amU
Aix Marseille Université

NNT : 2024AIXM0371

THÈSE DE DOCTORAT

Soutenue à Aix-Marseille Université
le 12 novembre 2024 par

Alexis JUVEN

Apprentissage automatique assisté par simulation pour la mesure infrarouge dans un tokamak

Discipline

Physique et Sciences de la Matière

Spécialité

Optique, Photonique et Traitement d'Image

École doctorale

ED 352 – Physique et Sciences de la Matière

Laboratoire/Partenaires de recherche

Commissariat à l'Énergie Atomique et aux énergies alternatives (CEA)

Institut de Recherche sur la Fusion par confinement Magnétique (IRFM)

Institut Fresnel

Composition du jury

Benoit CRESPIN Professeur, Université de Limoges, XLIM	Rapporteur
Laurent SIMON Professeur, Bordeaux INP, Université de Bordeaux, LaBRI	Rapporteur
Laure BLANC-FÉRAUD Directrice de Recherche CNRS, Laboratoire i3S de Sophia Antipolis	Présidente du jury
Roberto MIORELLI Docteur, Chercheur CEA, CEA/DRT/LIST	Examinateur
Renaud DEJARNAC Docteur, Chercheur à l'Institut de Physique de Plasma de Prague	Examinateur
Marie-Hélène AUMEUNIER Directrice de Recherche CEA, HDR, CEA/-DRF/IRFM	Directrice de thèse
Julien MAROT Maître de Conférences, HDR, Aix-Marseille Université, Institut Fresnel	Co-directeur de thèse

cea irfm

INSTITUT FRESNEL

Affidavit

Je soussigné, Alexis Juven, déclare par la présente que le travail présenté dans ce manuscrit est mon propre travail, réalisé sous la direction scientifique de Marie-Hélène Aumeunier et Julien Marot, dans le respect des principes d'honnêteté, d'intégrité et de responsabilité inhérents à la mission de recherche. Les travaux de recherche et la rédaction de ce manuscrit ont été réalisés dans le respect à la fois de la charte nationale de déontologie des métiers de la recherche et de la charte d'Aix-Marseille Université relative à la lutte contre le plagiat.

Ce travail n'a pas été précédemment soumis en France ou à l'étranger dans une version identique ou similaire à un organisme examinateur.

Fait à Saint-Paul-lez-Durance le 09/09/2024



Cette œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International.

Publications et conférences

Liste des publications réalisées dans le cadre du projet de thèse

1. Alexis JUVEN, Marie-Hélène AUMEUNIER, Romain BRUNET, Mickaël Le BOHEC, Mouloud ADEL, Roberto MIORELLI, Xavier ARTUSI et Christophe REBOUD. « Temperature Estimation in Fusion Devices using Machine Learning techniques on Infrared Specular Synthetic Data ». *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. Juin 2022, p. 1-5. DOI : 10.1109/IVMSP54334.2022.9816270
2. Marie-Hélène AUMEUNIER, Mickaël LE BOHEC, Romain BRUNET, A. JUVEN, M. ADEL, X. ARTUSI, R. MIORELLI, C. REBOUD et F. RIGOLLET. « Development of inverse methods for infrared thermography in fusion devices ». *Nuclear Materials and Energy* 33 (oct. 2022), p. 101231. DOI : 10.1016/j.nme.2022.101231
3. Romain BRUNET, Marie-Hélène AUMEUNIER, Roberto MIORELLI, Alexis JUVEN, C. REBOUD, X. ARTUSI et M. Le BOHEC. « Infrared measurement synthetic database for inverse thermography model based on deep learning ». *Fusion Engineering and Design* 192 (juill. 2023), p. 113598. DOI : 10.1016/j.fusengdes.2023.113598
4. Alexis JUVEN, Marie-Hélène AUMEUNIER et Julien MAROT. « U-Net for temperature estimation from simulated infrared images in tokamaks ». *Nuclear Materials and Energy* 38 (mars 2024), p. 101562. DOI : 10.1016/j.nme.2023.101562
5. M.-H. AUMEUNIER, A. JUVEN, J. GERARDIN, C-M. B. CISSE, S. PAMELA, R. MIORELLI, C. REBOUD, F. RETAILLEAU, J. MAROT, F. RIGOLLET et L. MAROT. « Surface temperature measurement from infrared synthetic diagnostic in preparation for ITER operations ». *Nuclear Fusion* 64.8 (2024), p. 086044. DOI : 10.1088/1741-4326/ad5a1f
6. Alexis JUVEN, Marie-Hélène AUMEUNIER et Julien MAROT. « Generative models and simulation to assess uncertainties for tokamak infrared thermography ». *2024 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. 2024. DOI : 10.1109/MLSP58920.2024.10734728
7. Q. TICHIT, M.-H. AUMEUNIER, Y. CORRE, J. GERARDIN, L. DUBUS, J. GASPAR, J.P. GUNN, A. GROSJEAN, R. DEJARNAC, A. PODOLNÍK, A. JUVEN, K. KRIEGER et E. TSITRONE. « Plasma heat load in the toroidal gaps of the ITER-grade plasma facing units in WEST tokamak ». *Nuclear Materials and Energy* (2024). DOI : 10.1016/j.nme.2025.101899

Participation aux conférences au cours de la période de thèse

1. *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. Juin 2022 (Oral). Orateur : Alexis JUVEN. Co-auteurs : Marie-Hélène AUMEUNIER, Romain BRUNET, Mickaël LE BOHEC, Mouloud ADEL, Roberto MIORELLI, Xavier ARTUSI et Christophe REBOUD.
2. *2024 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. Septembre 2024 (Poster). Orateur : Alexis JUVEN. Co-auteurs : Marie-Hélène AUMEUNIER et Julien MAROT.

Résumé

La protection de la première paroi est un enjeu majeur pour les opérations des machines de fusion thermonucléaire. Les composants face au plasma sont soumis à des flux de chaleur élevés (jusqu'à 10-20 MW/m^2), conduisant à des températures de 200 à 3600 °C. Dans la plupart des machines en opération et pour la future machine internationale ITER, la surveillance de ces composants est principalement assurée par un réseau de caméras infrarouges, directement relié au système de contrôle plasma.

Cependant, l'interprétation de ces images est compliquée par la présence de parois métalliques, d'émissivité basse et variable, générant un flux parasite dû aux multiples réflexions sur leur surface. Cette perturbation accroît le risque de sous-estimation de la température des parois, ce qui pourrait compromettre la sûreté de fonctionnement, ou au contraire, de surestimation, entraînant potentiellement l'arrêt non justifié de la machine. L'objectif de cette thèse est de développer des méthodes numériques capables de traiter ces phénomènes perturbateurs et d'estimer précisément la température des composants à partir des images infrarouges. Ces méthodes sont appliquées et testées sur la machine WEST. Pour résoudre ce problème inverse, l'approche adoptée durant la thèse est de développer un réseau de neurones profond entraîné à partir d'un jeu de données constitué exclusivement d'images simulées. Ainsi, la thèse s'est divisée en deux branches principales.

1. **Modélisation et simulation des phénomènes physiques en jeu (modèle direct).** Cela inclut la modélisation de la scène thermique d'un tokamak en opération, le rayonnement et la réflexion infrarouge, ainsi que le fonctionnement de la caméra. Cette partie a mené au développement d'un logiciel de lancer de rayon infrarouge par Monte Carlo, capable de simuler des images synthétiques prenant en compte les propriétés de réflexion complexes des matériaux (BRDF), et utilisant l'accélération GPU pour générer une grande quantité de données en un temps raisonnable. Ce développement est valorisé par son intégration dans une plateforme logicielle multiphysique dédiée au contrôle non destructif (CIVA), ainsi que dans la plateforme de modélisation et d'analyse d'ITER (IMAS).
2. **Utilisation des techniques d'apprentissage pour développer un modèle inverse.** Un jeu de données conséquent de scénarios a été généré grâce au modèle direct rapide précédemment développé, permettant d'obtenir une association entre image infrarouge capturée et température réelle des surfaces. Ce jeu de données a été utilisé pour entraîner un réseau de type U-Net à estimer les températures de surface à partir d'une image. Les performances de ce réseau sur des données simulées de test répondent aux spécifications d'ITER, avec des erreurs d'estimation sous la barre des 10 %. Le réseau présente aussi des résultats qualitatifs encourageants sur des images expérimentales de la machine WEST. Enfin, pour associer un niveau de confiance aux prédictions, l'utilisation de modèles non déterministes (de type génératifs bayésiens) est également étudiée. Ces modèles ont permis de gérer la nature mal posée des problèmes inverses (inhérents à la mesure de température par thermographie) ainsi que les incertitudes de mesure expérimentales en fournissant un intervalle de confiance autour de chaque prédiction de température.

