASI observations is created by the presence of clouds. Of course, the data in cloudy areas is not truly missing as it is just impacted by cloud (similar to data that can be impacted bt other geophysical parameters such as trace gasses etc.). However, because this thesis considers the retrieval of surface properties, only clear-sky observations can be retained. We therefore refer to cloudy data as missing during the thesis. This is not a particularity to IASI observations, but all infrared and visible sounding instruments, to which clouds are impervious. This leads to there being, on average, 70% of the data that is considered as missing.

Naturally, this is not too big of a problem when considering ML algorithms (such as NNs) that are operating at the pixel scale. In that case, only clear-sky pixels are used, and the rest are ignored. However, this becomes a significant challenge when considering image-processing techniques. Traditional image-processing methods often rely on complete datasets to extract meaningful information and patterns, making them less suitable for handling the substantial gaps introduced by cloud-covered regions. In more common applications of image-processing DL techniques, these missing data are relatively rare, and it is thus suggested to simply discard these gaps by filling the missing values by an average value. However, this may result in a loss of valuable information, especially in regions where cloud cover is persistent, and could even limit the performance of the model elsewhere. Indeed, if many pixels are surrounded by 70% of cloudy (missing) points, the model will tend to conclude that there is no useful information in the surroundings of an available data point. It is thus clear that addressing this challenge requires the development of specialized techniques tailored to handle missing data in the context of atmospheric observations.

Approaches include estimating the missing values based on the available data before applying the image-processing retrieval methods. The goal here is not to predict the atmospheric parameters in cloudy regions, but rather being able to use image-processing techniques despite the gaps. Several solutions were tested during the course of this PhD, some of which are presented in 4 (specifically in Section 4.2). Although not explored in this research project, an approach would be the exploitation of multi-sensor fusion techniques. Integrating data from complementary instruments that operate in different spectral ranges or have diverse sensing capabilities can provide a more complete picture of the atmosphere. This approach may mitigate the impact of missing IASI observations by incorporating information from other sources, such as microwave (such as the AMSU instruments, also onboard Metop satellites) or visible sensors, to enhance the overall accuracy of at-

mospheric property retrievals. Lastly, advancements in deep learning techniques, particularly convolutional neural networks (CNNs), offer promising solutions for handling missing data in image processing tasks. These models can learn intricate spatial relationships and patterns from available clear-sky observations, perhaps even enabling them to make informed predictions in regions with missing data. This is explored and explained in the second Part of this manuscript, in which partial convolutions are used.

In summary, addressing the challenge of missing data in IASI observations for image processing involves a combination of innovative imputation techniques, multi-sensor fusion, and the application of advanced deep learning models. These approaches aim to unlock the full potential of IASI data for extracting accurate and comprehensive atmospheric parameters, when using image-processing techniques, even in the presence of substantial cloud cover.

Part I

EXPLORATORY TECHNICAL DEVELOPMENTS

INTRODUCTION

The previous chapters have set the scene on the role of image-processing based AI models in remote sensing for analysing and interpreting IASI observations. Although these methods have a potential for improved retrieval of atmospheric and surface variables, a number of challenges must be overcome in order for the methods to be correctly applied and their behaviour understood.

This part of the thesis aims to create a first overview of possible solutions, and acts as a preliminary development stage, in view of creating long-term global retrievals of surface and atmospheric variables. The choice that was made here is to focus on a simple application case of surface temperature retrieval over a small area covering France. This is a simple enough application that allows to omit certain challenges faced, or at the very least facilitate their resolution.

This part is structured into three chapters. Chapter 4 aims to introduce different ways to reduce regional biases in the retrieval and extensively comparing the results to more state-of-the-art ML techniques. Bias-reduction is done by introducing a way for NNs (both at the pixel and image scales) to adapt their behaviour to local conditions. It also contains a thorough evaluation of the retrievals obtained with a localized CNN.

Chapter 5 introduces a methodology to estimate retrieval uncertainties, in addition to the retrieval of the geophysical variables. These uncertainties are required to make good use of the retrieved data.

Lastly, Chapter 6 presents an additional application where combining localization techniques to image-processing improves the estimation of extreme events.

In the conclusion of this part, we will go through all the developments implemented throughout the three chapters to determine the way forward for the global retrievals we want to develop.

4

INTRODUCING LOCAL INFORMATION INTO NEURAL NETWORKS

Contents

This chapter is based on the two following published papers: Eulalie Boucher and Filipe Aires (2023b). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part II – Assessment. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1593–1611. DOI: 10.1002/qj.4472; Eulalie Boucher, Filipe Aires, and Victor Pellet (2023). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part I – Methodology. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1180–1196. DOI: 10.1002/qj.4447.

Traditional NN models, also known as MLPs, as described in Chapter 3, are mainly built at the pixel level for instruments like IASI. This is because:

— Image processing techniques were not as developed in the early stages of NN remote sensing;

— It was commonly admitted that there is low or no interest in using neighbourhood pixels for the retrieval of a central pixel;
— There are many difficulties to it, as presented in Section 3.6.2 (Chapter 3).

Therefore, in traditional studies, a unique MLP is trained over the entire globe, and it is expected to perform well everywhere. This is an ambitious task because all relevant information to represent correctly the RT (in forward or inverse mode) all around the world would be required to achieve this goal, whatever the environment type or local conditions. This can be feasible for the atmosphere where the RT reaches an adequate level of precision (enough for assimilation) but it is less true for the Earth surfaces where the RT and the input data are still not entirely satisfactory (Aires et al., 2021). Some surface parameters describing the soil properties or the state of vegetation are not available at the global scale, so only a simplified RT is attainable. The MLP training will actually find the best compromise around the world, using a simplified relationship between the satellite observations and the variable(s) to retrieve. This can result in regional biases that can impact the quality of the retrieval and cause a problem when the retrievals are assimilated into NWP models. Therefore, the need to deal with these biases requires adopting a bias-correction strategy such as a variational bias-correction (Dee, 2004) or a Cumulative Distribution Function (CDF)-matching (Aires et al., 2021; Wagner, Lemoine, and Rott, 1999). CDF-matching consists in transforming at the pixel level, before the assimilation, the CDF of the satellite-based estimate towards the CDF of the corresponding variable in the model, which harmonizes the averages, ranges and even physical units of the variable. Whilst this method has its advantages, it is not a perfect solution as a simple statistical transformation cannot compensate for the lack of important physical information, and also, such process changes the spatial patterns present in the satellite data towards the spatial patterns of the model (Aires et al., 2021).

In order to avoid using such CDF-matching and improve the retrievals, it should be possible to reduce the regional biases directly within the retrieval algorithm. This would require to take into consideration, in the NN itself, some information about the specifics of each region in the global domain. “Localization” is defined here as a way to help the NN adjust its behaviour to local conditions.

Several localization approaches can be considered. This can be done at the pixel level, by adding information to the inputs, with the expectation that it helps the NN adjust to local conditions (Aires et al., 2001). This information should describe the local surface properties, for example it could be the surface emissivities, the vegetation type or the altitude.

As mentioned in previous chapters, it has recently become more common to consider satellite observations under the form of images, in which we are looking to exploit spatial structures. These AI or DL techniques are promising for the improvement of the next generation of remote sensing algorithms. Among the available AI models, CNNs are the predominantly chosen models, including for remote sensing (Kattenborn, Eichel, and Fassnacht, 2019; Lam et al., 2023a; Zhang et al., 2018;

Zhao et al., 2022) [1]. CNNs are well suited for high resolution data, which generally present dominant spatial features, contrasted textures with high spatial coherency, and should be tested for coarser instruments like IASI (Aires, Boucher, and Pellet, 2021; Malmgren-Hansen et al., 2019), although there are challenges, for example the processing of missing data, that come alongside these techniques. There exists possibilities to introduce localization into CNNs too. One way is to consider stationary domains, for which the location of a pixel in the image has a real physical meaning. Another way could be to force the CNN to have different spatial filters in different parts of the image, we will refer to these methods here as the "Localized" CNN architectures.

We propose to test and compare localization strategies, in particular in light of the new CNN models. We compare the forms of localization that are available both for MLP and CNN models. Section 4.1 presents the databases that are used in this study. Section 4.2 proposes a strategy to fill missing data before the use of CNNs. Various strategies to localize the neuronal modelling and their respective retrievals are presented in Section 4.3.

4.1 MATERIAL AND STUDY AREA

The data that is used for this part (this chapter and the next two chapters) are data extracted from the databases described in Chapter 2, but a quick summary is provided here. The processing applied to the databases is described in the following subsections.

4.1.1 *Datasets Used*

IASI Observations

We use a dataset of BT from the Metop-A IASI instrument [2] (see Section 2.1) colocated with ERA5 (see Section 2.2) Surface Temperature (ST) in order to intercompare different NN techniques (traditional MLPs and CNNs) to estimate ST, both over land and ocean. To this effect, we construct a database of images on a fixed domain. The notion of images here is important, since we will be comparing pixel-wise to image-processing approaches.

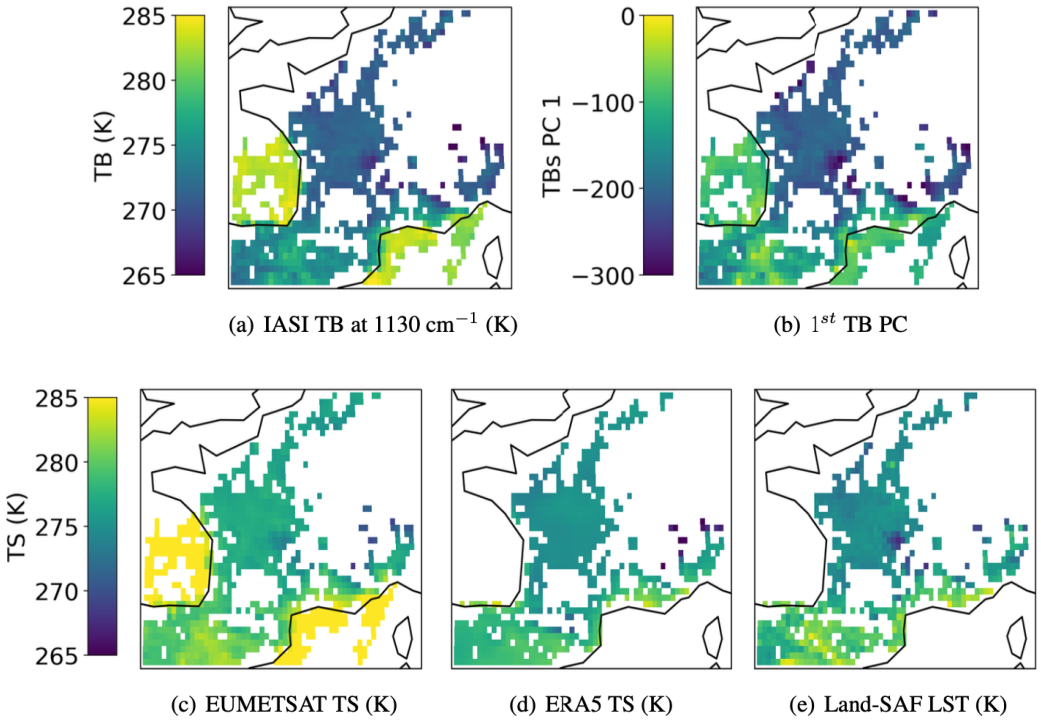
We rely on a processing of ten and a half years of IASI Metop-A data (July 2007–December 2018) performed by EUMETSAT [3] (Hilton et al., 2012). The EUMETSAT archive (see Sections 2.1.2 and 2.1.7) was provided in the form of orbit files containing Level 1c data (i. e. BTs of the 8461 IASI channels) and Level 2 data (i. e. retrieved geophysical variables including Land Surface Temperature (LST), Sea Surface Temperature (SST), atmospheric temperature profiles etc.). As mentioned previously, we have access to the 8461 IASI channels. This spectrum is divided into

1. Although much more recently, ViT's have begun to take over
2. DOI: 10.15770/EUM_SEC_CLM_0014
3. DOI: 10.15770/EUM_SEC_CLM_0014

three main bands: B1 from 645 to 1210 cm^{-1}, B2 from 1210 to 2000 cm^{-1}, and B3 from 2000 to 2730 cm^{-1}.

We concentrate on the window channels that contain information on surface properties. An information redundancy reduction principle was used to regularly extract 563 channels ranging between $770 - 980$ cm^{-1}, $1080 - 1150$ cm^{-1} and $1210 - 1650$ cm^{-1}, as it was done in Pellet and Aires (2018). PCA, as described in Section 2.1.5, is applied on the 563 selected channels independently over land and sea; the first three components represent 99.30%, 0.54% and 0.08% of the total variance for inland pixels; and 97.30%, 2.2% and 0.2% for marine pixels. It is therefore possible to perform the LST and SST retrievals using the first three PCs without any significant loss of information.

Figures 19(a) and (b) show an example of BT at 1130 cm^{-1} and the first extracted principal component. White pixels indicate cloudy sky conditions for which IASI BTs cannot be exploited for LST and SST retrievals.



(a) IASI TB at 1130 cm^{-1} (K) (b) 1^{st} TB PC

(c) EUMETSAT TS (K) (d) ERA5 TS (K) (e) Land-SAF LST (K)

Figure 19 – Example of the data used on January 1^{st}, 2015. Column (a) represents the IASI BT at 1130 cm^{-1}. Column (b) shows the first principal component obtained after the spectral PCA is applied. Column (c), (d) and (e) represent the ST from the EUMETSAT, ERA5 and Land-SAF archives respectively.

EUMETSAT IASI Temperature Product and Auxiliary information

The ST over ocean (SST) and land (LST) were extracted from the EUMETSAT Level 2 CDR (see Section 2.1.7), together with a cloud indicator, over the fixed domain. Figure 19(c) shows an example of the ST product. Again, white pixels are for cloudy conditions, as obtained from the EUMETSAT cloud mask.

Within the IASI archive, EUMETSAT provides auxiliary information that will be needed throughout this study[4]. The Land Fraction variable gives the ratio of land surface in the IASI field of view (i. e. it ranges from 0% to 100%). The Altitude (or Height) variable provides the average elevation within the IASI field of view. These variables come from a Land-Sea database and the Digital Elevation Model (DEM) GTOPO30[5].

ERA5 temperature product

We extract the same ten and a half years period (July 2007–December 2018) of the hourly ERA5 ST (see Section 2.2) under all sky conditions over our chosen domain. A cloud flag is however introduced according to the EUMETSAT cloud fraction. Figure 19(d) shows an example of the ST product. However, since only hourly data are available, we do not have exact time collocation between the two datasets. Therefore, for each orbit in our IASI database, we extract the closest hour from the ERA5 database (i. e. max $\pm$ 30-minute difference).

Land-SAF LST

Amongst various variables that can be retrieved from SEVIRI data, the Land Surface Analysis Satellite Application Facility (LSA-SAF or Land-SAF) provides a land surface temperature, shown in Figure 20. This product is derived from the clear-sky measurements of SEVIRI at channels 10.8 and 12.0 μm (926 and 833.3 cm^{-1}), and is available from 2004 onwards (Trigo et al., 2021; Trigo et al., 2008, 2011), with one measurement every 15 minutes, although it is only available to download with a time step of 1 hour.

For the evaluation of the results, one year (2015) of the LSA-SAF ST data is used. We proceed in the same way as for the previous ERA5 dataset, and choose the closest hour available of the Land-SAF LST data for each IASI orbit.

TELSEI Surface Emissivities

This study also uses the surface emissivities climatology derived from the emissivity interpolator Tool to Estimate Land Surface Emissivity in the Infrared (TELSEI)

4. see EUMETSAT IASI Level 2 Product Generation Specification Document EPS.SYS.SPE.990013

5. https://www.usgs.gov/centers/eros/science/usgs-eros-archive-digital-elevation-global-30-arc-second-elevation-gtopo30?qt-science_center_objects=0#qt-science_center_objects

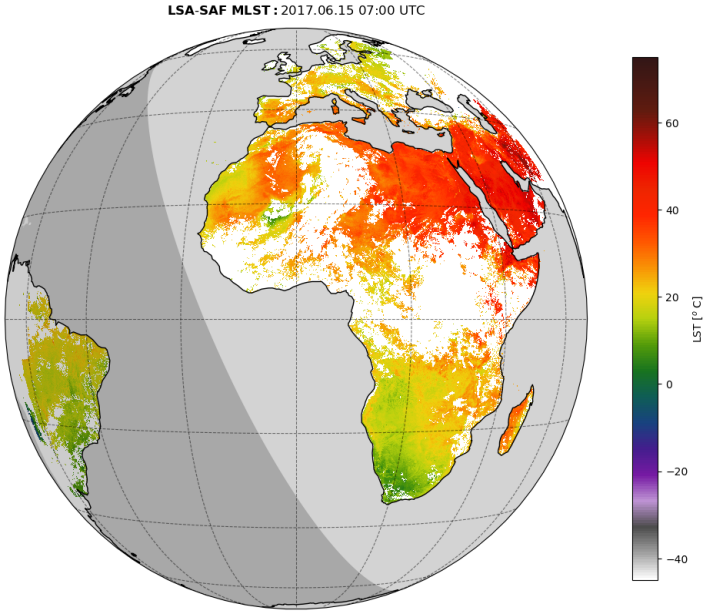


Figure 20 – Example of the Land-SAF Land Surface Temperature product on 2017/06/15 at 07:00 UTC. Source: https://landsaf.ipma.pt/en/data/products/land-surface-temperature-and-emissivity/

presented in Paul et al. (2012). To coincide with the IASI channels chosen for the database of BTs and to capture a good sample of the spectrum, we choose to keep the surface emissivity on three channels: 850, 900 and 1100 cm^{-1} (shown in Figure 21) to sample the emissivity spectrum at wavenumbers that help to discriminate surface type. The emissivity is fixed to 1 over the sea.

4.1.2 *Grid and Spatial Domain*

All the data involved in this study were re-gridded on a regular 0.25° grid over a fixed domain covering France and its surroundings, ranging between 41°N and 52°N in latitude and 5°W and 10°E in longitude. The domain can be seen in Figure 19. The choice of the domain is motivated by the necessity to reduce computational cost, the diversity of situations in such domain (mountains, sea, coastal and land pixels, variable cloud cover), and by the availability of datasets that will allow good evaluation of the results. The chosen domain is diverse and contrasting meteorological conditions can occur within a same scene. It is also often cloudy, which allows us to study the impact of missing data on the retrieval schemes. The choice of fixing a domain to work on is motivated by the image-processing techniques that will be tested. Images of size 44×60 are obtained in this domain, but images can be only partially filled depending on the IASI orbit.

On average, IASI observes a part of the chosen domain between two and four times per day. This leaves us with 15711 image samples over the ten-year period. This number is reduced to 8242 after screening out the images with a large cloud

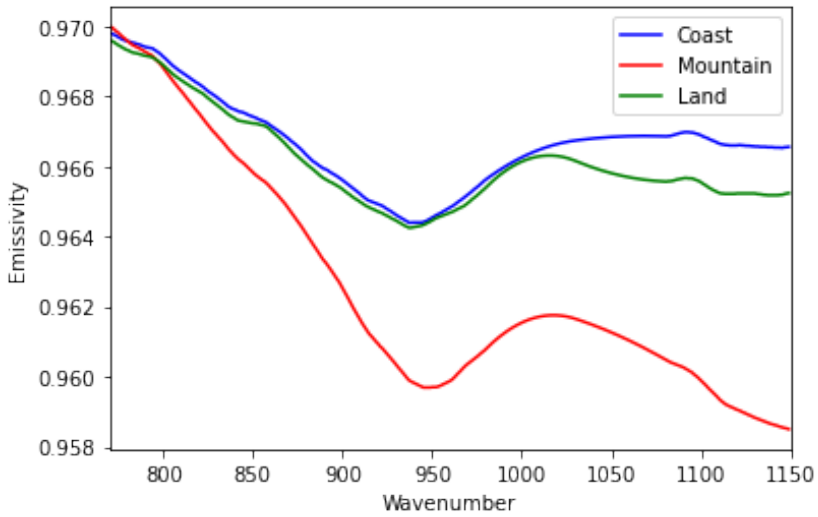


Figure 21 – A soil emissivity spectrum as a function of wavenumbers (in cm^{-1}) in January, over different areas of the chosen domain. In blue, a pixel over France's Western coast, in red a pixel over the French Alps, and in green a pixel in Central France.

fraction: For an image to be considered suitable, there should be at least 30 clear pixels in the image, and at least 2 clear pixels must be separated by 20 pixels (in Euclidean distance). This ensures that the available points are well distributed throughout the images. This results in two images per day on average.

4.2 FILLING MISSING DATA FOR CNNS

Using image-processing techniques has its advantages, notably for exploiting spatial coherency. However, these techniques also present disadvantages, among which is missing data, one of the main concerns for CNNs. The problem of missing data when using CNNs has not been addressed satisfactorily so far in the literature, in particular for satellite images.

Several sources of missing data can be listed. Firstly, clouds cover approximately 60% of the globe at any time (Rossow and Schiffer, 1999) although the exact number depends on spatial resolution. IASI observations are extremely sensitive to the presence of clouds, in particular for the retrieval of surface properties. Secondly, since we are focusing on a fixed domain, we have to face the issue of some orbits not covering the whole domain. Not processing images because missing data are present will mean that most of the IASI data flux would be lost for this type of CNN models. For current IASI processing chains, this has not been a problem since retrievals are performed at the pixel scale, it is therefore easy in that case to only process clear and available pixels. Missing data is a true limitation for the use of CNNs, and more generally image processing techniques, in the satellite commu-

nity as complete images are rarely available for visible/infrared instruments. An approach to fill missing data is therefore a necessity.

Even if a filling process is put in place in the inputs of the CNNs, the exploitation of the CNN retrievals can be done with confidence only for clear pixels where a true satellite observation exists. In this case, the filling is performed in order to not disturb too much the retrieval of the clear pixels (this will be commented in the following). The retrievals could also be exploited on missing pixels: in this case, it is expected that the use of the spatial dependencies allows to inter-/extrapolate the information available of the clear pixels into the missing pixels. This will be investigated in Section 4.4.3.

4.2.1 *Standard Spatial Interpolation and Extrapolation Methods*

Several spatial interpolation and extrapolation methods are considered here.

A Priori Filling

The common practice with CNNs is to insert a first guess value before the CNN is applied. Different a priori filling can be thought of:

1. Filling the missing data by inserting zeros values. Although this is the most widely used solution in the DL community, it is obviously a solution that is not satisfactory.
2. Filling the missing data by inserting the previous clear sky value on the particular pixel. This can be a little dangerous, notably in very cloudy regions since it is possible that the previous clear value dates from a few weeks in the past.
3. Filling the missing data with a climatological guess. This can come from monthly-mean values for each pixel location for example, but again, the monthly climatological value can be quite different from the true ST value.

These options are questionable as these artificial values can mislead the networks. This would then translate into reducing the spatial dependency we are trying to make use of in such methods.

Bilinear Interpolation and Nearest Neighbour Extrapolation

It is expected that it is more appropriate to interpolate the variable in such way to obtain plausible data in the missing pixels. This method should perturb less the CNN processing of the input image. A simple method for this is the linear or bilinear interpolation. Bilinear interpolation is an extension of the linear interpolation method for functions of two variables (i. e. x and y) on a 2D regular grid. It allows calculating the value of a function at any point, from its two nearest neighbours in each direction. The interpolation function is:

$$f(x, y) = ax + by + cxy + d,$$

where $f(x,y)$ is the interpolated value at the point (x,y) and a,b,c,d are coefficients computed from the four nearest neighbours. Bilinear interpolation is an easy-to-implement method. However, it is an interpolation method only, not suitable for extrapolation. This means that for the interpolation to be accurate and coherent, clear-sky pixels need to be available all around and in the vicinity of the considered missing pixel. Using this method on large clusters of missing data or missing parts of an image due to the orbit trajectory is thus not possible.

A possible extrapolation method associated to bilinear interpolation is the "nearest neighbour" extrapolation. This technique consists in applying bilinear interpolation where possible (i. e. filling the missing data points which have enough available neighbours) and filling the remaining of the missing data with the value of the nearest neighbour. However, if a large part of the image is missing in a corner of the image for instance, this method leads to the entire corner being filled with a single value. Again, this can mislead the CNN and can reduce the spatial information that such a network tries to leverage.

4.2.2 *Inter/Extrapolation Method Based on PCA*

As discussed earlier in Section 2.1.5, PCA is a very general statistical methodology that intends to describe the variability of multivariate observations using a new representation base defined by a limited set of patterns. For example, PCA was used to compress the IASI spectra. PCA has been widely used too in geophysics to isolate important spatial patterns, that is, the so-called Empirical Orthogonal Function (EOF), and this can be used for interpolation or extrapolation. The EOF's are equivalent to the eigenvectors, This approach was used for example in Aires, Prigent, and Rossow (2004b) for temporal interpolation and in Aires et al. (2014) for spatial interpolation.

PCA Representation of Images

Let

$$\mathbf{X} = \begin{pmatrix} X_{11} & \dots & X_{1n} \\ \vdots & \ddots & \vdots \\ X_{k1} & \dots & X_{kn} \end{pmatrix},$$

be the matrix of samples in our database, where k is the number of samples and n the number of features for each sample. The samples are images and the number of features is $n = 44 \times 60 = 2640$ pixels. Most samples

$$x_i = \left(x_{i1}, x_{i2}, \cdots, \text{NaN}, , \cdots x_{in}\right)$$

contain missing data, represented by NaN (i. e. Not a Number). We extract the EOFs (eigenvectors) $F_1, F_2, \cdots, F_n$ and eigenvalues $c_1, c_2, \cdots, c_n$ of the covariance matrix obtained from $\mathbf{X}$. The coordinates of this covariance matrix are calculated using only the clear pixels (excluding the NaNs). For generalisation purpose,

70% of the data is used, but making sure that a sufficient number of clear pixels are available over each grid point. Each image sample x_i data can therefore be written as:

$$x_i = \lambda_1 \cdot F_1 + \lambda_2 \cdot F_2 + \cdots + \lambda_m \cdot F_m, \tag{19}$$

where $m \leq n$ is the number of spatial EOFs chosen for the PCA representation (i. e. the new basis vectors) of the images, and λ_i are the coordinates of the projection of the data points in the new basis. The optimal number m to represent images needs to be optimized; it ought to be large enough to represent a sufficient part of the variability in $\mathbf{X}$, but small enough so that the following optimization scheme can find the right parameters $(\lambda_1, \lambda_2, \cdots, \lambda_m)$.

PCA Inter/Extrapolation Algorithm

An algorithm relying on this PCA representation, similar to what has been presented in Azcarate et al. (2011), is now presented. The goal is to find the coordinates $(\lambda_1, \lambda_2, \cdots, \lambda_m)$ (where m is the chosen number of EOFs) of an incomplete image $x_i = \left(x_{i1}, x_{i2}, \cdots, \text{NaN}, \cdots, x_{in}\right)$ (i. e. containing NaNs). These coordinates should be such that the representation (Equation 19) should be as close as possible to the incomplete image x_i over the clear pixels.

There are therefore two steps for this optimization scheme:

1. *First guess* - In order to find the coefficients λ_i (i. e. project the data points onto the new basis), the image needs to be complete. For this purpose, missing values are replaced by a first guess, for which several possibilities exist. The first one would be to use any of the methods proposed in 4.2.1. However, after several tests, the chosen solution adopted here is to find λ_1 such that $\lambda_1 \cdot F_1$ is as close as possible to the available data points. This is an optimization using only the first, most important, EOF and it provides a good first guess. The way to find the coordinates λ_1 is simply to test the several possibilities in a wide range [-500; +500] and chose the best solution by measuring the Root Mean Squared Error (RMSE) between $\lambda_1 \cdot F_1$ and x_i over the available clear pixels. Once λ_1 is chosen, the missing values are replaced by $\lambda_1 \cdot F_1$.
2. *Optimization algorithm* - Once the image is filled with a first guess, we are able to project the data points onto the EOF basis and thus obtain $\lambda_1, \cdots, \lambda_m$. m =5 is chosen. We can then replace the first guesses by $\lambda_1 \cdot F_1 + \lambda_2 \cdot F_2 + \cdots + \lambda_m \cdot F_m$ in the missing pixels. We then project this new image onto the EOF basis and obtain a new version of $\lambda_1, \cdots, \lambda_m$. This process is repeated until the predicted missing values have converged, which means we can interpolate or extrapolate over the missing values based on the spatial pattern obtained in the PCA. A complete image is then obtained. [6]

6. Another optimization algorithm that was tested was to reiterate the first guess method for all the λ_i, in a sequence. Results were similar, however computation time was much higher.

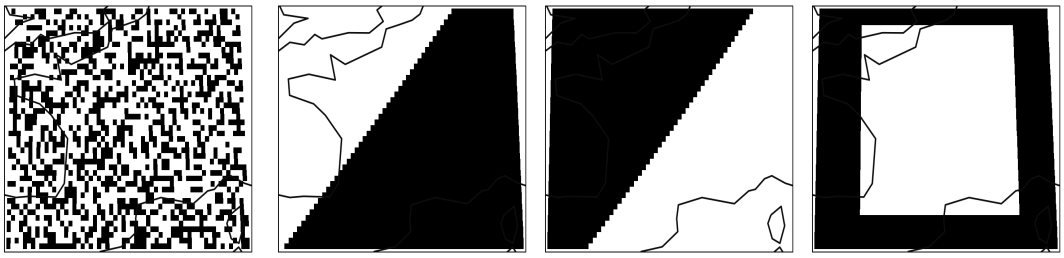
4.2.3 *Methods Comparison and Proposed Strategy*

We compare the previous PCA extrapolation technique with two of the presented solutions consisting of filling missing values with (1) monthly means (this will be the baseline), or (2) spatial bilinear interpolation followed by nearest neighbour extrapolation values.

In order to test the ability of the filling techniques, additional missing data are first introduced in our database so that results can be tested against real data. This will allow calculating the errors made by the interpolation techniques. For this purpose, three types of missing data are introduced:

- *Small holes*: randomly insert 6 missing pixels every 10 pixels in the image. They will be used to test the interpolation capacity.
- *Upper and lower triangles*: remove respectively the upper and lower triangle of the images. This will allow testing extrapolation on the edge of the image, without any guidance.
- *Central squares*: remove a square of 35×50 in the centre of the images. This serves to test large interpolation when pixels are available on the edges, providing some guidance on the interpolation ability.

Figure 22 illustrates this.



(a) Small holes (b) Upper triangle (c) Lower triangle (d) Central square

Figure 22 – Visualization of the missing data introduced into the database. White parts of the images are missing.

Table 4 shows the RMSE between the true values and the interpolated/extrapolated values. We calculate the RMSE on available pixels that were artificially removed following the three previously described missing data categories. Bilinear interpolation is more efficient on small holes while the PCA technique leads to lower errors than the two other techniques when it comes to extrapolation. This was to be expected since a simple nearest neighbour extrapolation is not ideal when there is few or no available pixels in the neighbourhood to guide it. The PCA interpolation scheme is able to find the m most important spatial features (i. e. the EOFs) and then use

them and their spatial dependency to fill large missing parts of the image based on the available information provided by the clear pixels.

This leads us to propose a strategy in two steps. First, we partially fill the images with a simple linear interpolation for the "small holes" in the image, defined as follows: there are three or more available pixels in the 5×5 window surrounding it. This allows to have more pixels available to help better constrain the following PCA inter-/extrapolation. To fill (interpolation or extrapolation) any larger holes (from clouds or missing parts caused by the orbit), the PCA-approach is used instead.

This technique is applied on both the BTs and the ST images. The final results of the process are illustrated in Figure 23. This figure shows the results when the process is applied to the ST images. This particular example is interesting to look at. It contains smaller holes in the Eastern and Southern parts of the image. Larger parts of the image are also missing due to both clouds and the orbit not covering the entire domain. The first step of the interpolation process is seen in Figure 23(b). No big discontinuities are observed, and regional specificities are well propagated in interpolated areas. The fully extrapolated image is seen in Figure 23(c). Again, it is observed that the general spatial patterns are respected, notably the gradient between land and sea and the distinction over the Alpine region, even though there are little or no pixels in those areas. A bilinear interpolation in such large areas would not respect the land/sea discontinuity, showing the interest of the PCA approach.

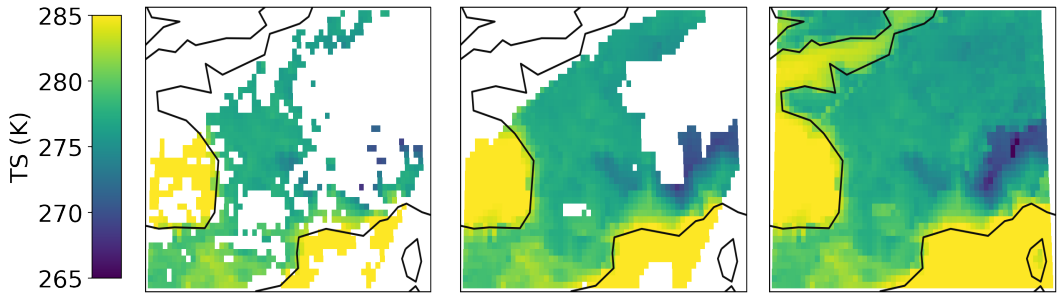
The image on the right is the image used for training the CNNs, the image on the left shows the pixels on which clear-sky retrieval statistics will be made. For pixel-wise approaches, we use only clear-sky pixels both for training and for calculating retrieval statistics.

	Interpolation on		***Extrapolation on***	
Technique	**Small holes**	**Central square**	**Upper triangle**	**Lower triangle**
Baseline	3.69	3.98	3.38	3.68
Bilinear interpolation	**1.49**	3.95	4.44	4.57
PCA inter/extrapolation	2.03	**2.65**	**2.67**	**3.02**

Table 4 – RMSE (in K) of interpolation/extrapolation techniques, on different zones (see Figure 22). The minimum RMSE is highlighted in bold for each feature.

4.3 LOCALIZATION STRATEGIES FOR REDUCING BIASES IN SURFACE PROPERTY RETRIEVALS

Using standard MLPs, as described in Chapter 3 (see Section 3.2 in particular), for the retrieval of ST has proven to be efficient (Aires et al., 2002a, 2001; Paul et al.,



(a) TS Raw (K) (b) TS partially interpolated (K) (c) TS fully extrapolated (K)

Figure 23 – Interpolation and extrapolation steps on a sample ST image starting with (a) the raw ST containing missing data, (b) the intermediary step after applying the bi-linear interpolation in small holes and (c) the completely extrapolated image obtained after the PCA inter/extrapolation procedure.

2012; Safieddine et al., 2020). However, when using a unique MLP for the retrieval around the world, regional bias errors can arise. This might be surprising as any form of NN (MLP and CNNs) is by definition an unbiased estimator. But this is an unbiased estimator on the whole learning dataset, but not locally, in each region of the spatial domain.

The remote sensing goal is to convert what is measured by the satellite into geophysical quantities that characterize the atmosphere or the Earth's surface using some ancillary data if necessary. This is an inverse problem that requires inverting the RT equation in the atmosphere. The RT equation does not require information about the location it is being applied to, since each relevant physical parameter is supposed to be provided to the RT. For atmospheric applications, this is often the case, but for surface applications the problem is much more complex. When it comes to surface retrievals, much information describing local conditions are often missing, and this will translate into regional biases. For instance, an error in the surface emissivity would lead to a regional bias in the retrieval (Paul et al., 2012).

4.3.1 *Localization in MLPs*

In the remote sensing community, MLP models are most widely used at the pixel level, meaning that the neural network is applied independently on each pixel. Generally, one network is trained on all pixels, no matter the localization of the pixel. The MLP is then expected to perform well everywhere, but this can lead to regional biases if not enough information is provided, especially if using only the IASI BTs for instance: The network has to find a compromise between all BTs-ST relations present in the whole database and a general compromise might not perform as well everywhere.

To reduce these biases, it is however possible to incorporate localization information within the inputs of the MLP. This can be done in several ways:

1 By introducing more physical variables into the inputs. This can be the land fraction, the surface altitude, or the emissivity of the surface of each pixel (Aires et al., 2001), in addition to the BTs. Adding more information in this way is an attempt to obtain a more physical MLP, able to deal with all possible situations around the world, thus obtaining a general inverse RT applicable to any location or ideally, all surface parameters that have a potential impact on the BTs (soil moisture, vegetation, soil properties, etc.) should be considered. This is an ambitious task.
2 By introducing geo-localization information on the pixel, e. g.. including the latitude and longitude of each pixel, in addition to the BTs. This is a less physical approach that we don't recommend here.
3 By training independent MLPs for each pixel of the domain. This allows the MLP to specialize on the relationship between BTs and ST at a certain location only, the missing physical variables are supposed in this case to not change for a particular pixel so that they can be omitted. The RT is therefore simplified, and the job of the MLP is made easier. However, this increases the risk of overfitting towards the training dataset compared to a global approach.

These three approaches bring additional information to the MLP, allowing it to somewhat localize the MLP behaviour for specific conditions. Each approach has its own advantages and drawbacks. The choice of the method should be based on the quality of the obtained results, and the type of use that will be made of the MLP.

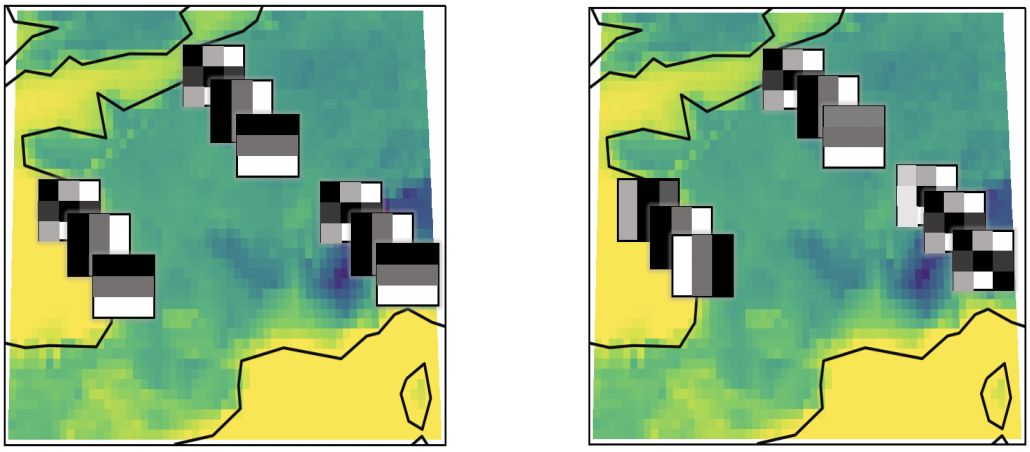
4.3.2 *Localization in CNNs*

Approaches for localization are available too for image-processing-based methods such as CNNs (see Section 3.4 in Chapter 3). They aim to incorporate the notion of spatial dependency directly into the network architecture. Generic CNNs, as described in Section 3.4, already incorporate some form of localization information from the fact that they can choose which spatial filters to apply (from a collection of learnt filters) in each part of the input image. However, generic CNNs are based on the concept of "weight sharing": The same convolution kernels are applied throughout the entire image, but react to different parts of the images. A small number of spatial filters might be sufficient for a particular location in the image, but a large stacking of convolution kernels is necessary to represent the whole image in a generic CNN.

Since we are working on a fixed spatial domain, the different parts of the image are always in the same physical location of the Earth's surface. This means that for each location of the image, a limited number of specific convolution kernels could be found instead of using a large number of generic kernels applicable everywhere. This is the role of the "locally connected layer" (Chen et al., 2015). This layer can be seen as an intermediate solution between a fully connected layer and a classic convolutional layer. It works exactly in the same way as the well-known convolu-

tional layer, apart from the fact that the parameters are not shared throughout an activation map. The optimal kernels that are obtained by the CNN will vary with the location. For instance, it can be different in the top right corner of an image than at the centre of the image, over a plain or a mountain, on a coastal area or over sea.

Figure 24 highlights the difference between the two types of layers, that was also explained in Section 3.4.3. We can see that for a classic convolutional layer, each of the three illustrated kernels is passed throughout the whole image. For the local convolution, the kernels differ for different locations, the CNN is therefore "localized". We will therefore call this new architecture a "Localized-CNN". For instance, in coastal areas, kernels will tend to spot a contrast between land and sea, whilst on mountainous regions, where LST is highly heterogeneous with a limited spatial coherency, spatial kernels will be focusing on the central pixel of the convolution pattern. Although each layer contains a lot more parameters than for a classic convolutional layer, fewer layers are needed to obtain similar results. Each layer is able to capture the spatial patterns that are important for each specific area of the input. Again, generic or Localized-CNNs have their own advantages and drawbacks and choice must be made based on results and practicality of use or interpretability of results.



(a) Classic Convolutional Layer (b) Local Convolutional Layer

Figure 24 – Left: standard CNN with same convolutional kernel layers convolved along the entire image. Right: localized convolution layer with different kernels in each part of the image.

4.3.3 *Neural Architectures and Training Specifications*

We undergo several experiments to retrieve ST over our fixed domain from the three PCs of the IASI selected BTs. The training is performed on pairs of colocated IASI BTs and ERA5 ST. The choice to use ERA5 ST for the training process is motivated by the fact that we want to facilitate the assimilation of IASI retrieved products at a later stage. Using the model-trained retrieved ST for assimilation presents multiple advantages and has been in discussion in the assimilation community for some years already (Aires, Prigent, and Rossow, 2005; Kolassa et al., 2013; Rodriguez-Fernandez et al., 2019). The first advantage is the compatibility of the retrieved product with the ERA5 model. Using ERA5 outputs for training means that the retrieved ST is coherent with the ERA5 ST. ST is a complex variable that can be difficult to define. Indeed, it is defined as the radiative temperature of the land/sea surface and depends on the thermal emission at the surface. The notion of surface can be defined differently, for example it can be at a depth of 1 mm or 1 cm. By training on the model outputs, there are no more ambiguities. The retrieved ST necessarily follows the same definition as the model ST and this hugely facilitates the assimilation of the product. The second advantage is that of course the obtained ST respects the spatio-temporal dynamics present in the IASI BTs. This is the most important aspect to facilitate assimilation. Obtained results will therefore be directly comparable to the EUMETSAT ST product which is also trained using ERA5 data, and the comparison will be made in Section 4.4.2.

Considering the relatively small number of samples in our database, we train the networks on 95% of the samples and use the remaining 5% for validation. As described in Section 4.3, multiple solutions are available to introduce localization information into the NNs. We therefore compare MLPs in which we introduce several degrees of localization information, and CNNs in which the extent of localization is defined within the architecture itself, depending on the chosen type of convolutional layer. We train the MLP models on clear pixels only and use the extrapolated images to train the CNNs[7]. In both cases, result statistics are computed on clear pixels only. We analyse the results by comparing the maps of the bias, STandard Deviation (STD) and RMSE (K), as well as correlation coefficient.

We define and describe here the architectures that will be used in the following.

Unique MLP: We consider first a MLP model as described in Section 3.2. This model is composed of one hidden layer with 100 neurons followed by a ReLU, as a non-linear activation function. The number of input neurons is determined by the number of input variables (i. e. the 3 BT PCs and any other localization variable).

Independent MLP: We propose a variant of this architecture with only 5 neurons in the hidden layer. This network architecture will be used for testing the extreme localization technique that consists in training an individual network for each pixel (i. e. $44 \times 60 = 2640$ networks).

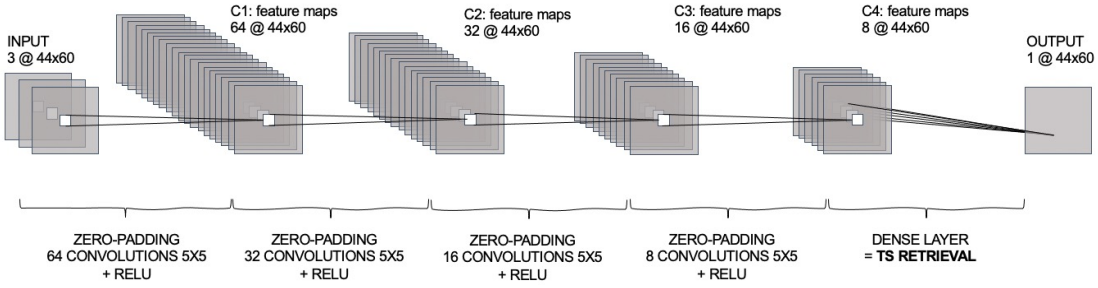
7. It should be noted that the MLPs can be trained on the original available data; whereas CNNs have to be trained on images with no missing data.