Mots-clés : Apprentissage Automatique, Lancer de Rayons, Problème Inverse, Imagerie Infrarouge, Fusion Nucléaire

Abstract

The protection of the first wall is a major issue for the operations of thermonuclear fusion machines. The Plasma Facing Components are indeed subjected to high heat fluxes (up to 10-20 MW/m^2), leading to high temperatures ranging from 200 to 3600 °C. In most operational machines and for the future international machine ITER, the monitoring of these components is mainly ensured by a network of infrared cameras, directly connected to the plasma control system.

However, the interpretation of these images is complicated by the presence of metallic walls, with low and variable emissivity, generating parasitic flux in the image due to multiple reflections on their surfaces. This measurement disturbance increases the risk of underestimating the wall temperatures, which could compromise operational safety, or conversely, overestimating this temperature, potentially leading to the unnecessary shutdown of the machine. The objective of this PhD thesis is to develop numerical methods capable of addressing these perturbative phenomena and accurately estimating the temperature of the components from infrared images. These methods are applied and tested on the WEST machine. To solve this inverse problem, the approach taken during the PhD thesis is to develop a deep neural network trained from a dataset of simulated images. Thus, the PhD thesis was divided into two main branches.

1. **Modeling and simulation of involved physical phenomena (forward model).** This includes modeling the thermal scene of an operational tokamak, infrared radiation and reflection, as well as the functioning of the camera. This part led to the development of a Monte Carlo infrared ray-tracing software, capable of simulating synthetic images taking into account the complex reflection properties of materials (BRDF), and using GPU acceleration to generate a large amount of data in a reasonable time. This new development is valued through its integration within a multi-physics software platform dedicated to non-destructive testing (CIVA), as well as its integration in the ITER Integrated Modelling & Analysis Suite (IMAS).
2. **Using learning techniques to develop an inverse model from simulated data.** A large dataset of scenarios was generated thanks to the previously developed fast forward model, thus allowing obtaining an association between captured infrared images and the actual surface temperatures. This dataset was used to train a neural network (U-Net) to estimate surface temperatures from an image. The performance of this network on simulated test data meets the ITER specifications, with estimation errors below 10%. The network also shows encouraging qualitative results on experimental images from the WEST machine. Finally, to associate a confidence level with the predictions, the use of non-deterministic (Bayesian generative) models was also studied. These models allowed managing the ill-posed nature of inverse problems (inherent to temperature measurement by thermography) as well as experimental measurement uncertainties by providing a confidence interval around each temperature prediction.

Keywords: Machine Learning, Ray Tracing, Inverse Problem, Infrared Imaging, Nuclear Fusion

Glossaire

Abréviations

— **BRDF :** *Bidirectional Reflectance Distribution Function*
— **BVH :** *Bounding Volume Hierarchy*
— **CFP :** Composants Face au Plasma
— **CPU :** *Central Processing Unit*
— **cWGAN-GP :** *conditional Wasserstein Generative Adversarial Network with Gradient Penalty*
— **FiLM :** *Feature-wise Linear Modulation*
— **GAN :** *Generative Adversarial Network*
— **GPU :** *Graphics Processing Unit*
— **ITER :** *International Thermonuclear Experimental Reactor*
— **ReLU :** *Rectified Linear Unit*
— **ROI :** *Region Of Interest*
— **WEST :** *W Environment in Steady-state Tokamak*

Notations mathématiques

Dans ce document, les scalaires sont notés en caractères fins, par exemple t pour un temps ou T pour une température. L'ensemble des matrices à coefficients réels de n lignes et p colonnes est noté $\mathbb{R}^{n\times p}$, et ses éléments sont notés en gras et majuscules, par exemple $\mathbf{A}$. L'ensemble des vecteurs de dimensions n à coefficients réels est noté $\mathbb{R}^n$, ses éléments sont notés en gras minuscule, par exemple $\mathbf{x}$, et sont considérés comme cas particuliers de matrices à n lignes et une colonne. La transposée d'une matrice $\mathbf{A}$ est notée $\mathbf{A}^\top$, l'application d'une matrice $\mathbf{A}$ à un vecteur $\mathbf{x}$ est notée $\mathbf{Ax}$, le produit de deux matrices $\mathbf{A}$ et $\mathbf{B}$ est noté $\mathbf{AB}$, et le produit scalaire entre deux vecteurs $\mathbf{u}$ et $\mathbf{v}$ est noté $\mathbf{u}^\top\mathbf{v}$. Si $\mathbf{x}$ est un vecteur, $\|\mathbf{x}\|$ désigne sa norme euclidienne (norme 2), et $\|\mathbf{x}\|_p$ sa norme p. Si $f : \mathbb{R}^p \to \mathbb{R}$ est une fonction différentiable de $\mathbb{R}^p$ vers $\mathbb{R}$, et $\mathbf{x} \in \mathbb{R}^p$, $\partial_j f(\mathbf{x})$ est la dérivée partielle de f en $\mathbf{x}$ par rapport à sa j-ème coordonnée, et $\nabla f(\mathbf{x})$ est le gradient de f en $\mathbf{x}$, c'est-à-dire le vecteur de ses dérivées partielles. Si $\mathbf{f} : \mathbb{R}^p \to \mathbb{R}^n$ est une fonction différentiable de $\mathbb{R}^p$ vers $\mathbb{R}^n$, et $\mathbf{x} \in \mathbb{R}^p$, on note $\left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right]$ la matrice jacobienne de $\mathbf{f}$ en $\mathbf{x}$, c'est-à-dire la matrice à n lignes et p colonnes formée des dérivées partielles $\left[\partial_j f_i(\mathbf{x})\right]_{i,j}$. Les variables aléatoires sont désignées en majuscule : si X est une variable aléatoire, p_X désigne sa fonction de distribution, $\mathbb{E}_X[X]$ son espérance, et $\mathbf{x} \sim p_X$ signifie que $\mathbf{x}$ est un élément obtenu par échantillonnage de la distribution p_X. L'ensemble des nombres complexes, $\mathbb{C}$, n'est jamais utilisé dans ce document : l'utilisation du terme "complexe" est à comprendre dans un sens littéral.

Table des matières

Affidavit **2**

Publications et conférences **3**

Résumé **4**

Abstract **5**

Glossaire **6**

Table des matières **7**

Introduction **10**

I. La thermographie infrarouge pour la fusion thermonucléaire **13**
I.1. La fusion thermonucléaire 14
I.1.1. La fusion nucléaire : une nouvelle source d'énergie 14
I.1.2. Le principe de la fusion thermonucléaire 14
I.1.3. État de la recherche, les réacteurs dans le monde 15
I.1.4. Fonctionnement d'un tokamak et enjeux techniques 16
I.2. La première paroi et le contôle de sa température 18
I.2.1. Composants face au plasma et divertor 18
I.2.2. Les diagnostics embarqués de la première paroi 19
I.2.3. La thermographie infrarouge dans WEST et ITER 20
I.2.4. Système d'asservissement temps réel dans WEST 21
I.3. La thermographie infrarouge quantitative : enjeux et état de l'art 22
I.3.1. Problème de la mesure de température dans un milieu métallique et chauffé 22
I.3.2. Modélisation directe par méthode des radiosités 25
I.3.3. Méthodes classiques de résolution de problème inverse appliquées à la thermographie infrarouge inverse 26
I.3.4. Limites de l'approche par radiosité et de ses méthodes d'inversion 27

Conclusion I **29**

II. Apprentissage automatique via des données simulées : méthode employée et preuve de concept **30**
II.1. Principe de l'inversion par réseau de neurone grâce à des données simulées 31
II.1.1. État de l'art des méthodes d'apprentissage dans le domaine du rendu inverse 31

II.1.2. Apprentissage automatique assisté par simulation pour la mesure de température dans les tokamaks . . . 32
II.2. Démonstration de la méthode sur un cas simplifié . . . 33
II.2.1. Modèle de lancer de rayons par réflexions purement spéculaire . . . 33
II.2.2. Génération du jeu de données synthétique . . . 35
II.2.3. Apprentissage automatique pour l'inversion . . . 36
II.2.4. Résultats sur données synthétiques purement spéculaires . . . 38

Conclusion II **39**

III. Simulation d'images infrarouge par lancer de rayons **40**
III.1. État de l'art des modèles directs existants . . . 42
III.2. Les différents éléments du modèle direct . . . 43
III.3. La caméra virtuelle . . . 43
III.3.1. Le modèle de caméra *pinhole* . . . 43
III.3.2. Modèle de distorsion . . . 45
III.3.3. Alignement caméra . . . 46
III.4. Équations de rendu infrarouge et modèle de BRDF . . . 46
III.4.1. La radiance émise : modèle du corps noir et émissivité . . . 47
III.4.2. La radiance réfléchie : modèle de la BRDF . . . 49
III.4.3. L'équation du rendu infrarouge à résoudre . . . 54
III.5. Résolution de l'équation du rendu par intégration de Monte Carlo . . . 54
III.6. Intersection accélérée entre un rayon et le modèle 3D . . . 57
III.6.1. Intersection entre un rayon et le modèle 3D . . . 58
III.6.2. Accélération des calculs : parallélisme sur carte graphique . . . 60
III.7. Validation du modèle direct . . . 61
III.7.1. Validation de l'implémentation de la BRDF dans le rendu Monte Carlo . . . 61
III.7.2. Validation sur une scène fermée à température uniforme . . . 63
III.8. Modélisation des scènes thermiques . . . 63
III.8.1. Le modèle 3D du tokamak et les caméras . . . 64
III.8.2. Température uniforme par composants . . . 65
III.8.3. Mélange de BRDF Phong et miroir . . . 66
III.8.4. Profil de température du divertor bas . . . 66
III.8.5. Profil de température des antennes de chauffage . . . 68
III.8.6. Non-uniformité des composants par bruit de gradient et points chauds . . . 68
III.8.7. Base de donnée d'images générées . . . 70

Conclusion III **72**

IV. Inversion des images infrarouge par réseau neurones à convolution **73**
IV.1. Introduction à l'apprentissage automatique . . . 75
IV.1.1. Apprentissage supervisé et fonction de perte . . . 75
IV.1.2. Optimiseurs pour minimiser la fonction de perte . . . 76
IV.2. Réseaux de neurones et apprentissage profond pour le traitement d'image . . . 79
IV.2.1. Architecture des réseaux de neurones . . . 79

IV.2.2. L'algorithme de rétropropagation du gradient . . . 82
IV.2.3. Couches adaptées au traitement d'image . . . 85
IV.2.4. Architectures de réseaux pour le traitement d'image . . . 87
IV.3. Estimation de température par U-Net sur des données simulées . . . 89
IV.3.1. Modélisation de la scène thermique . . . 89
IV.3.2. Génération des données . . . 91
IV.3.3. Apprentissage . . . 92
IV.3.4. Pertinence de l'ensemble de données . . . 93
IV.3.5. Résultats sur images simulées . . . 94
IV.3.6. Application à une image expérimentale . . . 97
IV.4. Limite de la méthode : problème mal posé sur des scènes en visée divertor . . . 98

Conclusion IV **102**

V. Inversion des images infrarouge par réseaux de neurones non déterministes **103**
V.1. L'inférence bayésienne dans le cadre de la résolution de problème inverse . . . 105
V.1.1. Formulation des probabilités et théorème de Bayes . . . 105
V.1.2. Exemple d'application de l'inférence bayésienne : le problème de Laplace . 106
V.1.3. Formulation du problème inverse dans le paradigme bayésien . . . 107
V.2. Méthodes de résolution de problèmes inverses dans le paradigme bayésien . . . 108
V.2.1. Les méthodes de Monte-Carlo par chaîne de Markov . . . 108
V.2.2. Utilisation de réseaux de neurones non déterministes . . . 110
V.3. Validation du cWGAN-GP dans un environnement contrôlé . . . 113
V.3.1. Génération de l'ensemble de données . . . 113
V.3.2. Architecture du réseau et apprentissage . . . 113
V.3.3. Résultats . . . 115
V.4. Application du cWGAN-GP à un problème mal posé sur une visée divertor . . . 117
V.4.1. Description de l'ensemble de données . . . 117
V.4.2. Architectures et apprentissage . . . 118
V.4.3. Résultats . . . 119

Conclusion V **123**

Conclusion générale **124**

Bibliographie **130**

Introduction

Un des enjeux majeurs de ce siècle est d'harmoniser nos conditions matérielles d'existence avec le respect du vivant et des limites planétaires. La crise environnementale, marquée par l'épuisement des ressources naturelles, la sixième extinction de masse, la pollution de la planète et le changement climatique, nous impose de réduire drastiquement notre impact sur l'environnement. Un aspect fondamental des changements de société nécessaires réside dans la production énergétique : il est crucial d'abandonner les sources d'énergie à fort impact environnemental et notamment à fortes émissions de gaz à effets de serre.

La fusion thermonucléaire, bien que ne faisant pas partie des solutions immédiates pour la rupture énergétique nécessaire dans les années à venir, représente un projet de recherche à long terme visant à fournir une nouvelle source d'énergie décarbonée d'ici la fin du siècle. En reproduisant les réactions qui ont sein au cœur du soleil, la fusion nucléaire promet une énergie abondante, sans les inconvénients des énergies fossiles ou de la fission nucléaire traditionnelle. Ce projet de recherche suscite une vaste collaboration internationale, notamment à travers le projet ITER, visant à démontrer la faisabilité de la fusion nucléaire par confinement magnétique à l'aide de tokamaks.

Les tokamaks sont des dispositifs utilisant des champs magnétiques pour confiner et maintenir un plasma à des températures extrêmement élevées, permettant ainsi la fusion des noyaux qui le constituent. Cependant, la protection de leur première paroi, les composants face au plasma, pose de nombreux défis. Ces composants sont exposés à des flux de chaleur intenses, pouvant atteindre jusqu'à 20 MW/m², entraînant des températures de surface allant de 200 à 3600 °C. Pour assurer la sûreté de fonctionnement des tokamaks et la longévité de leurs composants, il est crucial de surveiller en temps réel la température des composants. La thermographie infrarouge est le principal outil utilisé pour cette surveillance.

L'interprétation des images infrarouges est cependant complexe en raison de la nature réfléchissante des parois métalliques, d'émissivité faible et variable, responsable de flux parasites qui perturbent la mesure directe des températures. Cette perturbation accroît le risque de sous-estimation de la température des parois, ce qui pourrait compromettre la sûreté de fonctionnement, ou au contraire, de surestimation, entraînant potentiellement l'arrêt non justifié de la machine. Avec l'augmentation drastique de la puissance des prochaines machines, il sera essentiel de disposer d'un système de surveillance extrêmement fiable pour exploiter la machine de manière optimale sans risque d'endommagement des composants.

Dans ce contexte, ce travail de thèse a pour objectif de développer des méthodes numériques permettant d'estimer de manière précise les températures des composants à partir des images infrarouges, en surmontant les défis posés par l'émissivité variable des surfaces et la présence de réflexions. L'originalité de cette thèse réside dans une approche combinant simulation numérique et apprentissage automatique pour résoudre ce problème.