Generic CNN: The first proposed CNN model consists of the most commonly used convolutional layer. As described in Section 3.4, the main asset of this layer is the weight sharing that exists throughout the entire image. Each filter convolved through the entire image represents and detects a relevant spatial pattern (e. g.. a gradient in one direction). A relatively large amount of filters is therefore necessary because different patterns might be necessary to extract spatial features in different locations. So a whole alphabet of spatial filters needs to be found and utilized. Figure 25(a) shows the chosen architecture for this "generic" CNN. It begins with a series of convolutional layers, respectively of 64, 32, 16 and 8 kernels of 5×5 pixels size. Each convolutional layer is preceded by a step of zero-padding which controls the spatial dimension of the output and is followed by a ReLU activation function. Finally, we regress the 8 feature maps into the retrieved ST with a dense layer. This means that the ST retrieval is the result of each feature maps output multiplied by a certain weight. The weight given to each of the 8 feature maps remains constant regardless of the pixel location in the image. There is therefore not much localization in this type of architecture.

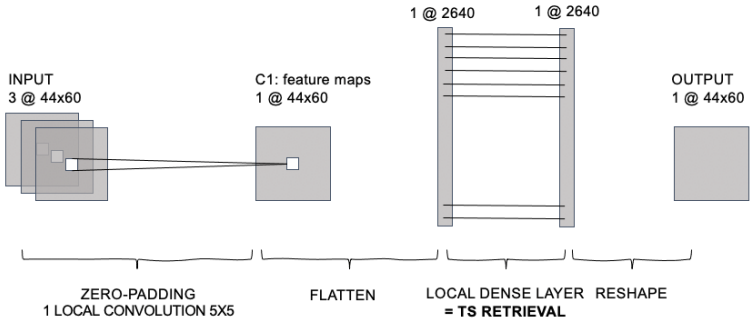
Localized-CNN: Lastly, we present a "localized" CNN. This model utilizes the locally connected layer described in Section 3.4.3. Since this layer is a version of a convolutional layer in which the weights are not shared, it is not necessary to stack many local convolutional layers. The convolution kernel will choose the most dominant spatial feature for each particular location. Experiments demonstrate that even just one convolutional layer with one kernel is enough to capture the spatial dependency present in the full image. Figure 25(b) shows the proposed architecture. We apply a single convolutional layer with a single kernel. This results in a unique feature map. In order to keep the localization aspect of this architecture, we follow by flattening this feature map and applying a local dense layer. This means that the ST retrieval weight varies according to the pixel location. We finish by reshaping the output into an image.

Hyperparameters in ML models are parameters that help to control the learning process. Neural networks are sensitive to hyperparameters. This is notably the case for very complex models, used to solve complex tasks. Thanks to the preprocessing steps applied to satellite data (i. e. the spectral PCA), the relationship between the inputs (the BT PCs) and the outputs (the ST) is simplified. The spectral PCA is a very efficient way to extract spectral information from the IASI spectrum, limiting the tasks' complexity. Therefore, the remaining inversion problem that we expect the models to solve is relatively simple. Consequently, the architectures that have been developed in this study are also quite simple. All of this means that the models presented in Section 4.3.3 are in fact only partly sensitive to hyperparameters. We recall that we train the networks on 95% of the samples and use the remaining 5% for validation.

For reference, Table 5 summarizes the training specifications for each of the four architectures used in this study.



(a) Classic CNN with 4 convolutional layers with 64, 32, 16 and 8 kernels respectively and 1 final dense layer to retrieve the TS.



(b) localized-CNN with only 1 convolutional layer and one kernel. This layer is followed by a local dense layer for the TS retrieval.

Figure 25 – A Classic CNN architecture versus a Localized-CNN architecture.

Hyperparameter	Unique MLP	Independent MLP	Generic CNN	Localized-CNN
Optimizer	Levenberg-Marquardt	Levenberg-Marquardt	Adam	Adam
Learning rate	0.001	0.001	0.001	0.001
Epochs	200	200	200	200
Batch size	128	32	16	16
Early stopping patience	2	2	2	2

Table 5 – Hyperparameters used to train the Unique MLP, Independent MLPs, Generic CNN and Localized-CNN architectures.

4.3.4 *Comparison of the Neural Retrievals*

Table 6 lists the localization strategies that are tested and analysed here. First, a comparison is made between an MLP architecture in which various localization variables are successively added, including the Land Fraction (LF), the Altitude (A), the surface Emissivity (Em), and the Latitude and Longitude (Lat/Lon). They are compared to an Independent MLP architecture with no localization inputs. Indeed, this architecture is already localized, since one network is trained specifically on each pixel. Lastly, two CNN architectures introduced previously are also considered: the generic and the Localized-CNNs.

Figure 26 compares the strategies by showing the RMSE, STD, bias and correlation maps between retrieved and target ST. First, we compare the localization

Architecture	Input variables
Unique MLP	BT PCs
	BT PCs, Land Fraction (LF)
	BT PCs, LF, Altitude (A)
	BT PCs, LF, Emissivities (Em)
	BT PCs, LF, A, Em
	BT PCs, Latitude (Lat), Longitude (Lon)
	BT PCs, LF, A, Em, Lat, Lon
Independent MLP	BT PCs
Generic CNN	BT PCs
Localized-CNN	BT PCs

Table 6 – Tested localization strategies. The left column refers to the network architecture used for the neuronal modelling. The right column indicates which variables are used as input to the network.

strategies for the MLP models. Opting for the simplest solution of training a unique MLP model on the whole spatial domain and using only the BT information leads to large regional biases, even if the spatial domain is small compared to a global scale retrieval. Naturally, this is an extreme scenario on the ladder of "localization". It remains nonetheless relevant to consider this scenario for comparison purposes. This network has to find a compromise between the physics present over the entire domain. A single network cannot capture the subtleties of the independent BTs/ST relation that lies within each pixel of the domain because there is missing data that would be necessary to fully describe the RT. The training minimizes the MSE whilst producing an unbiased output on the whole domain, which does not prevent regional biases. The bias map in Figure 26(a) suggests that these choices are not in favour of more complex locations such as the coastal and Alpine regions. This is visible too on the RMSE map where the highest errors occur in these complex regions. The retrievals over the complex regions need to be improved by introducing localization information. Introducing the land fraction as an additional input variable for example lowers the regional bias over the coastal regions, but this is not enough. The LF variable also helps to correct the systematic bias present over sea in Figure 26(a). A clear improvement can be observed when introducing additional localization variables such as the altitude and the surface emissivities. Indeed, columns (b), (c), (d) and (e) showcase an improvement in terms of RMSE, STD and bias over these complex regions. Theoretically, introducing the emissivities into the inputs should almost completely reduce regional biases, since this variable highly impacts the ST. However, the emissivities available are not perfect, especially over coastal areas, and this translates in higher biases over these areas. When adding latitude and longitude, the bias is improved over the Alps, but the coastal areas are degraded. This is not a good strategy. Lastly, the result of adding all the previous

localization variables at once in the inputs of the MLP can be observed in column (g): regional biases are significantly reduced, especially over complex areas. The localization information is beneficial for the MLP, however it is preferred here to add only "physical" variables to the input for reasons stated in Section 4.3.

A great improvement can be observed when using independent MLP models: with only 5 neurons in the hidden layer for each MLP model, a reduced RMSE is obtained and almost no regional bias is observed over the problematic regions (especially the Alpine region) because each MLP is specialized to a pixel location. These models do not have to make any compromise between regions and the remaining error and bias over notably the Alps are inherent to the true physical difficulty of the physical inversion process in these locations. For example, changes in surface emissivities can be faster over mountains and geolocalization errors are more likely to occur.

Although localized by nature of the image-processing approach, the generic CNN model produces considerable RMSE over the problematic regions, mostly from bias errors. This can be explained by the fact that:

1. Generic CNNs need a large number of convolutional layers in order to capture spatial patterns (possible only if a lot of samples are available in the learning database),
2. Spatial patterns need to be specialized in the various locations. Retrievals could benefit here from adding localization into the inputs.

However, the Localized-CNN model produces much lower bias errors. It is also the model which results in the least RMSE overall (i. e. considering both bias and STD errors). The localized convolutional layer allows capturing spatial patterns dedicated to each location, to help a specialized ST retrieval in a very simple and understandable way. For instance, it uses a spatial filter that is more "pixelized" in areas where spatial heterogeneity is high.

There is nonetheless a slight degradation of the correlations found with the image-processing approaches (i. e. Generic and Localized CNNs). This is a result of the convolutions which use the neighbouring pixels for the retrieval. This has the tendency to smooth out the retrieval fields, with the danger to slightly degrade the temporal correlations at the pixel level, due to the impact of the neighbouring pixels. This is something to be aware of when using such approaches, but overall, the retrieved dynamics is relatively constant no matter the localization method.

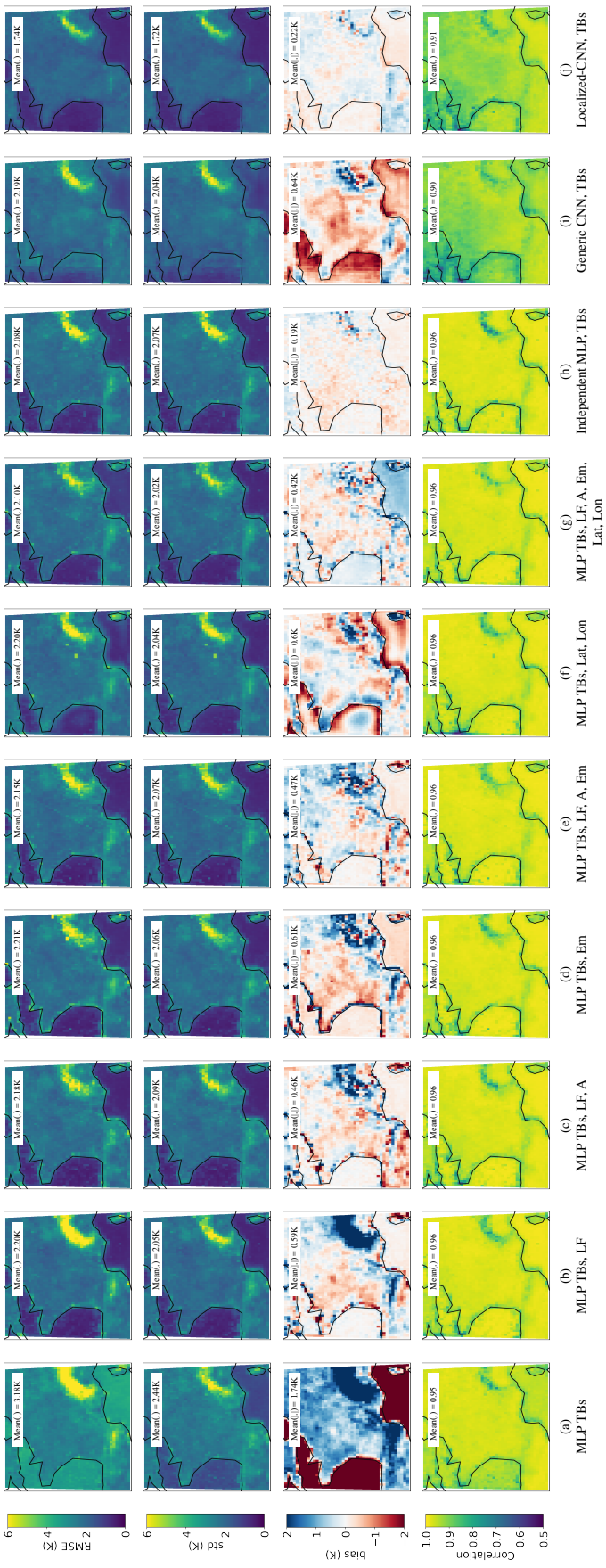


Figure 26 – RMS, STD, and bias error (K), together with the correlation coefficients between retrieved and target ST, calculated on the validation database. Columns (a) to (g) are for the global-scale MLP models, column (h) is for the Independent MLP, and columns (i) and (j) depict the generic and Localized-CNNs.

4.4 RESULTS AND EVALUATION OF THE LOCALIZED-CNN

The previous sections presented and compared several retrieval methods, each with their advantages and disadvantages. We focus here on the localized-CNN method results. This architecture is similar to a generic CNN, except that no weight sharing is applied, meaning that input spatial filters vary depending on the location in the image. Localized-CNNs do not require being as deep as generic CNNs because the obtained spatial filters are specialized for each part of the image. This allows the CNN to use specialized and dedicated spatial filters for each location, which is essential because the surface of the Earth is locally very diverse and specific. We apply a single localized-convolutional (Chen et al., 2015) layer with a single kernel of size 5×5. This results in a unique feature map which is then flattened and to which we apply a local dense layer. This way, the ST retrieval weight varies according to the pixel location. We finish by reshaping the output into an image. This architecture offers a good compromise as they use a pixel-wise approach in problematic areas (i. e. coastal and mountainous) while benefiting from spatial patterns in areas where spatial patterns are exploitable.

Two identical localized-CNN are trained respectively for ascending (i. e. orbits passing over the domain during the afternoon/evening (PM)) and descending (i. e. orbits passing over the domain during the morning (AM)) orbits. In the following, these ST estimates are referred to as the retrieved CNN-ST.

For both CNNs, the data from the year 2015 is kept aside as the testing set. 2015 was quite a standard year over France and lends itself well to validating results. The rest of the database (7 442 images) is then split on a 95/5% base for the training and validation sets respectively. We thus have The Adam algorithm (Kingma and Ba, 2014) is used to train the CNNs, and early stopping (with patience 2) is used to limit overfitting.

This section will therefore show the detailed results of this technique and evaluate them against ERA5, EUMETSAT and Land-SAF ST, and Section 4.4.3 intends to analyse the cloud impact on the ST.

4.4.1 *Analysis of the Surface Temperature Retrievals*

The localized-CNN model is made up of a localized convolution layer, consisting of a set of filters of size 5×5. During training, the filter is convolved over the input volume (the BT) to compute the dot product between the input volume and the filter weights. The weights are adjusted throughout training in order to minimize the MSE between predictions and target values. Once trained, looking more closely at the convolutional filters the CNN has learnt is a way to better understand how the CNN retrieves the ST estimate. The obtained local convolution kernels are similar in terms of patterns (but with different value ranges) for both morning and evening networks. The filters are shown for some pixels in Figure 27 for the morning network. Since weights are not shared, different weights are obtained in different parts of

the images. The obtained kernel clearly differs from one location of the image to another. For instance, depending on the input BT PC, on the Atlantic coast, the kernel focuses on the central pixel, as well as makes use of the land/sea gradient. Inland, the kernel is smoother: the network fetches information from neighbouring pixels and gives them almost as much importance as the central pixel. In the Alpine region, the spatial inhomogeneity translates into more chaotic kernel weights. The network fetches data from neighbouring pixels and tries to make a good mix of the obtained information to retrieve the ST in the central pixel. This is precisely why regional biases are reduced. The network really adapts its behaviour depending on the location of the pixel. Neighbouring pixels act as additional information to better constrain the solution, by exploiting recurrent patterns in this location, as well as increasing the signal-to-noise ratio.

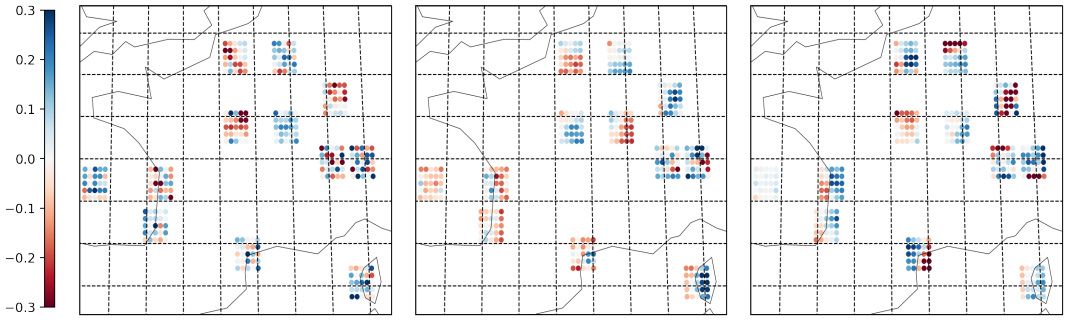


Figure 27 – Obtained weights of the 5 (lat) $\times$ 5 (lon) $\times$ 3 (PCA depth) local convolution filter of the Localized-CNN retrieval method, for the morning network, at 12 different locations in the image. From left to right, the filters for the first, second and third PCA PC of the spectral IASI information.

Four retrieval examples are shown in Figure 28. Only clear pixels are considered. One disadvantage of using a convolutional model is that this leads to a slightly smoother retrieval (compared to a retrieval that would be performed at the pixel-level), especially over land. For instance, the slightly warmer pixel present in Corsica on the second row of Figure 28(a) is smoothed in the retrieval map. This phenomenon is even more noticeable when using a generic CNN model (i. e. with shared weights and same spatial filters). The contrast between the hotter inland ST and the much cooler ST over the Alps is well respected. Despite using small and localized filters in the architecture, the larger spatial patterns are well retrieved by the localized-CNN. Comparing the reference ST (from ERA5) and the retrieved CNN-ST, we see that the retrieval is of very good agreement with the target. Over the whole ten-year period, the average STD error between the reference ERA5 ST and the retrieved CNN-ST is 0.85 K over oceanic regions, 1.83 K over land, 1.85 K over coastal regions and 2.50 K over Alpine regions in the morning. In the evening, the average STD error differs slightly with respectively 0.65 K, 2.02 K, 2.42 K and

2.00 K. Naturally, a larger error is made in problematic areas such as the Alps, especially in colder scenarios (i. e. autumn and winter seasons). This is due to the missing information that the CNN would need to perfectly explain the BT-to-ST dependencies in these areas. Overall, the retrieval is in good agreement with the ERA5 ST. Evaluation of this ST against other datasets is provided in the next section.

Figure 29 shows the time series from January 1^{st} 2015 and January 31^{st} of CNN and ERA5 temperatures on the left and, on the right, a scatter plot of the true versus retrieved CNN-ST over different regions of the image. Looking at the scatter plots, it can be seen that larger errors are obtained in the extreme ST values (i. e. hot and cold extremes), especially in pixels over the Alpine region. This can (only partly) be explained by the lower number of samples available to illustrate such scenarios, and/or can witness cloud contamination. A more in depth analysis of retrieval in the extremes is conducted in Chapter 6. In less extreme cases, the localized-CNN manages to retrieve well the expected ST. This figure shows that the localized-CNN is able to follow seasonal changes in ST as well as smaller variations within each season. These smaller variations are very well captured by the retrieval, in particular in summer months. The COR (i. e. correlation between the target ERA5 and CNN ST) and the COR of anomalies (i. e. correlation between target and CNN ST after removal of the seasonal mean computed over the whole database) are given in the top left corner of each time series. As expected, the correlation naturally diminishes marginally when the seasonal mean is removed from the observations.

4.4.2 *Evaluation*

Comparison with EUMETSAT and Land-SAF ST

In order to analyse the localized-CNN retrieval, we first compare the ERA5 target ST and the CNN-ST to the EUMETSAT ST. This database is described in Section 4.1.1. The comparison with EUMETSAT is very interesting because its PWLR3 (EUMETSAT, 2020) algorithm shares the same objective: to train a ST retrieval using the ERA5 ST target, from IASI observations. An important difference is that PWLR3 benefits from the help of microwave (AMSU and MHS) observations. A comparison is made with the Land-SAF ST too because this product is relatively independent of the ERA5 and EUMETSAT products since it comes from a different method (Generalized Split-Window (GSW) algorithm) and observations (clear-sky measurements of SEVIRI at channels 10.8 and 12.0 μm (926 and 833.3 cm^{-1})).

The Land-SAF ST (Trigo et al., 2008, 2011) product being available only over land, the comparisons with the Land-SAF database will be done only on LST. In the same way, only pixels that are clear according to both EUMETSAT and Land-SAF will be used in these comparisons. The overall RMSE, bias, STD and correlation coefficient between the four datasets (ERA5, EUMETSAT, Land-SAF, CNN retrievals) are computed over the year 2015 (i. e. the testing set). These statistics are obtained

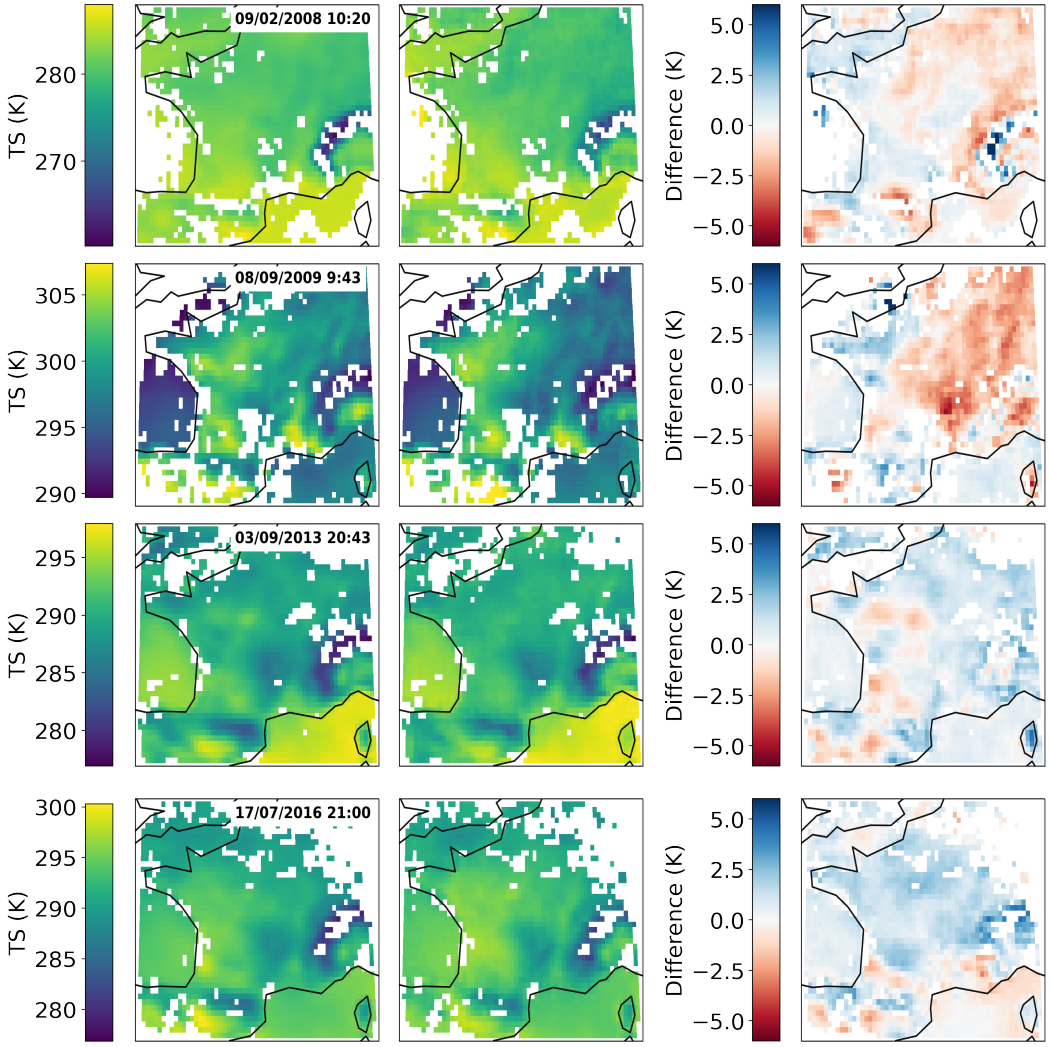


Figure 28 – Example of the ERA5 ST in column (a), the retrieved CNN-ST in column (b); followed by the difference in column (c). The date and time of the orbit is mentioned in column (a). The first two lines are examples from the morning and the two last lines are from the evening (all are taken from the respective validation sets).

by first calculating the error term, pixel per pixel, over the full domain, and then the average is computed. Statistics obtained over land and sea are distinguished. Results are given in Table 7.

Firstly, looking at results of descending orbits (i. e. in the morning) shows that the three datasets are just about equivalent in terms of STD error. The remaining STD random error can be considered inherent to the true physical difficulty of the physical inversion process, in particular the lack of some information (such as soil properties) to describe and constrain the RT equation in specific areas. The higher STD error obtained between ERA5 and the Land-SAF ST can be explained

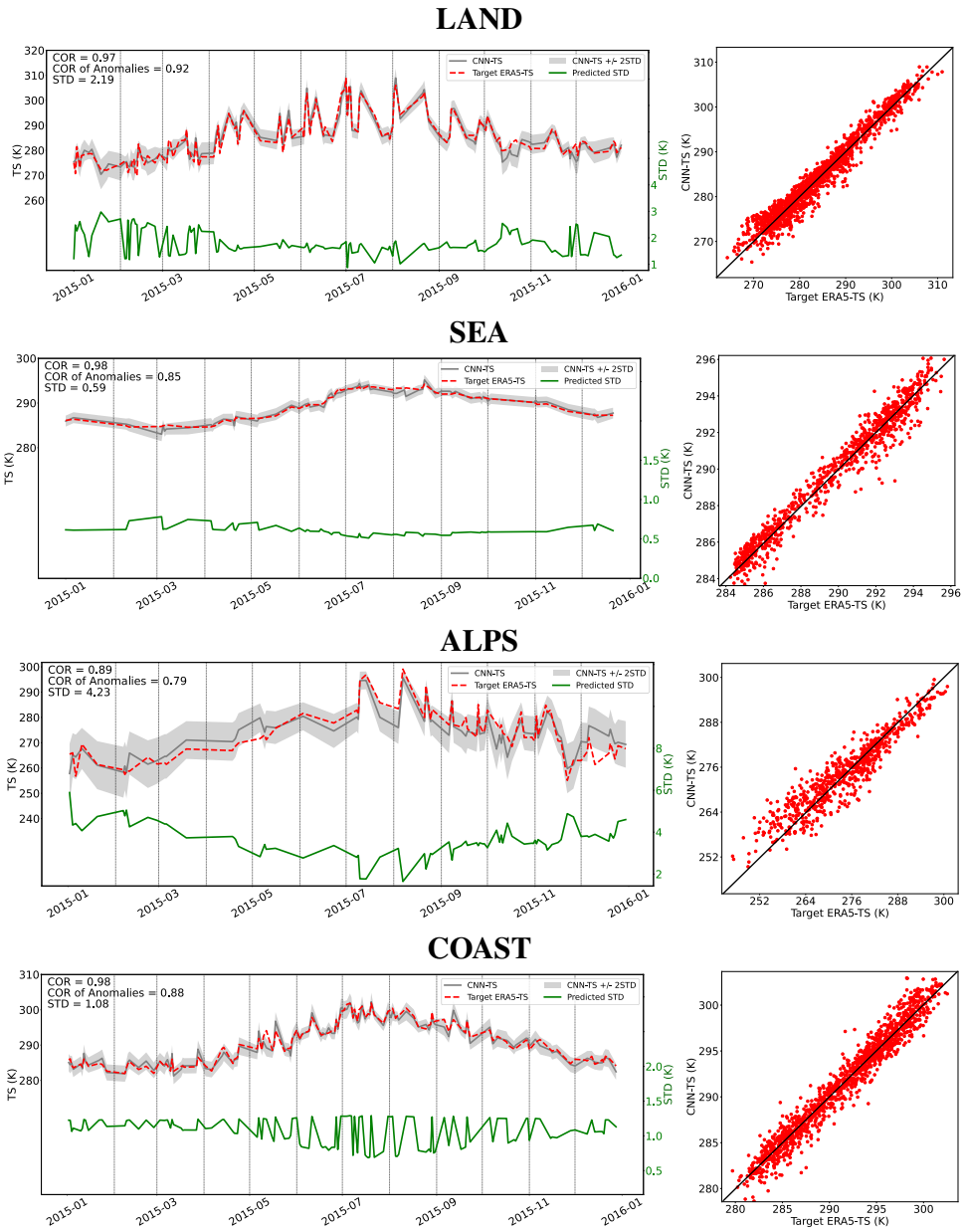


Figure 29 – Left: Time series of the target ERA5 (in red) and CNN (in grey) clear sky ST, in 2015, in pixels of different regions: over land, ocean, Alps and coast. Green lines are the predicted STD uncertainty estimates from Chapter 5. Greyed areas show the predicted $\pm$ 2STD uncertainty envelope. AM and PM retrievals were combined before plotting. Right: Scatter plot of the target ERA5 vs. CNN-ST (K). AM and PM retrievals were also combined before plotting.

Morning (AM)

Databases (A/B)	RMSE (K)		Bias (K) (A minus B)		Std (K)		Correlation	
	Land	Sea	Land	Sea	Land	Sea	Land	Sea
ERA5/Retrieved	1.87	0.87	0.08	0.11	1.82	0.85	0.98	0.97
ERA5/EUMETSAT	2.30	1.20	-1.26	0.38	1.87	1.12	0.98	0.95
ERA5/Land-SAF	2.65	N/A	-1.39	N/A	2.00	N/A	0.98	N/A

Evening (PM)

Databases (A/B)	RMSE (K)		Bias (K) (A minus B)		Std (K)		Correlation	
	Land	Sea	Land	Sea	Land	Sea	Land	Sea
ERA5/Retrieved	2.06	0.73	-0.03	0.29	2.03	0.65	0.95	0.98
ERA5/EUMETSAT	6.33	1.39	-5.19	0.36	3.49	1.27	0.83	0.91
ERA5/Land-SAF	2.52	N/A	1.00	N/A	1.70	N/A	0.96	N/A

Table 7 – Comparison between EUMETSAT, ERA5, Land-SAF and CNN ST in the morning (AM) (Top) and evening (PM) (Bottom) over the year 2015.

by the higher independence that exists between the Land-SAF and the other ST products. It can be seen that the lowest RMSE is obtained between the ERA5 ST and the CNN ST. This demonstrates a good quality retrieval on our part, since the model uses IASI data only. However, there is almost no bias between the ERA5 and CNN ST datasets, whilst biases of -1.26 K between the ERA5 and EUMETSAT products, and -1.39 K between the ERA5 and Land-SAF datasets exist. The localized CNN imposes significant localization constraints to the retrieval, thus reducing or even suppressing systematic regional bias errors. This illustrates the benefits of localization. EUMETSAT and Land-SAF LSTs are both warmer than ERA5 LST. The PWLR3 Validation Report provided by EUMETSAT (2020) compares the EUMETSAT IASI ST product to the ERA-Interim ST. This comparison shows regional differences ranging between ± 10 K. Notably, small differences over the oceanic region but larger differences over the land present in our domain are retrieved, with the EUMETSAT ST being warmer by about 1 to 5 K than the ERA-Interim ST. Part of the reason for this is probably that EUMETSAT has a global retrieval, which is more ambitious, but it is more difficult to achieve. The correlation coefficients between the four datasets shows that the temporal dynamics are well captured by all datasets. The correlation coefficients, in the morning, are all very high and similar (between 0.95 and 0.98). This indicates that all daytime products contain the dynamics of the ST.

Comparing the databases in the case of ascending orbits (i. e. in the evening) yields much different results. Again, a bias of around 1 K is still present between the Land-SAF and ERA5 LSTs, but the bias of EUMETSAT LST with the ERA5 product is of around 5K over land. Furthermore, the error STD (i. e. the random error) between EUMETSAT LST and the ERA5 LST is much higher than it was during the day (i. e. 3.49 K as opposed to 1.87 K). This suggests that there is a potential error in

the EUMETSAT training in the evening. Discussions with EUMETSAT confirmed this hypothesis, and it was suggested not to use the Metop-1 IASI L2 nighttime LST data until the next release. The latest release EUMETSAT (2022), that was only recently published, has corrected this error. In terms of bias, the localization information introduced in the CNN almost completely erases again the bias errors. This reduces the overall RMSE, making the CNN ST the closest dataset to ERA5.

Training on EUMETSAT Data: Intrinsic ST Information from IASI

The choice to use the ERA5 ST as the target for the training of the CNN is based on the will to facilitate the assimilation of IASI retrieved products later on. Training on the model has multiple advantages, one of them being the "compatibility" of the retrieved product with the model (Aires, Prigent, and Rossow, 2005; Kolassa et al., 2013; Rodriguez-Fernandez et al., 2019). This does not mean that the CNN is designed to reproduce the model. In particular, it does not reproduce the temporal behaviour because each time step is processed independently, and no temporal information is provided to the retrieval scheme. The dynamics of the retrieved product is important for assimilation. The goal of the learning algorithm is to extract the general sensitivities between BTs and ST from the learning database so that predictions are as accurate as possible. The NN will remove any behaviour not consistent with respect to the general sensitivities present in the database (Aires, Prigent, and Rossow, 2005) (i. e. remove any random error that has no link to the satellite observations).

To illustrate this, an experiment was conducted in which another localized-CNN model following the same architecture is trained using EUMETSAT ST targets instead of the ERA5 ones. The same training/validation/testing set split is used (i. e. the 2015 year is used as the testing set, 95% of the rest of the database is used for training, the remaining 5% is used for validation), and again two separate CNNs are trained for morning and evening orbits. The retrievals are then compared over the testing set. Not only is this experiment interesting to illustrate that the retrieved signal is stable (regardless of the outputs used in the training phase) but also to show the behaviour of a NN when training is done on an imperfect database (i. e. on the EUMETSAT ST database in which we know there are important errors at nighttime).

Figure 30 illustrates the comparison between the retrieved CNN-ST obtained with the CNNs trained on ERA5 and EUMETSAT. Results for morning and evening data have been combined. Both results over land and sea are considered. The distance between two datasets is calculated by $1-\text{corr}$, where corr is the correlation coefficient between the two considered datasets. The retrieved CNN-ST is slightly closer (by 0.01) to the database on which the CNN was trained. However, this is globally negligible, and it is possible to conclude that the two retrievals converge towards a unique retrieved CNN-ST. This both highlights the stability of the CNN ST and confirms that training the CNN with model outputs does not lead to a copy of the target dataset. The model finds the statistical relationship between the inputs and the

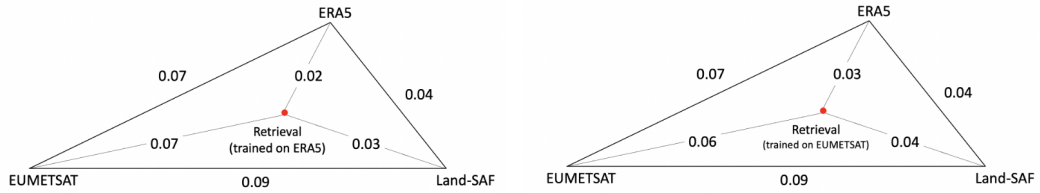


Figure 30 – Distance (defined as 1 minus the correlation) between EUMETSAT, ERA5, Land-SAF and CNN retrieval trained respectively on ERA5 (left) and EUMETSAT (right) and over the year 2015. Results for morning and evening data have been combined.

outputs, and retrieves the ST from the signal of the inputs (i. e. IASI observations only). It is also possible to conclude that an erroneous database (i. e. a database containing random errors) does not significantly mislead the CNN, the CNN is in fact able to correct these errors (Aires, Prigent, and Rossow, 2005).

4.4.3 *Preliminary Analysis of the Cloud Impact on ST*

So far, results have been analysed and evaluated against several ST databases. However, only clear pixels were considered for these evaluations, even though the IASI images were filled (interpolated), pixels below the clouds were not considered. Indeed, using infrared observations for the retrieval of surface variables is possible only for clear-sky conditions. Nonetheless, there might be here an opportunity to investigate the cloud impact on ST. The cloud impact is the impact of the clouds on the ST. We define the cloud impact as: $\mathrm{CI} = \mathrm{ST}_{\text{Cloudy}} - \mathrm{ST}_{\text{clear-like}}$, where $\mathrm{ST}_{\text{Cloudy}}$ is the true ST on cloudy conditions, available from microwave observations or numerical models. The $\mathrm{ST}_{\text{clear-like}}$ is the retrieval of ST under the cloudy pixels, but based on the information of the nearby clear pixels. This clear sky based retrieval is used as a virtual ST, as if no clouds were present. Such $\mathrm{ST}_{\text{clear-like}}$ can be obtained for example by retrieving the ST only over the clear pixels and then interpolating between the clear-sky pixels to estimate what would be the ST at a location if there were no clouds. However, when the input-to-output relationship (and/or the retrieval model) is non-linear, it is better to first interpolate the inputs and then apply the retrieval algorithm. This section will first show that the localized-CNN presents a true advantage to retrieve such a clear-like ST, before assessing and analysing the cloud impact.

Evaluating the Retrieval Under Clouds

A clear-sky ST can be obtained with the three following techniques:

1. Pixel-wise retrieval performed on clear-sky pixels followed by a spatial interpolation of the retrieved ST. The pixel-wise retrieval is performed by independent NNs for each pixel of the domain.

2. Interpolation of the clear-sky BTs followed by pixel-wise retrieval on all pixels. The pixel-wise retrieval is again performed by independent NNs for each pixel of the domain.
3. Interpolation of the clear-sky BTs followed by a localized-CNN retrieval on all pixels.

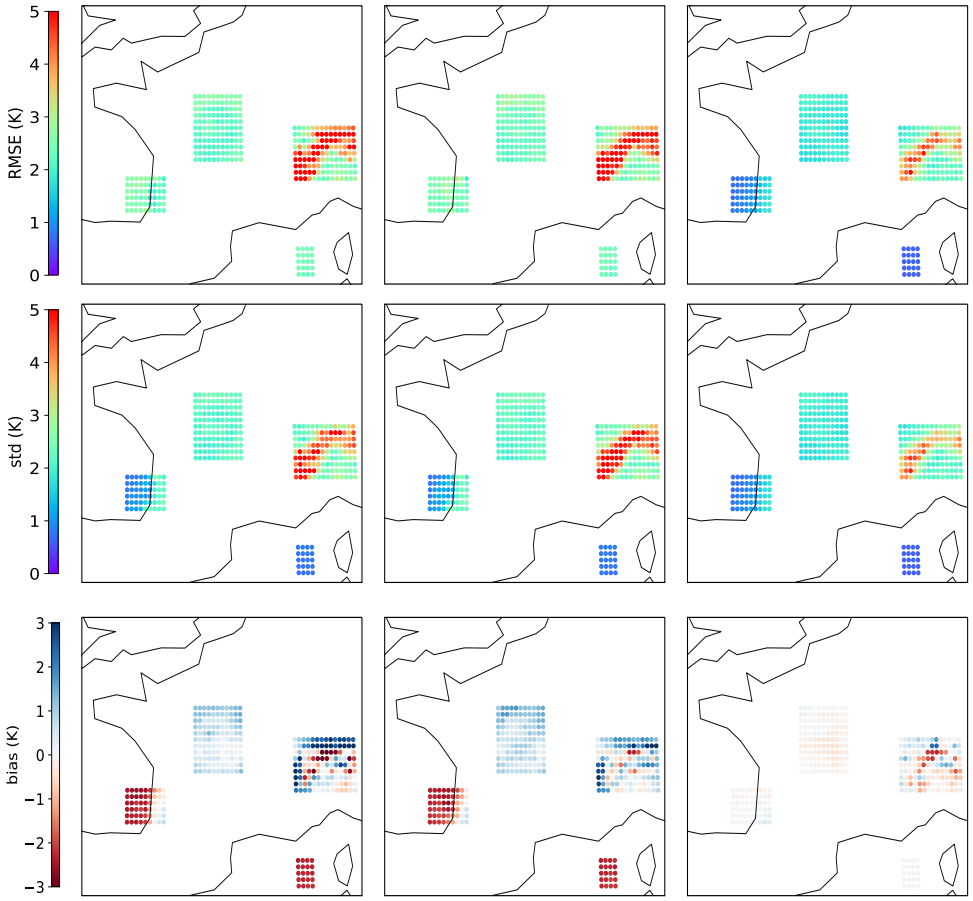
We use the inter-/extrapolation technique presented in Section 4.2, which consists in using a bilinear interpolation for small holes and a PCA-based extrapolation scheme for larger holes. The latter relies on the EOFs extracted from the clear pixels in the database. The goal is to optimize, for each image, the combination of the Empirical Orthogonal Functions (EOF) in such a way to stay as close as possible to the available clear pixels in the image. This way, the extrapolation benefits from the broad spatial patterns present in the EOFs, constrained by the available clear pixels. [8]

There is a necessity to evaluate how accurately the inversion algorithms retrieves the clear-like ST. To do this, four artificial holes are inserted into each image sample of the database over the clear pixels. One hole is centred in the middle of the domain, one is added over the complex region of the Alps, one over the Atlantic coast, and another over the Mediterranean Sea. It is then possible to train the models (on the training set, and using the validation set) and to compare the true clear-sky ST with the retrieved clear-like ST (on the testing set (i. e. the year 2015).

Figure 31 shows the RMS, STD and bias errors (K) of the three retrieval methods to compare the retrieved clear-like and the true clear-sky ST, over the four artificial holes that were added in the images. Not only is it possible to assess the reliability of the retrievals obtained underneath the clouds (i. e. clear-like ST), but also evaluate the quality of the localized-CNN method compared to the other approaches. Despite the fact that the three methods use the same spatial interpolation scheme, the localized-CNN shows results of better quality. This is because it also benefits from the use of localized spatial patterns it learnt during its training time. These localized spatial filters allow for a better interpolation than when using a pixelwise retrieval combined with an interpolation method.

Figure 31(b) shows the different statistics for the case with the inter-/extrapolation is performed on the BT inputs. The pixel-wise retrieval is then performed on all pixels of the images. An average of 3.8 K of RMSE is observed on the Alps and 2.4 K over the three other holes. These error terms can be decomposed between the STD and bias errors: over land, the error is mostly coming from random errors, whilst over the sea, it is mostly a systematic bias. Comparing with Figure 31(a), it can be seen that statistics are very similar. It can be concluded that in this specific application, the interpolation can equally be performed on the inputs and the outputs. The relationship between BTs and ST is simple and almost linear, however, using such techniques for other geophysical variables where the inversion is less straightforward could benefit from the interpolation being performed on the BTs. On

8. More details are provided in Section 4.2.2



(a) Pixel-wise retrieval, interpolation on outputs (TS)

(b) Interpolation on inputs (TBs) and pixel-wise retrieval

(c) Interpolation on inputs (TBs) and localized-CNN retrieval

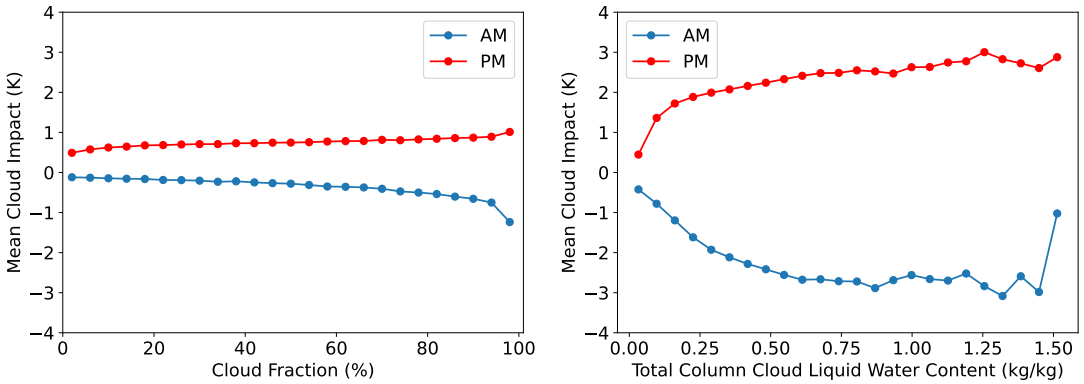
Figure 31 – From top to bottom: RMSE, STD and bias error (K) between the true clear-sky ST and the retrieved clear-like ST over the four artificial holes, over the year 2015. The interpolation of the artificial holes follow the aforementioned (a) (b) and (c) methods.

the other hand, an improvement is observed when looking at Figure 31(c). In fact, the overall RMSE over the Alps is of 2.9 K, 1.8 K over the central hole, 1.13 K over the Atlantic coast and 0.6 K over the Mediterranean Sea. Furthermore, the bias term does not exceed $\pm$ 0.1 K over any region. The difference between columns (b) and (c) almost only comes from the fact that the localized-CNN benefits from small, localized spatial patterns that help it to adapt to local conditions. This was already shown when analysing the results over clear-sky, but this experiment shows that this feature is also useful for the retrieval of a clear-like ST. This is the reason the land/sea frontier can be very clearly seen in the CNN retrieval over the Atlantic coast. This clearly shows the advantages of image-processing techniques that have specialized spatial filters according to the location in the image. This presents a real advantage when trying to obtain better spatial interpolations.