La première contribution de cette thèse est le développement d'un modèle de lancer de rayons

infrarouges par Monte Carlo accéléré par GPU, capable de générer des images synthétiques prenant en compte les propriétés de réflexion complexes des matériaux. Ce modèle permet de reproduire les phénomènes observés par les caméras infrarouges dans les tokamaks, et de créer ainsi une base de données riche et réaliste adaptée à l'entraînement des algorithmes d'apprentissage automatique.

La seconde contribution est l'application des techniques d'apprentissage profond, en particulier les réseaux de neurones convolutifs, pour l'estimation des températures de surface à partir des images infrarouges simulées. Des modèles déterministes sont explorés en premier lieu, puis des modèles génératifs bayésiens afin d'associer un niveau de confiance aux prédictions, permettant de gérer les incertitudes inhérentes aux mesures et au problème à résoudre.

Le chapitre I introduit le contexte général de la fusion thermonucléaire et les défis associés à la thermographie infrarouge dans les tokamaks. Il présente les principes de la fusion nucléaire, les enjeux techniques spécifiques à la protection de la paroi, les systèmes d'imagerie infrarouge présents dans les tokamaks pour la protection des composants, ainsi que le problème que pose l'estimation quantitative des températures de surface à partir des images capturées. Un état de l'art des méthodes numériques déjà explorées pour résoudre ce problème dans le cadre des machines de fusion est présenté et discuté. Les limites, mises en évidence par ces premières études, ont aussi motivé l'exploration d'une nouvelle approche, basée sur les techniques d'apprentissage automatique, et qui fait l'objet de cette thèse.

Le chapitre II présente une première démonstration de l'inversion par réseaux de neurones, pour l'estimation des températures à partir des images infrarouges. La méthode proposée s'appuie sur l'apprentissage d'un réseau de neurones à partir d'une base de données entièrement simulées. La méthode est ici appliquée dans le cas d'une scène thermique entièrement spéculaire (surfaces lisses agissant comme un miroir). Ce cas simplifié a permis de démontrer la faisabilité et la pertinence de l'approche proposée dans cette thèse, pour la problématique de la mesure infrarouge. Dans les chapitres suivants, les modèles de simulation (direct) et inverse seront complexifiés pour s'approcher des données réelles (expérimentales).

Le chapitre III détaille la méthode de simulation développée pour générer des données synthétiques. Le développement du logiciel de simulation est expliqué en profondeur, en présentant le rayonnement infrarouge et le modèle de réflexion, la génération d'images par propagation de rayons infrarouges par intégration de Monte Carlo, et l'accélération GPU pour générer une grande quantité de données en un temps raisonnable. La modélisation de la scène thermique d'un tokamak en opération, utilisée pour générer des jeux de données synthétiques, est présentée.

Le chapitre IV présente la technique d'apprentissage automatique utilisée pour l'inversion des images infrarouges. Il explore l'entraînement d'un réseau de neurones à convolution de type U-Net à partir des données synthétiques générées, et évalue les performances de ce réseau sur des données simulées de test. Les premiers résultats sur des images simulées montrent que l'approche par réseau de neurones est très prometteuse avec des températures estimées à mieux que 10 % dans un environnement entièrement réflectif et chaud avec une émissivité variable. La validité de cette approche est également démontrée de manière qualitative sur des images expérimentales de WEST. Enfin, ce chapitre met en évidence une limite de ce modèle, inhérente à l'utilisation de modèles déterministes dans le cadre de problèmes inverses mal posés, c'est-à-dire dans le cas où l'information est insuffisante pour discriminer une unique solution. Cette limite motive l'utilisation de modèles non déterministes.

Le chapitre V explore l'application des modèles non déterministes, basés sur un paradigme bayésien, permettant de mieux gérer la nature mal posée des problèmes inverses inhérents à la mesure de température par thermographie infrarouge. Un réseau de neurones génératif, de type cWGAN-GP, est présenté et évalué sur un jeu de données synthétiques posant un problème d'inversion mal posé. En apprenant automatiquement une distribution de probabilité qui capture l'incertitude persistante après une mesure, le modèle permet ainsi d'associer à chaque estimation de température un intervalle de confiance. Cette partie démontre les avantages des approches bayésiennes pour traiter les incertitudes expérimentales et le cas d'un problème mal posé. Elles donnent ainsi accès à des prédictions associées à un niveau de fiabilité, ce qui constitue une avancée majeure pour l'aide à la prise de décision pour la protection des machines de fusion.

Enfin, la conclusion générale de cette thèse résume les principales contributions et résultats obtenus, tout en explorant les perspectives futures. Les avancées réalisées, tant au niveau de la modélisation que des techniques d'apprentissage automatique, ouvrent de nouvelles possibilités pour la protection des machines de fusion, avec des applications potentielles pour ITER et les futures générations de tokamaks. Certaines étapes restent nécessaires pour appliquer aux données expérimentales ces méthodes actuellement évaluées sur des données synthétiques. La thèse se termine par une réflexion sur les améliorations possibles et les étapes à suivre pour la validation expérimentale des méthodes proposées.

I. La thermographie infrarouge pour la fusion thermonucléaire

Résumé

Ce chapitre introduit la fusion nucléaire, en présentant le contexte énergétique actuel, et les enjeux techniques liés aux réacteurs de fusion, notamment à la protection de la première paroi des tokamaks. La thermographie infrarouge est mise en avant comme un diagnostic clé pour la protection temps réel de la machine. Les défis rencontrés pour estimer avec précision les températures des surfaces à partir d'une image infrarouge sont mis en avant. Enfin, les méthodes numériques déjà explorées pour résoudre ce problème sont présentées, ainsi que leurs limites.

Sommaire

I.1. La fusion thermonucléaire 14
I.1.1. La fusion nucléaire : une nouvelle source d'énergie 14
I.1.2. Le principe de la fusion thermonucléaire 14
I.1.3. État de la recherche, les réacteurs dans le monde 15
I.1.4. Fonctionnement d'un tokamak et enjeux techniques 16
I.2. La première paroi et le contôle de sa température 18
I.2.1. Composants face au plasma et divertor 18
I.2.2. Les diagnostics embarqués de la première paroi 19
I.2.3. La thermographie infrarouge dans WEST et ITER 20
I.2.4. Système d'asservissement temps réel dans WEST 21
I.3. La thermographie infrarouge quantitative : enjeux et état de l'art 22
I.3.1. Problème de la mesure de température dans un milieu métallique et chauffé 22
I.3.2. Modélisation directe par méthode des radiosités 25
I.3.3. Méthodes classiques de résolution de problème inverse appliquées à la thermographie infrarouge inverse 26
I.3.3.1. Estimation des températures par méthode linéaire 26
I.3.3.2. Estimation de l'émissivité par méthode non linéaire 27
I.3.4. Limites de l'approche par radiosité et de ses méthodes d'inversion 27

I.1. La fusion thermonucléaire

I.1.1. La fusion nucléaire : une nouvelle source d'énergie

La production d'énergie grâce à la fusion nucléaire est considérée par beaucoup comme un des grands défis techniques du siècle dans le domaine de l'énergie. Le processus en jeu est celui qu'on retrouve au cœur des étoiles : il s'agit d'unir deux noyaux légers pour récupérer l'énergie libérée lors de cette fusion nucléaire. La technique de la fusion pourrait offrir de nombreux avantages : c'est une énergie bas carbone sans rejet de dioxyde de carbone ni d'autres gaz à effet de serre dans l'atmosphère, une énergie sans production de déchets nucléaires à longues périodes et haute activité, et sans risque d'emballement du cœur comme dans les réacteurs de fission nucléaire, la fusion du cœur du réacteur étant pratiquement impossible. Dernier atout : les combustibles nécessaires à la fusion, le deutérium et le tritium, sont disponibles en abondance et aisément accessibles puisque le deutérium peut être extrait dans l'eau de mers et le tritium produit à partir du lithium présent en abondance dans la nature. Aujourd'hui, la fusion nucléaire fait l'objet de nombreux travaux de recherche partout dans le monde, avec des avancées significatives ces dernières années pour reproduire sur Terre les conditions qui sont naturelles dans les étoiles, comme la densité et des températures extrêmement élevées. Pour répondre à ce défi scientifique et technique, cette recherche fait également l'objet d'une collaboration internationale dont l'expérience la plus importante est le projet de Réacteur expérimental thermonucléaire international (ITER), mené par sept membres (Chine, Corée du Sud, États-Unis, Fédération de Russie, Inde, Japon et Union européenne), en cours de construction à Cadarache (France).

I.1.2. Le principe de la fusion thermonucléaire

Lors de la fission nucléaire, c'est la scission des atomes qui produit de l'énergie. À l'inverse, lors de la fusion, c'est l'union de noyaux légers qui crée des noyaux plus lourds et dégage de l'énergie. Plusieurs réactions de fusion sont possibles, en fonction des éléments choisis. La réaction la plus prometteuse est la réaction D-T (I.1) qui visent à unir deux isotopes de l'hydrogène : le deutérium et le tritium.

$$ {}^{2}_{1}\mathrm{D} + {}^{3}_{1}\mathrm{T} \rightarrow {}^{4}_{2}\mathrm{He} + {}^{1}_{0}\mathrm{n} + 17.6\ \mathrm{MeV} \tag{I.1} $$

D'autres machines expérimentales utiliseront la réaction D-D, pour Deutérium-Deutérium (I.2), moins efficace que la réaction D-T, mais qui présente l'avantage de produire des neutrons moins énergétiques et donc moins endommageants pour les structures de la centrale.

$$ {}^{2}_{1}\mathrm{D} + {}^{2}_{1}\mathrm{D} \rightarrow {}^{3}_{1}\mathrm{T} + {}^{1}_{1}\mathrm{H} + 4\ \mathrm{MeV} \quad \text{ou} \quad {}^{2}_{1}\mathrm{D} + {}^{2}_{1}\mathrm{D} \rightarrow {}^{3}_{2}\mathrm{He} + {}^{1}_{0}\mathrm{n} + 3.3\ \mathrm{MeV} \tag{I.2} $$

Pour réaliser une fusion nucléaire, trois conditions sont nécessaires : une température (T) très élevée (100 millions de degrés soit environ 10 fois la température du soleil) pour provoquer des collisions de particules de haute énergie, une densité de particules (n) importante pour augmenter la probabilité de collision et un temps de confinement (τ) suffisant pour que les réactions de fusion puissent se produire en continu. Le critère de Lawson décrit les conditions dans lesquelles la fusion est soutenable : en pratique, cela se résume en une condition sur le

triple produit $nT\tau$, qui doit être suffisamment élevé.

Le confinement d'un plasma, qui a tendance à se dilater en raison de l'énergie libérée, requiert ainsi une force considérable. Dans une étoile comme le Soleil, la gravité immense suffit à créer les conditions nécessaires en son cœur. Sur Terre, deux approches sont envisagées :

— Le confinement inertiel : les éléments de fusion sont placés dans une capsule qui est brutalement chauffée par des lasers très puissants, provoquant son effondrement et créant des conditions de pression extrêmes propices à la fusion.

— Le confinement magnétique : cette méthode utilise des champs électromagnétiques très puissants, générés par des bobines, pour piéger les particules chargées du plasma. Cette approche est considérée comme la plus prometteuse pour la production industrielle d'électricité.

La technologie de confinement magnétique la plus répandue à ce jour est celle du tokamak (dont le nom vient d'un acronyme russe qui signifie « chambre toroïdale avec bobines magnétiques »), dispositif en forme de tore inventé dans les années 1950 qui permet de confiner le plasma au moyen de puissants champs magnétiques. Une autre voie également étudiée, notamment en Allemagne avec la machine Wendelstein 7-X, est celle des stellarators dont le confinement du plasma est assuré entièrement par un champ magnétique hélicoïdal, créé par un ensemble de bobines de géométrie très complexe. La Figure I.1 illustre les 2 types de dispositifs pour le confinement magnétique.

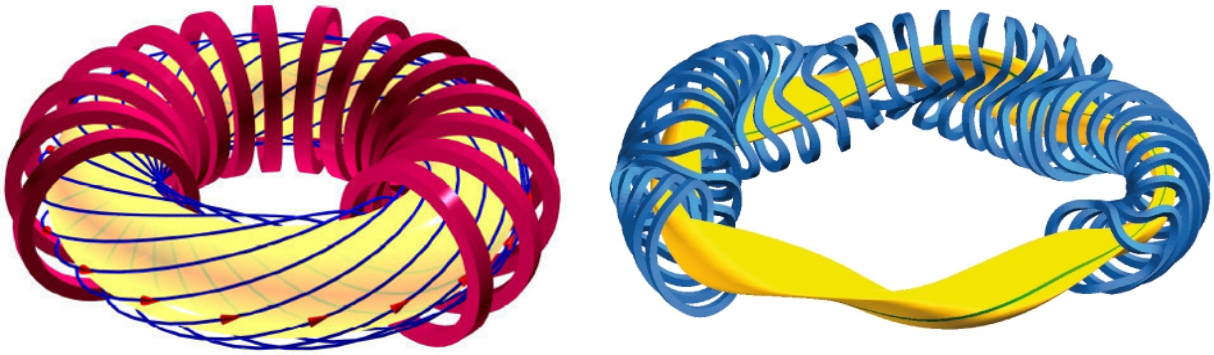


FIGURE I.1. – Gauche – Représentation des bobines toroïdales, du champ magnétique, et du plasma dans un tokamak [P. Moreau, MENA Fusion school 2021]. **Droite –** Représentation des bobines et du plasma dans le stellarator Wendelstein 7-X [Max-Planck Institut für Plasmaphysik].

I.1.3. État de la recherche, les réacteurs dans le monde

De nombreuses centrales de fusion expérimentales existent dans le monde. Cette section présente trois principales de ces machines dont les dimensions et principales caractéristiques sont illustrés en Figure I.2.

Le tokamak Européen JET (*Joint European Torus*) au Royaume-Uni, est une des rares machines à pouvoir faire des réactions D-T et détient actuellement le record de puissance. En France,

le tokamak Tore Supra, a été le premier à détenir le record de durée de plasma en 2003 de 6 minutes 30 secondes, grâce à ses bobines supraconductrices refroidies à quelques kelvins. Depuis 2016, Tore Supra a subi une profonde transformation pour devenir WEST, *W (Tungsten) Environment in Steady-state Tokamak*, une plateforme de test pour les composants d'ITER et explorer les problématiques de physique de plasma sur des longues durées en environnement tungstène [Buc+14 ; Buc+22]. Enfin, le projet international de réacteur thermonucléaire en cours de construction à Cadarache (France), ITER vise à produire des plasmas D-T avec une puissance d'amplification [1] de l'ordre de 10 pendant des périodes de 400 à 600 secondes.