Quantifying the Cloud Impact

The previous subsection confirmed that the localized-CNN model retrieves a clear-like ST in cloudy pixels, with an error of approximately 2 K. It is therefore worth attempting to use the clear-like retrieval in cloudy pixels as a way to analyse the Cloud Impact (CI) on the ST. As previously defined, this can be done by subtracting the clear-like ST from the cloudy ST obtained by NWP models (i. e. here the ERA5 re-analysis) or all-sky retrievals such as the one from the EUMETSAT PWLR3 Algorithm. The impact of clouds on ST depends on many parameters ranging from the weather conditions, the height of the cloud to its density, including also the amount of time the cloud stays over a given pixel. It is not possible here to assess all these complex relationships, but a preliminary analysis can be undertaken to show the potential of the IASI clear-like ST. Here, we choose to look at the sensitivities of the CI to cloud properties.

Figure 32 represents the CI against the Cloud Fraction (obtained from EUMETSAT) and the total column Cloud Liquid Water Content (CLWC) (obtained from the ERA5 re-analysis). The figure gives binned statistics: the mean of the difference between the ERA5 cloudy ST and clear-like ST is computed and plotted for each bin in the x-axis. Mornings and afternoons are distinguished because the CI is very dependent on the diurnal cycle. It must be noted that the ST reaches its diurnal minimum and maximum at respectively 1 am and 1 pm. Considering that IASI orbits pass over the chosen spatial domain between 9-11 am and 9-11 pm, we should be prepared to observe a lower CI than if the analysis was conducted at the peaks of the diurnal cycle.



(a) (b)

Figure 32 – Sensitivities of the Cloud Impact (CI) to cloud properties. The CI is calculated by the difference between the ERA5 cloudy ST and the IASI clear-like ST (K). The average CI is plotted for each bin of the Cloud Fraction (a) and Total Column Cloud Liquid Water Content (CLWC) (b). In blue is plotted the morning data (AM), and in red the afternoon data (PM).

In both figures, it should be noted that when the cloud property is at zero (i. e. no clouds), the CI is zero: this is a logical behaviour, and it shows the robustness of these bias measurements. Statistics computed over the mornings and over the afternoons/evenings are symmetric: during the day, clouds tend to cool the surface by reflecting a portion of the solar energy back into space; on the contrary, during the night, clouds limit the cooling of the surface since some heat is trapped and reflected back to the surface by the clouds, thus keeping the surface warmer that it would be in clear-sky conditions. This is illustrated in Figures 32(a) and (b) where the CI is negative in the morning and positive in the afternoon. Obtaining the physical, expected, behaviour suggests that the information present in the clear-like ST has a potential to be exploited.

Looking more closely at Figure 32(a) reveals that the higher the Cloud Fraction, the higher the cloud impact on the ST (whether that be positive or negative). The same is observed in Figure 32(b) when plotting the cloud impact against the total column CLWC. Again, the most impact is observed when CLWC is highest. However, a saturation can be seen when the CLWC reaches 1 kg/kg.

However, there are many sources of uncertainties in the cloud-to-ST relationship: First, there are large uncertainties on the cloud property products from EUMETSAT and ERA5. Secondly, the CI is calculated by the difference between the ERA5 re-analysis cloudy ST and the “clear-sky” ST obtained by the CNN, both having large uncertainties, propagating to the CI value. Lastly, there is also a large uncertainty on the existing relationship between the cloud properties and the CI. The CI depends on many parameters. Simple and averaged cloud properties, such as the two considered here, can hardly fully describe the complexity of clouds, so a one-to-one relationship does not exist.

A physical algorithm using the land surface energy balance model has been used to derive an all-sky retrieval of ST (Martins et al., 2019). It would be feasible to construct a NN to estimate the CI from the cloud properties. If precise enough, this estimate could then be used to correct the clear-like ST, turning it into an all-sky estimate of the ST, based solely on infrared IASI observations and the description of the cloud properties. However, as mentioned above, relationships between cloud properties and the CI are affected by many sources of uncertainties which would result in a very large uncertainty on the CI. A neuronal modelling would retrieve the deterministic statistic, but not its spread due to uncertainties. This experiment is left for future studies.

4.5 CONCLUSIONS OF THE CHAPTER

To conclude, the standard MLP global and pixel-wise retrievals were compared to the CNN image-processing approaches. For surface properties’ retrieval (such as surface temperature), missing information on the radiative transfer (e. g.. reliable surface emissivities) is a true difficulty for the retrieval. Therefore, a global statistical retrieval scheme needs to make a compromise on the ensemble of samples

of the learning dataset by choosing a simplified relationship. This often results in regional biases, limiting the use of such retrievals for assimilation. CDF-matching techniques can be used in order to reduce these biases, like done in Aires et al. (2021), but this approach is not ideal either because, by construction, they alter the spatial patterns of the satellite observations towards the spatial pattern of the model. An alternative solution to this is using a "localization" strategy. Diverse schemes of localization were proposed here to reduce these regional biases. These schemes include adding localization variables to a standard MLP model, training independent MLPs for each pixel (this is close to using CDF-matching), or using image-processing techniques. Each approach has its own advantages and drawbacks, and results depend on the application.

Localization is an effective tool to reduce regional biases between retrieved products and the targets. It is nonetheless important to note that whilst this can be a huge advantage when hoping to assimilate retrieved products into NWP models, it can also be problematic if the said NWP model contains bias errors that are not considered the truth. By localizing, a risk is taken to reproduce certain regional biases that would not have been reproduced using a global model.

However, CNN techniques, alongside all image-processing approaches, comes with some obligatory pre-processing steps: Firstly, it is necessary to transform the orbital data into images. The choice to focus on a fixed domain over a particular domain has proven to be a reasonable choice. Indeed, it allows using a CNN or localized CNN to make full use of the local features. Secondly, it is essential to fill the missing data that are related to cloudy skies or to parts of the orbits not covering the domain. An effective algorithm was presented to fill the images both in the interpolation and the extrapolation modes. The extrapolation respects the large scale spatial patterns present in the database whilst constraining the extrapolation by staying close to available data in each image, and the interpolation follows the local conditions of the observations. The missing data scheme is essential for building the training database, but also in operational mode to actually perform the retrievals.

In depth evaluation of the localized-CNN retrieval approach was conducted. This choice was motivated by the goal to better understand and analyse image-processing techniques for the processing of remote sensing with coarse resolution instruments such as IASI. It was shown that the localized-CNN architecture is a promising DL solution that provides ST retrievals of good quality. Overall, an average STD error of 0.77 K is obtained over the sea (i. e. for the retrieval of SST), and 1.99 K over land (i. e. for the retrieval of LST) in comparison to the ERA5 re-analysis. These statistics are an improvement compared to the EUMETSAT PWLR3 retrieval that benefits from much more input information such as microwave observations. The retrieval also seems to compare to the Land-SAF similarly than ERA5 does, witnessing of the quality of the Localized-CNN retrieval. The localized-CNN architecture uses the spatial dependencies to improve the retrieval in pixels, mixed with a more pixelwise approach in complex areas where little or no spatial coherency is found. Contrarily

to more generic CNNs that require large number of filters, Localized-CNNs benefit from specialized spatial filters. This feature is a true advantage when trying to exploit spatial dependencies on a fixed domain because this way the RT is constrained for each location. The training is capable of finding the BTs-to-ST dependencies present in the IASI observations, in a relatively independent way from the ERA5 targets it was trained on. This is confirmed when comparing results produced by the same CNN trained on another target database (i. e. the EUMETSAT ST).

The evaluation of the retrievals obtained with the localized-CNN presented in this chapter paves the way towards the use of such image-processing techniques for IASI data. The next step is to manage to retrieve a global scale estimate of the ST using image-processing techniques, and then to build a long-term record from IASI observations into an operational framework.

5

ESTIMATING RETRIEVAL UNCERTAINTIES

Contents

The study presented in this chapter was published in Eulalie Boucher and Filipe Aires (2023b). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part II – Assessment. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1593–1611. DOI: 10.1002/qj.4472.

Estimating CNN retrieval uncertainties on regression problems is not an easy task, however it is an absolutely necessary one if we want to make the most out of ML techniques. In fact, uncertainties of retrieved products are important not only to assess the quality of the retrievals but also for instance to assimilate them into NWP frameworks.

In this chapter, we will present a simple and easy-to-implement scheme based on the partitioning of the input space, originally proposed in Aires and Pellet (2021). We adapt the approach to make it suitable for the localized-CNN architecture, but the algorithm could easily be applied to all the NN methods presented throughout this part of the thesis (more particularly in Chapter 4, Section 4.3.4).

Section 5.1 reminds us briefly of the data that is used and which architecture the retrieval of the ST is based upon. Section 5.2 will detail the sources of uncertainties

that need to be modelled, and Section 5.3 provides a non-exhaustive list of state-of-the-art methods that exist for NN uncertainty estimation. Finally, Section 5.4 will present our methodology and show the results on the ST application.

5.1 DATA AND RETRIEVAL METHODS

This chapter uses the same training, validation and testing databases as in the previous chapter, over the same fixed domain covering France. Details can be found in Section 4.1.

Although the methodology to estimate uncertainties is applicable to any NN-based retrieval scheme, we illustrate it here with the Localized-CNN model that is developed and evaluated in the previous Chapter (please refer to Section 4.4). To recap, the Localized-CNN is composed of a single localized-convolutional layer with a single 5×5 kernel. Experiments demonstrate that one convolutional layer with one kernel is enough to capture the spatial dependencies present in the full image. This is due to the highly localized property of the layer. This results in a unique feature map (i. e. the ST estimate).

5.2 SOURCES OF UNCERTAINTY

When retrieving the ST from IASI observations or any other geophysical variable from satellite observations, it is crucial to identify and distinguish the various sources of uncertainty that contribute to the final error. The methods presented in the previous section should be evaluated to determine if they account for these errors.

Some of these sources of uncertainty are:

1. *Errors in the (input) observations*: Firstly, the BTs measured by IASI are contaminated by some instrumental noise. This error also propagates into the spectral PCA and thus into the principal components used here, although the PCA considerably reduces this noise (Aires et al., 2002a,b).
2. *Errors in the target outputs*: The choice of the target output data (in this case ERA5 temperatures) can also introduce errors into the retrieval. Of course, the target is considered to be the "truth" by the NN, and if there are important bias errors for instance in this truth, then the model will attempt to recreate this error (however, as mentioned in Section 4.4.2, a NN is also capable of correcting certain types of error if they cannot be related to the inputs).
3. *Collocation Errors*: The satellite observations have to be collocated in time and space with the target outputs (coming from ERA5 in this case, but the targets could also come from *in situ* measurements for instance). This collocation is rarely perfect, and this introduces some errors into the input-to-output relationship we are modelling.
4. *Modelling Errors*: The retrieval method (in this case the NN) is of course not perfect itself and this creates errors in the retrieved variables. This error

can come from a number of factors: the architecture of the model being too complex or not complex enough, the optimizer not being optimal, the training database being too small etc.

5. *Errors in the RT inversion*: The RT equation links the BTs to the geophysical parameters, but is conditioned to the perfect knowledge of the conditions (i. e. it requires knowing the soil's composition for instance). Therefore, when trying to induce geophysical variables like the ST from only the BTs, there is an error term that can arise. In sum, this error basically represents the imperfect relationship between inputs and outputs.

5.3 STATE OF THE ART

A major question when working with AI techniques, and especially NNs is the common thought that such methods are a "black box", meaning that they lack interpretability and transparency. This begins with the lack of uncertainties regarding the NN's outputs. This poses several problems, in the remote sensing world (where the data is inharmonious and often unlabelled (Rußwurm et al., 2020)) but also for many other safety-critical applications (such as medical diagnosis and autonomous driving), as well as fields using reinforcement learning where uncertainties are part of the learning techniques (Lütjens, Everett, and How, 2019). Indeed, without uncertainty estimates, it is challenging to gauge the reliability of model predictions. This also impacts the decision-making process in a case where a human must make a decision based on model predictions. Uncertainty estimates also help to detect out-of-distribution samples, and in general terms help to increase model interpretability by providing insights into when and why the model might be uncertain about its predictions. The uncertainty of the output is of course not solely based on the networks' prediction ability, but is a mixture of both uncertainty included in the data and the model uncertainty (Abdar et al., 2021; Yarin, Gal, 1998). The challenge is to capture all sources of uncertainty and, even better, to differentiate the sources of uncertainty.

Because it is so crucial to have access to these uncertainties to make the most use of AI techniques in real world applications, trying to estimate NN uncertainties has been of growing interest to researches in recent years (Gawlikowski et al., 2023). This section aims to present some of the main works in this field. These can be split into three main branches.

However, as explained in the previous section, multiple sources contribute to uncertainty in the retrieved products. It is also important for proposed methods to include as many sources as possible in the uncertainty estimates. Table 8 specifies the sources of uncertainty considered in the estimate of each method we will present in the following subsections.

5.3.1 *Ensemble Methods*

Ensemble methods that are based on the predictions from several runs of (almost) the same model on perturbed input data. Perturbing the input data provides the uncertainty coming from the input data itself, whilst slightly different models gives an idea of the model uncertainties. In the contrary, errors that come from the imperfect relation between inputs and outputs are not modelled. This can be seen in Table 8.

As an example this is what is done at the ECMWF; their forecasting model (the Integrated Forecasting System (IFS)) is run a number of times with a set of perturbed initial conditions and each ensemble member uses a slightly different realisation of the physical forecast model. This method has recently been adapted for the ECMWF AI-based forecasting model (the AIFS), but because there is no longer a physical model that can be perturbed, the ensemble forecasts are generated by injecting random noise [1].

Method	**Source of Uncertainty**				
	Errors in the Inputs	Errors in the Targets	Collocation Errors	Modelling Errors	Errors in the Inversion
Ensemble Methods w. input perturbation	✓	✗	✗	✗	✗
Ensemble Methods w. model perturbation	✗	✗	✗	✓	✗
Bayesian Neural Networks	✓	bias: ✗/ STD: ✓	bias: ✗/ STD: ✓	✓	✓
Quantile Regression Neural Networks	✓	bias: ✗/ STD: ✓	bias: ✗/ STD: ✓	✗	✓
Two distinct models	✓	bias: ✗/ STD: ✓	bias: ✗/STD: ✓	✓	✓

Table 8 – A summary of the sources of uncertainty that are considered or not in the presented methods.

5.3.2 *Bayesian Neural Networks*

Bayesian Neural Networks (BNN)s model the parameters as probability distributions instead of learning a single set of fixed parameters (weights and biases) as in conventional NNs. During training, instead of optimizing for a single set of parameters, the entire distribution over parameters (that is consistent with the training data) is learnt, expressing uncertainty about the values of parameters (Kononenko, 1989; Ma et al., 2021; Wilson and Izmailov, 2020).

BNNs treat both the network weights and biases as random variables, with prior distributions placed over them. Through Bayesian inference, these priors are updated to posterior distributions using observed data during training. This is typically done using techniques such as Markov Chain Monte Carlo sampling, and this captures complex dependencies between inputs and outputs, including nonlinear relationships and interactions between variables, while properly accounting for uncertainty.

1. https://www.ecmwf.int/en/about/media-centre/news/2024/capturing-uncertainty-forecasts-based-ai

During inference, BNNs provide not only a point estimate prediction but also a distribution of predictions, reflecting the uncertainty in the model's parameters and output. This modelled uncertainty encompasses all sources of uncertainty discussed in the previous section, assuming perfect Bayesian hypothesis.

This approach was tested in a remote sensing setting in Aires, Prigent, and Rossow (2004a).

5.3.3 *Deterministic Methods*

Deterministic NNs that are trained to give a prediction of the uncertainty directly. This is possible since NNs are, once trained, deterministic functions that deliver the same results each time the same sample is fed. Ultimately, there are two solutions to quantifying the model uncertainties this way:

1. Directly trying to quantify the uncertainties inside the network (i. e. trying to predict the distribution of the predicted quantity rather than a deterministic value (Nandy, Hsu, and Lee, 2020). This is automatically done with the softmax function for classification tasks, but the probabilistic output is often misinterpreted (Vasudevan, Sethy, and Ghias, 2019). For regression tasks, an example is the Quantile Regression Neural Network (QRNN), (Pfreundschuh et al., 2018), although this option does not take into account modelling errors;
2. By using two distinct models; one to predict the desired quantity, and the other to estimate the uncertainty of the first network's prediction (Raghu et al., 2019). This technique has the advantage of being simple to implement, and all sources of uncertainty (a part from bias in target values) are modelled. This is because we consider that the first model is robust and the errors it makes is the sum of the errors coming from all sources. Since this is the error we are trying to learn (with the second network), the final estimate also incorporates all of these sources. This is the technique that is considered in this chapter.

5.4 ESTIMATING CNN UNCERTAINTIES: METHODOLOGY

5.4.1 *Univariate Case*

We start by considering the case of only one pixel of the domain for illustrative purpose.

Figure 33(a) represents a scatter plot of the target ERA5 ST against the first principal component (PC1) of the IASI BTs that includes the most information on ST. The relationship between the two variables is very strong, almost linear in fact. The horizontal axis is divided into 5 bins based on the quantiles of the distribution, and

the mean ST for each bin is represented as a black square. This represents what a NN would try to learn with this univariate input-to-output configuration.

We see a vertical dispersion in the red dots (this is represented by plot $\pm$ STD envelope around the mean). The vertical dispersion represents the missing information to retrieve perfectly the ST only from the BT PC1. Part of this dispersion is in fact explained by the two other PCs used in the inputs of the NN. This can be seen by looking at the retrieved ST, plotted in blue. However, the remaining vertical spread (i. e. the part of the vertical spread that is not explained by the other PCs and thus not present in the retrieval) describes the inherent difficulty of the inversion problem that we would like to estimate during the retrieval itself. This difficulty comes from the several sources of uncertainty that were described in the previous section.

All of these error sources, and any other sources of error which may have been omitted here, add up to the remaining vertical spread, and lead to the retrieval errors shown in red in Figure 33(b).

It is seen in Figure 33(a) that the retrieval is not able to completely explain the vertical spread in the cold extremes, however, a lot of the vertical spread in hot extremes is well retrieved by the CNN. This explains the structure of the CNN retrieval error in Figure 33(b).

If we consider that the CNN retrieval error is a summary of all the sources of uncertainty, we can extract two types of information about it: The *Bias:* $B = \text{mean}(\widehat{ST} - ST)$, where $\widehat{ST}$ is the surface temperature retrieval from the localized-CNN, and ST is the "true" ST (here taken from ERA5); and the *Error STD:* $\text{STD} = \text{STD}(\widehat{ST} - ST)$.

The error made by the CNN could be separated into bias and STD errors, but we concentrate on estimating only the CNN's error STDs. The method proposed here is however applicable to other error statistics, as it is done in Aires and Pellet (2021).

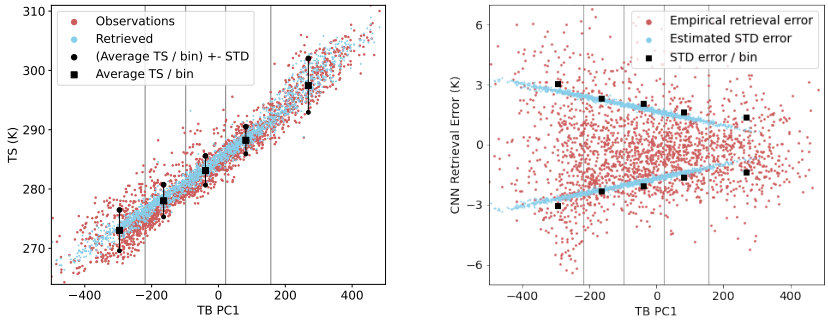
5.4.2 *Partitioning of the Input Space*

To estimate this error STD, we must define sets of samples in which we can actually empirically calculate the CNN's error STD diagnostic. We must define these sets of samples in such a way that:

1. they are adapted to each location of the image and
2. they can be obtained from the information that is available at the time of the retrieval, namely the BTs.

This will make the uncertainty estimates state-dependent.

We choose to create five bins based on the BT distribution quantiles, as is shown in Figure 33. It is considered here that it is sufficient to consider the bins on the first BT PC only since it represents more than 95% of the input information. The problem becomes more complex when the partitioning has to be done on a multivariate input



(a) (b)

Figure 33 – (a) The red dots show the ERA5 ST against the BT PC1. The BT PC1 space has been divided into 5 bins according to the quantiles of the BT PC1 distribution. The black squares represent the average ST in each bin, and the two black dots are $\pm$ STD. Blue dots are the CNN-Retrieved ST against the BT PC1. (b) The red dots show the BT PC1 versus the CNN retrieval error (ERA5 ST - CNN ST). Black squares are the error STD per bin and the blue dots are the CNN estimate of the error STD.

space, but a solution for the multivariate case is also proposed in Aires and Pellet (2021) using a clustering instead of a binning approach. Since uncertainties vary with the location in the domain, the binning is done independently for each one of the 2640 pixels of the chosen spatial domain. In each of these bins, the STD of the error obtained with the localized-CNN is calculated, as shown for one pixel in Figure 33.

5.4.3 *Training the Second CNN*

We therefore obtain images in which we represent the error STD for each pixel. We can then utilise these error STD images to train a second CNN, identical to the one that was used to calculate the errors in the first place. Figure 35 illustrates the final framework.

We use a Localized-CNN following the same architecture as the one previously used for the ST retrieval, except it is asked to retrieve simultaneously the ST and its associated error STD.

Although the Localized-CNN is trained with targets on only five discrete error STD values, it actually simulates a continuous mapping between the inputs and the retrieval error STDs, so continuous STDs are obtained when the CNN is used in operational mode. This process is shown in Figure 33(b): For this specific pixel, the CNN is asked to retrieve the black square for all samples in each bin, and the CNN learns a continuous mapping, by interpolating the binned values.

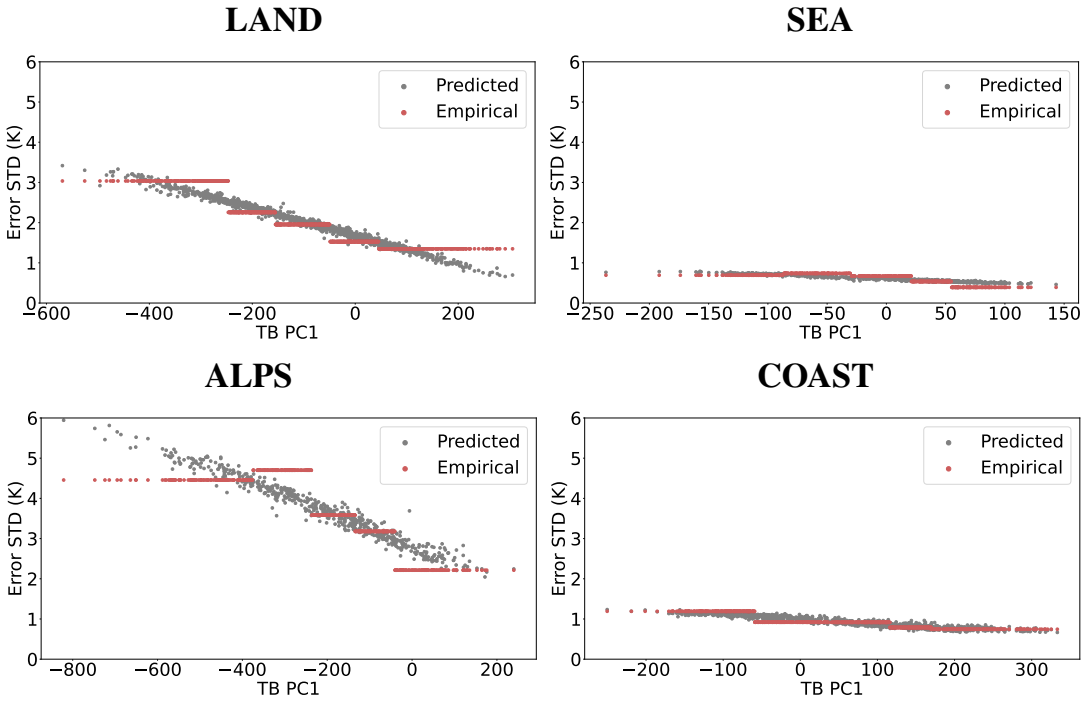


Figure 34 – Scatter plot of empirical error STD (in red) and predicted error STD (in grey) vs. the BT PC1. Results over four pixels of different regions are shown.

5.5 RESULTS AND CONCLUSIONS

5.5.1 *State-Dependent Uncertainty Estimates*

Figure 34 shows the target, empirical error STD in red and the estimated error STD in grey. These errors STD are represented in kelvins (K) and plotted against the first BT PC. Results are shown over four representative pixels. As expected, larger errors occur in the more complex parts of the image, namely the alpine region and inland. Retrieval errors over sea and coastal regions are smaller and relatively stable, respectively ranging between 0.8-1 K, and 2.2-3 K. The network predicts well the general behaviour of the uncertainties: it is continuous, and the lower the BT PC1, the higher the error. Also, looking back at the left column of Figure 29, it can be seen that the prediction, plotted as the grey line, lies mostly in the ± 2 STD uncertainty envelope. The green line shows the predicted error STD throughout the 2015 year. The fact that this line is varying suggests that the error estimate is state-dependent, which is a necessary feature. This is also shown in Figure 36.

Fundamentally, the retrieval errors (or uncertainties) form a distribution[2]. Here, the CNN is asked to only retrieve the STD of the error distribution. It is therefore important to check whether the predicted STD is in fact the STD of the empirical errors. For this, we compare the number of times the empirical error is inside the

2. Some techniques such as QRNN exist to approximate this distribution (Pfreundschuh et al., 2018).

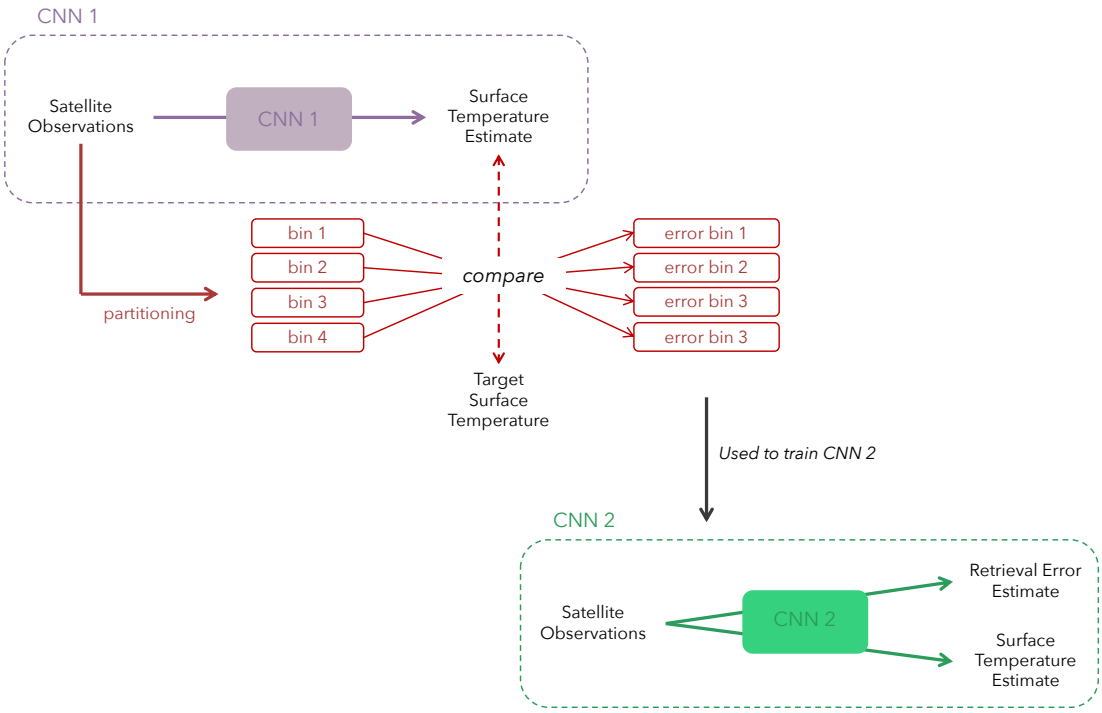
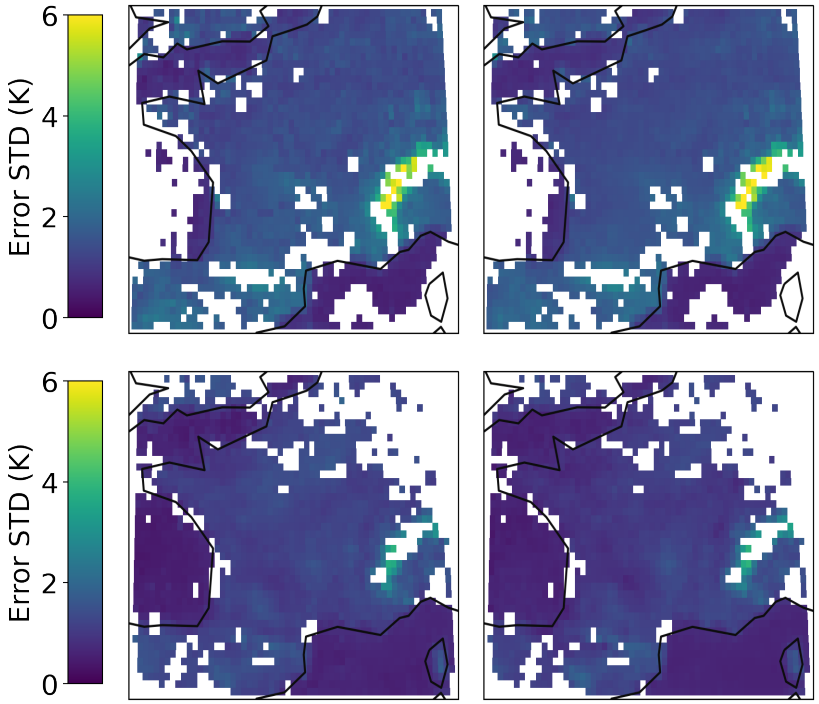


Figure 35 – Illustration of the framework to estimate uncertainties. A first CNN (in purple) is trained, and errors committed by that CNN are calculated on each bin of the input space partitioning. These errors are used to train the second CNN (in green) that estimates both the ST estimate and the uncertainty.

predicted STD and 2 STD envelopes. The empirical error is inside the 1 STD uncertainty envelope around 80 % of the time (68 % for a Gaussian), and inside the ± 2 STD uncertainty envelope around 96 % of the time (95 % for a Gaussian), depending on the localization. This gives an indication that the predicted STD describes relatively well the uncertainties of the CNN retrieval, since error PDFs can depart from a Gaussian distribution. The predicted error STD is a coherent measure of the retrieval uncertainty. The method here thus proves to be robust to correctly predict the STD of the error distribution.

5.5.2 *Conclusions*

When working with NNs (both MLPs and CNNs), it is useful to obtain an estimate of the retrieval uncertainties. Of course, this is the case for safety-critical applications, but also in a remote sensing context, especially for data assimilation where the model versus observations compromise is weighted depending on them. A solution based on Aires and Pellet (2021) is proposed here to estimate a state-dependent retrieval uncertainty. By partitioning the input space, we are able to calculate the retrieval error, and we consider that this error is a summary of all sources of uncertainty. A second CNN is then trained, giving us a simultaneous retrieval of the ST and its associated retrieval error. Although simple, this approach encompasses most uncertainty sources, is easy to implement, and provides valuable uncertainty



(a) Empirical error STD (K) (b) Predicted error STD (K)

Figure 36 – Example of the empirical (a) and predicted (b) error STD on the testing set. The first row is an example of morning data, the second row is an example of afternoon data.

estimates. Such uncertainty estimates contribute to the operational use of the NN retrievals.

6

LOCALIZATION AS A TOOL TO HELP RETRIEVALS OF EXTREMES

Contents

This chapter is based on Eulalie Boucher and Filipe Aires (2023a). « Improving remote sensing of extreme events with machine learning: land surface temperature retrievals from IASI observations. » In: *Environmental Research Letters* 18.2, p. 024025. DOI: 10.1088/1748-9326/acb3e3.

Extreme events are defined as unusual and severe weather, climate, or environmental conditions that generally lie in the tail of the observed historical distributions. A better monitoring of extreme states is essential for many socioeconomic sectors. For example, droughts have a large impact on food production and will require changes in water management strategies in the agricultural sector (Bindi and Olesen, 2011; Grillakis, 2019; Iglesias and Garrote, 2015). Prolonged heatwaves have a direct impact on health and mortality (Baccini et al., 2008; Cissé et al., 2022; Patz et al., 2014; Weilnhammer et al., 2021), with mortality from heat extremes between 1980 and 2009 which is double what it would have been without climate change (Oudin Åström et al., 2013). These ongoing changes have an impact on the public and private sectors. It is therefore urgent to better monitor such events to limit their socioeconomic impact. Currently, NWP models are the best tool to predict weather extremes. NWP relies on our ability to assimilate observations and in particular weather extremes observations.

As mentioned in previous chapters, satellite remote sensing is the ideal tool to monitor the atmosphere and the land/ocean surfaces at a global scale and with a high revisit time (e. g.geostationary satellites such as Meteosat Second Generation have

a revisit time of 15 minutes). To do so, the satellite community has increasingly relied on statistical methods, including NNs, to process Earth observations (Aires, Prigent, and Rossow, 2004b; Aires et al., 2001; Blackwell, 2005; Blackwell et al., 2008; Rodgers, 2000). However, the very nature of such statistical regression methods is to make compromises as they are trained on a large database of “pooled" samples. Data pooling here refers to the process of combining data samples from either several sources or, in the case of remote sensing, different locations around the globe. Thus, they have the tendency to better perform on average conditions and less so on more extreme conditions. A regression model provides a trivial example, because it always explains less than 100% of the true variance of a dataset (Hastie, Tibshirani, and Friedman, 2009), meaning that extremes cannot be perfectly estimated. This is what we call the “dampening” of the retrieval ranges of variability. In parallel, to simplify the implementation of the models, data from different locations are often pooled together to train a unique and global model, meaning that the same NN is used to retrieve variables over somewhat different environments and is required to make compromises. This leads, although less frequently, to what we refer to as the “inflating" of the retrieval ranges of variability. It is crucial to understand and control these two types of errors (i. e. dampening and inflating) in order to make the best possible use of emerging ML methods, in particular (but not only) for a better quantification of weather extremes.

“Localization” is a presented here to help the NN model adjust its behaviour to local conditions (see Chapter 4 and Boucher and Aires (2023b) and Boucher, Aires, and Pellet (2023)). Several forms of localization, such as additional inputs or pixel-wise models, can be used to improve NN retrievals. More recent image-processing techniques such as CNN allow for the use of spatial dependencies that both help the retrieval and reduce the data pooling at the origin of the inflating: This is another form of localization. Such localization strategies should help limit the dampening/inflating issues of statistical models. This study aims to document and improve ML retrievals by using localization to reduce the dampening and inflating errors when retrieving extreme values.

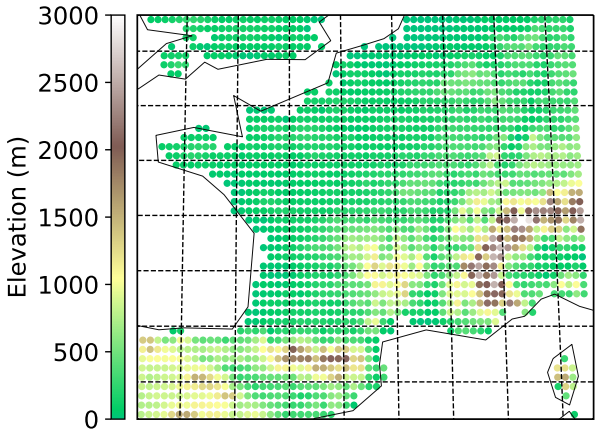
In this chapter, we focus again on the Earth’s LST which is listed as an Essential Climate Variable under the Global Climate Observing System (GCOS), and rely on IASI to estimate it (Aires et al., 2002b; Parracho et al., 2021; Paul et al., 2012; Safieddine et al., 2020). LST is a product that is operationally assimilated in NWP models (Collard and McNally, 2009; Hilton et al., 2009; Matricardi and McNally, 2014); this application is used as an example to show the improvements on the retrieval of extreme states, but the method presented here is general and can be transposed to any remote sensing application.

Section 6.1 presents the data used in this study, whilst Section 6.2 provides a detailed explanation of the dampening and inflating effects and the methods proposed here to limit them. Results and method comparisons are presented in Section 6.3. Conclusions and perspectives are given in Section 6.4.

6.1 DATA AND METHODS

6.1.1 *Datasets*

For this study, we use the same processing as described in Chapter 4, but omitting the ocean pixels: We use a database of colocated BT observations from IASI (Metop A) and ECMWF ERA5 reanalysis of LST (Hersbach et al., 2020; *IASI Level 1C Climate Data Record Release 1 - Metop-A* 2018). A surface emissivities climatology will be used as an ancillary database. Both the BT observations and the emissivities are regridded onto a 0.25° regular grid. Regridding is done by averaging the BT (or emissivity) values of IASI pixels whose centres fall in each grid point. The ERA5 LST is downloadable on the same regular grid. We use data ranging from July 2007 to December 2018 over a fixed domain covering France and its surroundings only, offering a rich variety of surfaces (including coasts, lowland and mountains). The domain ranges from 41°N to 52°N in latitude and 5°W to 10°E in longitude. Figure 37 outlines the domain by showing the elevation (in meters) of each grid point). Such a limited domain was chosen to facilitate the use of novel image-processing DL techniques.



(a)

Figure 37 – The domain chosen for this study, ranging from 41°N to 52°N in latitude and 5°W to 10°E in longitude. The elevation (in meters) is plotted to show the varying surfaces.

Satellite Observations

We regularly extract channels from the IASI spectrum between 770-980 cm^{-1}, 1080-1150 cm^{-1} and 1210-1650 cm^{-1} (with a total of 563 channels). This subset of channels is a good compromise that includes most of the information contained in these window channels while keeping some redundancy to help reduce the in-

strument noise. We then apply a PCA, an effective method to reduce dimensionality (Aires et al., 2002a,b; Calbet and Schlüssel, 2006) prior to using the ML/DL algorithms. The information is compressed using a PCA into three PCs representing 99.30, 0.54 and 0.08% of the total variance (for a total of 99.92%). These principal components contain most of the information about surface properties from the IASI spectrum. Such a high percentage for the primary PC can be explained by the high spectral correlation between the extracted channels.

We extract an image each time a Metop A polar orbit partially covers the chosen domain (approximately twice a day), which results in 15 711 "images" over the period. Only 8 242 images of the domain over the ten-year period are kept after screening out the images only covering a small part of the domain or those with a large cloud fraction. The acceptance criterion is that there should be at least 30 clear pixels in the image, and an Euclidean distance of at least 20 pixels between a pair of clear pixels. Naturally, the extracted images only contain clear-sky pixels, since no information on the surface can be obtained for cloudy pixels. An analysis of the extracted database was performed to ensure that both summer and winter months were fairly represented.

ERA5 Land Surface Temperature Database

The ECMWF ERA5 reanalysis (Hersbach et al., 2020) is used to estimate the state of the surface and atmosphere with an hourly time-step. The LST product over the domain is extracted from this reanalysis, and the observed clear-sky IASI BTs are colocated with the closest hour of the LST product.

TELSEI Emissivity Climatology

A climatology of surface emissivities derived from TELSEI (Paul et al., 2012) is used. This tool gives monthly global estimates of infrared emissivity with IASI spectral resolution. To stay coherent with the extracted IASI BT channels, three channels are kept for the emissivity database. As in Boucher, Aires, and Pellet (2023), we choose here the 850, 900 and 1100 cm^{-1} wave-numbers to be representative of the overall spectrum.

6.1.2 *LST Retrieval Methodology*

As detailed in the previous chapters, statistical models have been used in the remote sensing community for over three decades. Such methods, and more specifically NNs, have proven to be particularly efficient to retrieve geophysical variables from satellite observations, see IASI applications in Aires et al. (2002b), Aires, Rossow, and Chédin (2002), August et al. (2012), Hilton et al. (2012), Masiello, Serio, and Antonelli (2012), Parracho et al. (2021), Paul et al. (2012), Safieddine et al. (2020), and Stubenrauch et al. (2017). However, a well-known problem when using statistical models is its tendency to dampen the range of values of the retrieved

variables. This issue is not specific to NNs but is general to most statistical retrieval schemes, including random forests, linear and non-linear regressions, optimal interpolation, etc. It should be pointed out that this is also the case for non-linear regression models. We first recall the details of the models that will be used. They are described more thoroughly in Chapter 4.

For the LST retrieval, we use several models previously presented in Chapter 4 to compare the effect of localization on the approximation of extremes. Localization limits the need for data pooling and will therefore limit the inflating effect. In addition, localization improves the retrieval across all horizontal bins, and should therefore limit the dampening effect too.

A first possibility is to add more information into the NN inputs. These auxiliary variables help the NN to understand better the physical conditions on each pixel, by better constraining the RT on this location. It is therefore able to better adapt its behaviour to varying environments. In this study, we solely use surface emissivities as a localization variable. This approach is possible for both MLP and CNN models.

Alternatively, it is possible to use independent MLP models for every pixel of the domain. This way, the models do not have to make any compromise since they focus only on the input-to-output relationship over one particular pixel. By localizing using pixel-based NN, less information is required to fully describe the RT on each location: some auxiliary variables might be omitted, in particular if they are constant on each pixel.

Another localization consists in a modified version of CNNs: the *Localized-CNN*, in which the convolution layer is thus replaced with a locally connected layer (i. e. specialized spatial filters are derived for every pixel of the domain instead of using the same filters everywhere).

In the following, several retrieval methods with different localization strategies will be compared:

- A *Unique MLP* model with BTs and emissivities in the inputs. This model is composed of one hidden layer with 100 neurons followed by a linear activation function. The Levenberg-Marquardt optimization algorithm is used for training.
- *Independent MLP* models for each pixel of the domain. A separate model is trained over each pixel of the domain, composed of one hidden layer with 5 neurons followed by a linear activation function. The inputs to this model are the BT PCs. The Adam optimization algorithm (Kingma and Ba, 2014) is used for the training.
- A *Localized-CNN* model with BT PCs and emissivities. The architecture of this model is composed of a single localized-convolutional layer with a single 5×5 kernel. Experiments demonstrate that one convolutional layer with one kernel is enough to capture the spatial dependencies present in the full image. This is due to the highly localized property of the layer. This

results in a unique feature map. The Adam optimization algorithm (Kingma and Ba, 2014) is used for the training.

Training hyperparameters for each model are shown in Table 9. Please note that different optimization algorithms are used because in the case of the unique MLP, the problem is more complex: a lot of different local conditions are pooled into one big database and optimizing the model with the Adam algorithm did not lead to satisfactory results. The Levenberg-Marquardt algorithm is more efficient in this case. However, this algorithm was not available to train convolutional neural networks in Python, therefore we opted for the well-known Adam algorithm in this case. Furthermore, all models are trained with separate training and validation sets and early stopping was used to ensure that no overfitting occurs. The training time of the Unique MLP model is 5 minutes. Each independent MLP model takes 1 to 2 seconds to train, depending on the pixel location. Lastly, the localized-CNN model takes 8 minutes to train.