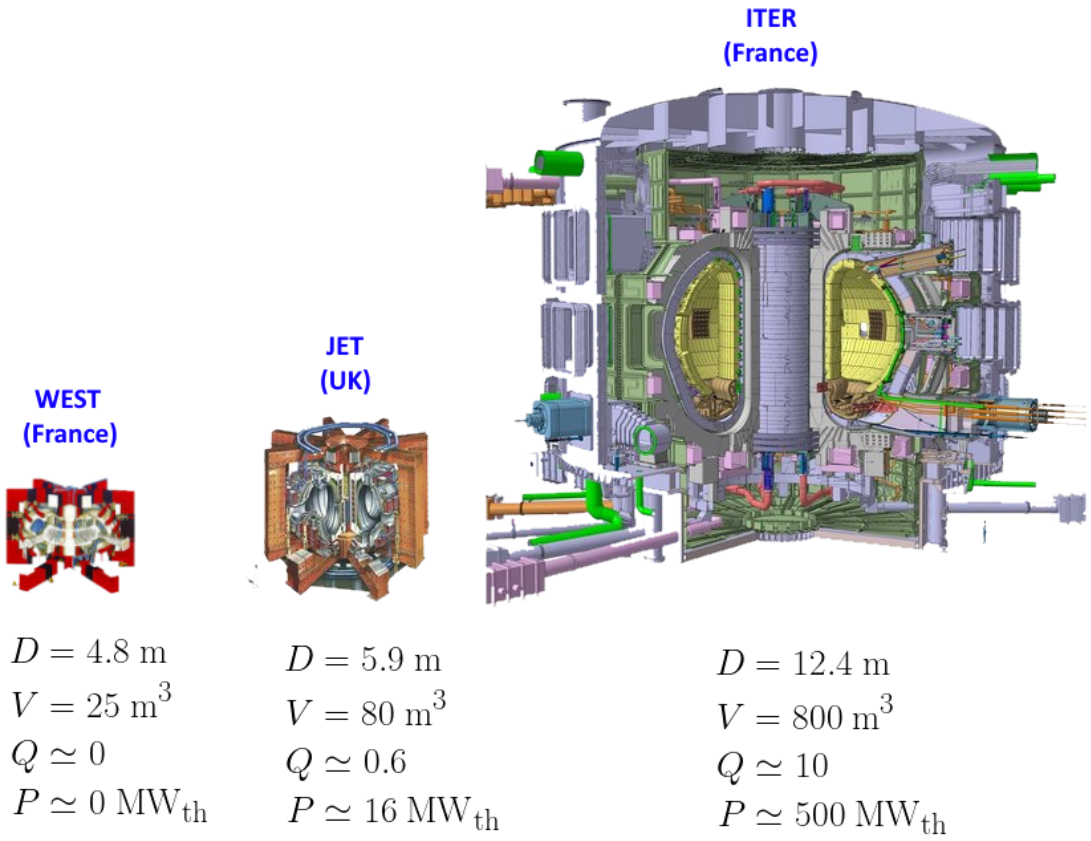


FIGURE I.2. – Dimensions et caractéristiques des tokamaks Tore Supra/WEST, JET et ITER.

I.1.4. Fonctionnement d'un tokamak et enjeux techniques

Le premier enjeu d'un tokamak est le confinement efficace et stable du plasma, via un champ magnétique puissant de plusieurs Tesla. Ce champ magnétique est créé par un ensemble de bobines représentées en Figures I.1. Le solénoïde central initialise un courant de plasma par

1. Le facteur de gain énergétique, noté Q, est le rapport entre l'énergie produite par le plasma et l'énergie injectée pour maintenir le plasma. Le seuil de $Q = 1$, correspondant au "break-even", où le plasma produit autant d'énergie qu'il en reçoit, a été approché par le JET.

induction, générant ainsi un champ magnétique toroïdal qui enroule le plasma en une configuration en forme de tore. Les bobines toroïdales, disposées en anneau autour du tokamak, génèrent un champ magnétique toroïdal qui circule autour du tore. Ce champ magnétique principal maintient les particules chargées sur des trajectoires hélicoïdales autour de l'axe torique, empêchant ainsi qu'elles ne divergent vers l'extérieur. Enfin, les bobines poloïdales produisent un champ magnétique vertical, servant à façonner et stabiliser le plasma en contrôlant sa position et sa forme. En ajustant le courant dans les bobines poloïdales, il est possible de manipuler les profils de densité et de température du plasma pour optimiser les conditions de confinement.

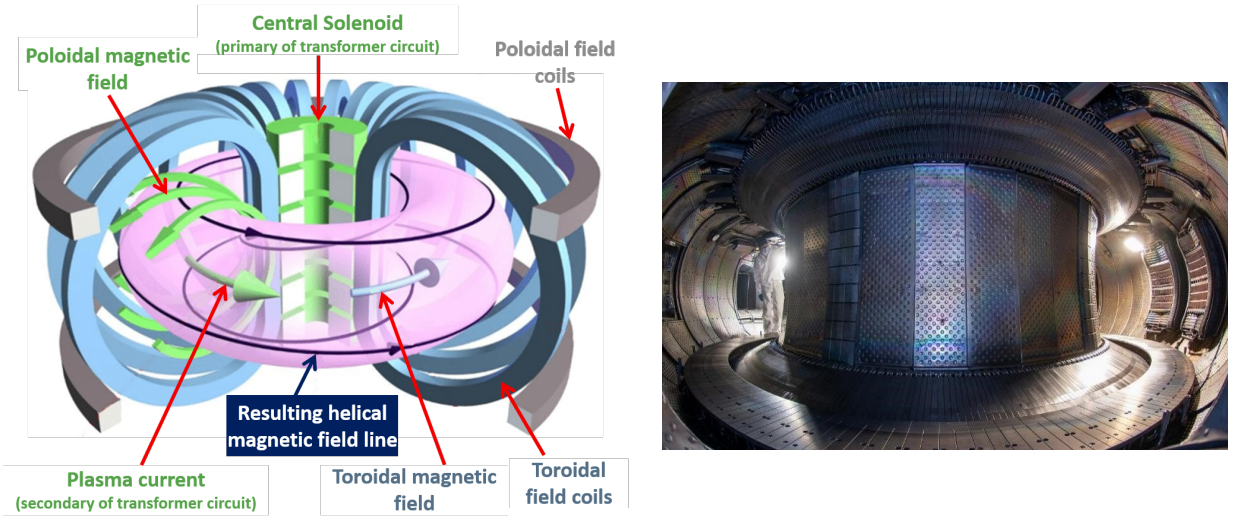


FIGURE I.3. – Gauche – Représentation des différentes bobines permettant de réaliser le champ magnétique qui confine le plasma dans un tokamak [P. Moreau, MENA Fusion school 2021]. **Droite –** Photo de la chambre du tokamak WEST.

Le deuxième enjeu est le chauffage du plasma, réalisé conjointement par plusieurs éléments. Le chauffage ohmique est le processus initial, basé sur la résistance électrique du plasma au courant induit par le solénoïde central. Bien que cette méthode soit efficace au début de la décharge, son efficacité diminue à mesure que le plasma se réchauffe et que sa résistance électrique initiale baisse. Une autre méthode de chauffage est l'injection de faisceaux neutres (NBI). Cette technique consiste à bombarder le plasma avec des atomes de deutérium. Ces atomes pénètrent dans le plasma et, par des collisions, augmentant ainsi la température du plasma. Enfin, le chauffage par ondes électromagnétiques (RF) est également utilisé. Des systèmes d'antennes émettent des ondes électromagnétiques à des fréquences spécifiques. Ces ondes sont absorbées par les particules chargées du plasma, augmentant ainsi la température globale du plasma.

Enfin, le dernier enjeu rencontré dans les tokamaks, et qui constitue le sujet de recherche de ce manuscrit, est la protection de la première paroi, c'est-à-dire la première barrière matériel exposée au plasma, et dont les éléments la constituant sont également appelés composants face au plasma (CFP). En effet, le confinement magnétique tel qu'exposé dans les sections précédentes, n'est pas parfait : des particules de haute énergie peuvent s'en échapper et être responsables de flux de chaleur importants sur les composants face au plasma. La surveillance de

la température de cette première paroi et la protection contre les dommages potentiels sont alors essentielles pour le bon fonctionnement et la durabilité de la machine. Pour répondre à cet enjeu, un ensemble de diagnostics de surveillance est mis en place, dont les systèmes de thermographie infrarouge, qui font l'objet de cette thèse et décrits dans la section suivante.

I.2. La première paroi et le contôle de sa température

I.2.1. Composants face au plasma et divertor

La première paroi a trois fonctions principales : garantir l'intégrité de la chambre à vide, extraire en continu la puissance provenant du plasma, et éviter toute contamination du plasma par des impuretés qui se détacheraient.