Hyper-parameter	Unique MLP	Independent MLP	Localized-CNN
Optimizer	Levenberg-Marquardt	Levenberg-Marquardt	Adam
Learning rate	0.001	0.001	0.001
Epochs	200	200	200
Batch size	128	32	16
Early stopping patience	2	2	2

Table 9 – Hyperparameters used to train the Unique MLP, Independent MLPs, Generic CNN and Localized-CNN architectures.

6.2 THE "DAMPENING" EFFECT OF STATISTICAL METHODS VS. THE "INFLATING" EFFECT OF DATA POOLING

This section aims to explain the reason behind this problem and provides a tool to ameliorate the retrievals of extreme values. With this in mind, we introduce the dampening effect, defined by the range of values in the retrieved LST dataset being smaller than the range of values in the original (ERA5) LST dataset. Similarly, we introduce the inflating effect as the opposite; the range of values in the retrieved LST dataset being larger than the range of values in the original dataset.

Mathematically, let us define the set of samples of extreme low and high values of LST at the pixel level (pixel i) from the 10^{th} lower ($Q_{0.1}^{ERA5}(i)$) and higher ($Q_{0.9}^{ERA5}(i)$) percentiles:

$$S_L(i) = \{y^t \mid y_{ERA5}^t(i) \leq Q_{0.1}^{ERA5}(i)\} \text{ and} \tag{20}$$

$$S_H(i) = \{y^t \mid y_{ERA5}^t(i) \geq Q_{0.9}^{ERA5}(i)\}, \tag{21}$$

where t spans the time dimension. This definition is valid both for the original (ERA5) and retrieved dataset. We then define an estimate of the range over a pixel i, for a particular dataset, as:

$$R_i = \left\{ \overline{S_H(i)} - \overline{S_L(i)} \right\}. \tag{22}$$

The range R_i can thus be understood as the average LST value of samples in $S_H(i)$ minus the average LST value of samples in $S_L(i)$. Dampening is the process by which the retrieved range, computed from the $\widehat{y}$ values, is smaller than the real range, computed from the y_{ERA5} values:

$$R_{i,RET} - R_{i,ERA5} < 0. \tag{23}$$

Because the retrieval errors are larger for these extreme values, it is often argued that the problem is due to a lack of extreme conditions in the training dataset. However, enriching the training dataset of the NN with additional extreme conditions does not solve the problem because sampling is not the real issue.

Figure 38(a) aims to illustrate what happens. We consider a NN with only one input (BT PC1) and one output (LST). Let us consider a cloud of points (in red) representing the actual true input-to-output relationship (i. e. the BT-to-LST relationship). As previously described, a NN is an algorithm that tries to approximate a complex function $f(\text{BT}) = \text{LST}$. The NN can be seen as an injective function with only one output for each input. This function is shown in blue in the figure. Since the relationship is monovariate, the NN approximates the cloud of points for every little horizontal bin along the x-axis with an average value of the target LST (the black squares on the figure). It can be seen that the NN approximation (in green) runs through these points. The extreme values (both smallest and largest) in the shaded areas will thus not be present in the retrieval range. Adding samples to represent these extreme values will not reduce the vertical dispersion. In fact, no matter the statistical method used (linear regression, random forests, NN, etc.), this vertical dispersion cannot be reduced. The only possible solution would be to introduce more PCs into the model with the expectation that the additional information will help explain and predict the existing vertical dispersion in all horizontal bins.

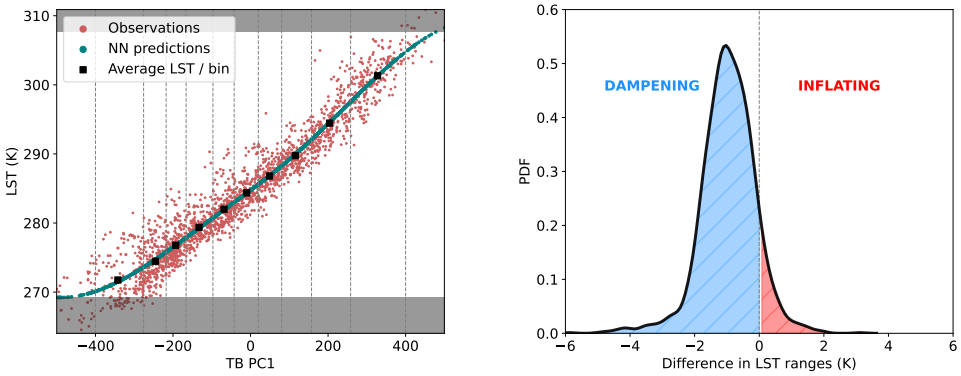
When working on a spatial domain like it is the case in this study, a form of data pooling is commonly done. Data pooling refers to combining multiple pixels of the domain to train a model, over which the BT-to-LST relationships differ. This data pooling creates an overall dampening on the global range. However, at a local scale, the LST ranges can in fact be wider than what they should be. This happens because the obtained NN performs a compromise among all the pixels. The expectation when using a global NN is that the retrieval performs well on each location (i. e. over the Alps, over coastal areas etc.). This is an ambitious task, all the more so when working at the global scale. This would be possible if we had all the necessary information to correctly represent the RT (forward or inverse) all around the world, whatever the environment type or local conditions, but it is difficult for the Earth surfaces where the RT is still not entirely satisfactory (Aires et al., 2021).

Some surface parameters describing the soil properties, or the state of vegetation are not available at the global scale, so only a simplified relationship is attainable. A global NN would in fact find the best compromise, using a simplified relationship between the satellite observations and the variable(s) to retrieve. Due to this, we may obtain an inflating effect:

$$R_{i,RET} - R_{i,ERA} > 0. \tag{24}$$

This inflation is contradictory with the nature of regression models. This is why there are fewer pixels with inflating than dampening. Figure 38b shows the dampening and inflating effects by plotting the PDF of the difference in ranges ($\Delta_R = R_{RET} - R_{ERA5}$) for all pixels. There is a net dampening when $\Delta_R < 0$, and a net inflating when $\Delta_R > 0$.

Both the dampening and inflating effects have negative consequences on the retrieval (i. e. they contribute to the error of the model), especially on extreme events. Whether one wishes to focus on minimizing one or the other depends on the particular application. For instance one may choose to rather focus on reducing the dampening error to not miss an extreme event or the inflating effects to avoid false alarms. The goal in this study is to improve the estimation of extreme cases by limiting both the dampening and inflating effects as much as possible.



(a) (b)

Figure 38 – (a) Simplified NN model to illustrate the “dampening” effect. The observation samples (LST vs BT) are plotted in red. The input space is binned, and the average output (LST) in each bin is plotted in black squares. A NN retrieval model is shown in blue, it follows well the binned averages. Shaded areas represent LST values that cannot be retrieved by the NN. (b) Illustration of the “dampening” and “inflating” effect when data is pooled. The PDF of the difference in ranges ($\Delta_R = R_{RET} - R_{ERA5}$) shows the “dampening” ($\Delta_R < 0$) and the “inflating” ($\Delta_R > 0$) effects

6.3 APPLICATION TO LST RETRIEVAL

Three diagnostics will be used in the following to quantify the two effects: (1) the percentage of time in which we observe a net dampening and inflating, (2) the average amount of dampening and inflating (expressed in K), and (3) the Mean Absolute Error (MAE) that takes into account both types of errors. The goal is to improve the estimation of extreme cases by limiting both the dampening and inflating effects.

6.3.1 *Analysis of the Dampening and Inflating Effects*

SET	VARIABLE	DIAGNOSTIC	Unique MLP with emissivities	Independent MLPs	Localized-CNN with emissivities
EXTREME LOWS S_L	Average LST	% inflating	5.7	0.1	0.7
		% dampening	94.3	99.9	99.3
		average inflating	-0.7	-0.0	-0.3
		average dampening	2.0	1.8	1.6
		MAE	1.9	1.8	**1.6**
EXTREME HIGHS S_H	Average LST	% inflating	26.8	1.6	25.5
		% dampening	73.2	98.4	74.5
		average inflating	0.6	0.1	0.4
		average dampening	-1.3	-1.1	-0.6
		MAE	1.1	1.1	**0.5**
EXTREME LOWS S_L AND HIGHS S_H	LST RANGES	% inflating	19.3	0.5	9.8
		% dampening	80.7	99.5	90.2
		average inflating	1.2	0.1	0.5
		average dampening	-2.0	-1.4	-1.1
		MAE	1.9	1.4	**1.1**

Table 10 – Summary of the dampening and inflating effects on extreme lows, extreme highs and LST ranges. The extreme lows and highs (S_L and S_H) are respectively defined as the average LSTs on pixels that lie outside the 10^{th} to 90^{th} percentiles of the ERA5 LST, and the ranges as the average value in the extreme highs minus the average value in the extreme lows (see Section 6.2). For each, we show the fraction of the time in which dampening/inflating is observed (in %), the average dampening and inflating (in K), and the MAE that takes into account both types of errors.

We show the results in Table 10, which includes the diagnostics mentioned above. This table decomposes the error originated by the dampening and inflating effects. Each diagnostic is performed on the extreme lows and highs (the retrieved vs. ERA5 LST values themselves) and the LST ranges.

It is seen that the more pooling is done in the learning database, the higher the inflating becomes. Indeed, on all sets (i. e. extreme lows, extreme highs, and LST ranges) we observe that the *Unique MLP* model inflates the most. This is due to the compromise such a model must make. The model has to find similarities between

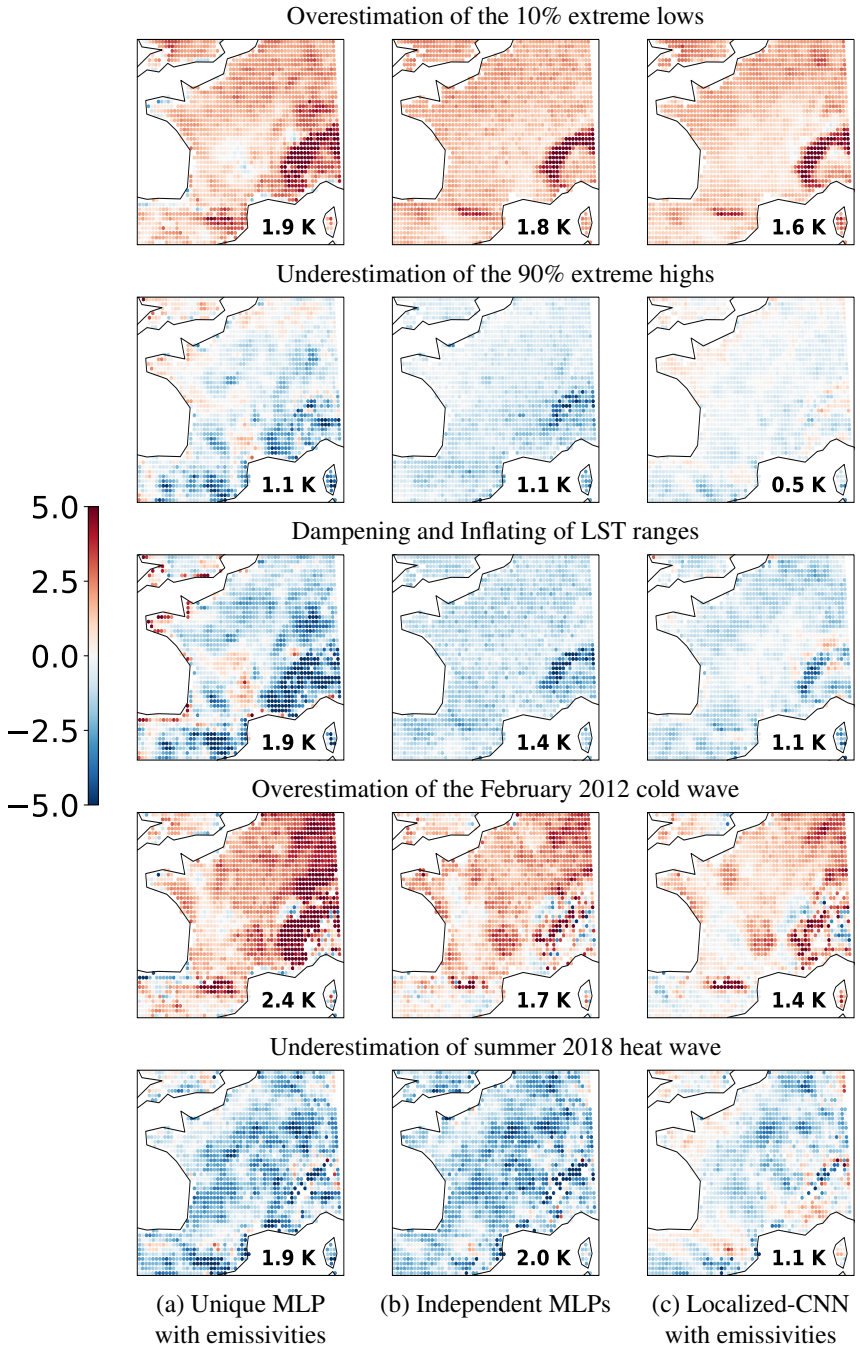


Figure 39 – Maps (in K) of the average bias in the retrieval of extreme lows (row 1), extreme highs (row 2), the ranges of LST (row 3), LST during the February 2012 cold spell (row 4), and the summer 2018 heatwave (row 5). The extreme lows (S_L) and highs (S_H) are defined as the pixels that lie outside the 10^{th} to 90^{th} percentiles of the ERA5 LST, and the ranges are the difference in average values between the extreme highs and the extreme lows (see Section 6.2. The MAE over the domain is given on each map.

the BTs-to-LST relationship on very different pixels. If one pixel has a much wider range than the other, this larger range will tend to be present in the retrieval of the

pixel whose range was smaller, leading to an inflating effect of the range. The *Independent MLPs* hardly suffer from the inflating effect: by construction, there is no pooling done for these models. The remaining inflating is just due to stochastic variations in the optimization performed during the training. The *Localized-CNN* model struggles a little more from the inflating effect in terms of percentage. However, the average inflating that occurs remains very low, below 0.5 K. The dampening effect (naturally) appears to be the most problematic source of error. Ideally, we would like the fraction of dampening to be 100% (i. e. no inflating effect), as this is the normal behaviour of a regression model, whilst having the average inflating as close to 0 as possible. The *Localized-CNN* offers a very satisfying compromise considering this objective. The *Localized-CNN* reduces drastically the average dampening that is observed in comparison to the *Unique MLP*, and even compared to the *Independent MLPs* (especially on extreme highs).

The MAE considers both types of errors, giving them an equal weight. This diagnostic gives a good overview on the performance of the different models for extreme events. The *Localized-CNN* demonstrates the best performance.

The table can be translated into a spatial representation in Figure 39. The first three rows are maps showing the average dampening and inflating effects of extreme lows (i. e. overestimation in most cases), of extreme highs (i. e. underestimation in most cases) and of LST ranges. This is done for each method and each pixel of the domain. Overall, it is over the Alps and Pyrenees mountains that the largest error is made, especially for the extreme lows. As mentioned previously, using *Independent MLPs* as opposed to a *Unique MLP* helps reduce the inflating and thus the MAE. However, there remains a significant amount of dampening. The *Localized-CNN* image-processing allows the model to utilize the spatial dependencies present in the images. The retrieval in each pixel uses the 5×5 neighbouring pixel filters. Since information is extracted from the neighbouring pixels, the model is able to better detect extreme cases. This leads to a drastic reduction of the dampening effect, but re-introduces a slight inflating, mostly in pixels surrounding the coldest pixels (i. e. mountains). In general terms, a difference in performance of the three methods can be observed between extreme highs and extreme lows. Summer and winter months being equally represented in the training database, this difference is not due to the methodology, but rather comes from an inherent physical difficulty for retrieval over the low extremes. This may be due to the cloud contamination, that is more likely to occur in cold situations (Stillinger et al., 2019).

Overall we can see that diminishing the pooling that is done in the database by localizing the retrieval algorithm as much as possible results in smaller errors in the extreme cases, both by limiting the artificial inflating effect and minimizing the dampening effect.

6.3.2 *Two Case Studies: Retrievals During Past Cold Spells and Heatwaves*

Looking at past cold spells and heatwaves over the chosen domain is another way to appreciate the quality of extreme value retrievals. It is important to be able to retrieve this type of events from the satellite observations, in particular for weather prediction.

A focus is made on the cold spell that hit France in February 2012. This cold spell lasted from the 1st to 13th February 2012 and was one of France's strongest during the last 50 years [1]. Extreme cold temperatures were recorded over most of the country, due to an anticyclonic situation which favoured the circulation of icy air. The fourth row of Figure 39 shows, for each retrieval scheme, the maps of the average overestimation during the cold spell. We can see that once again, the more localized the network is, the lower the dampening effect becomes. Localization allows to better adapt the retrieval for each pixel, and the model that use neighbouring pixels performs even better during the cold spell, showing that using spatial patterns helps to better categorize extreme situations. Comparing columns (a) and (c) shows that retrievals during the cold spell are improved by 1 K despite the same input variables being used. Using neighbouring pixels that also gravitate towards a cold temperature gives more weight to the resulting value being as cold as expected.

Similarly, another focus is made on the heatwave that France faced at the end of July - beginning of August 2018 [2]. The average underestimation of the LST during the heatwave is shown in the last row of Figure 39. This heatwave occurred during what was, at the time, the second-hottest summer since the beginning of climate records, coming just after the 2003 summer. It was one of the longest heatwaves since the post-war period. Overall, the more localized the NN is, the better it performs on the heatwave retrieval. The MAE is reduced from 1.9 K when using the *Unique MLP* to 1.1 K when using the *Localized-CNN*.

6.4 CONCLUSIONS OF THE CHAPTER

Extreme events are essential for weather and climate impact studies. The satellite retrieval of extreme events by statistical models is a true challenge. In this study, we introduce two types of errors that occur when estimating extreme events: the dampening and inflating effects. Dampening comes from the fact that, by construction, statistical methods reduce the range of the retrieval, as they hardly can explain 100% of the variance of a database. Contrarily, inflating occurs when pooling is done (i. e. mixing several datasets into one; here, the different locations of the domain) so that the behaviour over one pixel is transposed into another pixel of the domain.

1. https://public.wmo.int/en/media/news/cold-spell-europe-late-winter-20112012
2. https://www.thelocal.fr/20180829/summer-2018-frances-second-hottest-july-and-august-on-record/

This study shows that the novel concept of localizing the retrieval helps to reduce drastically the dampening and inflating effects, and therefore reduces the overall error that is made by the NNs on extreme events. This was shown both on the LST extreme values themselves and on the LST ranges. We show that using an image-processing approach such as the *Localized-CNN* has the advantage of limiting the inflating, and of drastically reducing the dampening on the ranges of variability to 1.1 and 0.5 K respectively.

This chapter focused on the LST variable on a limited spatial domain only, but perspectives include extending this work to a global scale retrieval (Aires, Boucher, and Pellet, 2021; Malmgren-Hansen et al., 2019), other surface and atmospheric variables and future instruments such as IASI-NG (Crevoisier et al., 2014) or SEVIRI instruments. The dampening effect is inevitable when using statistical models, and with the emergence of AI techniques, it is important to understand and limit its effect on the retrieval of extreme events. Monitoring extreme weather events includes looking at other atmospheric variables such as volcanic SO_2 (Karagulian et al., 2010) and CO_2 (Xu et al., 2013) for example for the detection of fires and hurricanes. Applying the presented techniques to such variables has the potential to facilitate the detection of such extreme weather events if assimilated into NWP models (Xu et al., 2013). More generally, the method presented throughout this chapter is easily transferable to other applications of NN models. Perspectives also include combining localization methods (i. e. image processing techniques and localization strategies).

EXPLORING SOLUTIONS FOR GLOBAL RETRIEVALS AND CONCLUSION

The objective of this part was to introduce and experiment with several ways of using image-processing techniques in IASI-based satellite remote sensing. This was done by experimenting ST retrievals on a small domain focusing on France and its surroundings. Targeting a regional application facilitated the development of such techniques by automatically getting some images. This way, the application was already getting closer to more traditional image-processing applications, and numerous challenges specific to remote sensing could be omitted. For example, the geometry of acquisition of IASI data was ignored, and the techniques could be applied to images of regularly spaced pixels. One remaining challenge was for instance the presence of missing data, but the small study area made it easier to deal with, and solutions could be found to interpolate the missing areas.

This simplified application permitted to develop the use of CNNs for IASI observations, and even allowed to introduce an additional goal, which was not only to improve ST retrievals but also to reduce regional biases that can arise from statistical inversions. This was done by introducing the concept of "localization", a way for the NN to adapt its behaviour to local conditions. This can be done in several ways, both for models at the pixel level and for image-processing techniques; by injecting localization variables into the model inputs or by training the model to specialize its weights for each area. Whatever form it takes, localization has the advantage to reduce regional model biases, improving the overall quality of the ST retrievals. Of course, this improvement is specific to the variable we are looking to retrieve, and the impact of localization, as well as the best way to do it will naturally vary depending on this.

However, it was shown how complex processing orbit data can be, despite the simplification made (i. e. process orbit data over a fixed domain). It is even more complicated to proceed at the global scale, but it is a necessity. We therefore discuss the potential strategies to set up a global retrieval from IASI observations:

(A) A UNIQUE GLOBAL MLP

In global retrieval schemes, it is very natural to use a unique global MLP, expected to retrieve the geo-physical variable globally (Aires, Rossow, and Chédin, 2002; Bouillon et al., 2022; Parracho et al., 2021; Safieddine et al., 2020). For this approach, the database consists of pixelwise inputs (namely BTs) located over the whole globe. A unique MLP model is then trained on samples covering the entire

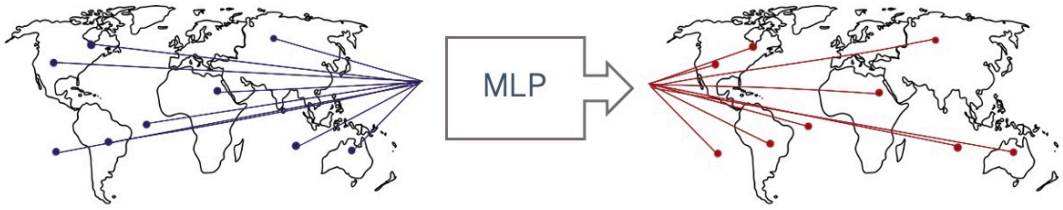


Figure 40 – Diagram of the unique MLP strategy for global retrieval.

globe, and it is expected to retrieve the pixelwise ST globally. This approach is referred to as “a unique global MLP” (Figure 40).

This approach is theoretically what the community would like to use because it is very simple to implement and to use in operational mode. Missing pixels can be omitted and there is no need for missing data filling. In the context of global assimilation, it can be convenient when retrieved variables are directly assimilated into NWP models, which is still rare (Rodriguez-Fernandez et al., 2019). The task asked to the MLP (i. e. to perform well, over very diverse environments, and often with limited available information) is however complex and ambitious. The quality of the results depend strongly on the variable to retrieve. For surface properties such as the ST, this strategy can lead to the model having to make significant compromises. To facilitate the MLP task, it is possible to introduce localization variables into its inputs, helping to adjust its behaviour to local conditions. This improves the results by reducing regional biases. Global pixelwise MLP retrieval can also be helped by using CDF-matching (Aires et al., 2021), another form of localization. This is useful, especially for assimilation of the retrieved data because it calibrates the CDF of the satellite-based estimate towards the CDF of the corresponding variable in the model. The variables are therefore harmonized and there is increased compatibility between the model and the retrieved variable.

However, using localization and/or CDF-matching often does not resolve all the problems the unique MLP model must face, especially if not enough information is provided to it to fully constrain the RT.

(b) Independent MLPs for each pixel

An alternative for pixelwise retrieval is to use independent models for each pixel of the domain (see Section 4.3). In this approach, every pixel is considered independently, and an independent MLP model is trained over each pixel. This means that for each pixel on the globe, a separate database is built, and an MLP model is trained only for this location. Each model is then asked to retrieve the ST only

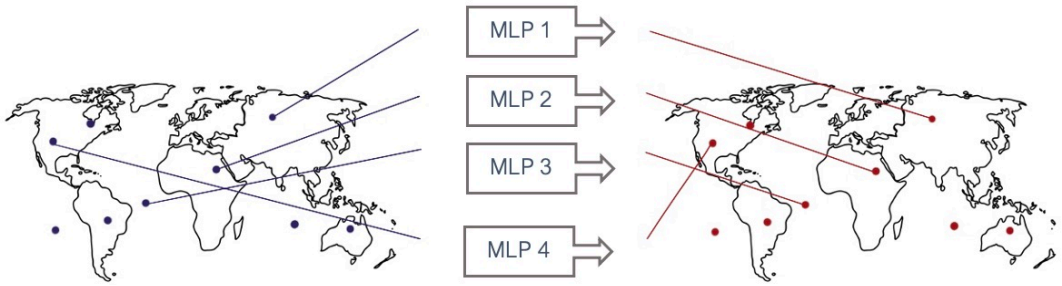


Figure 41 – Diagram of the independent MLPs for each pixel strategy for global retrieval.

for its specific location. We refer to this technique as “independent MLPs for each pixel”, see Figure 41.

In this technique, each MLP model focuses on the input-to-output relationship over a specific location only. Therefore, a very simple MLP is sufficient to capture the dependencies over each pixel, providing unbiased results. Each model requires very little parameters. Although the number of learnable parameters when imagining a global processing scheme where each pixel of the globe is characterized by its own model is larger than for the case where we consider a unique global MLP, the difference in computation time is negligible, especially because this processing can easily be parallelized. This technique is also very simple to implement and again, missing data can easily be omitted from the processing scheme.

However, the training of MLPs can be unstable on certain locations, and using independently trained models can increase the granularity of the retrievals and may introduce some noise in the retrieved product. The impact this may have on the quality of the results depends on the variable to retrieve.

(c) MULTIPLE STATIC DOMAINS AND CNNS

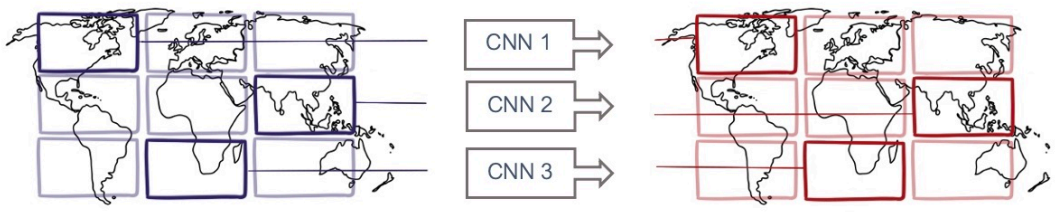


Figure 42 – Diagram of the multiple CNNs over static domains strategy for global retrieval.

(a) (b) (c)

Figure 43 – Diagram of the possible static domains.

Another solution is to consider one or multiple static domains over the globe, and to train independent CNN models on each of the static domains. Each CNN model is expected to retrieve the ST (or any other variable) only over its static domain. This approach is referred to as using “several static domains”. Figure 42 illustrates this approach. The static domains could be under three forms, that are also illustrated in Figure 43:

— Regularly spaced tiles can be cut out (as shown in Figure 43(a)), and the orbits can be projected inside (as it was done over France). In the figure for instance two tiles are shown, and the orbit is projected inside each tile. However, this could lead to potential discontinuities between each domain, causing problems for example in the assimilation process. This problem could however be resolved by considering overlapping domains.

— A large global domain (i. e. one large image covering the full globe) but the difficulty here is when using polar-orbiting satellites with which it is impossible to obtain a global image at the same observation time. Obtaining a global image would require either concatenating several orbits, and some points would be retrieved several times at different times of the day (dissolving the spatial coherency), or, as shown in Figure 43(b), projecting a single orbit onto the map and leaving the rest as missing pixels.

— The orbit can be conserved under their original format; The IASI instrument has a swath width of 2200 km, decomposed into 30 field of views each. Each of the four simultaneous sounding pixel collects the radiance spectrum formed of the 8461 IASI channels. The number of swaths m depends on the orbits. Each IASI orbit can therefore be considered as an image of size $2m \times 60$. From this, two orbits can be considered the same if they observe the same pixels at the same time of the day. IASI has a revisit period of 29 days. With 14 orbits a day, there are therefore $14 \times 28 = 392$ different orbits possible. This is seen in Figure 43(c), in which two orbit domains can be seen.

This approach has the advantage of allowing the use of both generic CNNs (if the domain is small enough to avoid regional biases) and a more localised version in which filters specialize to certain parts of the images (see Section 3.4.3).

Depending on the size of the chosen static domains, pre-processing steps such as filling missing data can be more or less easy to implement. If domains are of

similar size to the one used in this study (i.e. 44×60 on a 0.25° regular grid), inter-/extrapolation algorithms based on EOFs can be developed for each domain, and cloudy pixels can be filled. For larger domains (like the global image or the orbits), the downside is that using an EOF-based inter-/extrapolation method is problematic to use a spatial pattern in the form of EOFs that will link phenomena in the North and South Pole. Therefore, a new filling strategy should be put in place to pre-process the images. This will be discussed in the next chapters.

(D) VARYING SPATIAL DOMAINS AND A UNIQUE CNN

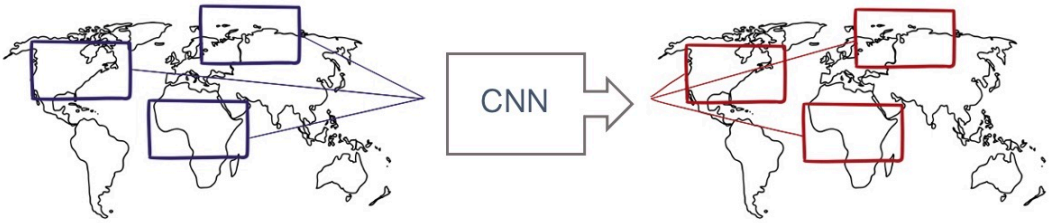


Figure 44 – Diagram of the unique CNN over varying spatial domains strategy for global retrieval.

(a) (b)

Figure 45 – Diagram of the possible varying domains.

The most logical technique when using image-processing is to use images covering varying spatial domains. This is the standard approach for most uses of CNNs (Aires, Boucher, and Pellet, 2021). This would consist in fixing an image size and extracting images of such size all over the globe. A unique CNN model would then be trained on generic images from all around the world (Aires, Boucher, and Pellet, 2021). This CNN is expected to retrieve the ST for images over the entire globe. We refer to this approach as “varying spatial domain”, see Figure 44.

Two forms of spatial domains could be considered:

— Regular sized-images can be extracted directly from the orbits, as shown in Figure 45(a). For instance, here, 5 images would be extracted.

— The orbits can be kept under their original format; The IASI instrument has a swath width of 2200 km, decomposed into 30 fields of view each. Each of the four simultaneous sounding pixel collects the radiance spectrum formed of the 8461 IASI channels. The number of swaths m depends on the orbits. Each IASI orbit can therefore be considered as an image of size $2m \times 60$. This is shown in Figure 45 in which we have three images, corresponding to the three orbits shown.

In both cases, it is however necessary to deal with missing data since CNN models require complete images. This step can be difficult, especially in the case where orbits are conserved as such, because using an EOF-based inter-/extrapolation method is problematic since it will link phenomena in the North and South Pole. Perhaps a new filling strategy should be put in place to pre-process the images.

Theoretically, the stacking of many convolutional layers should allow the CNN to learn to detect spatial patterns that are general enough to help the retrieval on any of the spatial domains present in the database. In this case, the strategy is close to the unique global MLP because a general model is asked to perform well in very different environments. The difference is that instead of a pixel information, the CNN uses a local spatial domain and pertinent local spatial patterns helping the retrieval must be found during the training.

In this case, localization can be done under the form of auxiliary (i. e. localization) variables, that can facilitate the CNN's retrieval.

This technique was tested in an SST retrieval context in Aires, Boucher, and Pellet (2021) on square images, and in Malmgren-Hansen et al. (2019) using the orbits directly as images. It is more complicated for the CNN to retrieve spatial patterns because the images represent much contrasted environments, but with a deeper architecture (that contains many convolutional layers to be able to cover all kind of spatial patterns present in the database), this should be feasible.

GENERAL CONCLUSIONS

The objective of this thesis is to create global and improved IASI-based retrievals of surface and atmospheric temperatures with the use of CNN models. The second part of this thesis will focus on this, and a dedicated approach needs to be found. The previous sections summarized the few approaches that could be considered to reach this goal. We are convinced that some form of localization is vital to create the best IASI-based retrievals, even more-so if the focus is made on surface property retrievals. However, although the localized-CNN showed the best results in terms of reducing biases and improving the quality of results in general, it is also one of the most restrictive in terms of development (i. e. it requires a static domain and

much computational power if that domain is large). Unfortunately, it is complicated to have access to necessary resources to perform global localized-CNN retrievals.

The following part of the thesis will use traditional CNN modelling approaches, on orbit spatial domains. The localization will be done by feeding additional information to the networks (i. e. surface emissivities and land/sea mask). The final work frame will be presented in full in the following chapters.

Part II

TOWARDS A GLOBAL CNN PROCESSING CHAIN OF IASI ORBITS

INTRODUCTION

The first part of the thesis helped us to get familiar with the best ways to apply image-processing techniques to IASI observations. To do this, a relatively small area (covering France and its surroundings) of the Earth was considered, considerably facilitating the processing.

It was shown in Part I that localization was an efficient too to reduce regional biases on ST retrievals, and that an innovative way to do so was to use localized-CNNs, that adapt their spatial filters to the area of the image. The conclusion of Part I showed that adapting the localized-CNN to the globe can pose additional challenges, especially due to the missing values.

Partly due to the limitation of computational resources, the choice was made to take another approach for the global retrieval of atmospheric and surface temperatures, and to localize the retrievals using additional input information only. In this second part of the thesis, we will focus on retrieving cloud properties as well as atmospheric temperature profiles. Both of these variables are less sensitive to local conditions, and not localizing the CNNs with locally-connected layers will have less of an impact than it did on surface properties.

The chosen approach will be detailed in this part of the thesis; It is based on using partial convolutions to deal with missing data. This method, that was discovered during the second year of the PhD, has highly facilitated the use of image-processing techniques for global retrievals based on IASI observations. In the remainder of this thesis, an image will be defined as a long strip equalling to an ascending or descending orbit.

This part is split into two chapters. Chapter 7 will introduce the chosen approach focusing on the classification of cloud phase. Chapter 8 will adapt the approach for a regression task. In this chapter we will estimate atmospheric temperature profiles.

7

CLOUD CLASSIFICATION

Contents

This chapter was the subject of a publication by Eulalie Boucher, Filipe Aires, and Marie Doutriaux-Boucher (2024). « Introducing a new partial Convolutional Neural Network for IASI cloud classification [Manuscript submitted for publication]. » In: *Submitted to IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Clouds are an important part of the climate system because of their impact on the Earth's energy balance. They always cover approximately 60% of the globe (Rossow and Schiffer, 1999). Clouds can however be an obstacle for the use of satellite data (Hartmann, Ockert-Bell, and Michelsen, 1992), especially for visible and infrared observations, impeding their use for surface applications. Their detection and classification is then essential for the analysis of remote sensing data and the validation of climate models.

IASI provides crucial observations to initialize NWP models through data assimilation techniques. In clear-sky conditions, it provides vertical profiles of the atmospheric temperature and humidity with good accuracy. The Metop data (IASI plus the other instruments) account for 40% of the impact of space-based observations in NWP forecasts (Eyre et al., 2022; Hilton et al., 2009).

Detection of clouds generally relies on visible and infrared satellite observations as this part of the electromagnetic spectrum is very sensitive to their presence (Mahajan and Fataniya, 2020). Such measurements are often available on geostationary meteorological satellites that offer, both spatially and temporally, continuous measurements at high resolution. This is the case of Meteosat's SEVIRI that fully observes the diurnal cycle (Seethala et al., 2018) of clouds or surface variables, for which it has proven to be very efficient (Derrien and Le Gléau, 2005; Sobrino and Romaguera, 2004). In particular, the Optimal Cloud Analysis (OCA) SEVIRI-derived Climate Data Record (CDR) (EUMETSAT, 2022b) provides a reference quality cloud phase classification, capable to detect multi-layered clouds.

However, SEVIRI products are only available in its 0° geostationary orbit centred over Europe and Africa. In an attempt to maximize the exploitation of IASI data, it is crucial to develop an accurate cloud mask and classification dedicated to IASI. Cloud property retrievals exist for the IASI instrument (Stubenrauch et al., 2017), but often use classical approaches that, even when based on neural networks, are usually based on a pixel-scale retrieval. Using the spatial coherency within the IASI observations could be beneficial for the detection of spatially structured variables such as clouds. However, as it was shown in Part I, the utilization of modern CNNs on IASI images is not straightforward, and the following points need careful consideration:

1. It is necessary to represent the measurements in a way that facilitates the use of spatial patterns, and
2. a method needs to be found to deal with missing data that are recurrent in any spatial representation of IASI orbits.

The first objective of this chapter is therefore to develop a cloud classification from IASI measurements, at the global scale; based on the SEVIRI Optimal Cloud Analysis (OCA) product. In this way, as shown in Aires et al. (2011) and Favrichon et al. (2019), it is possible to take advantage of SEVIRI's capability to correctly classify cloud types to train a global cloud classification.

The second objective is to develop and test a modern image-processing AI-based approach to exploit the spatial coherency of clouds like it has already been done in Kazantzidis et al. (2012), Shao et al. (2019), and Xie et al. (2017), but specifically for IASI data, on which applying image-based processing is complex.

The datasets are described in Section 7.1 and the methodology is introduced in Section 7.2. Section 7.3 presents the results, emphasizing on the cloud phase classification product, but also deriving a Cloud Fraction (CF) product. Evaluation is conducted using the AVHRR CF product.

7.1 MATERIAL

7.1.1 *IASI Data and Spatial Representation*

As stated in Section 2.1.6, EUMETSAT has recently released the IASI Principal Component Scores (PCS) Fundamental Data Record for Metop-A and -B (EUMETSAT, 2022a). Only Metop-B measurements are used in this study, but the methodology proposed here could be directly extended to Metop-A and -C. Although the full spectrum can be reconstructed using the eigenvectors, we choose to directly use the PCS, which is common practice in retrieval algorithms (Aires et al., 2002a,b; EUMETSAT, 2022; Huang and Antonelli, 2001; Jolliffe, 1986). Only PCS for IASI Band 1 (between 645–1210 cm^{-1}) are selected, because clouds are mostly detected in these window channels. Although Bands 2 and 3 potentially provide additional information, no significance improvement was seen in results, whilst much more computational power was required. The number of PCS that are kept in the retrieval depends on the application. An example of the first PCS is shown in Figure 46(a).

We will here present the final work frame that was selected in the conclusion of Part I; The IASI instrument acquires its measurements with 30 cells (or Field of Views (FOV)) across each scan line. Each FOV is analysed simultaneously using a 2 $\times$ 2 pixel grid. These pixels are then "flattened" into the data files, and each orbit is represented as an array of size: $n_{scanlines} \times$ 120, where $n_{scanlines}$ is the number of scan lines in an orbit. IASI-based retrievals (whether for atmospheric or surface properties) are generally performed at the pixel scale, using single point measurements, and therefore do not take any spatial structure into account, so the IASI distribution format is acceptable for most applications. However, we intend to develop an image-based retrieval scheme, so we choose to restructure the orbit data by restoring the original geometrical arrangement. Orbits can therefore be represented as a rectangular grid of size: $2 \times n_{scanlines} \times 60$. The IASI images are processed separately for the ascending and descending parts of each orbit. Every orbit becomes two images that can be treated as such. We will refer to them as the ascending orbit and descending orbit from now on, and $n_{scanlines}$ now refers to the number of scanlines in the ascending/descending part. This transformation is illustrated in Figure 47. Ascending and descending orbits are split to create two distinct images because their acquisition conditions differ (therefore an orbit no longer has an average of 765 scanlines but half of that). The final images are of a fixed shape of 850 $\times$ 60. The size $850=2\times425=2\times n_{scanlines}$ corresponds to twice the maximum number of scanlines (425) (because one scanline is extended in two lines since each FOV is 2×2) in an ascending or descending orbit. The orbits with fewer scanlines are padded with missing values. This fixed size of the images allows for an easy use of image-processing techniques in the following.

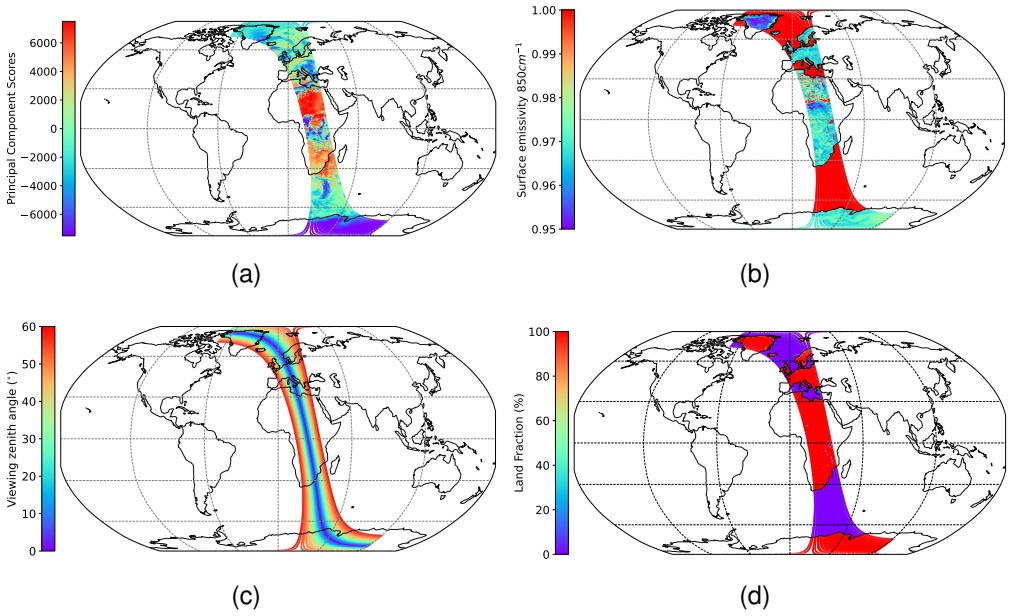
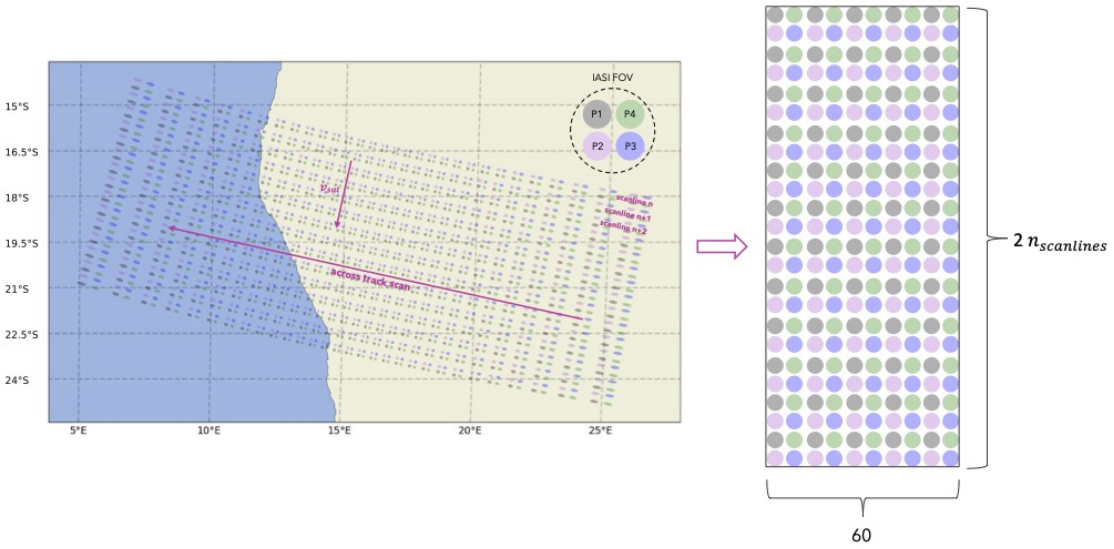


Figure 46 – Example of an ascending IASI orbit on 2016/05/01, 19:30 UTC. (a) First Principal Component Score for Band 1, (b) surface emissivity for the wavelength 850 cm^{-1} from the TELSEI database, (c) satellite viewing zenith angle, and (d) land-sea mask.

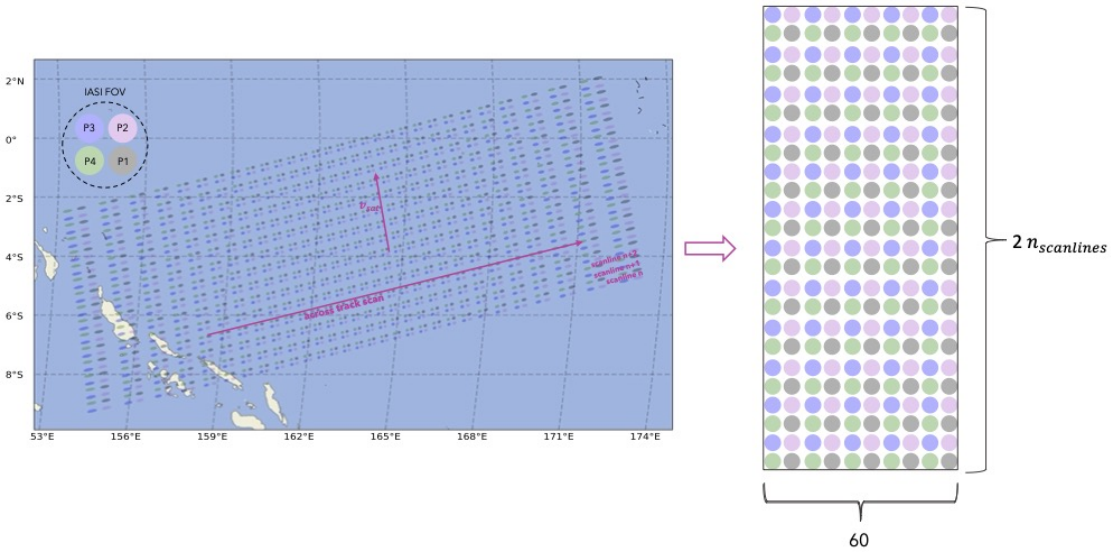
7.1.2 *SEVIRI Data and IASI/SEVIRI Collocation*

The SEVIRI OCA CDR provides a homogeneous time series of cloud properties with a 15-minute temporal resolution between 2004 and 2019 over the SEVIRI Meteosat disk centred at 0°E (EUMETSAT, 2022b). An example is shown in Figure 48. The OCA algorithm was first developed by the Rutherford Appleton Laboratory in 1997, and was further developed by EUMETSAT, allowing for multi-layered cloud situation retrievals (Watts, Bennartz, and Fell, 2011). The algorithm relies on reflectances from SEVIRI visible images (in daytime only); radiances from SEVIRI's infrared channels; ERA-Interim forecasts; a cloud mask at the pixel resolution; a daily clear-sky reflectance map; and an emissivity map (from MODIS), to retrieve cloud properties over the Meteosat disk. Variables retrieved in the CDR include the cloud top pressure, cloud optical thickness, cloud effective radius, as well as a four-class (i. e. clear, water, ice, two-layer (2L)) cloud phase classification, for each pixel of the SEVIRI image. This latter classification variable will be central to our study. The OCA product refers to cloud layers from top to bottom (i. e. , the first layer is the highest layer) and by definition, the second layer (i. e. , the lower layer), if any, is less visible from the satellite. The 2L classification corresponds to ice on the top layer and an undetermined cloud phase in the lower layer.

Although often considered a simple task, collocation is a central step in this study to which we have paid special attention. Approximate collocation schemes, based for instance on nearest neighbours, can be satisfactory in some applications, but



(a) Descending orbit



(b) Ascending orbit

Figure 47 – Example of the acquisition geometry of IASI pixels and their transformation into an image for a descending (a) and an ascending (b) orbits.

may prove insufficient for our application. Indeed, our objective is quite ambitious, as it is intended here to estimate the heterogeneity of cloud phases within each IASI pixel. A rigorous collocation (of the multiple SEVIRI pixels in each IASI pixel) is therefore a crucial step of the experiment presented here.

We perform spatial and temporal collocation between IASI and SEVIRI FOVs for every IASI ascending and descending orbits that partly or fully cover the Meteosat disk. This leads to collocation for circa 24,000 orbit files (half of which are ascending, and half descending) over the 2014 to 2019 time span.

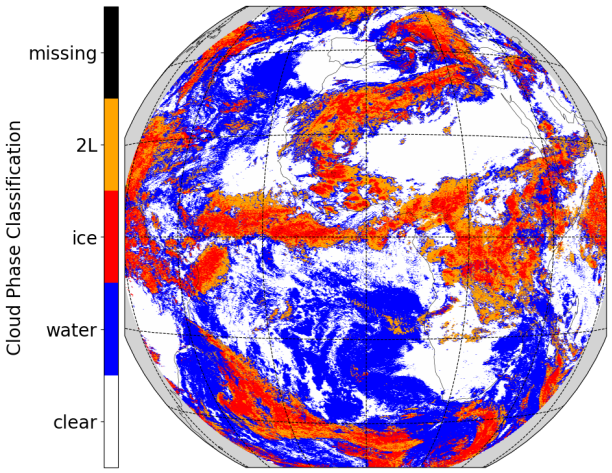
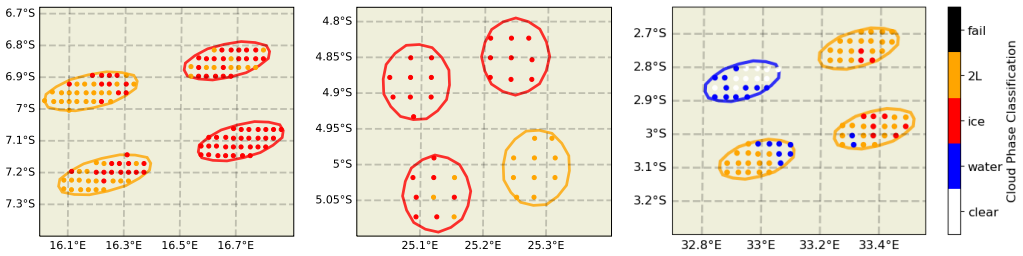


Figure 48 – Example of the SEVIRI Optimal Cloud Analysis four-class (clear in white, water in blue, ice in red, two-layer ice in orange) cloud phase classification on 2016/05/01 at 19:30 UTC

Temporal collocation is done by extracting the measurement time of the IASI pixel in the image falling closest to the Meteosat disk centre. We then choose the SEVIRI image that is temporally closest to this time. Temporal collocation is not performed independently for each pixel because it is required in the following to use images with continuous spatial patterns.



(a) FOV 1 (b) FOV 15 (c) FOV 30

Figure 49 – Spatial collocation of IASI and SEVIRI data (a) FOV 1, (b) FOV 15 and (c) FOV 30 along one IASI scanline. Each IASI FOV is composed of its 4 IASI pixels. We show the footprints of each pixel, and inside, the SEVIRI points. The ellipse edge colour of the IASI footprints corresponds to the class assigned to the IASI pixel based on the dominant class of SEVIRI pixels inside the footprint.

For spatial collocation, the footprint of each IASI pixel is found by calculating the ellipse equation using the latitude, longitude and satellite azimuth and zenith angles. Then, the cloud phase class assigned to the IASI pixel is the most common class amongst all SEVIRI points falling in the footprint. For IASI pixels of the orbit that do not fall over the SEVIRI disk, the cloud class is marked as missing. Figure 49

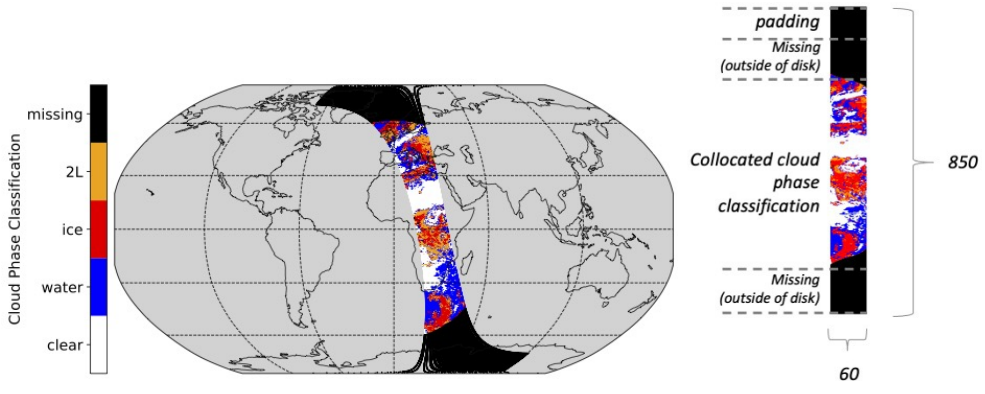
	$> 90\%$	$< 90\%$
Clear	0.79	0.21
Water	0.59	0.41
Ice	0.49	0.51
2L	0.40	0.60

Table 11 – Fraction of IASI pixels of each class that include more (left column) or less (right column) than 90% of SEVIRI points of that same dominant class.

shows the ellipses for each of the four IASI pixels in different FOVs along a scanline, and the SEVIRI points falling in the footprint. A IASI pixel is around 12 km at nadir, but around 40 km on the scanline edges (long-axis), and thus more SEVIRI points fall in each IASI footprint on the IASI scan edges.

Figure 50 shows (a) the projection of one ascending orbit and (b) the final image formatting. Missing points are marked in black.

The fraction of SEVIRI points belonging to each class for each IASI pixel is saved and will be used in Section 7.3.2. Table 11 represents the fraction of the IASI pixels of each class that have an internal dominant fraction higher or lower than 90%. This diagnostic indicates the degree to which each class is ambiguous inside the IASI pixels. Pixels classified as ice or 2L are less spatially uniform than those classified as clear or water, as only 49%(40%) of IASI pixels of these two classes are covered by more than 90% of ice(2L) SEVIRI points. This means that IASI retrievals over this kind of clouds will be less reliable, solely because of spatial heterogeneity. This shows how important it could be to have information of intra-variability in each IASI pixel.



(a) Projection onto map (b) Image formatting

Figure 50 – Example of the collocation between IASI ascending orbit and the SEVIRI cloud phase classification on 2016/05/01 at 19:30 UTC. (a) shows the projection of the collocated orbit and (b) describes the image formatting.

7.1.3 *Auxiliary Information*

TELSEI Surface Emissivities

A climatology of surface emissivities derived from the TELSEI emissivity interpolator (Paul et al., 2012) is used here. The database provides monthly global infrared emissivities at the IASI spectral resolution. The surface emissivities over three channels (850, 900 and 1100 cm^{-1}) are selected to discriminate surface type. Over the ocean, surface emissivities are fixed to 1. The same collocation procedure, presented in Section 7.1.2, is used to regrid the data from a 0.5° regular grid to the IASI FOV resolution. An example of the surface emissivities at wavelength 850 cm^{-1} is shown in Figure 46(b).

ERA5 Snow and Sea Ice Cover

To determine snow-covered surfaces that may have an impact on the classification of the cloud phase, the snow depth parameter from the ECMWF atmospheric reanalysis ERA5 is used (Hersbach et al., 2020). It is expressed in meters of water equivalent over a grid box (0.5°). It can partly or fully cover the grid box. Similarly, to determine sea-ice-covered surfaces, the sea ice concentration parameter which is the fraction of a grid box which is covered by sea ice (defined as frozen seawater which floats on the surface of the ocean) is used. We extract only one measurement per day (at 10 UTC to be close to Metop overpass). The same collocation procedure, presented in Section 7.1.2, is used to regrid the data to the IASI FOV resolution. A IASI pixel is marked as sea-ice or snow-covered if either the snow depth or the sea ice concentration is larger than 0.

AVHRR Cloud Fraction

Alongside the IASI L1c measurements, the EUMETSAT dataset provides a cloud fraction measure ranging between 0% (clear) and 100% (fully cloudy). The cloud mask comes from the L1B AVHRR integrated within the IASI FOVs. This cloud fraction will be used for evaluation purposes.

7.1.4 *Building the NN Database*

To feed the cloud classifier, observations and variables related to the cloud phase are considered. After some significance tests for the SEVIRI cloud-phase classification, the first 20 IASI PCs relative to Band 1 (the first of which is shown in Figure 46(a)); the three surface emissivities extracted from TELSEI at channels 850, 900 and 1100 cm^{-1} (shown in Figure 46(b)); the satellite viewing angle (shown in Figure 46(c)); and the land fraction (shown in Figure 46(d) were selected. This results in input matrices (or images) of size $850 \times 60 \times 25$.

Our objective is to perform multi-class classification, where each pixel in an image must be classified into one of $C = 4$ cloud phase categories: clear (0), water (1),

ice (2), or 2L (3). To train the network, a target output is created using the one-hot encoding. This leads to a vector that has one unit class and three null classes. Therefore, the target image is a 3D-tensor of size $850 \times 60 \times 4$, where the third dimension $c_{i,j,k}$ represents each pixel (i,j)'s membership to class k. Specifically, $c_{i,j,k}$=1 only if pixel (i,j) belongs to class k, and 0 otherwise.

With the image-processing scheme chosen here, missing values are inevitable. Four main sources of missing values arise:

1. The target data (i. e. , the SEVIRI-based OCA cloud phase) is only available over the Meteosat disk centred at 0°E, meaning that all IASI orbit measurement points falling outside this disk have to be flagged as missing.
2. The IASI orbits, that are now considered a rectangular image, are not always made up of the same number of scan lines. As a result, the images have been padded to ensure a fixed size, resulting in some missing data for some orbits.
3. It can happen that either the IASI or SEVIRI instruments encounter some sort of problem that results in actual missing data.
4. Lastly, due to the inherent confusion between sea-ice or snow-covered surfaces and clouds (especially ice clouds), sea-ice or snow-covered pixels are also flagged as missing. In fact, sea-ice/snow covered surfaces and clouds have similar spectral signatures (Cimini et al., 2006; Stubenrauch et al., 1999). We tested (not shown) this statement by performing a small experiment to retrieve the AVHRR cloud fraction from IASI measurements over sea-ice and snow. Our results were inconclusive, which suggest that the IASI PCS measurements do not distinguish sea-ice/snow and cloud cases. This means that we will not attempt, in the following, to perform any cloud classification over snow/ice covered regions. For such regions, we suggest using the AVHRR CF in the IASI Level 1C product, to confirm only the presence of clouds, not classifying the cloud phase.

Combining all of these sources of missing data, on average, 54% of an orbit is missing when building the training database, and 16% during the actual processing: In the training mode, all pixels that do not fall inside the Meteosat disk are marked as missing. While in the processing mode, only the pixels that are missing for reasons (2), (3) and (4) will be marked as missing, and a prediction is made for all other pixels, whether over the Meteosat disk or not. This must be kept in mind as it creates challenges both in the training and the evaluation procedures.

7.2 A CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE FOR CLASSIFICATION

Moving away from the classical pixel-wise retrievals requires using Machine Learning models that can exploit the neighbouring pixels' information. CNNs are considered here as they are one of the main classes of image-processing methods,

capable of resolving complex tasks ranging from 3D medical imaging segmentation (Singh et al., 2020) to facial emotion detection (Jaiswal, Raju, and Deb, 2020).

7.2.1 *Partial-CNN Architecture*

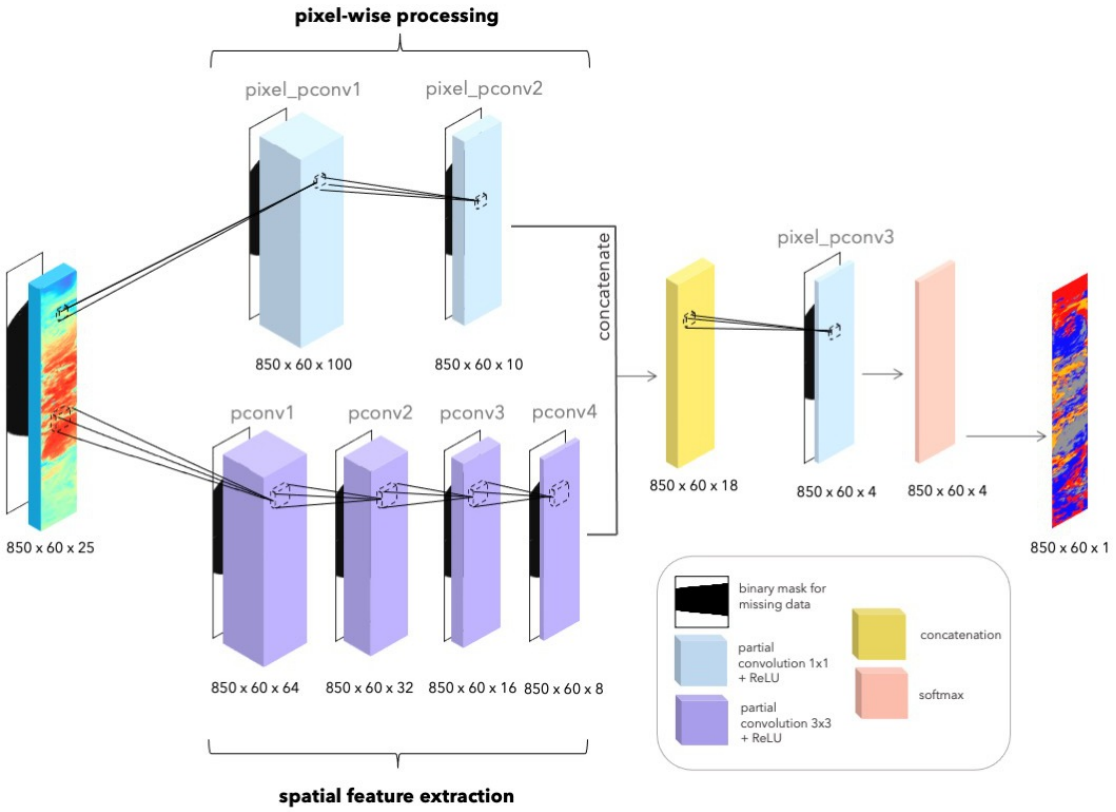


Figure 51 – Chosen architecture for the Partial Convolutional Neural Network.

Figure 51 illustrates the specific architecture used for this study. As any CNN, this model is a succession of convolutional layers. One of the limitations of traditional convolutional layers, is that they do not tolerate missing values in the input images, both in training and inference mode. Common practices consist in replacing missing values by a constant value (zero or the average value for example), or adding a pre-processing step to interpolate missing data (Boucher, Aires, and Pellet, 2023). However, this is only possible when there are a very few missing values in the training database. If too many constant values are inserted into the database, this defeats the purpose of using neighbouring pixels since the CNN will learn not to trust a pixel's surrounding.

For this reason, we use instead "partial convolution layers". Such layers were first introduced in Harley, Derpanis, and Kokkinos (2017) and Liu et al. (2018), but they are exploited here for the first time as a tool to deal with missing data present in

the input images. We expect this to become a very general and important tool for the satellite remote sensing community. A more detailed explanation was given in Section 3.4.4, but we recall the basis in the following paragraph:

The partial convolution layer operates in just the same way as a classic convolution, except the convolution operation is performed only on valid pixels and is normalized by the number of valid pixels. This is made possible by adding a binary mask m to the layers' inputs, where $m_{i,j} = \begin{cases} 0 & \text{if missing} \\ 1 & \text{otherwise} \end{cases}$, and transforming the traditional convolution operation (25) into (26):

$$y_{i,j} = \sum_{k=0}^{L-1} \sum_{l=0}^{L-1} x_{i+k,j+l} w_{k,l} + b, \tag{25}$$

$$y_{i,j} = \frac{1}{\sum_{k=0}^{L-1} \sum_{l=0}^{L-1} m_{i+k,j+l}} \sum_{k=0}^{L-1} \sum_{l=0}^{L-1} (x_{i+k,j+l} \odot m_{i+k,j+l}) w_{k,l} + b, \tag{26}$$

where $y_{i,j}$ is the activation map (the output of the convolution layer) at position (i,j), $x_{i,j}$ is the input signal at position (i,j), w is the weight of the filter with size $L \times L$, b is the bias term, m is the binary mask for missing data, and $\odot$ denotes element-wise multiplication. The activation maps of one layer are then passed on to the next layer of the network. After each layer, it is common to apply an activation function. Here the ReLU is chosen.

We use a succession of partial convolutions and ReLU layers that operate in two distinct parts: one that exploits spatial features (p_convs in Figure 51) from 3×3 filters; and one working at the pixel scale (pixel_pconvs), using 1×1 filters. The choice of the size of the filters (i. e. 3×3), should be highly dependent on the application. For instance, IASI pixels range between 12km and 40km diameter, it is thus unnecessary to use 7×7 filters because this would imply looking at 100km wide spatial patterns. The fusion of these two branches enables the model to make use of spatial features found within the image, whilst limiting the smoothing effect this can have by exploiting pixel-wise processing.

In the final layer of the network, the softmax activation function is applied to the scores output by the previous layers of the network x. For a given pixel x, and a given class c, we obtain $\text{softmax}(x_c) = \frac{e^{x_c}}{\sum_k e^{x_k}}$. This computation depends on the scores obtained for the other classes k so that the softmax outputs have a sum equal to 1, and can therefore be interpreted as the probability of a pixel belonging to each class.

7.2.2 *Training Approach*

During training, the categorical cross-entropy loss function $L_{CE} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$ is minimized, where y is the target image (i. e. the class of each pixel) and $\hat{y}$

is the image of predicted probabilities. The loss is computed only on available pixels, therefore missing pixels have no impact on the update of the network weights during back-propagation. The Adam (Kingma and Ba, 2014) algorithm is used. The networks are initialized to be trained over 100 epochs, however, early stopping (on the validation set's loss) is used to ensure no overfitting occurs. This limits the training to approximately 40 epochs.

Two distinct CNNs, but with the same architecture, are created and trained for the ascending and descending orbits. 2014–2018 data is used for training (of which 25% is reserved for validation), whilst 2019 data is used for testing. This results in databases of size 5730 (5651) for training, 1909 (1884) for validation, and 1254 (1222) for testing for ascending (descending) orbits respectively. These databases only include orbits that cover partly cover Meteosat disk.

7.3 CLOUD CLASSIFICATION AND CLOUD FRACTION RESULTS

We present here the results of the cloud phase classification using the partial CNN model presented in the previous section. Full evaluation is done on the testing set (January to August 2019). This data was not used during training. However, since the target classification is only available on the Meteosat disk, but the presented method allows for a global classification, this section is split into two parts: the results and evaluation over the Meteosat disk (using the OCA product), and its extension over the IASI global coverage. Results for ascending/descending orbits are assessed independently. As mentioned in Section 7.2.1, the partial CNN model outputs the probability of a pixel belonging to each class. We therefore output two products: the cloud phase classification, in which the predicted class is chosen as the one with the highest probability; and a CF based on the network probabilities, representing the % of the IASI pixel covered by clouds.

7.3.1 *Cloud Phase Classification over the Meteosat Disk*

Evaluation over the Meteosat disk can be done using the target OCA cloud phase classification, and direct comparison is possible. Overall, the model accuracy scores are presented in Table 12. We distinguish results over all points; homogeneous points (i. e. IASI points that are covered by more than 90% of the same SEVIRI dominant class); land; ocean; points at nadir (i. e. , where the viewing angle is close to 0°); and lastly, points on the scanline edges (i. e. , where the viewing angle > 45°). The rate of correctly classified points is of 76% (resp. 77%) for ascending (resp. descending) orbits. It can be observed that results are consistent between points at nadir (i. e. , centred on the scanline) and on the scanline edges, meaning that the incidence angle does not perturb the retrieval. Experiences showed that when the network did not have access to the information of the viewing zenith angle, a larger discrepancy could be observed between the two cases. Similar behaviour can be seen over land, ocean and coastal points. Again, this is not unexpected, since the

	All	Homogeneous	Land	Ocean	Coast	nadir	Scanline edges
Ascending	0.77	0.88	0.79	0.76	0.75	0.77	0.77
Descending	0.78	0.88	0.82	0.76	0.78	0.78	0.78

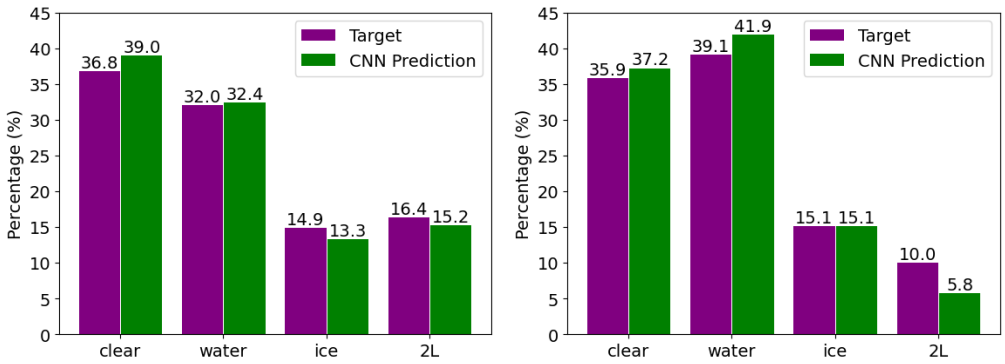
Table 12 – Model accuracy scores for ascending and descending orbits. Scores are differentiated over point subsets.

contrast between atmospheric and surface contributions are degraded over such areas. Adding such auxiliary information (i. e. , viewing angle and land fraction) into the networks is acceptable, as long as these variables have a known physical impact on the radiative transfer (Boucher, Aires, and Pellet, 2023).

Table 12 shows a net improvement when only considering homogeneous IASI pixels (i. e. , very predominant SEVIRI cloud type in the IASI pixel), as the percentage of well-classified points is increased from 77% (resp. 78%) to 88% (resp. 88%) for ascending (resp. descending) orbits. Homogeneous points have a less ambiguous spectral signature that helps to distinguish cloud types. This means that part of the classification errors is not related to the retrieval method (architecture, learning database, or learning algorithm) but rather on the ambiguity of the IASI observations themselves, and more specifically the intra-variability inside each IASI pixel.

Figure 52 shows the proportions of each cloud phase class in the testing set in purple, and CNN prediction in green, for ascending and descending orbits. Comparing the two distributions for ascending orbits, it can be seen that the model reproduces well the proportions of each cloud phase class. The CNN model has the slight tendency to overestimate clear and liquid water clouds, whilst omitting a small amount of ice and 2L clouds. This is also the case for descending orbits, but the model omits a little bit more (about 4%) of the 2L clouds.

Looking at the confusion matrices of the models helps to understand the model errors. A confusion matrix is a $m \times m$ matrix (where $m = 4$ is the number of classes) that assesses the classification performance of a model over all classes, by examining its error distribution. It provides the fraction of correctly classified points, and the distribution of the misclassified ones in the other classes. The top row of Figure 53 shows the normalized confusion matrix for ascending (left) and descending (right) orbits. The sum along each row is equal to 1 and therefore the entry in row A/column B can be interpreted as the percentage of pixels of class A predicted as class B. The diagonal entries therefore show the (normalized) fraction of pixels correctly classified by the model. The model distinguishes the cloud phase classes well, especially the clear and liquid water clouds. 90% of clear pixels and 75% (resp. 90% and 83%) of liquid water clouds are correctly classified for ascending (resp. descending) orbits. Correctly classifying ice clouds, and even more-so 2L clouds, is found to be more difficult for the CNN model; 36 (resp. 65)% of the time, a 2L cloud is mistaken for a liquid water or ice cloud in ascending (resp. descending) orbits.



(a) Ascending (b) Descending

Figure 52 – Proportions of cloud phase classes (clear, water, ice, 2L) in the testing set (purple) and inferred by the CNN (green), for ascending (a) and descending (b) orbits.

Although this misclassification rate can seem large, it is in fact not surprising. Two major factors can explain these errors.

1. *"The ambiguity within the 2L IASI pixels"*: As mentioned in Section 7.1.2, the ice and 2L classes are less spatially uniform. A IASI pixel classified as 2L by the collocation procedure can in fact be covered by a rather large fraction of SEVIRI points that are of another cloud phase class. This ambiguity can translate into a misclassification by the CNN because the IASI data are ambiguous. For example, it could simultaneously be seeing 51% of 2L ice clouds, and 49% of liquid water clouds over the same footprint; as a result, whilst the collocation database would assign the class 2L to this pixel, the CNN model might however output water, with a 50% confidence. This can also be seen in Table 12, in which the overall accuracy scores of the models are shown. In this table, we distinguish results over all points and over homogeneous points only. The model accuracy shoots up from 77% to 88% when looking at IASI pixels that are covered by more than 90% of SEVIRI points of the dominant class. We will further analyse this phenomenon in Section 7.3.2.
2. *"Missing information coming from the visible":* For daytime descending orbits, the OCA algorithm used to derive the target cloud phase classification uses SEVIRI's visible images. This means that the target cloud classification is of better quality during the day than for nighttime. Our CNN model uses only IASI infrared observations as input, which impacts the quality of the 2L cloud class. This problem is less of an issue for the nighttime because the target SEVIRI algorithm uses only infrared measurement. This explains the performance difference for the retrieval of ice and 2L clouds of the ascending (nighttime) and descending (daytime) orbits.

To highlight the advantage of using an image-processing approach for this task, another model was trained, this time omitting the spatial-feature extraction branch

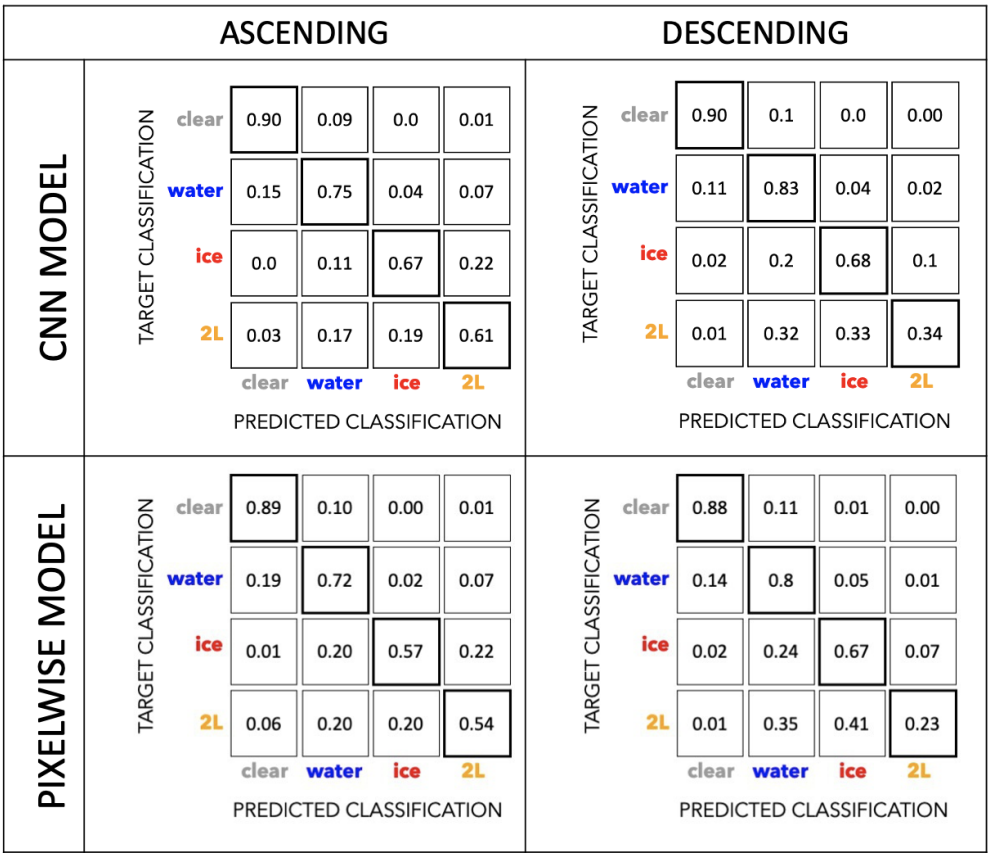
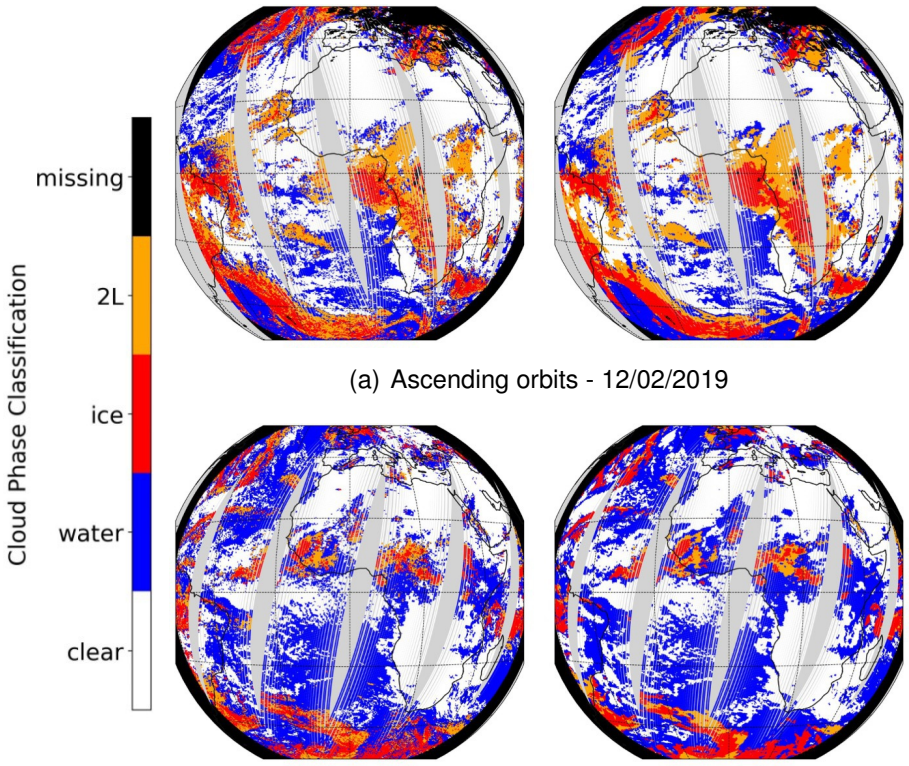


Figure 53 – Normalized confusion matrices for the testing set, using the CNN model (top) and the pixel-wise model (bottom), for ascending (left) and descending (right) orbits. The diagonal entries show the fraction of correctly classified pixels.

of the architecture presented in Figure 51. This new model is a traditional pixel-wise approach, more common in remote sensing applications. It was trained using the same datasets and training specificities. The confusion matrices obtained with this new network are presented in the bottom row of Figure 53. The comparison of the two models reveals the advantage of using an image-processing approach for such application, with a significant improvement of the classification results despite using the same input information. To our knowledge, it is the first time that IASI measurements are capable of recognizing multi-layered clouds. Using single point measurements for classification almost systematically degrades the performance of the model over all classes. This is especially true for ice and 2L clouds, for which performance is affected by 10% for descending orbits. This suggests that the use of spatial patterns is useful for the classification of cloud types. Indeed, clouds have strong spatial coherency and the use of neighbouring pixels is beneficial for their detection, crucial, even, for multi-layered clouds. However, the drawback of this is that the obtained products exhibit a higher level of smoothness, as demonstrated in Figure 54. This is a standard behaviour for image-based processing.

Figure 54 shows specific examples of cloud classification, for both ascending (top) and descending (bottom) orbits. Each plot shows the concatenation of all orbits



(a) Ascending orbits - 12/02/2019

(b) Descending orbits - 12/06/2019

Figure 54 – Example of target (left) and inferred (right) cloud phase classification for ascending orbits on 12/02/1019 (a) and descending orbits on 12/06/2019 (b). Black pixels indicate missing pixels outside the Meteosat disk, sea-ice/snow cover, or instrument failure. Grey areas are pixels between orbit passes.

falling over the Meteosat disk on 12/02/2019 for ascending orbits, and 12/06/2019 for descending orbits. Missing pixels are shown in black, and are either due to snow cover, instrument failure, or because they fall outside the disk. Grey colour indicates that no point was located over the area. There is an excellent agreement between the target and predicted cloud classification, both for ascending and descending orbits. Overall spatial patterns are almost identical between the target and predicted scenes. The CNN retrieval appears to be slightly smoother. This is natural as the CNN uses convolutional spatial filters that inevitably smooth out the output solution. This effect is reduced thanks to the pixel-wise branch of the CNN architecture (Section 7.2.1). Isolated pixels with a different class than its surroundings might be smoothed out because of the spatial filters. This is mostly the case for ice and 2L pixels.

7.3.2 *Intra-Pixel Cloud Phase Fractions*

As shown in the previous section, the majority of the errors made by the classifier are due to ambiguity arising from the heterogeneity of SEVIRI pixels inside each IASI pixel field-of-view. However, the network (thanks to the softmax activation func-

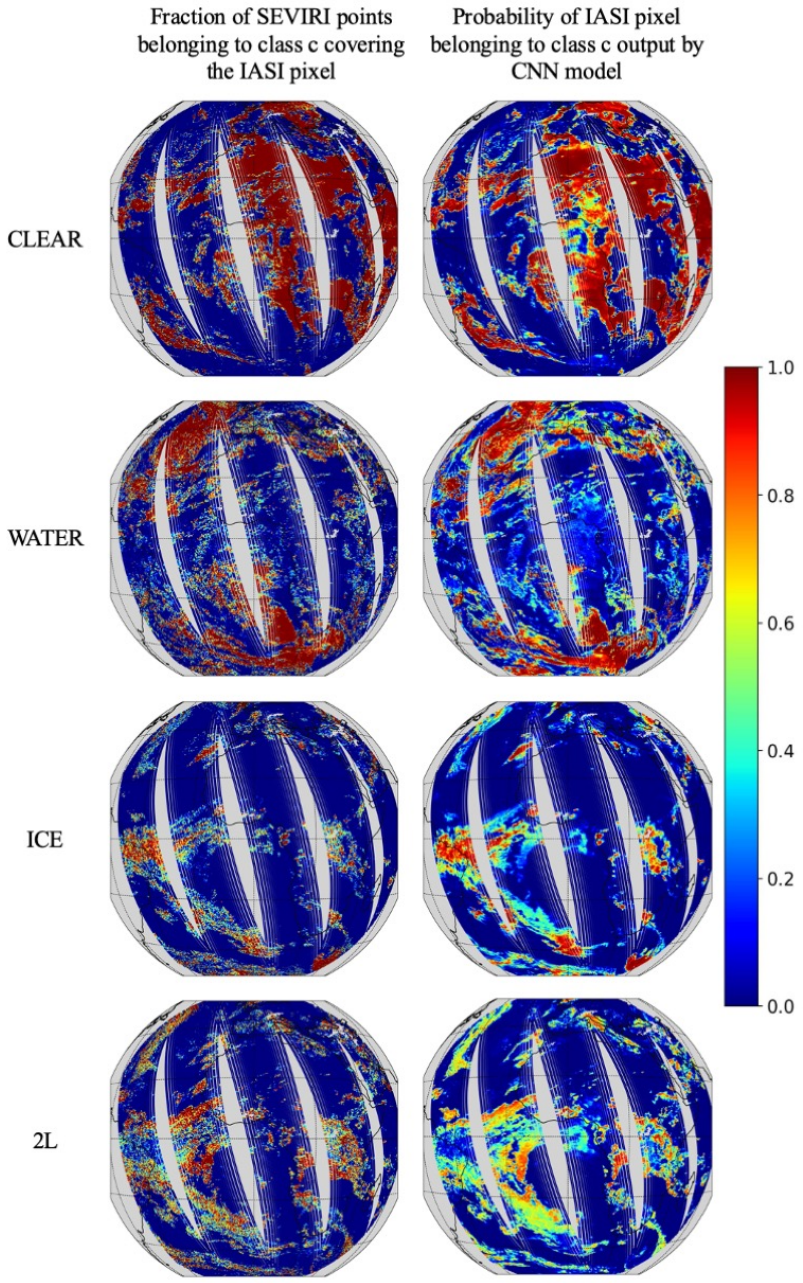


Figure 55 – Left: Cloud class fraction of SEVIRI points in the IASI pixels. Right: Probability of each IASI pixel belonging to each class according to the CNN model. The orbits shown are ascending orbits on 01/04/2019, only available pixels are shown.

tion) outputs the class (clear, water, ice, 2L) probability in each IASI pixel which are usually interpreted as the level of confidence of the classification, but it is possible to analyse them further.

In our case, this level of confidence is also the level of ambiguity within a IASI pixel. Indeed, for each class, we obtain remarkable correlations between the class

membership probability p_{class} given by the model, and the fraction of SEVIRI points that fall over the IASI pixel belonging to each class CF_{class}. This was not *a priori* to be expected. Correlations between the two are 0.88 for clear, 0.77 for water, 0.78 for ice and 0.72 for 2L clouds. These are values for ascending orbits, but the same results are obtained on descending orbits. Such correlations strongly indicate that the CNN model is capable of detecting ambiguities within the integrated IASI observations of one pixel. The class membership probabilities p_{class} can actually be used as a proxy for the *CF* of each class. The model is, to some extent, capable of downscaling the IASI data towards SEVIRI resolution, because it is able to provide an information at a higher spatial resolution than the IASI data.

Figure 55 compares the fraction CF_{class} of original SEVIRI points covering the IASI pixel belonging to each class (left) and the probability p_{class} of IASI pixels belonging to each class (right) for the ascending orbits of 01/04/2019. Although the agreement is not perfect, their strong correlation is clear.

This is a remarkable feature of the CNN method presented here, that goes beyond the initial cloud classification task. The network was trained without the intra-IASI-pixel heterogeneity information. Despite this, the network is able to detect ambiguous situations, and is even able to quantify which classes are mixed inside each IASI pixel. The next step towards downscaling would be to position, within each IASI pixel, the proportion of each distinct cloud class, creating a 3 km product (i. e. SEVIRI resolution). However, downscaling is obviously not an easy task and would require thorough thought, especially because interpolation between two neighboring IASI pixels will be required.

Since we have access to the fraction of each cloud class inside the IASI pixels, it would be possible to train the CNN directly with these outputs. For example, the target vector for a IASI pixel classified as water, but covered only by 70% of SEVIRI water pixels, could be $(0.1, 0.7, 0.1, 0.0)$ instead of $(0, 1, 0, 0)$. This approach was tested (not shown) but led to degraded model accuracy scores, from 77% (resp. 88% for homogeneous points) to 70% (resp. 83%) for ascending orbits. This is most likely because this approach makes the NN task more complex. This is a good illustration of how training with degraded data, such as binary classification, can still yield a NN model capable of surpassing these problems.

7.3.3 *Extension Over the Globe*

Despite being trained only over the Meteosat disk (where the target product is available), our NN can actually be used at the global scale. The binary mask accounting for missing data is set up only if the pixel is snow-covered or encountered an instrumental error. Figure 56(a) shows the target and predicted cloud phase classification, for ascending orbits of 14/04/2019. Black pixels are missing due to instrumental failure and (for the most part) sea-ice/snow cover. Over the Meteosat disk, where direct comparison with the target classification is possible, it can be seen that spatial patterns between the two products are almost identical. Large

clear areas over Northwest Africa are well retrieved, as well as more isolated water clouds. Similarly, the separation between ice (red) and 2L (orange) clouds are also well respected. Outside the Meteosat disk, no direct comparison is possible, but the figure shows clearly that there is no discontinuity around the Meteosat disk edges. Spatial patterns seem reasonable in terms of spatial and statistical features, even in the higher latitudes. For example, some Northern Europe pixels that are not covered by sea-ice or snow are classified as clear. This means the model is capable of correctly distinguish cloudy and clear cases outside the Meteosat disk.

In addition to the cloud phase classification product, we compute a cloud fraction (*CF*) for each IASI pixel, and for each class: CF_{clear}, CF_{water}, CF_{ice} and CF_{2L}. These *CF*s can either refer to the true fraction of the SEVIRI points covering the IASI points or be derived by the network probabilities p (that act as a proxy). We will denote the CF_{class} as p_{class} when the proxy is used, and CF_{class} when the true fraction is used.