Pour gérer l'interaction plasma-paroi, une première approche est d'utiliser des limiteurs : un objet massif en contact direct avec le plasma et conçu pour supporter les hauts flux. Une autre approche est la configuration divertor ou limiteur magnétique "immatérielle", où la "frontière plasma" n'est plus en contact avec la paroi, mais est localisée en un point virtuel dans la machine (point X). La configuration divertor est, depuis les années 90, largement privilégiée dans les machines de fusion. Concrètement, cette configuration point X est réalisée grâce à une bobine additionnelle placée sous un composant appelé divertor. Elle crée une annulation du champ magnétique au-dessus du divertor, générant des lignes de champ magnétique qui se croisent, formant ainsi une sorte de "X" lorsqu'elles sont vues en coupe toroïdale, comme illustré en Figure I.4. Dans la configuration de plasma standard, on observe alors, sur le divertor, deux zones de forts flux, appelées *strike lines*, de part et d'autre du point X, (au niveau des jambes des lignes de champ magnétiques), comme nous pouvons l'observer sur les images infrarouges de la visée divertor de la Figure I.6.

Les autres composants sont aussi exposés à des forts flux, le confinement des particules de plasma n'étant pas parfait. En effet, les particules qui suivent les lignes de champ fermées à l'intérieur du plasma sont piégées, mais celles qui dérivent vers l'extérieur de la dernière surface fermée entrent dans une région appelée la *scrape-off layer* (SOL). Dans cette région, les lignes de champ ne sont plus fermées et finissent par heurter les composants de la machine, exposant les composants à des fort flux de chaleur. Typiquement, les dépôts de puissance de chaleur sur les composants face au plasma peuvent s'élever jusqu'à 10 $\mathrm{MW \cdot m^{-2}}$ en régime continu à 20 $\mathrm{MW \cdot m^{-2}}$ en régime transitoire.

Ces conditions extrêmes nécessitent donc une conception robuste et des matériaux résistants pour supporter les contraintes thermiques. Le matériau aujourd'hui privilégié est le tungstène (W), qui présente à la fois une température de fusion supérieure à 3 000 °C, mais aussi une bonne tenue aux radiations et une activation faible pour les réactions D-T. C'est également un matériau résistant avec un seuil de pulvérisation élevée qui permet de limiter la contamination du plasma par les impuretés lors des décharges.

La Figure I.5 présente les différents composants de la machine WEST évoqués précédemment : on retrouve notamment le divertor bas (plancher de la machine), l'élément critique destiné à recevoir le maximum de la puissance, le divertor haut, les antennes de chauffage, et les limiteurs sur la paroi interne. Il est également à noter l'existence de trois types de famille de composants

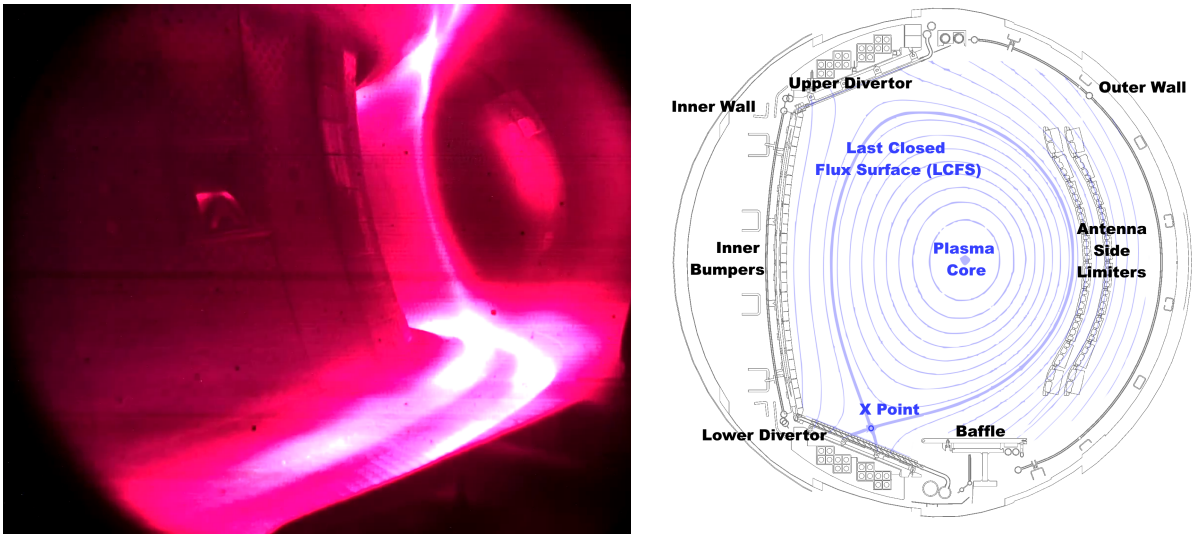


FIGURE I.4. – Gauche – Capture du plasma par une caméra visible lors d'un choc dans le tokamak WEST. **Droite –** Coupe toroïdale représentant les différentes lignes de champs ainsi que les composants de WEST. Les particules du plasma suivent principalement les lignes de champs fermées. Un point X est formé au niveau du divertor bas par la dernière surface magnétique fermée. Au-delà, les lignes de champs ne sont plus fermées : les particules du plasma de bord viennent heurter les CFP, principalement au niveau du divertor bas.

intégrés dans la machine : inertiel (non refroidis), semi-inertiel (en contact avec un support froid), et activement refroidi avec un canal de refroidissement intégré.

Chacun de ses composants aura une surveillance plus ou moins appuyée suivant leur criticité. Cette surveillance pourra se faire à travers des diagnostics embarqués (section I.2.2) et/ou par thermographie infrarouge, l'objet de ces travaux (section I.2.3).

I.2.2. Les diagnostics embarqués de la première paroi

Les diagnostics intégrés aux composants des tokamaks permettent d'estimer leurs températures internes. Dans le cas de WEST, vingt thermocouples sont intégrés aux composants, permettant d'estimer la température de surface à travers une étape d'inversion des données collectées. En complément de ces thermocouples, les composants sont également équipés de Fibres de Bragg (FBG). Les FBG sont des fibres optiques qui exploitent des phénomènes d'interférence pour mesurer la dilatation des matériaux, fournissant ainsi des informations sur leur température. Quatre sondes de ce type sont installées dans WEST.

Ces diagnostics sont qualifiés d'intrusifs, car leur mise en œuvre nécessite un usinage spécifique des composants pour intégrer ces capteurs en leur sein. De plus, ces mesures fournissent des données ponctuelles de température, et une étape d'inversion, afin d'obtenir la température de surface à partir de la température interne, introduit des incertitudes.

Ces outils sont principalement utilisés pour des analyses physiques *a posteriori* et ne sont pas adaptés à la protection en temps réel. Pour disposer d'un outil de mesure capable de surveiller

de grandes zones de la machine en temps réel et de manière non intrusive, un système de thermographie infrarouge est utilisé.

I.2.3. La thermographie infrarouge dans WEST et ITER

La thermographie infrarouge est un diagnostic essentiel pour mesurer les températures de surface de la première paroi du tokamak, tant pour la protection en temps réel que pour les analyses physiques. Ce type de surveillance est non intrusif et permet de couvrir de vastes zones de la machine.

Dans WEST, un réseau de 12 caméras infrarouges est installé. Ces caméras offrent une bonne résolution temporelle, capturant des images à une fréquence de 50 Hz, et ont une résolution de 640 par 512 pixels, offrant ainsi une résolution spatiale inférieure au centimètre par pixel. Elles fonctionnent dans des plages de longueur d'onde correspondant à l'infrarouge moyen (*MWIR*, [3–5 µm]), permettant ainsi de surveiller des températures allant de 200 °C à 2000 °C. La Figure I.5 montre les différents champs de vue des caméras WEST : 5 caméras observent directement le divertor, 5 autres les antennes de chauffage. WEST comprend également deux autres types de visées : une visée tangentielle grand angle (relevante aux visées infrarouges prévues dans ITER) et une visée avec un champ de vue étroit pour surveiller quelques éléments (tuiles) du divertor avec une très haute résolution.

L'installation de systèmes d'imagerie similaires à ceux prévus pour ITER dans WEST en fait une plateforme idéale pour la mise en place et le test de systèmes de surveillance.