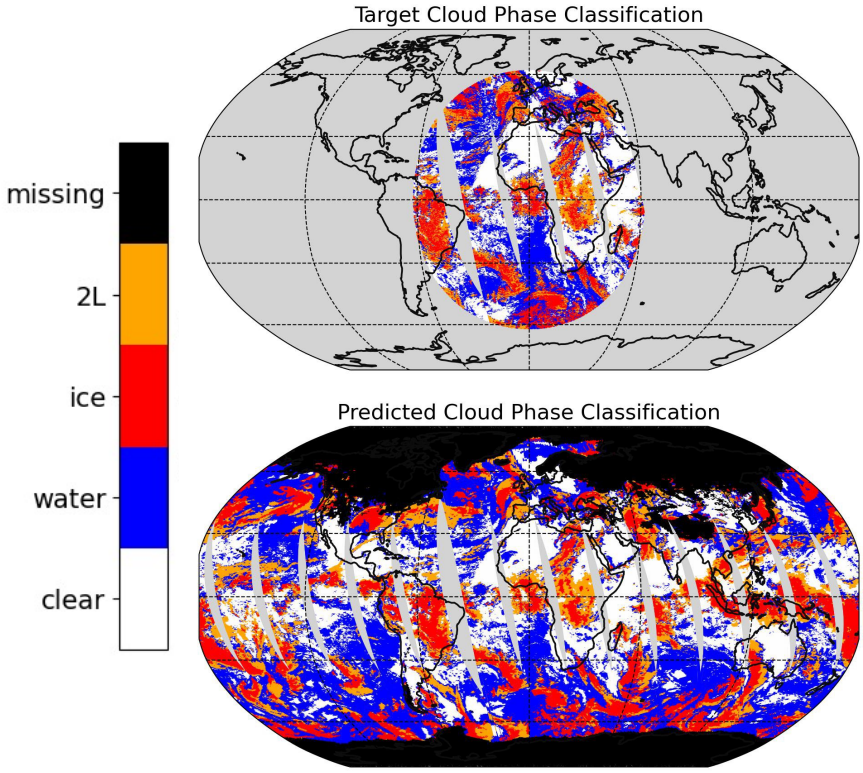
We define $CF_{CNN} = p_{water} + p_{ice} + p_{2L} = 1 - p_{clear}$. This is shown on the bottom row of Figure 56(b). Grey pixels here correspond to the missing pixels, where no *CF* is derived. The *CF* product is directly related to the probability of each IASI pixel being clear (i. e. the fraction of the IASI pixel that is clear) output by the network. Purple/blue pixels are the clearer pixels, whilst the red colour indicates a high *CF*. Not surprisingly, some pixels classified as cloudy do not have a 100% *CF* since they may be partly covered by clear sky. This is often the case on the cloud edges.

7.3.4 *Evaluation of the Global CF Product Using AVHRR*

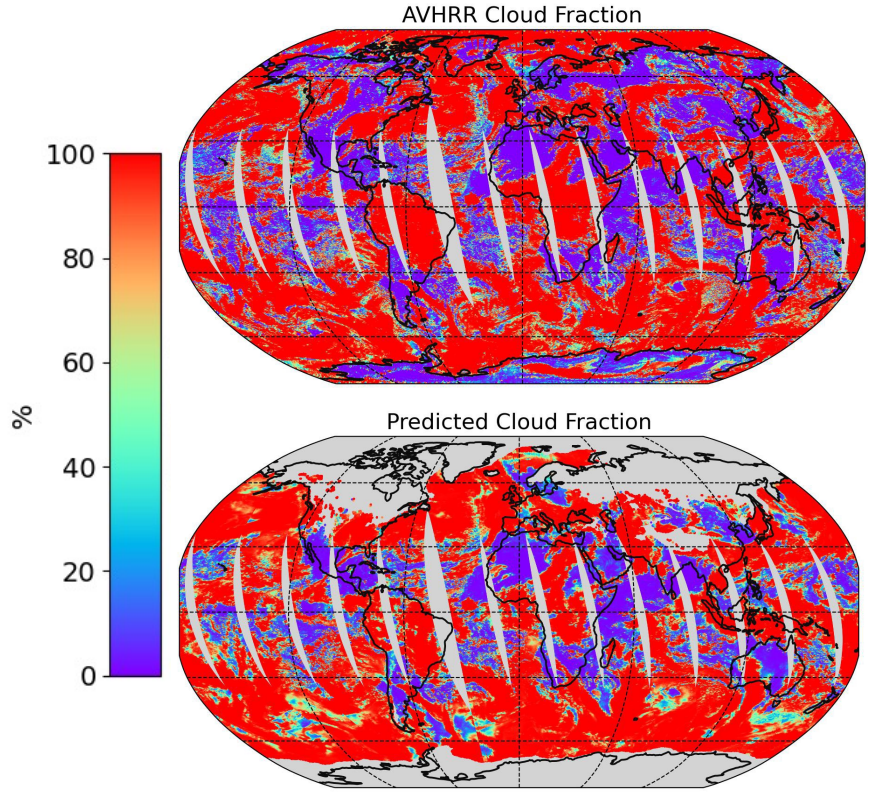
As commented in the previous section, the estimation of the *CF* in each IASI pixel for each cloud type, allows computing the CF_{CNN} for each IASI pixel. This cloud fraction can then be evaluated by comparison to the AVHRR CF_{AVHRR} (presented in Section 7.1.3). Although this does not evaluate the classifier itself, it allows for comparison between intra- and extra-Meteosat disk predictions.

We have three measures of the *CF*: (1) the $CF_{SEVIRI} = 1 - CF_{clear}$ calculated using the "true" fractions of SEVIRI points covering the IASI pixels. This measure is only available over the Meteosat disk. (2) The CF_{AVHRR} is the AVHRR product integrated over the IASI FOVs. This measure is available globally, at the IASI resolution, and is shown in Figure 56(b). (3) The CF_{CNN}, which is available globally in Figure 56.

Table 13 shows the inter-comparison of these three *CF* measures, over the testing set. Comparisons with CF_{SEVIRI} are only possible inside the Meteosat disk. We compute the Root Mean Squared (RMS) error $\sum(CF_1 - CF_2)^2$, that is decomposed as the STD error, and the bias error. These absolute errors are expressed in % and highlight the differences between the CFs. The first row of the table shows the "true" difference between SEVIRI and AVHRR *CF*s. This error serves as a guideline on what to expect between the CNN-derived and the AVHRR *CF*s. The RMSE is approximately 31% for ascending and descending models. The RMSE metric gives



(a) Cloud Phase Classification



(b) Cloud Fraction

Figure 56 – (a) Target and predicted cloud phase classification and (b) AVHRR and predicted Cloud Fraction, over the ascending orbits on 14/04/2019. Black pixels are missing due to instrumental failure or sea-ice/snow cover. Grey areas are missing data between orbits.

Comparison	Model	Inside Meteosat disk			Outside Meteosat disk		
		RMSE(%)	*STD(%)*	*BIAS(%)*	*RMSE(%)*	*STD(%)*	*BIAS(%)*
CF_{AVHRR}/CF_{SEVIRI}	Ascending	30.5	28.9	-9.8	n/a		
	Descending	32.4	29.9	-12.5	n/a		
CF_{AVHRR}/CF_{CNN}	Ascending	25.6	24.9	-8.7	26.6	25.4	-8.2
	Descending	27.3	24.3	-12.0	27.5	24.8	-12.0

Table 13 – Inter-comparison of *CF* measures (CF_{SEVIRI}, CF_{AVHRR} and CF_{CNN}) inside and outside the Meteosat disk, for ascending and descending models, over the testing set.

a high weight to large errors (as a comparison, the Mean Absolute Error between these two *CF*s is only 14%). The high RMSE comes from pixels marked as almost clear in one and fully cloudy in the other. Most of this error comes from the STD error (i. e. random error), as the bias error only accounts for about 10%. The bias error is negative, which means that on average, SEVIRI detects more clouds than AVHRR. The second row of the table compares the CNN-derived and the AVHRR *CF*s. Inside the Meteosat disk, we obtain an RMSE of 25.6% (resp. 27.3%) for the ascending (resp. descending) model. This error is again mostly due to random errors, and the bias error accounts for 8.9% (12%). Very similar errors are obtained outside the Meteosat disk, as the RMSE is 26.6% (27.5%). On average, the error is increased by less than 1% outside the Meteosat disk. This confirms the capability of the model to perform satisfactorily at the global scale. The bias error stays almost identical inside and outside the Meteosat disk, meaning that the model does not over- or under-estimate cloudy pixels differently inside or outside the disk.

Looking back at Figure 56(b) that shows the AVHRR CF_{AVHRR} (top) and the CNN-derived CF_{CNN} (bottom), it can be seen that the general spatial patterns of the *CF* are very well respected. CF_{CNN} is smoother than CF_{AVHRR}. This is a natural consequence of the use of convolutional spatial filters in the model. Most differences are noticed on the outskirts of cloudy areas (i. e. transitions between clear and cloudy) that are much sharper in the AVHRR *CF*. The two *CF*s are nonetheless very similar, inside and outside the Meteosat disk.

7.4 CONCLUSIONS

In this chapter, a CNN-based cloud phase classifier was developed from IASI observations. The SEVIRI's visible/infrared cloud classification is used to train the CNN. Our approach allows extending SEVIRI's cloud classification to the globe.

We introduced, to our knowledge for the first time in the atmospheric remote sensing community, partial convolutional layers into a CNN architecture. This is the result of the effort put into fully understanding how we should go about global processing, that was presented throughout Part I of this thesis. The partial convolutions

allow for the use of image processing techniques, despite the considerable amount of missing data in input images.

In the context of cloud phase classification, the fact that images are used instead of pixels, makes it possible to better distinguish multi-layered clouds based solely on IASI measurements. This is a first for IASI-based cloud detection. Clear, water, ice and 2L clouds are correctly classified with more than 75% accuracy. This number increases to 87% when considering only spatially homogeneous IASI pixels, showing that important differences are for a large part actually related to IASI/SEVIRI resolution discrepancy and thus large heterogeneity inside IASI pixels. As expected, most errors occur when discriminating 2L clouds/ice clouds, that are harder to differentiate for the model, especially during daytime. Indeed, during the daytime, the SEVIRI mask will be more accurate (because it is using the visible) and hence discrepancies with IASI may be larger since the infrared signal is not strong enough to perfectly infer the 2L clouds. However, thanks to the image processing approach and the cloud spatial coherency, these errors are reduced compared to a pixel-wise retrieval.

The availability of our IASI cloud product extends globally, and can be generated in near real time. Only a simple step is required to format the IASI orbit into a rectangular image, that can then be instantaneously fed into the partial CNN classifier.

The network output probabilities that are a good proxy of the percentage of the IASI pixel covered by each cloud type. Therefore, our product not only classifies the cloud phase but also estimates the cloud type fraction covering each IASI pixel, regardless of the cloud type. This is a first step towards the downscaling of IASI data towards SEVIRI resolution: Higher resolution pixels inside the IASI FOVs could then be classified. This also shows in a more general way that neural networks are capable of identifying the correct signal in a database that is not perfect (here, the binarization of the SEVIRI cloud type inside each IASI pixel).

Access to a cloud mask and cloud properties is primordial for the use of IASI data. Removing cloud contaminated scenes is a necessary pre-processing step for atmospheric and, more importantly, surface temperature retrievals. Our cloud mask will therefore be a preliminary step for land surface and atmospheric (temperature and humidity profiles) retrieval in line with the methods we developed in the first part of the thesis, and in Boucher and Aires (2023b) and Boucher, Aires, and Pellet (2023). More generally, the model presented in this chapter is the first to make use of partial convolutions that are capable of learning spatial features, despite the presence of missing data. The model can be easily modified to retrieve other geophysical variables such as surface or atmospheric temperatures, where missing data is inevitable due to the presence of clouds. This technique makes the use of image-processing much more accessible, and is in fact an essential tool for the treatment of IASI data under the form of images. This will be done in the next chapter.

8

GLOBAL TEMPERATURE PROFILE RETRIEVALS

Contents

Building upon the methodology established in the previous chapter, the focus now shifts towards the global retrieval of atmospheric temperature profiles.

The IASI instrument captures data in hyperspectral resolution, with which we are able to obtain temperature measurements across various layers of the atmosphere. Accurate monitoring of temperature variations is vital, not only for enhancing weather forecasting capabilities but also for comprehensively analysing and predicting future climate phenomena, particularly in the context of global warming. With two simultaneously flying instruments, the IASI observations therefore have huge potential to create long-term time series of atmospheric temperature profiles that perfectly complement the traditional *in situ* atmospheric temperature records (Bouillon et al., 2022; Safieddine et al., 2020). Indeed, such records are of excellent quality, but are spatially very sparse (not to mention unevenly distributed).

As for many other geophysical parameters, the retrieval of atmospheric temperature is mostly done statistically, but at the pixel level of IASI measurements (Aires and Pellet, 2021; Bouillon et al., 2022; Clerbaux et al., 2009; Pellet and Aires, 2018). As we have shown in the previous chapters of this thesis, looking beyond a single pixel may be of use, especially where spatial heterogeneity is large. This might be the case with atmospheric temperature in the lower layers of the atmosphere, as this is where many heterogeneous continental surfaces have the strongest input,

and where most weather phenomena occur (i. e. in the troposphere that extends on average 10 km above sea-level). This is due to the large amount of water vapour that is found in the troposphere (in which 99% of the atmosphere's water vapour is found). As we move upwards into the stratosphere, the atmosphere is stratified and there is less turbulence due to the ozone layer that absorbs radiation from the Sun. This chapter will try to capitalize on the spatial patterns this brings along to improve the quality of temperature profile retrievals at a global scale.

The work presented in this chapter is therefore the continuation of the previous chapter. The global retrieval of atmospheric temperature profiles is done with the same processing scheme as for the cloud classification; IASI orbits are considered a long strip that can be viewed as an image, on which traditional image-processing techniques can be applied.

The datasets are described in Section 8.1 and the model architecture is introduced in Section 8.2. Section 8.3 will present the results, comparing the final retrieval with the pixelwise approach, and evaluating the final results against *in situ* radiosonde measurements.

8.1 MATERIAL

8.1.1 *IASI Data*

The same data is used in this chapter as in the previous one. As a quick recap, we use the Metop-B EUMETSAT IASI PCs Fundamental Data Record presented in Section 2.1.6 between 2014 and 2021. Similarly, as for the cloud classification, we use only a subset of the PCs, however, both bands 1 and 2 are here retained.

The IASI orbits are split into ascending and descending categories, and each orbit is restructured as a rectangular strip of shape 850×60. The rationale behind this is explained in Section 7.1.1.

8.1.2 *ERA5 Atmospheric Temperature*

The hourly ERA5 atmospheric temperature profiles are used as target values for the training of the CNN, because it is the database that represents the best knowledge of the atmosphere. ERA5 provides hourly temperature over 37 pressure levels between 1000 and 1 hPa. The 37 pressure levels are defined as shown in Figure 57. The temperature profiles are available on a 0.25° grid, at an hourly time step. Figure 58(a) is an example of the temperature at 1000 hPa on February 5^{th} 2023 at 14:00 UTC on this grid. A temperature profile over the 37 levels extracted from the same map (located over the Himalayas) is plotted in Figure 57. Because the reanalysis data is interpolated from 137 model levels to the 37 fixed pressure levels, it gives values that can be below the surface. The data that is below the surface (lev-

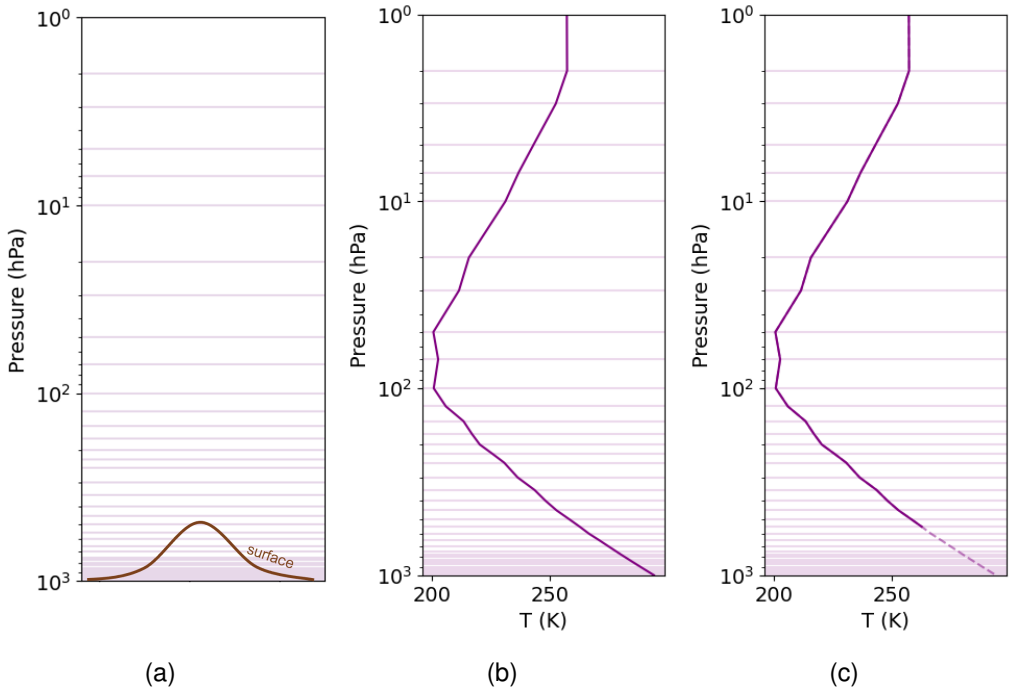


Figure 57 – The 37 pressure levels are shown in purple (in hPa). (a) shows the levels, (b) plots a temperature profile extracted from ERA5 over the Himalayas and (c) shows that values can be given below the surface due to interpolation, but should be masked out (in dotted lines).

els where the pressure is larger than the surface pressure, shown in Figure 58(b)) should/can be masked out.

We perform spatial and temporal collocation between the IASI observations and the ERA5 temperature profiles for every IASI orbit. Temporal collocation is done by selecting a IASI pixel in the centre of the orbit, and associating the ERA5 map at the closest hour. To ensure spatial continuity (which is important when neighbouring pixels are used for the retrieval), temporal collocation is not performed independently for each IASI pixel. For spatial collocation, the same procedure as performed in Section 7.1.2 is used; the footprint of each IASI pixel is calculated, and the mean ERA5 temperature is assigned. Spatial collocation can be seen in Figure 60, for an orbit on June 12^{th} 2019.

8.1.3 *Auxiliary Databases*

TELSEI

The same climatology of surface emissivities derived from the TELSEI emissivity interpolator (Paul et al., 2012) as in the previous chapter is used here. We select surface emissivities over three channels (850, 900 and 1100 cm^{-1}). Over the ocean, surface emissivities are fixed to 1. The same collocation procedure, pre-

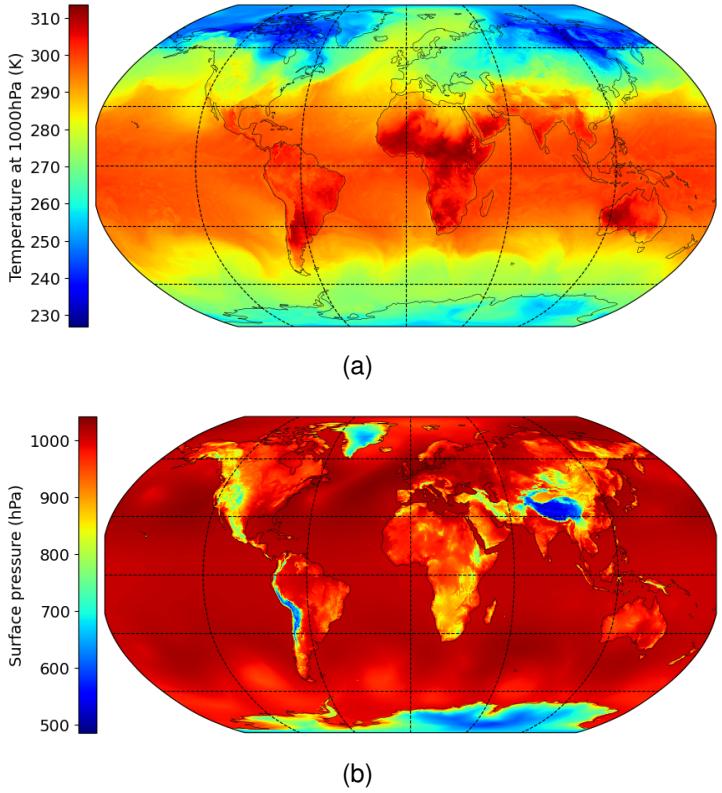


Figure 58 – ERA5 (a) temperature at 1000 hPa and (b) surface pressure on 05/02/2023 at 14:00 UTC.

sented in Section 7.1.2, is used to regrid the data from a 0.5° regular grid to the IASI FOV resolution (see Figure 46(b) for an example).

ARSA Radiosonde Measurements

Worldwide, there are over 800 active radiosonde stations that report twice per day at 00 and 12 UTC. The Analysed Radio Soundings Archive (ARSA) database, created by the Laboratoire de Météorologie Dynamique (LMD) is based on observations from those radiosonde stations all around the world. It is a database of reprocessed radiosonde measurements between 1979 and August 2019 with an extensive quality control. ARSA thus provides temperature profiles worldwide, on 42 atmospheric pressure levels. Figure 59 shows the number and location of recorded profiles.

EUMETSAT Level 2 Temperature Profiles

The EUMETSAT Level 2 IASI temperature profiles distributed by EUMETSAT (2022) (see Section 2.1.7 for more information) are also used for validation over the year of 2019. The PWLR3 algorithm is used to derive atmospheric temperature profiles from IASI observations, collocated with some microwave (AMSU and MHS) observations.

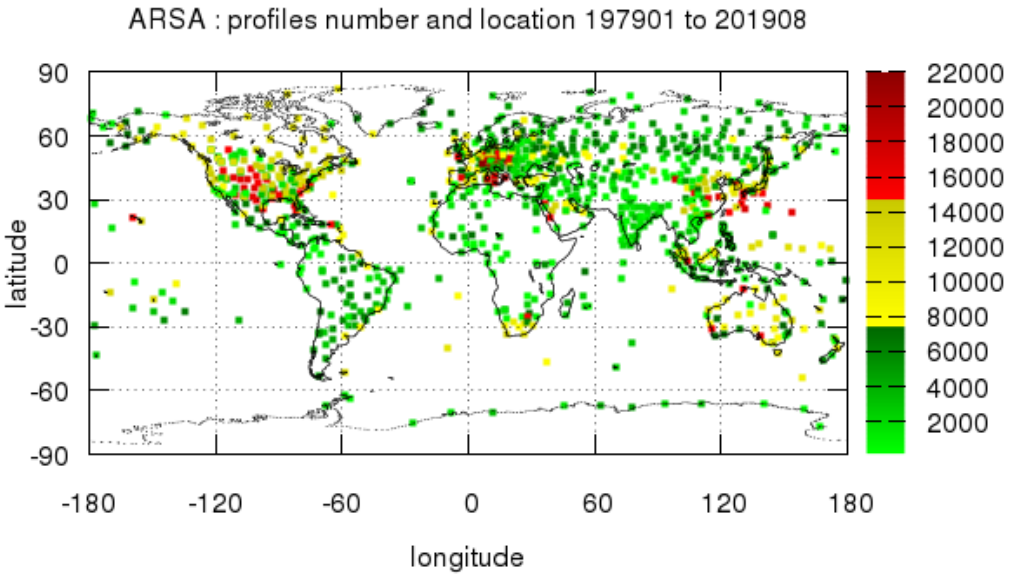


Figure 59 – Number of ARSA profiles and their location between January 1979 and August 2019. Source: https://ara.lmd.polytechnique.fr/index.php?page=arsa

8.1.4 *The Training Database*

The retrieval is performed independently over ascending and descending orbits (i. e. two networks are trained).

To perform the retrieval of the atmospheric temperatures over the 37 pressure levels, the neural network is fed a number of observations and variables that will help it to learn the statistical link there is between the observations and the atmospheric temperatures. To this effect, we select the first 10 IASI PCs relative to Band 1; the first 10 PCs relative to Band 2; the three surface emissivities extracted from TELSEI at channels 850, 900 and 1100 cm^{-1}; the surface pressure; the satellite viewing angle; and finally, the land fraction. This results in input images of shape $850 \times 60 \times 26$. The output is of shape $850 \times 60 \times 37$, corresponding to the temperature measurement at 37 levels.

Since infrared measurements are extremely sensitive to the presence of clouds, the retrieval of atmospheric temperature across all 37 pressure levels is done in clear-sky conditions only. This implies that there is a significant amount of missing data. Figure 60 shows a representative example of collocated ERA5 temperature at 100 hPa on a IASI orbit, in which missing pixels are shown in hashed areas. The cloud fraction product created in the previous chapter is used to mask out cloudy scenes. For pixels with snow/sea-ice cover, for which the latter was not derived, the AVHRR cloud fraction is used. Clear-sky conditions are defined with a threshold

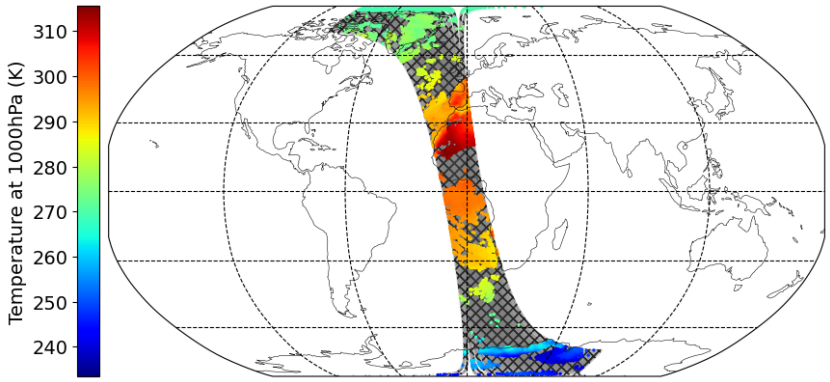


Figure 60 – Example of collocated ERA5 temperature at 1000 hPa on an ascending IASI orbit on 1st June 2019. Hashed areas represent cloudy pixels of the orbits.

of 5% (i. e. if less than 5% of the IASI pixel is covered by clouds, then the pixel is considered clear).

In addition to the missing data created by cloudy scenes, the interpolation that is done between the model pressure levels and the ERA5 pressure levels generates some “vertically” missing values in the target profiles, as shown in Figure 57(c). Indeed, for a pixel that is located on a surface where the surface pressure is smaller than 1000 hPa (i. e. the altitude is not null, on the figure we show a point over the Himalayas, that have a surface pressure of 500 hPa), some of the 37 temperature values have to be marked as missing. This must be taken into account when training the neural networks.

The collocation between the IASI input database and the ERA5 temperature profiles results in over 50 000 samples (one sample = one orbit = one image) over the 2014-2017 time period. However, only 10 000 are randomly selected for training; 5000 of which are ascending, and 5000 descending. For each model we use 3500 for training and 2000 for validation. 2018 and 2019 years are kept aside for evaluation, and the year 2019 is kept for validation against other databases.

8.2 THE NEURAL NETWORK RETRIEVAL APPROACH

In the upper atmosphere, there is less turbulence and temperature is therefore relatively smooth. This means that the IASI retrievals done at the pixel level (with a pixelwise MLP for instance) are already of good quality. Indeed, since the neighbouring pixels are in a similar state, there are no true spatial patterns that an image-processing technique can take advantage of. However, as it was shown at several occasions, in Aires, Boucher, and Pellet (2021), and also in Part I of this thesis, the use of neighbouring pixels has a positive impact on the retrievals of surface properties. The challenge in this chapter is therefore to develop a NN model that is able to perform just as well as an MLP in the upper atmosphere, and surpass it in the lower atmosphere and closer to the surface. This section aims to describe the method used to achieve this objective.

Although CNNs are most known for classification tasks, they are of course able to perform regression tasks, as it was done in Part I of this thesis. The last layer of the network and the chosen loss function are the parameters that control the output volume and type.

The model that we use in this chapter is based on the foundations of the model presented in Chapter 7, but is adapted for regression tasks. Figure 61 shows the detailed final architecture. It is composed of two main branches:

1. The first branch is exploiting spatial features (p_convs) using a succession of convolutional layers with filters of size 3×3, followed by ReLU activations.
2. The second branch uses only pixel scale information. This is done with convolutions with filters of size 1×1 (marked as the pixel_pconvs on the figure), also followed by ReLU activations.

As it will be shown in this chapter, the two branches each have their own advantages, and thus we fuse the two branches to create a model that is capable of detecting spatial features and also using the most of the pixelwise information. In Chapter 7, the training of the two branches was done simultaneously. Here, the two branches are independently pre-trained, and the final model is then trained by freezing the weights of the two branches, and training only the part of the network responsible for the weighting of the two outputs. The training of the three parts are detailed in the next subsections.

The reason behind training each part separately comes from the use of the MSE loss during training, that penalizes large errors. Since the surface and lower atmosphere will benefit from the use of neighbouring pixels the most, whilst the higher atmosphere is smoother and advantages will be less obvious, training both MLP and CNN branches together may have trouble to converge, as different parts of the temperature profile will benefit from different combinations of the two branches. The use of the MSE will therefore have the tendency to prioritize the improvement of the image-processing branch (that has the potential to correct large errors in the lower levels of the atmosphere), which may degrade results in the higher atmosphere. This will be shown in more details later on in the chapter.

8.2.1 *The Convolutional Model Branch*

We train the image-processing branch of the network independently. This branch is shown in green in Figure 61. As previously mentioned, this branch relies on a succession of convolutional layers with filters of size 3×3.

As always when using IASI observations, missing data impedes the correct use of convolutional layers. In this application, there are more than 60 % of missing data on average in an image (i. e. orbit). This architecture uses partial convolutions

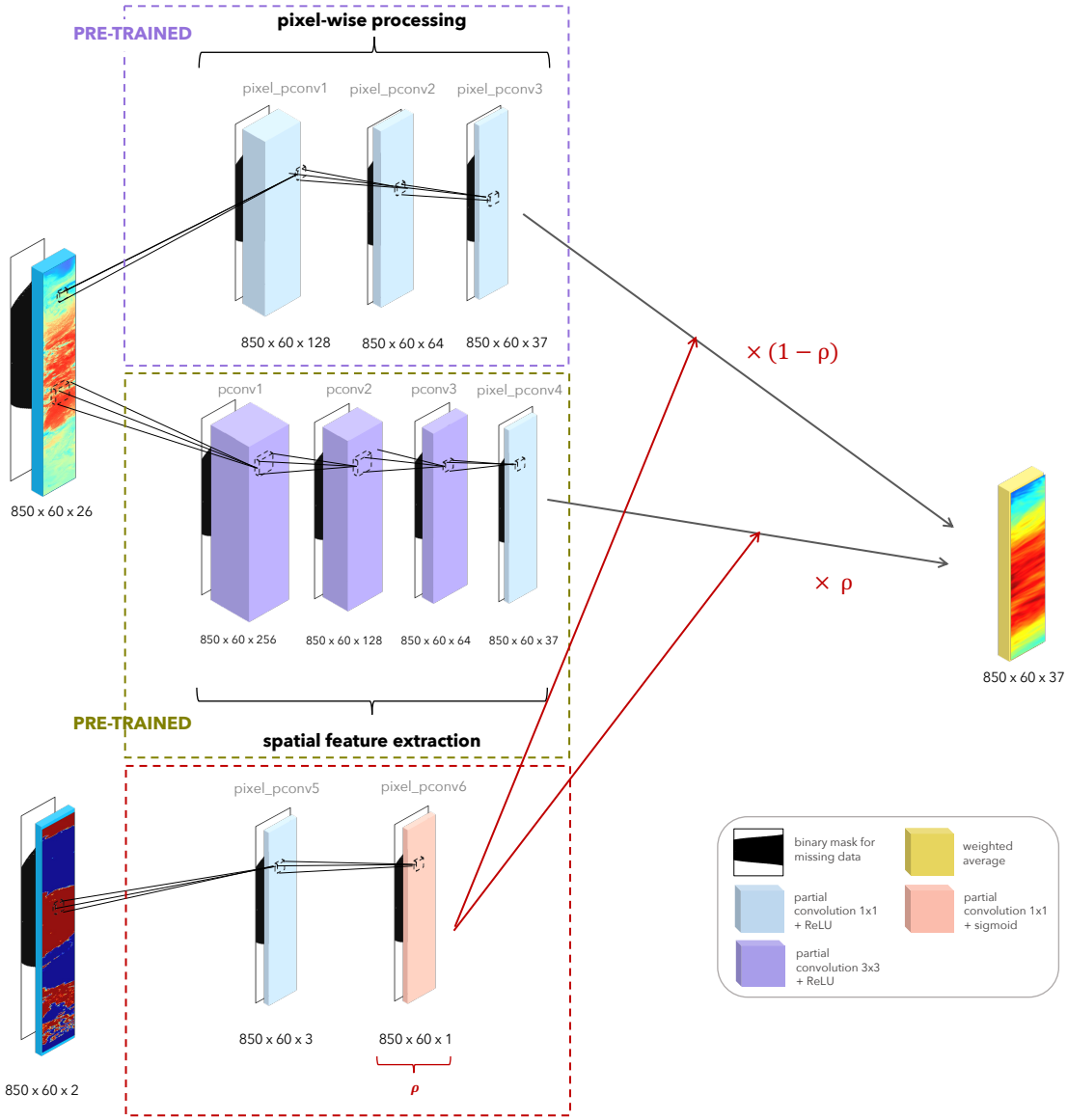


Figure 61 – Architecture of the Partial Convolutional Neural Network for regression.

to offset this issue (as in the previous chapter). We repeat in a few words the implementation of the partial convolution:

$$y_{i,j} = \frac{1}{\sum_{k=0}^{L-1}\sum_{l=0}^{L-1} m_{i+k,j+l}} \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} (x_{i+k,j+l} \odot m_{i+k,j+l}) w_{k,l} + b, \tag{27}$$

where $y_{i,j}$ is the activation map (the output of the convolution layer) at position (i,j), $x_{i,j}$ is the input signal at position (i,j), w is the weight of the filter with size $L \times L$, b is the bias term, m is the binary mask for missing data, and $\odot$ denotes element-wise multiplication.

In the final layer of the network, 37 activation maps are output. These 37 2D-activation maps are the 37 temperature estimates over the pressure levels, for all pixels of the image. The novelty here is that, unlike for the cloud classification ap-

plication presented in the previous chapter, missing values are also found vertically within the outputs (due to terrain elevation). This means that for a pixel classified as clear-sky, we may not have the 37 temperature values available, and this must be taken into account during the training process. To do so, we minimize the MSE that is only calculated on available pixels during training; For pixels that are not available spatially (i. e. that are in cloudy conditions), the loss is set to be 0. For pixels that have a few vertically missing values due to elevation (i. e. mountainous pixels), the loss value is weighted according to the number of pressure levels that are available.

As a summary, this branch outputs an estimate of the atmospheric temperature T_{CNN} over the available pressure levels for each clear-sky pixel in the image, solely based on image-processing feature extraction.

8.2.2 *The Multi-Layer Perceptron Model Branch*

Alongside this CNN branch, we also train the MLP part of the network independently. The pixelwise branch is shown in purple in Figure 61. It is made up of a succession of 1×1 convolutions that act just like fully connected layers. The same technique as for the image-processing branch is used to mask out missing pixels.

This branch also outputs an estimate of the atmospheric temperature T_{MLP} over the available pressure levels for each clear-sky pixel in the image, not taking into account neighbouring pixels (i. e. it does not exploit spatial patterns).

8.2.3 *Concatenation of the Two Branches and Final Network*

The final network is the concatenation of the two latter branches. The way the two estimates of temperature are merged is debatable, and several techniques could be considered. After fully evaluating the performances of both branches (detailed in Section 8.3.1), it was found that two main characteristics influence the performance of each branch:

— *The number of available (i. e. clear) neighbouring pixel*: This is simple to understand as the CNN branch was trained to find spatial patterns in the images and is dependent on their presence in inference mode. If no neighbouring pixels are available, the CNN retrieval is incomplete. Although this does not stop us from using the network on an isolated pixel (this is the advantage of using partial convolutions), the network will not perform as well as if neighbouring pixels were available. Using the MLP retrieval over isolated pixels makes more sense in this case.

— *The land fraction*: This variable describes, in essence, the spatial heterogeneity of the temperature profile over each pixel. Over the ocean, there is no real contrast between neighbouring pixels, and neighbouring pixels are not of much use. In contrast, patterns are much more predominant over coastal areas and, to a lesser extent, over land.

The goal of the final network is therefore to find a parameter ρ that finds the best combination of the two estimates: $T_{COMB} = \rho T_{CNN} + (1 - \rho) T_{MLP}$. The parameter ρ is found using the fraction of available neighbouring pixels (how many pixels amongst, the 9 pixels the CNN uses, are available) and the land fraction, and is the result of two successive pixelwise (i. e. 1×1) convolutions followed by a sigmoid activation layer that forces the output between 0 and 1 (this is equivalent to a small MLP model with 1 hidden layer of 3 neurons). This is seen in the red box of Figure 61. The parameter ρ is therefore the result of a non-linear combination of the fraction of available neighbours and the land fraction. We will investigate this relationship in the following section.

The final architecture is shown in Figure 61. During the training of the final network, the CNN and MLP branch weights are frozen (i. e. no updates are made to them), and only the weights that infer ρ are learnt. The MSE loss is used to compare the target temperature with T_{COMB}, and updates the weights accordingly.

8.3 ATMOSPHERIC TEMPERATURE RETRIEVALS

In this section, we will consider and compare the retrievals of atmospheric temperature coming from the pixelwise branch, the image-processing branch as well as the final combination model.

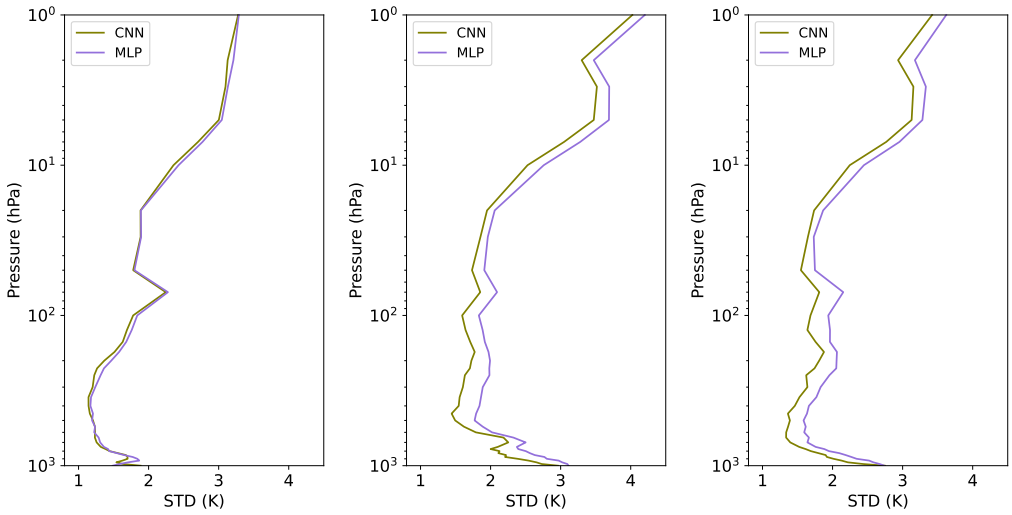
8.3.1 *Looking at the Impact of Using Neighbouring Pixels*

As previously mentioned, the impact of using neighbouring pixels in the retrieval is highly dependent on the variable to retrieve. For the atmospheric temperature profiles, it is also dependent on the pressure level, as the atmosphere is more or less stable depending on the layer. For this reason, we will begin by comparing the two independently trained branches; the MLP branch and the CNN branch.

The statistics are shown with respect to the ERA5 target temperatures, and are computed over the testing dataset (years 2018 and 2019). Although there are two independently trained networks for ascending and descending orbits, since the results follow similar behaviours, we group the results and analyse the entirety of the orbits together.

Figure 62 shows the STD of the temperature retrieval errors over the testing dataset. The green line shows that of the CNN branch, whilst the purple is for the MLP branch. The statistics are separated over land (i. e. where the land fraction is larger than 90%), coasts (i. e. where the land fraction is between 40 and 60%) and oceans (i. e. where the land fraction is less than 10%). This figure shows exactly that the local conditions have a very large effect on the performance of image-processing techniques, and their potential to improve retrievals.

Firstly, it can be seen that over the ocean, where temperatures, especially in the lower layers of the atmosphere, are very smooth spatially, both models are of simi-



(a) over oceans (b) over land (c) over coasts

Figure 62 – STD of the temperature retrieval errors (target - retrieval) over (a) oceans, (b) land and (c) coasts. The STD is shown for all 37 pressure levels. In green is the error made by the CNN branch and in purple the error of the MLP branch.

lar quality, and the CNN does not bring any particular advantage. The CNN offers a slight advantage just below 100 hPa and between 10 and 1 hPa. Over the oceans the STD varies from 1 K in the lower layers and 3.1 K at the top of the atmosphere. However, things look slightly different over coasts and land. Overall, the STD varies from 1.5 K to 4 K. The performance of the CNN model surpasses the MLP model by reducing STD error by almost 0.5 K over land and 0.3 K for coastal areas. This represents a 10% improvement, which is not negligible. The difference in performance is pretty consistent in lower layers of the atmosphere, but becomes less significant for higher layers, where the atmosphere is smoother (and, as will be discussed later, on which IASI contains less information). This highlights once again the importance of spatial heterogeneity being present for image-processing techniques to perform. Using information from neighbouring pixels in this case helps the model to adapt the retrieval in a way that takes into account spatial heterogeneity.

The same plots are shown, differentiating the performances of the two branches over the number of available neighbours in a 3×3 window. We show in Figure 63 the STD of retrieval errors for pixels when there is no neighbour (i. e. the clear pixel is isolated), 2 neighbours, 5 neighbours and 8 neighbours (this is the maximum number of neighbours the CNN uses). Naturally, the more neighbours available, the better the CNN becomes compared to the MLP. Its performance is considerably hindered when no neighbours are available. This is due to the partial convolution architecture, that, despite the presence of missing neighbours, will find 3×3 spatial patterns that help the inversion. These filters converge towards the same filters that would have been learnt if training was done on complete images. Therefore, if information is missing, the retrieval loses quality. This is lessened starting from

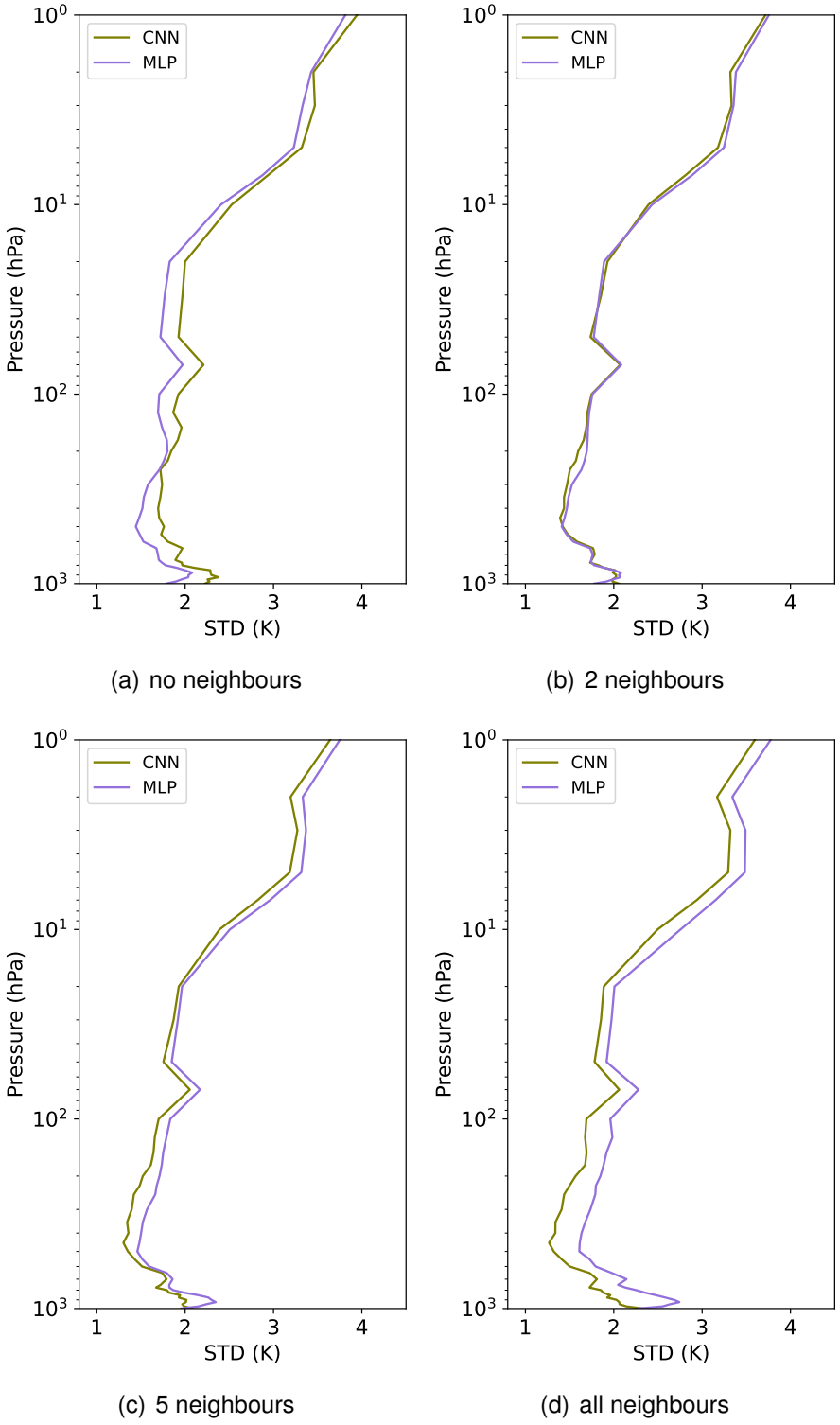


Figure 63 – STD of the temperature retrieval errors (target - retrieval) over points with (a) no, (b) 2, (c) 5, and (d) all neighbours. The STD is shown for all 37 pressure levels. In green is the error made by the CNN branch and in purple the error of the MLP branch.

2 neighbours, where the two branches happen to have similar performances. For points with 5 neighbours and more, the CNN brings a significant advantage in terms of retrieval error.

8.3.2 *Evaluation of the Final Combination Model*

Knowing that the MLP model is better performing in conditions where a clear-sky pixel is isolated from any other, it can be interesting to build a model that is capable of taking the best of both branches. This is exactly what the final model, presented in Section 8.2.3, does. As shown in the previous section, the two main factors that influence the performance of each branch are the land fraction and the fraction of available neighbours. The more neighbours available, the more interest we gain in using the CNN branch. Similar reasoning can be applied but to a lesser extent, with the land fraction. Thus, the two branches are combined in the following way: $T_{COMB} = \rho T_{CNN} + (1-\rho) T_{MLP}$, where ρ is the output of a small MLP model directly linked to these two factors, as shown in Figure 61.

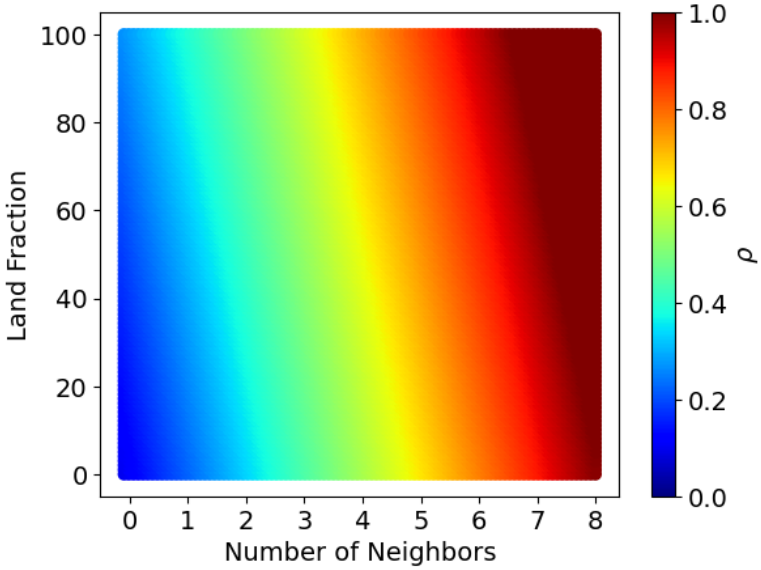


Figure 64 – The parameter ρ with respect to the number of available neighbours (ranging from 0 (none) to 8 (all), and the land fraction in %.

In Figure 64, we show the evolution of the ρ parameter with respect to the number of neighbours on the x-axis and the land fraction on the y-axis. It can be seen that the relationship is quite simple and linear; the more neighbours are available, the more weight is given to the CNN branch. Even more weight is given to the CNN branch if the point is more inland. For instance, for a point that has 50% land fraction and all neighbours, ρ will be equal to 0.8, whilst for a point that is 100% inland and has all neighbours, ρ will be 0.95.

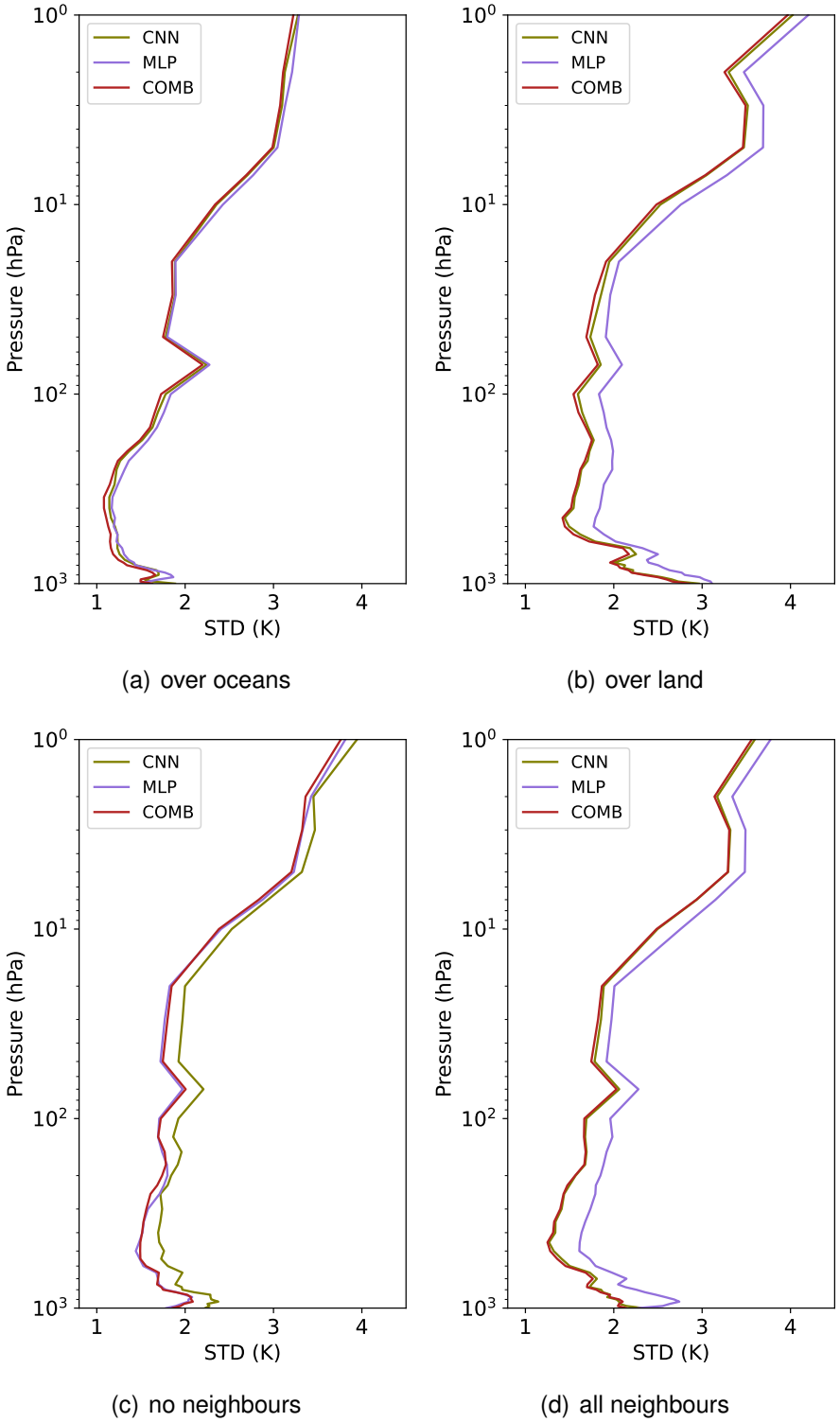


Figure 65 – STD of the temperature retrieval errors (target - retrieval) over (a) oceans, (b) land, and for points with (c) no and (d) all neighbours. The STD is shown for all 37 pressure levels. In green is the error made by the CNN branch, in purple the error of the MLP branch and in red is the error of the combination model.

We now show, in Figure 65, the STD of retrieval errors for the CNN (in green), MLP (in purple) branches, as well as the final model (shown in red), over oceans and land, and for points with no and all (8) neighbours available.

It can be seen that the final model is consistently better over all scenarios, as it is taking a combination of the models, making sure that the most appropriate one gets the more weight. In fact, in some instances, the final model is even better than both models (for instance over oceans, shown in Figure 65), showing once again that taking a mixture of the two branches can prove useful. All models have a peak in error that can be observed just above 100 hPa. The switch between the troposphere and stratosphere occurs around this altitude, with an inversion in the temperature profile. The exact altitude at which this happens varies depending on the region, adding some difficulty for the model to correctly grasp the switch in temperatures.

In Figure 66, we show the bias statistics over the same configurations. For all models, the overall bias over the 37 levels is close to zero; this is by construction of the neural network approach. Neural networks are unbiased estimators, but, on average and over the entire database. When looking at specific subsets of data points, as it is done here, we can observe that a slight positive bias is consistent in the lower layer of the atmosphere, whilst a negative bias is seen in the top layers. The turning point of these two biases occurs around the 100 hPa layers, which is again linked to the inversion in temperature happening between the troposphere and stratosphere. The bias is therefore a result of the harder estimation.

This final model, is the one we consider for the temperature retrieval T_{COMB}, and all further evaluation will be performed on this model. We begin by showing, in Figure 67, the maps of average STD and Bias of the retrieval errors between T_{ERA5} and T_{COMB} on a 1° grid, at different pressure levels (1, 10, 100, 350, 750 and 1000 hPa). Larger errors (close to 4 K) are present in the higher atmosphere. This is due to the IASI spectrum not being very sensitive to the temperature in the higher layers of the atmosphere (as it was shown in Section 2.1.3, more particularly in Figure 9). In fact, many IASI-based atmospheric temperature retrievals, such as in Bouillon et al. (2022), do not consider these higher layers of the atmosphere because of this. Considering this, it is evident that larger errors will arise in those layers. The STD error decreases (reaching an STD as low as 1 K) as we move down the atmosphere, but start to rise again at the surface, due to the heterogeneity of the Earth's surface. Although the CNN branch of the network already helps to limit these errors, there remains an inherent difficulty. The biases that were observed in Figure 66 are found again in the bottom rows of Figure 67. Negative biases occur in the top layers, whilst the bias is positive in the lower atmosphere.

It can also be observed that regional biases are larger close to the surface. This is the case around the Eastern coasts of America and Africa, for instance. The framework that has been chosen in this second part of the thesis relies on localization through auxiliary variables such as the surface emissivities and the land fraction

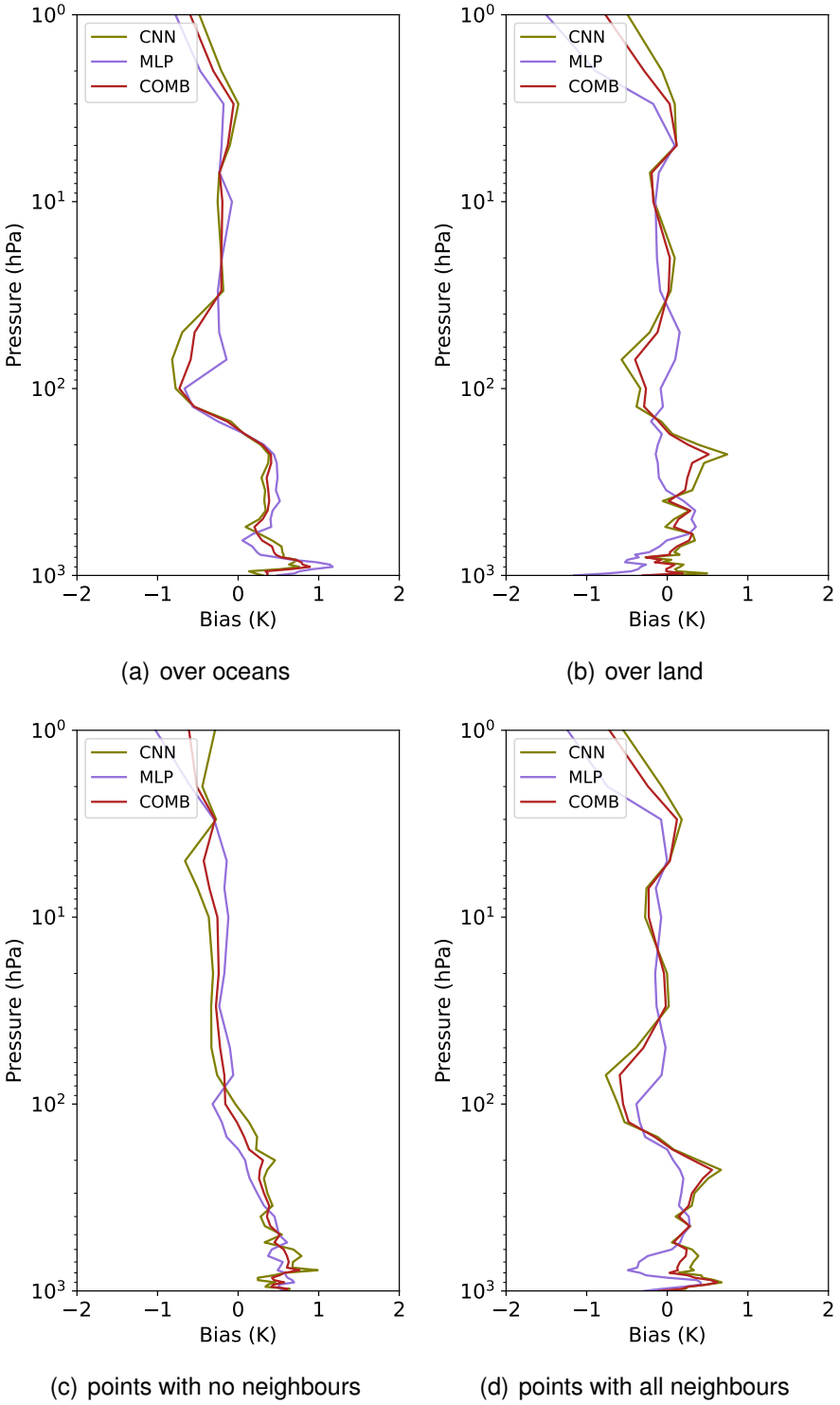


Figure 66 – Bias of the temperature retrieval errors (target - retrieval) over (a) oceans, (b) land, and for points with (c) no and (d) all neighbours. The Bias is shown for all 37 pressure levels. In green is the error made by the CNN branch, in purple the error of the MLP branch, and in red is the error of the combination model.

only. As it was shown in Part I of this thesis, this may not be sufficient to completely exclude regional biases from arising. Indeed, a global model means that the network has to find a compromise between all regions of the Earth. With more computational power, it would be possible to consider static orbits (taking into account IASI's 29-day revisit period) to create more localized CNNs that differentiate their weights for different regions, at least in the lower layers. As it was shown in Chapter 4, this is the best strategy to reduce the regional biases we can observe in the figure.

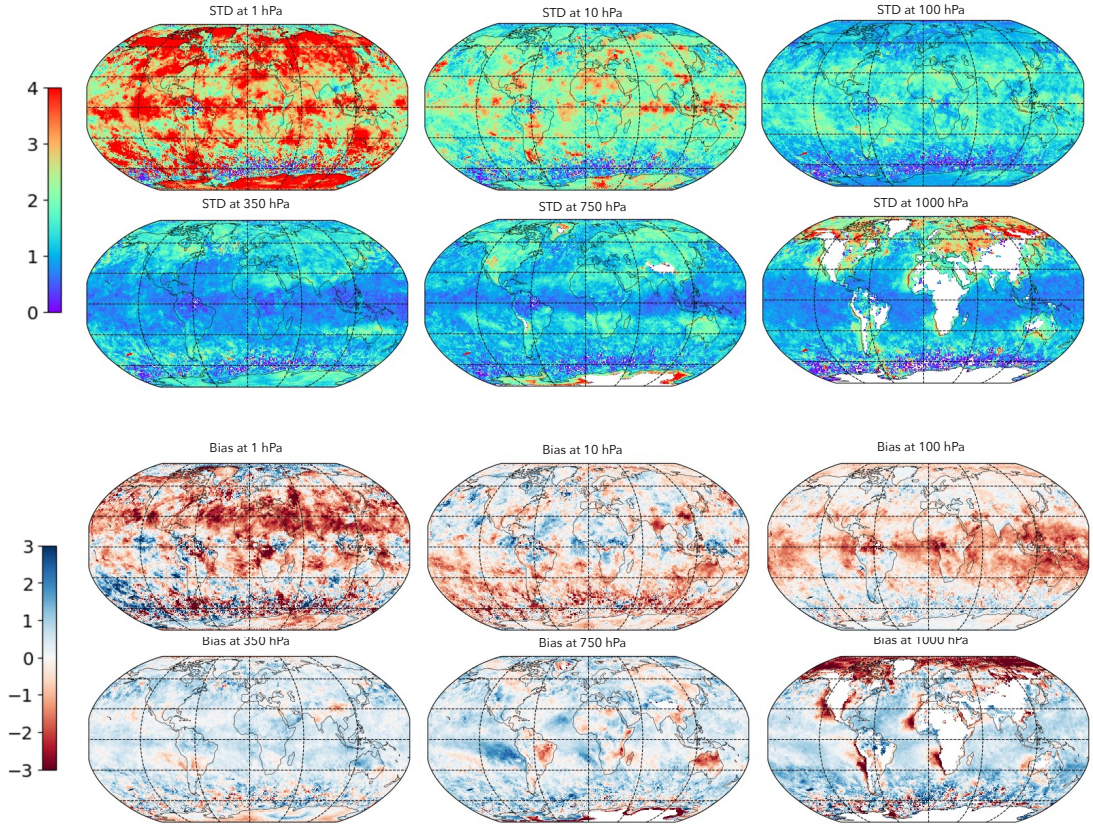


Figure 67 – Average STD (top) and Bias (bottom) errors between the retrieval and ERA5 temperatures at 1, 10, 100, 350, 750 and 1000 hPa in kelvins on a 1° grid. Fewer pixels are plotted in the lower layers due to changes in surface pressure.

The average profiles retrieved by the final model over, in blue, polar latitudes (below 60°S and above 60°N; in orange, mid-latitudes (between 30°S and 60°*S* and 30°N and 60°N); and in yellow, in the tropics (between 30°S and 30°N) are shown in Figure 68(a). These profiles are perfectly in line with the ERA5 average profiles, shown in dotted lines. There is very little difference between the two lines. All regions are correctly retrieved on average. Figure 68(b) shows the difference in the two average profiles over the same regions. The fact that the biases are different depending on the region is the result of the model being global. The model has to

make compromises in order to estimate, at its best, what is happening all over the Earth. However, these differences are less than 1 K in all situations, showing that regional biases are not too important.

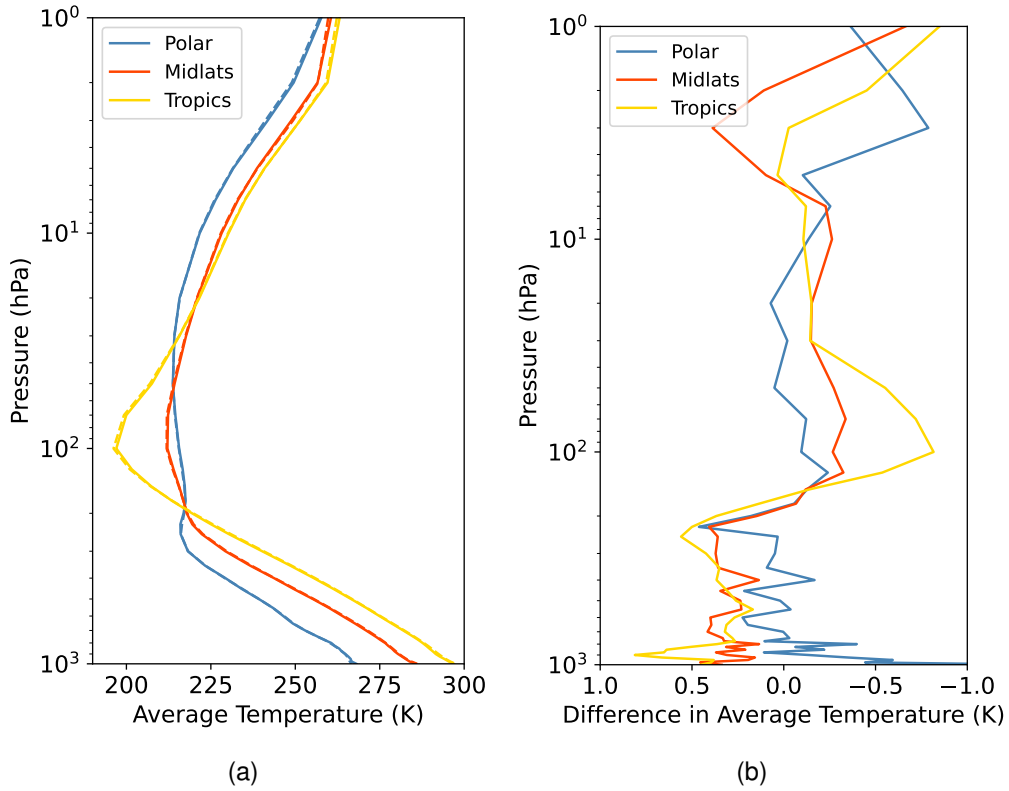


Figure 68 – (a) In solid lines, average temperature profiles retrieved by the final model over polar latitudes (in blue), mid-latitudes (in orange) and tropics (in yellow). In dashed lines, average ERA5 profiles over the same regions. (b) The difference in average profiles (ERA - T_{COMB}).

As an example of retrieval, we also illustrate in Figure 69 the target and estimate temperatures of the 14 daily ascending orbits extracted from 1st April 2018. The same pressure levels are shown, as in the last figure. As we come closer to the Earth's surface, the number of plotted pixels diminishes due to surface pressure variation (i. e. terrain elevation). Looking at the retrievals, it is evident that the model has a harder time retrieving the correct temperatures in higher layers of the atmosphere. For instance, at 1 hPa the bias is visible (i. e. the model overestimates the temperature). This is consistent with the previous plots. Overall, and even more in the mid-layers of the atmosphere, the model and the ERA5 temperatures are very similar, and spatial patterns are present in both maps.

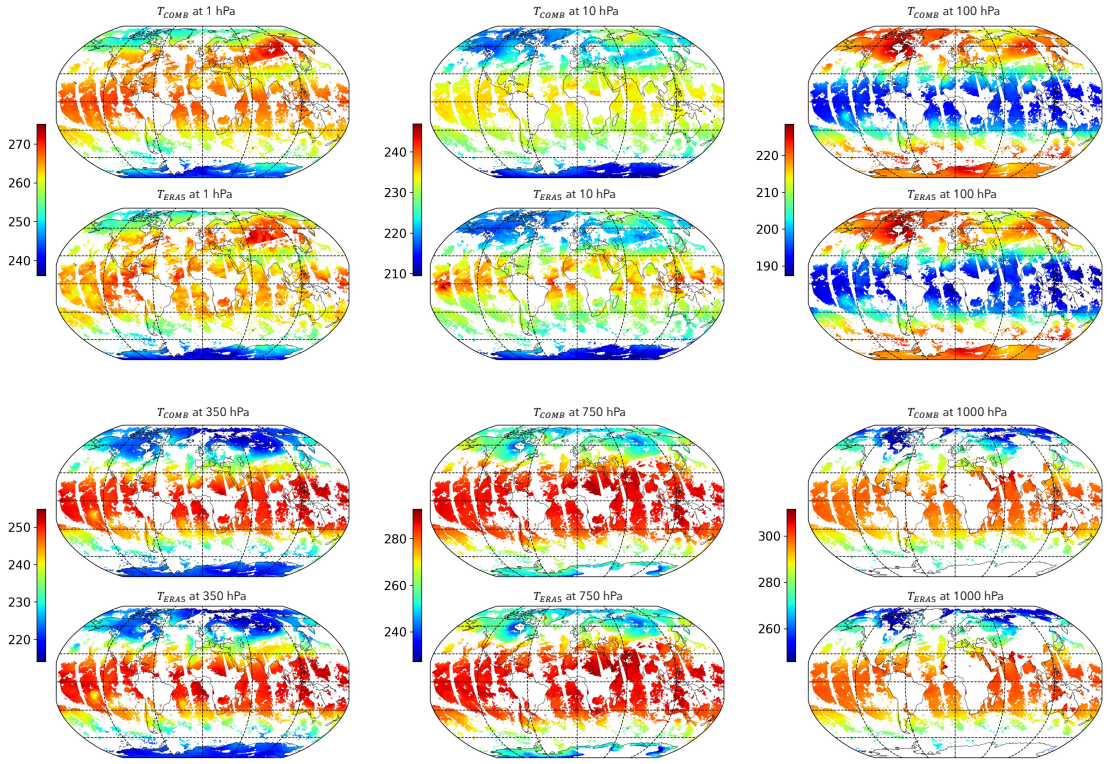


Figure 69 – Examples of retrieval (top) and target (bottom) clear-sky temperature at 1, 10, 100, 350, 750 and 1000 hPa in kelvins. Fewer pixels are plotted in the lower layers due to changes in surface pressure. The 14 daily ascending orbits from April 1st 2018 are shown.

8.3.3 *Validation Using the ARSA Database*

The previous sections have evaluated our model against the target, i. e. temperature profiles from ERA5. We now validate the retrieved product by comparing the retrieved temperature profiles to *in situ* observations from radiosondes, that show the "real" temperatures of the atmosphere at a certain time. As explained in Section 8.1.3, the ARSA database contains worldwide reprocessed radiosonde measurements that we collocate with the IASI measurement points. Firstly, the ARSA temperature profiles are interpolated onto the 37 ERA5 pressure levels. Collocation is performed using a maximum time difference of 30 minutes and a spatial distance of less than 0.1°. For each radiosonde, the nearest IASI measurement point is kept if it exists. In addition to this, for each IASI point that has been associated with a radiosonde measurement, the EUMETSAT Level 2 temperature profile is retrieved (and interpolated to the same 37 pressure levels) from the EUMETSAT (2022) database.

This results in 2223 clear-sky profiles collocated in 2019, the locations of which are shown in Figure 70. There is quite a large disparity in the number of points in different regions due to time zones. For example, there are no collocated points over North America; IASI passes over the West coast of America at around 10 am and

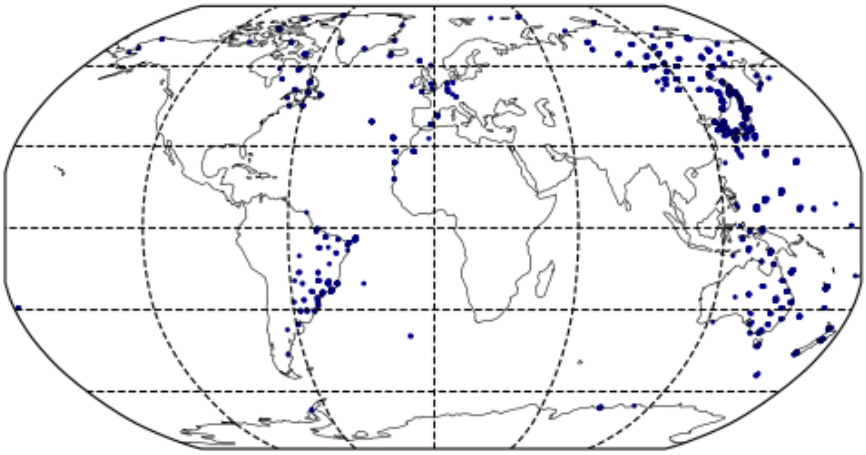


Figure 70 – Location of IASI/ARSA collocated clear-sky profiles between January and August 2019. A IASI point must find itself at a distance of less than 1° and have a maximum time difference of 30 minutes to be collocated with an ARSA measurement. The disparity in the number of points is due to this constraint. More details are given in the text.

10 pm local time, but radiosondes are sent at 00 and 12 UTC which corresponds to 5 am and 5 pm local time. Collocation in such areas is therefore very limited (not to say impossible if no interpolation is done). For the Eastern coast, it would be required to extend the collocation period to over 90 minutes to obtain a significant amount of collocated points.

Over these profiles, we are therefore able to compare the radiosonde measurement T_{ARSA} to our retrieval T_{COMB}, and the EUMETSAT T_{EUM}. The objective is for the retrieval to be as close as possible to the *in situ* measurement.

Figure 71(a) shows the STD error between T_{ARSA} and respectively the retrievals T_{COMB} (shown in red) and T_{EUM} (shown in yellow), over all the collocated points. The smaller the STD error, the closer the retrievals are to the *in situ* measurements. The figure shows that our CNN-based retrieval T_{COMB} is in fact significantly closer to the ARSA database for lower levels of the atmosphere. The STD error between the retrieval and ARSA is reduced by close to 1 K at 800 hPa in comparison to the EUMETSAT retrieval. This is clearly shown by Figure 71(b), in which we show the same STD errors but without the log-scale on the y-axis (emphasizing on the lower atmosphere).

However, the trend reverses as we go through the tropopause and enter into the stratosphere, where the EUMETSAT retrieval seems to be of better quality. We remind here that the EUMETSAT PWLR3 algorithm does not only use the IASI observations to retrieve the temperature profiles, but also some microwave information coming from AMSU and MHS. This can explain the difference in retrievals in the higher atmosphere; the IASI spectrum is less sensitive to the temperature in the higher layers of the atmosphere (as it was shown in Section 2.1.3, more partic-

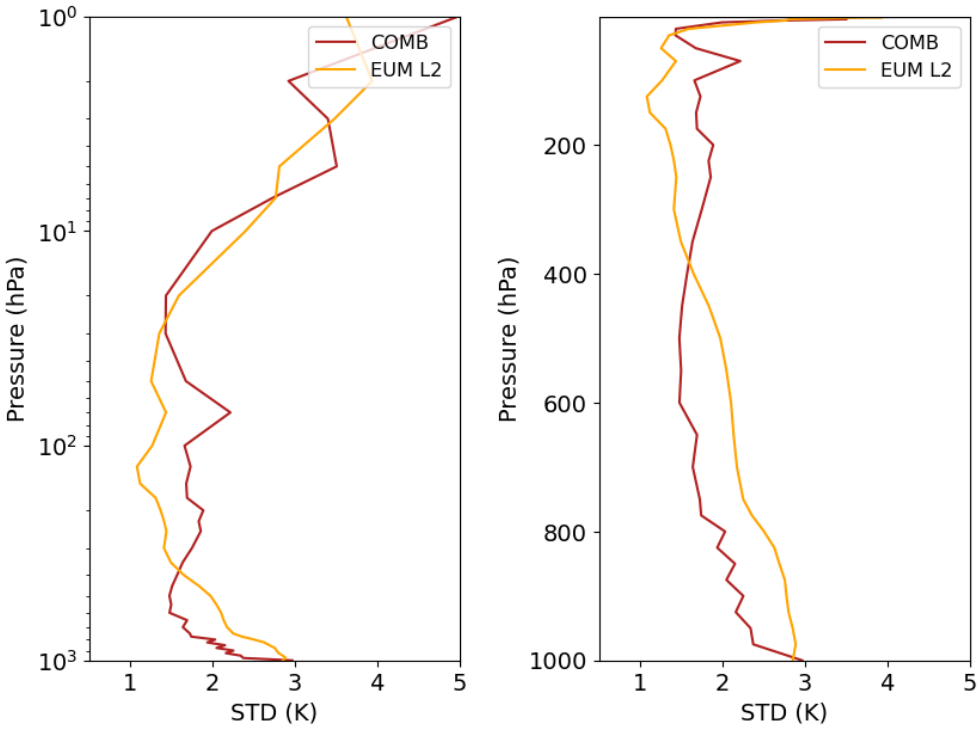


Figure 71 – STD errors (in K) between the collocated ARSA profiles and (in red) the retrieved T_{COMB}, and (in yellow) the Level 2 EUMETSAT profiles.

ularly in Figure 9), whereas the information contained in the additional microwave observations are more sensitive to the higher atmosphere.

Finally, we show in Figure 72 two example profiles, taken from the year 2019. The figure shows locations in (a) North America, (b) Eastern Asia. The profiles are, overall, well retrieved by both algorithms; It can once again be seen in Figure 72(a) that the EUMETSAT retrieval is more representative in the higher layers, whilst our model has an advantage at the bottom of the atmosphere.

In conclusion, we have demonstrated that the image-processing based model proposed in this chapter is capable of producing accurate temperature profile estimates that are comparable to the EUMETSAT Level 2 product. The model experiences more difficulty in the top layers of the atmosphere, where less information is present in the IASI observations. However, the obtained retrieval is in fact better than the EUMETSAT one in the lower layers of the atmosphere. In those layers, the impact of using neighbouring pixels is higher and significant.

8.4 CONCLUSIONS

In this chapter, we designed a CNN-based regression model (that also incorporates a pixelwise approach) to retrieve global clear-sky atmospheric temperature profiles

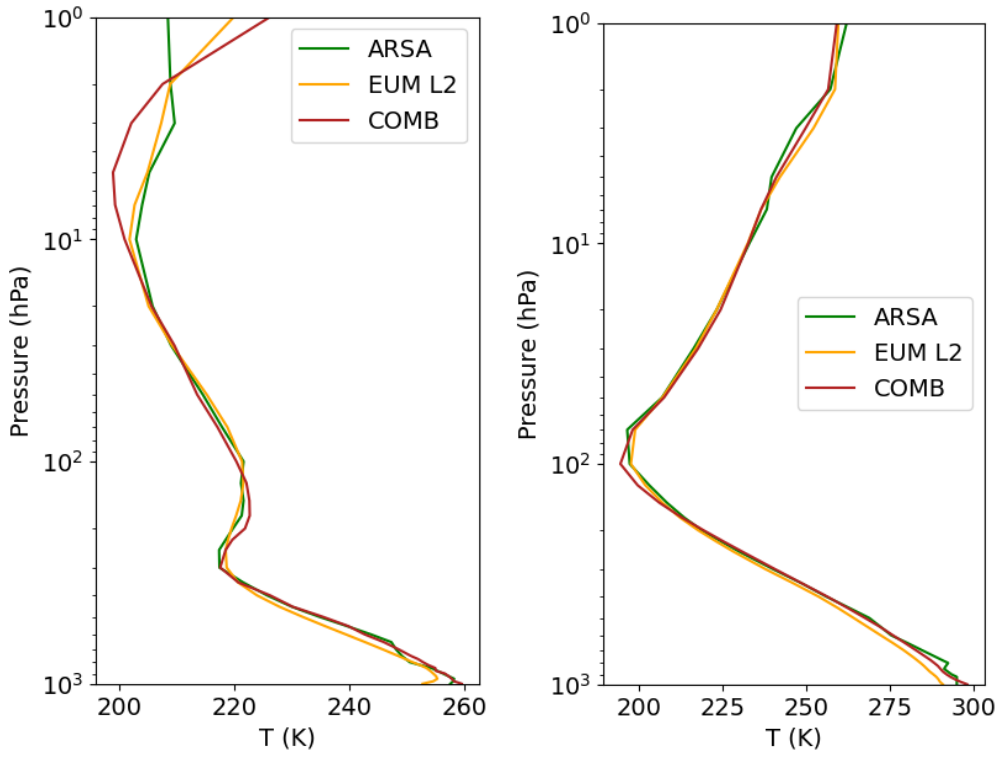


Figure 72 – Examples of temperature profiles over (a) North America and (b) Eastern Asia in 2019. The red lines show our retrieval, the yellow line represents the EUMETSAT Level 2 retrieval, and the green line is the ARSA *in situ* profile.

over 37 pressure levels. This is possible through the use of partial convolutions introduced in the previous chapter, that allow to train an image-processing technique despite the presence of missing data (i. e. cloudy areas). This is a necessity if we want to use such techniques on IASI observations, that are very sensitive to the presence of clouds.

It has been strengthened in this chapter that the impact of using neighbouring pixels for the inversion of IASI observations is very much dependent on the variable we are trying to retrieve, and even more so its spatial coherency. For example, we show that over oceans and in higher altitudes, the difference with a pixelwise approach is lessened. However, in heterogeneous zones, where spatial dependencies are high, namely over coasts, lands and in lower layers of the atmosphere, the use of surrounding pixels brings useful information to retrieve the temperature.

We have also shown that using image-processing approaches have the capacity to do as well as operational Level 2 retrievals done at EUMETSAT. These operational products mostly rely on IASI observations, but also use microwave information, that make up for deficiencies in IASI observations, for example at the top of the atmosphere. For layers of the atmosphere where the IASI spectrum is very sensitive to the temperature, the model presented in this chapter creates temperature estimates

that are actually 1 K closer to *in situ* measurements than those of the EUMETSAT retrievals.

Overall, the method proposed in this chapter is simple to use, only requiring to reset the IASI orbits into their original format. Although in this study the three separate branches were trained independently, with some initialization it should be possible to train all the branches simultaneously. In all cases, the inference is done in one single pass, and takes only a couple of seconds for the processing of one month of observations.

CONCLUSIONS ON THE GLOBAL RETRIEVAL PROCESSING SCHEME

The objective of this second and last part of the thesis was to put to use the understanding gained on how to use image-processing techniques on IASI observations. In the first part of the thesis, potential strategies for conducting global retrievals were discussed. However, the significant challenge of coping with extensive missing data obstructed the straightforward application of image-processing techniques at a global scale. To address this issue, the concept of partial convolutions, originally designed for image in-painting, was repurposed to handle missing data effectively.

This part of the thesis serves to prove that a global retrieval is possible, and shows how they can benefit from the use of CNNs. The global processing is done by restructuring orbits into their original geometry, and considering each orbit to be an image. The image is a sort of long rectangular strip where each pixel is related to its neighbours. A IASI measurement point (i. e. one pixel of the IASI FOV) serves as a pixel of an image.

We propose, in Chapter 7, a CNN-based classifier, trained on the SEVIRI reference cloud product, that gives us a global classification of the cloud phase. We show that using image-processing techniques in which spatial patterns are recognized and used helps to differentiate two layer clouds from single layer ones. The technique also demonstrated that a form of downscaling could be done, as an estimate of the percentage of clouds within each IASI pixel could also be derived.

This global cloud fraction product was then used to mask out any cloudy areas and perform global retrieval of atmospheric temperatures. It was shown in Chapter 8 that using information contained in neighbouring areas could be beneficial for the temperature retrieval, especially in the lower atmosphere, where our retrievals find themselves closer to *in situ* observations than the operational products distributed by EUMETSAT (2022).

The two application cases presented in this part of the thesis have the potential to be seamlessly integrated into an operational processing chain. The ultimate objective would be to create a unique processing pipeline to deal with IASI orbits and obtain, in near-real time, estimates of several surface and atmospheric parameters. Upon reception of the IASI orbits at the ground station, they could be fed directly into the cloud classifier. Subsequently, clear sky surface and atmospheric temperatures could be retrieved in real time as part of this streamlined process.

9

CONCLUSIONS AND PERSPECTIVES

Contents

The objective of this thesis was to exploit image-processing techniques for the purpose of (1) better understand how they can be correctly and optimally applied to IASI observations and (2) create improved, global retrievals of some geophysical variables. This chapter will summarize the main results that were presented in Parts I and II of this thesis. We will then finish by describing the prospects that this work opens up.

9.1 TECHNICAL ADVANCES

Part I of this thesis focused on preliminary technical developments that paved the way for global retrievals. We started by focusing on a small spatial domain. This choice was initially made to simplify the use of CNNs by allowing us to omit some of the challenges of a global approach. For instance, the IASI data was re-gridded onto a chosen small domain, which helped to address the issues posed by missing data. Indeed, the domain was sufficiently small to allow for inter- and extrapolation over cloudy areas.

The simplified use case meant that there was room to explore several image-processing techniques and how they should be used to really benefit IASI-based retrievals. This was done in Chapter 4 through the adaptation of the behaviour of a neural network to local conditions, both at the pixel and image scales. The most promising development was the localized-CNN, in which weights are not shared. This means that the model is able to find particular spatial patterns for each location in the image, and provide an improved estimate of the surface temperature, by considerably reducing local biases. It was also shown in Chapter 6 that such a localization helped to better retrieve extreme quantities because local conditions were better understood. Better understanding local conditions also means better anticipating what kind of extreme values the ST can reach, and how to integrate them into the retrieved ranges.

Chapter 5 also leveraged on this simple use case to estimate and provide retrieval uncertainties. Although neural models have proved to facilitate and improve remote sensing tasks over the years, a deficiency is their lack of interpretability. The absence of uncertainty associated to an estimated quantity contributes to this. In this chapter, we developed a method to estimate both ST and the associated uncertainty. This was done by training a first model before computing its errors (that we consider to be the sum of all sources of uncertainties) on small bins of the input space. A second model is then trained to output this target error as well as the temperature estimate. This method has the advantage of creating state-dependent uncertainty estimates that are a true asset for the operational use of statistical methods in remote sensing and their potential use in data assimilation methods.

To summarize, the first part of this thesis has set the scene on how to use image-processing approaches to extract the best out of them. We showed that the main challenge with infrared observations was the amount of missing data that can be problematic to train CNN models. This was resolved firstly by inter and extrapolating the holes to create full images, on which we train models. However, it was evident that this solution was not suitable for global retrievals. Reflecting on how to adapt the techniques presented in the first part to global processing led to adapting the partial convolutional layer, initially meant for image in-painting, to learn convolutional filters despite the presence of cloudy areas. This was introduced in Part II of the thesis and represents perhaps the most important technical advance proposed in this thesis. Without such a tool, global processing of large images containing a large amount of missing pixels would be close to impossible. With our new developments, we are now able to train a CNN while omitting missing pixels.

To conclude, the first objective of this thesis was to learn (and teach) how to use image-processing techniques on what seemed to be a very impractical use case: the IASI orbits. Solving some of the challenges we met at the beginning really allowed to determine the best way to use the CNN models on such observations, so that there was a real improvement on the retrieved products. The expertise gained by working on this was useful to design several solutions for the global processing of the observations. The selected overall framework is to consider an orbit as an image, use partial convolutions and localize the retrievals by inputting additional information to the model; this is perhaps the easiest and most straightforward way to deal with the complexity of the IASI orbits.

9.2 IMPROVED IASI RETRIEVALS

The second objective of this thesis was to create improved, global retrievals of geophysical variables using the developed techniques. Indeed, since the launch of the IASI instrument, retrievals have been done at the pixel scale, which puts some limits on spatial consistency of the retrievals. Of course, this was partly due to the lack of popularity of image-processing techniques at the time, but could it also be due to the limited spatial resolution of the IASI instrument, which limits

the need to look at what is happening in neighbouring pixels? The main takeaway message from the different applications presented in this thesis is the importance of considering the variable being retrieved. It is not only vital to thoroughly understand how image-processing techniques work, but also what we want to do with them.

Chapter 4 showed that for heterogeneous surface temperature over land, it was useful to look at neighbouring pixels and even more so to adapt the behaviour of the algorithm over each part of the surface. Chapter 7 showed that, on the contrary, when trying to find clouds, it was important to train a model that was capable of finding very general spatial patterns that are specific to each cloud type. Finally, Chapter 8 focused on atmospheric temperature profiles, which represented a perfect example to demonstrate how the advantages of image-processing really depends on the conditions. We showed that the CNN was of real interest in the lower layers of the atmosphere, and even more clearly over heterogeneous regions (i. e. mountains, land and coasts).

Using CNN models at the image scale versus employing a pixelwise model is not a clear-cut decision. However, I truly hope that this thesis can serve to convince future users of IASI-orbit data that incorporating some image-processing techniques into their approach is essential for achieving improved retrievals.

9.3 PROSPECTS

The work presented in this thesis is part of a methodological development in the use of image-processing techniques for IASI observations, and more generally for polar orbiting and infrared observations. This thesis aimed at exploring possibilities on how to best apply such techniques. Naturally, only a subset of DL architectures could be evaluated across a restricted set of geophysical variables. Future directions include assessing novel models and anticipating the expanded capabilities that are enabled by upcoming launches of new hyperspectral infrared sounder instruments.

9.3.1 *Future Methodological Developments*

Firstly, it was evident in this thesis that choices were made depending on the computational limitations we were faced with. For instance, the final developments for the classification of the cloud phase, presented in Chapter 7, and the retrieval of atmospheric temperature profiles, presented in Chapter 8, were performed using what we call a traditional CNN (i. e. one that does not adapt its weights depending on the image area). Ideally, we would have liked to repeat the exercise using more complex architectures, such as those developed in Part I of this thesis, especially for the lower layers of the atmosphere. A future development would therefore be to extend the localized-CNN model globally. This would be possible without too much trouble, providing we have sufficient computational resources.

Furthermore, the thesis concentrated exclusively on the use of CNNs as the main model for image-processing. This was because it was, at the time, the most used algorithm that achieved outstanding results across many fields of applications. Since then, a new approach known as vision transformers has emerged. These models were briefly described in Section 3.5. ViTs could prove beneficial in this context, particularly because we often deal with large images representing orbital views. Transformers possess the ability to discern patterns between distant patches across an image, which aligns well with the spatial characteristics of our data. Testing and adapting ViTs to our purpose is therefore another prospect of this thesis.

9.3.2 *Extension to Other Geophysical Parameters*

This thesis focused on retrieving ST in Part I, and cloud properties and atmospheric temperature profiles in Part II. Focusing on these geophysical variables already started to show how differently image-processing techniques behave depending on the variable of interest. For instance, it was shown in Aires, Boucher, and Pellet (2021) that using CNNs for the retrieval of SST showed only little or no advantage, as the SST fields are very smooth. Chapter 4 however put in light that using neighbouring pixels had an undeniable advantage over land, especially where spatial heterogeneity is evident (i. e. over coastal and mountainous areas). Chapter 7 also conveys that image-processing techniques bring large advantages for the identification and classification of clouds, for example by being able to distinguish two-layer clouds. Chapter 8 also highlighted the difference in performance of a CNN model depending on the level of the atmosphere in which to retrieve the temperature.

With this in mind, it would be interesting to analyse which other geophysical variables could benefit from having image-processing retrievals. For instance, one could consider retrieving humidity profiles simultaneously with atmospheric temperatures. Since the humidity has a close relationship to the temperature, it could be argued that retrieving both at the same time could help correct certain errors obtained when retrieving only the temperature profiles, for example at 100 hPa as we have seen. The same thing could be envisaged to retrieve the ST, in addition to the 37 atmospheric temperatures.

9.3.3 *Contributions of Future Instruments*

IASI is, as of today, one of the most important sounders for Earth observation, and it accounts for a vast majority of the satellite data that is assimilated into NWP models. However, new and future missions are sending image-processing-prone infrared spectrometers into space.

On the one hand, IASI-NG, with augmented spectral resolution, is expected to be launched in 2025. The IASI-NG FOV will double, from a 2×2 pixel grid covering 50×50 km^2, to a 4×4 pixel grid covering 100×100 km^2. Although the spatial resolution remains the same, this, coupled with the radiometric noise being reduced

by a factor 2, will increase spatial coherency between IASI pixels and will favour the use of image-processing techniques to take into account neighbouring pixels.

On the other hand, the IRS instrument that is planned to launch in 2025 on MTG-S. Being the first operational hyperspectral infrared sounder in geostationary orbit, IRS observations will be a perfect trade-off between spectral resolution and spatial and temporal resolutions due to the geostationary orbit, which will benefit from the use of, for example, the localized-CNN presented in this thesis.

APPENDIX

A

APPENDIX A: SUPPLEMENTARY MATERIAL

A.1 IASI NEDT NOISE CALCULATIONS

A.1.1 *Derivation of Planck's Law*

$$\frac{dB_\lambda(T)}{dT} = \frac{2\,c^3\,h^2\,e^{\frac{ch}{k\lambda T}}}{k\,\lambda^6\,T^2\left(e^{\frac{ch}{k\lambda T}}-1\right)^2} \tag{28}$$

A.1.2 *Measured IASI NEDT Noise at 280 K*

The measured IASI NEDT noise at 280 K noise is downloadable by clicking here.

A.2 IASI FOOTPRINT CALCULATIONS

A.2.1 *Calculations*

To calculate the footprints of IASI pixels, it is necessary to obtain the dimensions of the minor and major axes of the IASI footprints (ellipses) as a function of the viewing angle. These calculations were taken from Souichiro Hioki and Aurélien Chauvigné (Lille University). Assuming the Earth is a perfect sphere, we have the following:

with Θ being the viewing angle, R the Earth's radius and h the satellite altitude. Following the law of the sines we have

$$\frac{R}{\sin\alpha} = \frac{R+h}{\sin(\pi-\Theta)} \tag{29}$$

This gives:

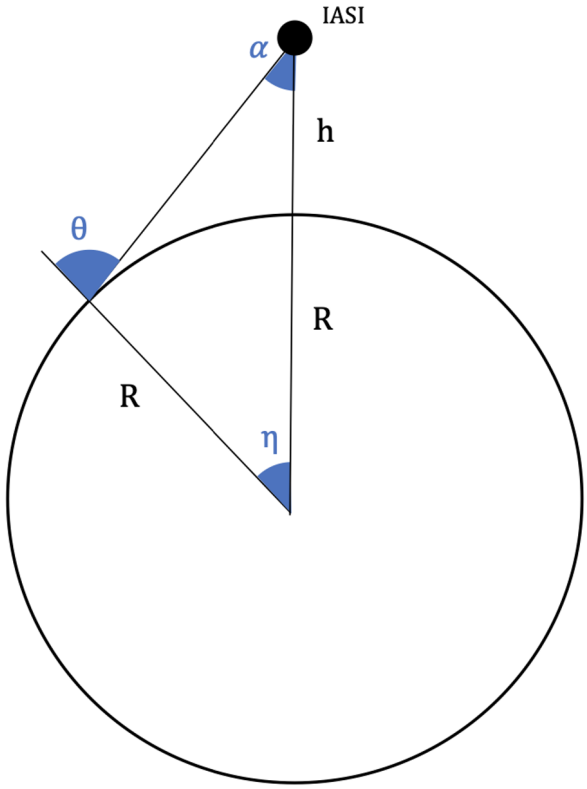
$$\alpha = \sin^{-1}\left(\frac{R}{R+h}\sin(\pi-\Theta)\right) \tag{30}$$

and

$$\Theta = \sin^{-1}\left(\frac{R+h}{R}\sin(\alpha)\right). \tag{31}$$

Since $\eta = \Theta - \alpha$, from Equation 31 we have:

$$\eta = \sin^{-1}\left(\frac{R+h}{R}\sin(\alpha)\right) - \alpha = \gamma(\alpha) \tag{32}$$



Finally, we have the angular dimensions of the small (m) and large (M) axes of the IASI ellipse in radians:

$$m = \frac{1}{2}\big(\gamma(-\beta) - \gamma(\beta)\big)$$

and

$$M = \frac{1}{2}\big(\gamma(\alpha - \beta) - \gamma(\alpha + \beta)\big)$$

A.2.2 *Python Code*

The python code below calculates the ellipse coordinates for a IASI pixel given its FOV (1 to 30), the pixel's latitude and longitude, and the satellite zenith and azimuth angles.

```
import matplotlib.pyplot as plt
import matplotlib
from mpl_toolkits.axes_grid1 import make_axes_locatable
import matplotlib.axes as maxes
import cartopy as cart
import cartopy.crs as ccrs
import numpy as np
import math

def gamma_iasi(alpha_iasi,satalt=817):
    earth_radius_km = 6378.100
```

```
    calc =((earth_radius_km+satalt)*np.sin(alpha_iasi))/earth_radius_km
    inds = np.where(((calc<1)&(np.around(calc,1)==-1)) |((calc>1) & (np.around(calc,1
                                            )==1)))
    calc[inds] = np.around(calc[inds],1)

    return np.arcsin(calc)-alpha_iasi

def IASI_footprint_ellipses(fov,latc,lonc,satZenc,satAzimc):
    earth_radius_km = 6378.100
    satalt = 817
    angdiam = 0.82250
    beta_iasi = math.radians(angdiam)/2.
    saz = np.radians(satZenc)
    alpha_iasi = np.arcsin((earth_radius_km/(earth_radius_km+satalt))*np.sin(np.pi-
                                            saz))

    # Angular dimensions of minor and major axes

    r_major = np.degrees(0.5*abs(gamma_iasi(alpha_iasi-beta_iasi)-gamma_iasi(
                                            alpha_iasi+beta_iasi)))
    r_minor = np.degrees(0.5*abs(gamma_iasi_scalar(-beta_iasi)-gamma_iasi_scalar(
                                            beta_iasi)))

    # Transform to cartesian equation (to plot)

    theta_minor = np.radians(90)
    theta_major = np.radians(0)

    x_major = r_major*np.cos(theta_major)
    y_major = r_major*np.sin(theta_major)

    x_minor = r_minor*np.cos(theta_minor)
    y_minor = r_minor*np.sin(theta_minor)


    return  [2*x_major, 2*y_minor, (90-satAzimc)]

# Plot
ellipse = IASI_footprint_ellipses(fov,
                              latitude,longitude,satZenith,satAzim)

fig = plt.figure(figsize=(16,16))
ax = plt.axes(projection=ccrs.Robinson())
ax.coastlines(linewidth=0.4,color='0.05')

ax.add_patch(mpatches.Ellipse(xy=[longitude,latitude], width=ellipse[0], height=
                                        ellipse[1], angle = ellipse[2],transform=
                                        ccrs.PlateCarree(), facecolor='
                                        mediumorchid',alpha=0.3))
ax.scatter(longitude,latitude,transform=ccrs.PlateCarree(), c=black, s=.1, alpha=0.
                                        001)

ax.set_xlabel('longitude')
```

```
ax.set_ylabel('latitude')
sm = plt.cm.ScalarMappable(cmap=c_colour, norm=plt.Normalize(vmin=0, vmax=5))
sm._A = []
gl = ax.gridlines(crs=ccrs.PlateCarree(), linewidth=2, color='black', alpha=0.2,
                                          linestyle='--', draw_labels=True)
gl.xlabels_top = False
gl.ylabels_left = False
gl.ylabels_right=True
gl.xlines = True
ax.add_feature(cart.feature.OCEAN)
plt.savefig('ellipse.pdf',dpi=200,bbox_inches='tight')
```

BIBLIOGRAPHY

Abdar, Moloud, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. (2021). « A review of uncertainty quantification in deep learning: Techniques, applications and challenges. » In: *Information fusion* 76, pp. 243–297.

Aires, Filipe, Eulalie Boucher, and Victor Pellet (2021). « Convolutional neural networks for satellite remote sensing at coarse resolution. Application for the SST retrieval using IASI. » In: *Remote Sensing of Environment* 263, p. 112553. DOI: 10.1016/j.rse.2021.112553.

Aires, Filipe, Alain Chédin, Noëlle A. Scott, and William B. Rossow (2002a). « A regularized neural net approach for retrieval of atmospheric and surface temperatures with the IASI instrument. » In: *Journal of Applied Meteorology* 41.2, pp. 144–159. DOI: 10.1175/1520-0450(2002)041<0144:ARNNAF>2.0.CO;2.

Aires, Filipe, Francis Marquisseau, Catherine Prigent, and Geneviève Sèze (2011). « A land and ocean microwave cloud classification algorithm derived from AMSU-A and-B, trained using MSG-SEVIRI infrared and visible observations. » In: *Monthly weather review* 139.8, pp. 2347–2366. DOI: 10.1175/MWR-D-10-05012.1.

Aires, Filipe, Fabrice Papa, Catherine Prigent, Jean-François Crétaux, and Muriel Berge-Nguyen (2014). « Characterization and space–time downscaling of the inundation extent over the inner Niger Delta using GIEMS and MODIS data. » In: *Journal of Hydrometeorology* 15.1, pp. 171–192. DOI: 10.1175/JHM-D-13-032.1.

Aires, Filipe and Victor Pellet (2021). « Estimating retrieval errors from neural network inversion schemes — Application to the retrieval of temperature profiles from IASI. » In: *IEEE Transactions on Geoscience and Remote Sensing* 59.8, pp. 6386–6396. DOI: 10.1109/TGRS.2020.3026944.

Aires, Filipe, Catherine Prigent, and William B Rossow (2004a). « Neural network uncertainty assessment using Bayesian statistics: A remote sensing application. » In: *Neural Computation* 16.11, pp. 2415–2458. DOI: 10.1162/0899766041941925.

Aires, Filipe, Catherine Prigent, and William B. Rossow (2004b). « Temporal interpolation of global surface skin temperature diurnal cycle over land under clear and cloudy conditions. » In: *Journal of Geophysical Research: Atmospheres* 109.D4. DOI: 10.1029/2003JD003527.

Aires, Filipe, Catherine Prigent, and William B. Rossow (2005). « Soil moisture at a global scale: 2. Global statistical relationships. » In: *Journal of Geophysical Research: Atmospheres* 110. DOI: 10.1029/2004JD005094.

Aires, Filipe, Catherine Prigent, William B. Rossow, and Mathew Rothstein (2001). « A new neural network approach including first guess for retrieval of atmospheric water vapor, cloud liquid water path, surface temperature, and emissivities over land from satellite microwave observations. » In: *Journal of Geophysical Research* 106, pp. 14887–14907. DOI: 10.1029/2001JD900085.

Aires, Filipe, Wiliam B. Rossow, Noëlle Scott, and Alain Chédin (2002b). « Remote sensing from the Infrared Atmospheric Sounding Interferometer instrument 1. Compression, denoising, and first-guess retrieval algorithms. » In: *Journal of Geophysical Research: Atmospheres* 107, pages = ACH 6–1–ACH 6–15. DOI: 10.1029/2001JD000955.

Aires, Filipe, William B. Rossow, and Alain Chédin (2002). « Rotation of EOFs by the Independent Component Analysis: Toward a solution of the mixing problem in the decomposition of geophysical time series. » In: *Journal of the Atmospheric Sciences* 59.1, pp. 111–123. DOI: 10.1175/1520-0469(2002)059<0111:ROEBTI>2.0.CO;2.

Aires, Filipe, Peter Weston, Patricia de Rosnay, and David Fairbairn (2021). « Statistical approaches to assimilate ASCAT soil moisture information — I. Methodologies and first assessment. » In: *Quarterly Journal of the Royal Meteorological Society* 147.736, pp. 1823–1852. DOI: 10.1002/qj.3997.

Andler, Daniel (2023). *Intelligence artificielle, intelligence humaine: la double énigme*. Gallimard.

August, Thomas, Dieter Klaes, Peter Schlüssel, Tim Hultberg, Marc Crapeau, Arlindo Arriaga, Anne O'Carroll, Dorothee Coppens, Rose Munro, and Xavier Calbet (2012). « IASI on Metop-A: Operational Level 2 retrievals after five years in orbit. » In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 113.11, pp. 1340–1371. DOI: 10.1016/j.jqsrt.2012.02.028.

Azcarate, Aïda, Alexander Barth, Damien Sirjacobs, Fabian Lenartz, and Jean-Marie Beckers (2011). « Data Interpolating Empirical Orthogonal Functions (DINEOF): a tool for geophysical data analyses. » In: *Mediter. Mar. Sci.* 12. DOI: 10.12681/mms.64.

Baccini, Michela et al. (2008). « Heat effects on mortality in 15 European cities. » In: *Epidemiology*, pp. 711–719. URL: http://www.jstor.org/stable/25662618.

Bi, Kaifeng, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian (2022). « Pangu-weather: A 3D high-resolution model for fast and accurate global weather forecast. » In: *arXiv preprint arXiv:2211.02556*.

Bindi, Marco and Jørgen E Olesen (2011). « The responses of agriculture in Europe to climate change. » In: *Regional Environmental Change* 11.1, pp. 151–158. DOI: 10.1007/s10113-010-0173-x.

Blackwell, William J (2005). « A neural-network technique for the retrieval of atmospheric temperature and moisture profiles from high spectral resolution sounding data. » In: *IEEE Transactions on Geoscience and Remote Sensing* 43.11, pp. 2535–2546. DOI: 10.1109/TGRS.2005.855071.

Blackwell, William J, Frederick W Chen, Laura G Jairam, and Michael Pieper (2008). « Neural network estimation of atmospheric profiles using AIRS/IASI/AMSU data in the presence of clouds. » In: *IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium*. Vol. 1. IEEE, pp. I–122. DOI: 10.1109/IGARSS.2008.4778808.

Blumstein, Denis, Gilles Chalon, Thierry Carlier, Christian Buil, Philippe Hebert, Thierry Maciaszek, G Ponce, Thierry Phulpin, Bernard Tournier, Denis Simeoni, et al. (2004). « IASI instrument: Technical overview and measured performances. » In: *Infrared Spaceborne Remote Sensing XII*. Vol. 5543. SPIE, pp. 196–207. DOI: 10.1117/12.560907.

Boucher, Eulalie and Filipe Aires (2023a). « Improving remote sensing of extreme events with machine learning: land surface temperature retrievals from IASI observations. » In: *Environmental Research Letters* 18.2, p. 024025. DOI: 10.1088/1748-9326/acb3e3.

Boucher, Eulalie and Filipe Aires (2023b). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part II – Assessment. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1593–1611. DOI: 10.1002/qj.4472.

Boucher, Eulalie, Filipe Aires, and Marie Doutriaux-Boucher (2024). « Introducing a new partial Convolutional Neural Network for IASI cloud classification [Manuscript submitted for publication]. » In: *Submitted to IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Boucher, Eulalie, Filipe Aires, and Victor Pellet (2023). « Towards a new generation of artificial-intelligence-based infrared atmospheric sounding interferometer retrievals of surface temperature: Part I – Methodology. » In: *Quarterly Journal of the Royal Meteorological Society* 149, pp. 1180–1196. DOI: 10.1002/qj.4447.

Boucher, Olivier (2015). *Atmospheric aerosols*. Springer.

Bouillon, M., S. Safieddine, S. Whitburn, L. Clarisse, F. Aires, V. Pellet, O. Lezeaux, N. A. Scott, M. Doutriaux-Boucher, and C. Clerbaux (2022). « Time evolution of temperature profiles retrieved from 13 years of infrared atmospheric sounding interferometer (IASI) data using an artificial neural network. » In: *Atmospheric*

Measurement Techniques 15.6, pp. 1779–1793. DOI: 10.5194/amt-15-1779-2022.

Bouillon, Marie (2021). « Températures atmosphériques homogènes dérivées des observations satellitaires IASI : restitution, variations spatio-temporelles et événements extrêmes. » Theses. Sorbonne Université.

Calbet, X and P Schlüssel (2006). « Analytical estimation of the optimal parameters for the EOF retrievals of the IASI Level 2 product processing facility and its application using AIRS and ECMWF data. » In: *Atmospheric Chemistry and Physics* 6.3, pp. 831–846. DOI: 10.5194/acp-6-831-2006.

Cayla, Francois and Pascale Javelle (1995). « IASI: Instrument overview. » In: *Infrared Spaceborne Remote Sensing III*. Vol. 2553. SPIE, pp. 316–328. DOI: 10.1117/12.221368.

Chen, Yu-hsin, Ignacio Lopez-Moreno, Tara N. Sainath, Mirkó Visontai, Raziel Álvarez, and Carolina Parada (2015). « Locally-connected and convolutional neural networks for small footprint speaker recognition. » In: *INTERSPEECH*.

Chepfer, Hélène, Laurence Picon, Marine Bonazzola, Hélène Brogniez, Marjolaine Chiriaco, and Solène Turquety (2023). *Principes de la télédétection: Applications à l'observation du système climatique terrestre*. Dunod.

Cimini, D, F Romano, E Ricciardelli, and V Cuomo (2006). « On the role of surface emissivity in polar night-time cloud detection. » In: *Proceedings of 15th International TOVS Study Conference, Maratea, Italy*.

Cissé, G. et al. (2022). « Health, Wellbeing, and the Changing Structure of Communities. » In: *Climate Change 2022: Impacts, Adaptation, and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change [H.-O. Pörtner, D.C. Roberts, M. Tignor, E.S. Poloczanska, K. Mintenbeck, A. Alegría, M. Craig, S. Langsdorf, S. Löschke, V. Möller, A. Okem, B. Rama (eds.)]* Cambridge University Press.

Clerbaux, Cathy, Anne Boynard, Lieven Clarisse, Maya George, Juliette Hadji-Lazaro, Hervé Herbin, Daniel Hurtmans, Matthieu Pommier, Ariane Razavi, Solène Turquety, et al. (2009). « Monitoring of atmospheric composition using the thermal infrared IASI/MetOp sounder. » In: *Atmospheric Chemistry and Physics* 9.16, pp. 6041–6054.

Collard, AD and AP McNally (2009). « The assimilation of infrared atmospheric sounding interferometer radiances at ECMWF. » In: *Quarterly Journal of the Royal Meteorological Society* 135.641, pp. 1044–1058. DOI: 10.1002/qj.410.

Crevoisier, Cyril et al. (2014). « Towards IASI-New Generation (IASI-NG): impact of improved spectral resolution and radiometric noise on the retrieval of ther-

modynamic, chemistry and climate variables. » In: *Atmospheric Measurement Techniques* 7.12, pp. 4367–4385. DOI: 10.5194/amt-7-4367-2014.

Dalagnol, Ricardo, Fabien Hubert Wagner, Lênio Soares Galvão, Daniel Braga, Fiona Osborn, Polyanna da Conceição Bispo, Matthew Payne, Celso Silva Junior, Samuel Favrichon, Vinicius Silgueiro, et al. (2023). « Mapping tropical forest degradation with deep learning and Planet NICFI data. » In: *Remote Sensing of Environment* 298, p. 113798. DOI: 10.1016/j.rse.2023.113798.

Dee, D. (2004). « Variational bias correction of radiance data in the ECMWF system. » In: *ECMWF Workshop on Assimilation of High Spectral Resolution Sounders in NWP*, pp. 97–112.

Derrien, M and H Le Gléau (2005). « MSG/SEVIRI cloud mask and type from SAFNWC. » In: *International Journal of Remote Sensing* 26.21, pp. 4707–4732. DOI: 10.1080/01431160500166128.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). « Bert: Pre-training of deep bidirectional transformers for language understanding. » In: *Proceedings of NAACL-HLT, arXiv preprint arXiv:1810.04805*, pp. 4171–4186.

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. (2021). « An image is worth 16x16 words: Transformers for image recognition at scale. » In: *International Conference on Learning Representations (ICLR), arXiv preprint arXiv:2010.11929*.

EUMETSAT (2016). « IASI L2 PPF v6: Validation Report. » In: DOI: EUM/TSS/REP/14/776443.

EUMETSAT (2022). « IASI-A and -B climate data record of all sky temperature and humidity profiles Release 1.1, European Organisation for the Exploitation of Meteorological Satellites. » In: DOI: 10.15770/EUM_SEC_CLM_0063.

EUMETSAT (2022a). *IASI Principal Components Scores Fundamental Data Record Release 1 - Metop-A and -B, European Organisation for the Exploitation of Meteorological Satellites*. DOI: 10.15770/EUM_SEC_CLM_0084.

EUMETSAT (2022b). *Optimal Cloud Analysis Climate Data Record Release 1 - MSG - 0 degree, European Organisation for the Exploitation of Meteorological Satellites*. DOI: 10.15770/EUM_SEC_CLM_0049.

EUMETSAT Doutriaux-Boucher Marie, August Thomas (2020). « IASI-A and -B climate data record of all sky temperature and humidity profiles Release 1, European Organisation for the Exploitation of Meteorological Satellites. » In: DOI: 10.15770/EUM_SEC_CLM_0027.

Eyre, J. R., W. Bell, J. Cotton, S. J. English, M. Forsythe, S. B. Healy, and E. G. Pavelin (2022). « Assimilation of satellite data in numerical weather prediction. Part II: Recent years. » In: *Quarterly Journal of the Royal Meteorological Society* 148.743, pp. 521–556. DOI: 10.1002/qj.4228.

Favrichon, Samuel, Catherine Prigent, Carlos Jimenez, and Filipe Aires (2019). « Detecting cloud contamination in passive microwave satellite measurements over land. » In: *Atmospheric Measurement Techniques* 12.3, pp. 1531–1543. DOI: 10.5194/amt-12-1531-2019.

García-Sobrino, Joaquín, Joan Serra-Sagristà, and Joan Bartrina-Rapesta (2017). « Hyperspectral IASI L1C data compression. » In: *Sensors* 17.6, p. 1404. DOI: 10.3390/s17061404.

Gawlikowski, Jakob, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. (2023). « A survey of uncertainty in deep neural networks. » In: *Artificial Intelligence Review* 56.Suppl 1, pp. 1513–1589.

Grillakis, Manolis G (2019). « Increase in severe and extreme soil moisture droughts for Europe under climate change. » In: *Science of The Total Environment* 660, pp. 1245–1255. DOI: 10.1016/j.scitotenv.2019.01.001.

Hadji-Lazaro, Juliette, Cathy Clerbaux, and Sylvie Thiria (1999). « An inversion algorithm using neural networks to retrieve atmospheric CO total columns from high-resolution nadir radiances. » In: *Journal of Geophysical Research: Atmospheres* 104.D19, pp. 23841–23854. DOI: 10.1029/1999JD900431.

Harley, Adam W., Konstantinos G. Derpanis, and Iasonas Kokkinos (2017). « Segmentation-aware convolutional networks using local attention masks. » In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 5038–5047.

Hartmann, Dennis L, Maureen E Ockert-Bell, and Marc L Michelsen (1992). « The effect of cloud type on Earth's energy balance: Global analysis. » In: *Journal of Climate* 5.11, pp. 1281–1304. DOI: 10.1175/1520-0442(1992)005<1281:TEOCTO>2.0.CO;2.

Hastie, Trevor, Robert Tibshirani, and Jerome H Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer, 746 pp.

Haykin, Simon (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 842 pp.

Hersbach, Hans et al. (2020). « The ERA5 global reanalysis. » In: *Quarterly Journal of the Royal Meteorological Society* 146.730, pp. 1999–2049. DOI: 10.1002/qj.3803.

Hilton, Fiona, Andrew Collard, Vincent Guidard, Roger Randriamampianina, and Marc Schwaerz (2009). « Assimilation of IASI radiances at European NWP centres. » In: *Proceedings of Workshop on the assimilation of IASI data in NWP, ECMWF, Reading, UK*, pp. 39–48.

Hilton, Fiona et al. (2012). « Hyperspectral Earth observation from IASI: Five years of accomplishments. » In: *Bulletin of the American Meteorological Society* 93.3, pp. 347–370. DOI: 10.1175/BAMS-D-11-00027.1.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). « Long short-term memory. » In: *Neural Computation* 9.8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Huang, Hung-Lung and Paolo Antonelli (2001). « Application of Principal Component Analysis to high-resolution infrared measurement compression and retrieval. » In: *Journal of Applied Meteorology* 40.3, pp. 365–388. DOI: 10.1175/1520-0450(2001)040<0365:AOPCAT>2.0.CO;2.

IASI Level 1C Climate Data Record Release 1 - Metop-A (2018). DOI: 10.15770/EUM_SEC_CLM_0014.

Iglesias, Ana and Luis Garrote (2015). « Adaptation strategies for agricultural water management under climate change in Europe. » In: *Agricultural Water Management* 155, pp. 113–124. DOI: 10.1016/j.agwat.2015.03.014.

Jaiswal, Akriti, A Krishnama Raju, and Suman Deb (2020). « Facial emotion detection using deep learning. » In: *2020 International Conference for Emerging Technology (INCET)*. IEEE, pp. 1–5.

Jolliffe, I.T. (1986). *Principal Component Analysis*. Springer Verlag.

Karagulian, Federico, Lieven Clarisse, Cathy Clerbaux, Alfred J Prata, Daniel Hurtmans, and Pierre-François Coheur (2010). « Detection of volcanic SO_2, ash, and H_2SO_4 using the Infrared Atmospheric Sounding Interferometer (IASI). » In: *Journal of Geophysical Research: Atmospheres* 115.D2. DOI: 10.1029/2009JD012786.

Kattenborn, Teja, Jana Eichel, and Fabian Ewald Fassnacht (2019). « Convolutional Neural Networks enable efficient, accurate and fine-grained segmentation of plant species and communities from high-resolution UAV imagery. » In: *Scientific reports* 9.1, p. 17656. DOI: 10.1038/s41598-019-53797-9.

Kazantzidis, Andreas, Panagiotis Tzoumanikas, Alkiviadia F Bais, Spiros Fotopoulos, and George Economou (2012). « Cloud detection and classification with the use of whole-sky ground-based images. » In: *Atmospheric Research* 113, pp. 80–88. DOI: 10.1016/j.atmosres.2012.05.005.

Kingma, Diederik P and Jimmy Ba (2014). « Adam: A method for stochastic optimization. » In: *International Conference on Learning Representations (ICLR) 2015, arXiv preprint arXiv:1412.6980*.

Kolassa, J., Filipe Aires, Jan Polcher, Catherine Prigent, Carlos Jimenez, and José Pereira (2013). « Soil moisture retrieval from multi-instrument observations: Information content analysis and retrieval methodology. » In: *Journal of Geophysical Research* 118. DOI: 10.1029/2012JD018150.

Kononenko, Igor (1989). « Bayesian neural networks. » In: *Biological Cybernetics* 61.5, pp. 361–370.

Lam, Lisa, Maya George, Sébastien Gardoll, Sarah Safieddine, Simon Whitburn, and Cathy Clerbaux (2023a). « Tropical cyclone detection from the thermal infrared sensor IASI data using the deep learning model YOLOv3. » In: *Atmosphere* 14.2, p. 215. DOI: 10.3390/atmos14020215.

Lam, Remi et al. (2023b). « Learning skillful medium-range global weather forecasting. » In: *Science* 382.6677, pp. 1416–1421. DOI: 10.1126/science.adi2336.

Liu, Guilin, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro (2018). « Image inpainting for irregular holes using partial convolutions. » In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 85–100.

Lütjens, Björn, Michael Everett, and Jonathan P How (2019). « Safe reinforcement learning with model uncertainty estimates. » In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 8662–8668. DOI: 10.1109/ICRA.2019.8793611.

Ma, Yuchi, Zhou Zhang, Yanghui Kang, and Mutlu Özdoğan (2021). « Corn yield prediction and uncertainty analysis based on remotely sensed variables using a Bayesian neural network approach. » In: *Remote Sensing of Environment* 259, p. 112408. DOI: 10.1016/j.rse.2021.112408.

Mahajan, Seema and Bhavin Fataniya (2020). « Cloud detection methodologies: Variants and development — A review. » In: *Complex & Intelligent Systems* 6, pp. 251–261. DOI: 10.1007/s40747-019-00128-0.

Malmgren-Hansen, David, Valero Laparra, Allan Aasbjerg Nielsen, and Gustau Camps-Valls (2019). « Statistical retrieval of atmospheric profiles with deep convolutional neural networks. » In: *ISPRS Journal of Photogrammetry and Remote Sensing* 158, pp. 231–240. DOI: 10.1016/j.isprsjprs.2019.10.002.

Martins, João P. A., Isabel F. Trigo, Nicolas Ghilain, Carlos Jimenez, Frank-M. Göttsche, Sofia L. Ermida, Folke-S. Olesen, Françoise Gellens-Meulenberghs, and Alirio Arboleda (2019). « An all-weather land surface temperature product based on MSG/SEVIRI observations. » In: *Remote Sensing* 11.24. DOI: 10.3390/rs11243044.

Masiello, Guido, Carmine Serio, and Paolo Antonelli (2012). « Inversion for atmospheric thermodynamical parameters of IASI data in the principal components space. » In: *Quarterly Journal of the Royal Meteorological Society* 138.662, pp. 103–117. DOI: 10.1002/qj.909.

Mathieu, Jordane (2018). « Modèles d'impact statistiques en agriculture : de la prévision saisonnière à la prévision à long terme, en passant par les estimations annuelles. » Theses. Université Paris Sciences et Lettres.

Matricardi, Marco and AP McNally (2014). « The direct assimilation of principal components of IASI spectra in the ECMWF 4D-Var. » In: *Quarterly Journal of the Royal Meteorological Society* 140.679, pp. 573–582. DOI: 10.1002/qj.2156.

Mcculloch, Warren and Walter Pitts (1943). « A logical calculus of ideas immanent in nervous activity. » In: *Bulletin of Mathematical Biophysics* 5, pp. 127–147. DOI: 10.1007/BF02478259.

Murphy, D (2013). « EUMETSAT geostationary meteorological satellite programs. » In: *Handbook of Satellite Applications, Springer, New York*, pp. 991–1019.

Nandy, Jay, Wynne Hsu, and Mong Li Lee (2020). « Towards maximizing the representation gap between in-domain & out-of-distribution examples. » In: *Advances in neural information processing systems* 33, pp. 9239–9250.

Oudin Åström, Daniel, Bertil Forsberg, Kristie L Ebi, and Joacim Rocklöv (2013). « Attributing mortality from extreme temperatures to climate change in Stockholm, Sweden. » In: *Nature Climate Change* 3.12, pp. 1050–1054. DOI: 10.1038/nclimate2022.

Parracho, Ana Claudia, Sarah Safieddine, Olivier Lezeaux, Lieven Clarisse, Simon Whitburn, Maya George, Pascal Prunet, and Cathy Clerbaux (2021). « IASI-derived sea surface temperature dataset for climate studies. » In: *Earth and Space Science* 8.5, e2020EA001427. DOI: 10.1029/2020EA001427.

Patz, Jonathan A, Howard Frumkin, Tracey Holloway, Daniel J Vimont, and Andrew Haines (2014). « Climate change: challenges and opportunities for global health. » In: *Jama* 312.15, pp. 1565–1580. DOI: 10.1001/jama.2014.13186.

Paul, M., F. Aires, Christophe Prigent, I.F. Trigo, and Frédéric Bernardo (2012). « An innovative physical scheme to retrieve simultaneously surface temperature and emissivities using high spectral infrared observations from IASI. » In: *Journal of Geophysical Research: Atmospheres* 117.11, p. D11302. DOI: 10.1029/2011JD017296.

Paul, Maxime (Sept. 2013). « Synergie infrarouge et micro-onde pour la restitution atmosphérique. » Theses. Université Pierre et Marie Curie - Paris VI.

Pellet, Victor and Filipe Aires (2018). « Bottleneck channels algorithm for satellite data dimension reduction: A case study for IASI. » In: *IEEE Transactions on Geoscience and Remote Sensing* 56.10, pp. 6069–6081. DOI: 10.1109/TGRS.2018.2830123.

Pfreundschuh, S., P. Eriksson, D. Duncan, B. Rydberg, N. Håkansson, and A. Thoss (2018). « A neural network approach to estimating a posteriori distributions of Bayesian retrieval problems. » In: *Atmospheric Measurement Techniques* 11.8, pp. 4627–4643. DOI: 10.5194/amt-11-4627-2018.

Phulpin, Thierry, Didier Renaut, Herve Roquet, and Claude Camy-Peyret (2024). *Satellites for Atmospheric Sciences 1: Meteorology, Climate and Atmospheric Composition*. Wiley. ISBN: 9781789451405. URL: https://books.google.co.uk/books?id=kmzrEAAAQBAJ.

Radford, Alec, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. (2018). « Improving language understanding by generative pre-training. » In.

Raghu, Maithra, Katy Blumer, Rory Sayres, Ziad Obermeyer, Bobby Kleinberg, Sendhil Mullainathan, and Jon Kleinberg (2019). « Direct uncertainty prediction for medical second opinions. » In: *International Conference on Machine Learning*. PMLR, pp. 5281–5290.

Rodgers, Clive D (2000). *Inverse methods for atmospheric sounding: theory and practice*. Vol. 2. World scientific.

Rodriguez-Fernandez, Nemesio, Patricia Rosnay, Clement Albergel, Philippe Richaume, Filipe Aires, Catherine Prigent, and Yann Kerr (2019). « SMOS neural network soil moisture data assimilation in a land surface model and atmospheric impact. » In: *Remote Sensing* 11, p. 1334. DOI: 10.3390/rs11111334.

Rosenblatt, F. (1958). « The perceptron: A probabilistic model for information storage and organization in the brain. » In: *Psychological Review* 65.6, pp. 386–408. DOI: 10.1037/h0042519.

Rossow, William B. and Robert A. Schiffer (1999). « Advances in understanding clouds from ISCCP. » In: *Bulletin of the American Meteorological Society* 80.11, pp. 2261–2288. DOI: 10.1175/1520-0477(1999)080<2261:AIUCFI>2.0.CO;2.

Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams (1986). « Learning representations by back-propagating errors. » In: *Nature* 323, pp. 533–536. DOI: 10.1038/323533a0.

Rußwurm, Marc, Mohsin Ali, Xiao Xiang Zhu, Yarin Gal, and Marco Körner (2020). « Model and data uncertainty for satellite time series forecasting with deep recurrent models. » In: *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, pp. 7025–7028.

Safieddine, Sarah et al. (2020). « Artificial neural networks to retrieve land and sea skin temperature from IASI. » In: *Remote Sensing* 12.17. DOI: 10.3390/rs12172777.

Schwartz, Martin et al. (2024). « High-resolution canopy height map in the Landes forest (France) based on GEDI, Sentinel-1, and Sentinel-2 data with a deep learning approach. » In: *International Journal of Applied Earth Observation and Geoinformation* 128, p. 103711. DOI: 10.1016/j.jag.2024.103711.

Seethala, C., J. F. Meirink, Á. Horváth, R. Bennartz, and R. Roebeling (2018). « Evaluating the diurnal cycle of South Atlantic stratocumulus clouds as observed by MSG SEVIRI. » In: *Atmospheric Chemistry and Physics* 18.17, pp. 13283–13304. DOI: 10.5194/acp-18-13283-2018.

Shao, Zhenfeng, Yin Pan, Chunyuan Diao, and Jiajun Cai (2019). « Cloud detection in remote sensing images based on multiscale features-convolutional neural network. » In: *IEEE Transactions on Geoscience and Remote Sensing* 57.6, pp. 4062–4076. DOI: 10.1109/TGRS.2018.2889677.

Singh, Satya P, Lipo Wang, Sukrit Gupta, Haveesh Goli, Parasuraman Padmanabhan, and Balázs Gulyás (2020). « 3D deep learning on medical images: A review. » In: *Sensors* 20.18, p. 5097. DOI: 10.3390/s20185097.

Sobrino, JA and M Romaguera (2004). « Land surface temperature retrieval from MSG1-SEVIRI data. » In: *Remote Sensing of Environment* 92.2, pp. 247–254. DOI: 10.1016/j.rse.2004.06.009.

Stillinger, Timbo, Dar Roberts, Natalie Collar, and Jeff Dozier (2019). « Cloud masking for Landsat 8 and MODIS Terra over snow-covered terrain: Error analysis and spectral similarity between snow and cloud. » In: *Water Resources Research* 55. DOI: 10.1029/2019WR024932.

Stubenrauch, Claudia J, Artem G Feofilov, Sofia E Protopapadaki, and Raymond Armante (2017). « Cloud climatologies from the infrared sounders AIRS and IASI: Strengths and applications. » In: *Atmospheric Chemistry and Physics* 17.22, pp. 13625–13644. DOI: 10.5194/acp-17-13625-2017.

Stubenrauch, Claudia J., Ralph Holz, Alain Chédin, David L. Mitchell, and Anthony J. Baran (1999). « Retrieval of cirrus ice crystal sizes from 8.3 and 11.1 μm emissivities determined by the improved initialization inversion of TIROS-N Operational Vertical Sounder observations. » In: *Journal of Geophysical Research: Atmospheres* 104.D24, pp. 31793–31808. DOI: 10.1029/1999JD900828.

Talagrand, O (2014). « 4D-VAR: four-dimensional variational assimilation. » In: *Advanced Data Assimilation for Geosciences: Lecture Notes of the Les Houches School of Physics: Special Issue*, p. 1.

Trigo, Isabel F, Sofia L Ermida, João PA Martins, Célia M Gouveia, Frank-M Göttsche, and Sandra C Freitas (2021). « Validation and consistency assessment of land surface temperature from geostationary and polar orbit platforms: SEVIRI/MSG and AVHRR/Metop. » In: *ISPRS Journal of Photogrammetry and Remote Sensing* 175, pp. 282–297.

Trigo, Isabel F., Isabel T. Monteiro, Folke Olesen, and Ewa Kabsch (2008). « An assessment of remotely sensed land surface temperature. » In: *Journal of Geophysical Research: Atmospheres* 113.D17. DOI: 10.1029/2008JD010035.

Trigo, Isabel F. et al. (2011). « The Satellite Application Facility for land surface analysis. » In: *International Journal of Remote Sensing* 32.10, pp. 2725–2744. DOI: 10.1080/01431161003743199.

Vasudevan, Vishal Thanvantri, Abhinav Sethy, and Alireza Roshan Ghias (2019). « Towards better confidence estimation for neural models. » In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 7335–7339.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin (2017). « s. » In: *31st Conference on Advances in Neural Information Processing Systems*. Vol. 30.

Wagner, Wolfgang, Guido Lemoine, and Helmut Rott (1999). « A method for estimating soil moisture from ERS scatterometer and soil data. » In: *Remote Sensing of Environment* 70.2, pp. 191–207. DOI: 10.1016/S0034-4257(99)00036-X.

Watts, PD, R Bennartz, and F Fell (2011). « Retrieval of two-layer cloud properties from multispectral observations using optimal estimation. » In: *Journal of Geophysical Research: Atmospheres* 116.D16. DOI: 10.1029/2011JD015883.

Weilnhammer, Veronika, Jonas Schmid, Isabella Mittermeier, Fabian Schreiber, Linmiao Jiang, Vedran Pastuhovic, Caroline Herr, and Stefanie Heinze (2021). « Extreme weather events in Europe and their health consequences – A systematic review. » In: *International Journal of Hygiene and Environmental Health* 233, p. 113688. DOI: 10.1016/j.ijheh.2021.113688.

Wilson, Andrew G and Pavel Izmailov (2020). « Bayesian deep learning and a probabilistic perspective of generalization. » In: *Advances in Neural Information Processing Systems* 33, pp. 4697–4708.

Xie, Fengying, Mengyun Shi, Zhenwei Shi, Jihao Yin, and Danpei Zhao (2017). « Multilevel cloud detection in remote sensing images based on deep learning. » In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10.8, pp. 3631–3640. DOI: 10.1109/JSTARS.2017.2686488.

Xu, Dongmei, Zhiquan Liu, Xiang-Yu Huang, Jinzhong Min, and Hongli Wang (2013). « Impact of assimilating IASI radiance observations on forecasts of two tropi-

cal cyclones. » In: *Meteorology and Atmospheric Physics* 122.1, pp. 1–18. DOI: 10.1007/s00703-013-0276-2.

Yarin, Gal (1998). « Uncertainty in deep learning. » Thesis. University of Cambridge.

Zhang, Meng, Hui Lin, Guangxing Wang, Hua Sun, and Jing Fu (2018). « Mapping paddy rice using a convolutional neural network (CNN) with Landsat 8 datasets in the Dongting Lake area, China. » In: *Remote Sensing* 10.11. DOI: 10.3390/rs10111840.

Zhao, Feng, Rui Sun, Liheng Zhong, Ran Meng, Chengquan Huang, Xiaoxi Zeng, Mengyu Wang, Yaxin Li, and Ziyang Wang (2022). « Monthly mapping of forest harvesting using dense time series Sentinel-1 SAR imagery and deep learning. » In: *Remote Sensing of Environment* 269, p. 112822. DOI: 10.1016/j.rse.2021.112822.