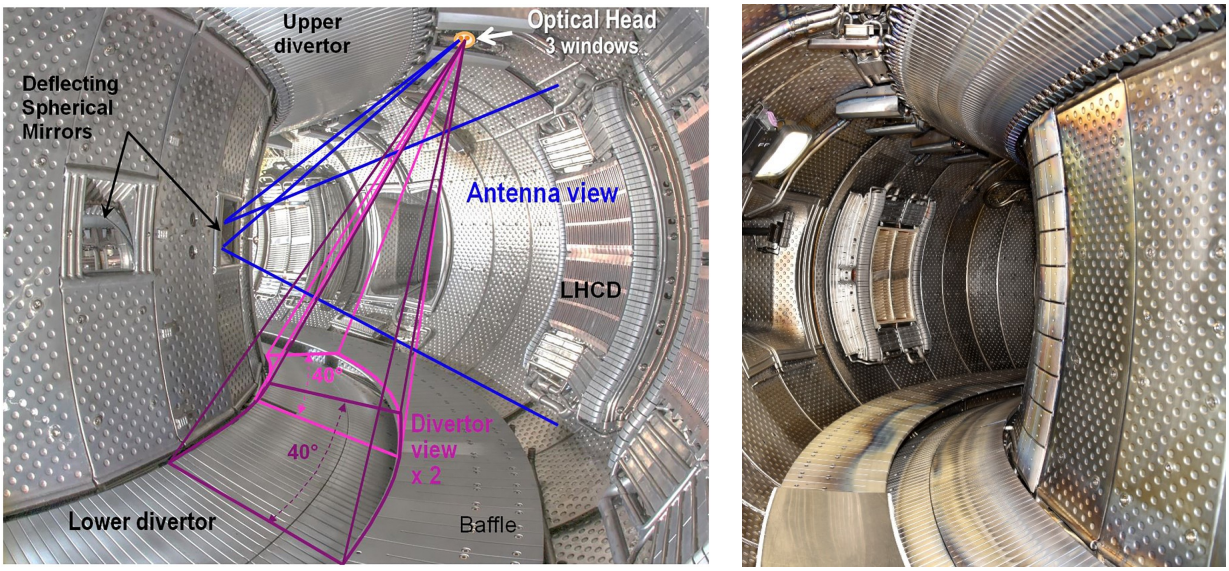


FIGURE I.5. – Champ de vue des différentes caméras infrarouge installées dans WEST. **Gauche –** Représentation des champs de vue pour les visées antenne et divertor. **Droite –** Photo reproduisant le champ de vue de la caméra tangentielle grand angle.

I.2.4. Système d'asservissement temps réel dans WEST

La surveillance du tokamak WEST est assurée par un système de traitement des images infrarouges en temps réel, utilisant des cartes FPGA. Ce système envoie des signaux d'asservissement pour réduire la puissance de fonctionnement de la machine, ou pour arrêter celle-ci en cas de surchauffe excessive d'un des composants.

La surveillance repose sur la définition de régions d'intérêt (ROI) dans les images, où des seuils de température sont établis. Si les pixels les plus intenses au sein d'une ROI dépassent un certain pourcentage de ce seuil, le système temps réel envoie un signal au système de contrôle de la machine pour réduire sa puissance par asservissement. Si le seuil maximal toléré est dépassé, un signal ordonnant l'arrêt de la machine est envoyé. La définition de différentes ROI permet d'ajuster les seuils en fonction des différents composants et de leur tolérance à la chaleur. Par exemple, le divertor en tungstène a une tolérance thermique bien plus élevée que des tubes en cuivre.

Un logiciel de visualisation, ThermaVIP [Mon+23], permet de visualiser les films capturés pendant les décharges, ainsi que de réaliser diverses analyses sur ces films ou images. Il permet aux opérateurs de vérifier le déroulement d'un choc et de prendre des décisions de sécurité si nécessaire. Une vue du logiciel, montrant les ROIs sur différentes images provenant de diverses perspectives, est présentée en Figure I.6.

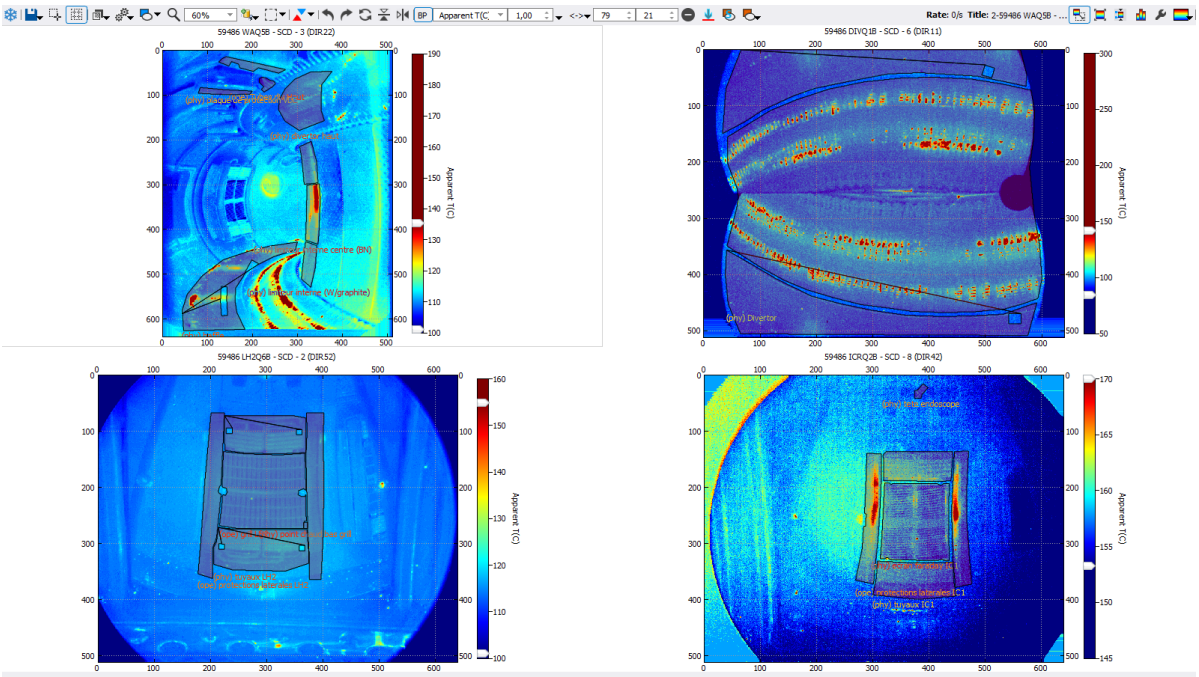


FIGURE I.6. – Capture d'écran du logiciel ThermaVIP. Les films de 4 caméras infrarouges sont visualisées, ainsi que les délimitations en région d'intérêt (ROI) sur chaque image. En haut à gauche : image de la caméra tangentielle grand angle. En haut à droite : images de visée divertor. En bas : images de caméras observant les antennes de chauffage.

Bien que ces systèmes permettent une surveillance en temps réel efficace de la machine, ils ne sont pas parfaits. La nature du rayonnement infrarouge fait que les caméras ne mesurent

pas directement la température des surfaces observées. En réalité, la mesure correspond à une quantité de lumière perçue par pixel, qui est liée à la température des surfaces mais n'est pas entièrement déterminée par celle-ci. La lumière perçue est perturbée par un flux réfléchi, et la lumière émise par les surfaces est modulée par leur émissivité. Ainsi, les seuils définis pour chaque ROI sur l'image peuvent ne pas correspondre strictement à un seuil de température des surfaces. Cela peut conduire à un arrêt injustifié d'un choc ou à une sous-estimation de la température de surface, ce qui peut avoir des conséquences importantes sur la sécurité de la machine en cas de dommages.

La section suivante décrit en détail le problème que pose l'émissivité des surfaces et le flux réfléchi parasite, ainsi que les méthodes numériques qui ont été envisagées pour le corriger.

I.3. La thermographie infrarouge quantitative : enjeux et état de l'art

Dans cette section, nous présentons la problématique de la thermographie infrarouge quantitative dans un environnement chaud et entièrement réflectif, avec des propriétés de surface changeante. Pour estimer les températures de surface à partir des images infrarouges dans un tel environnement, une première approche basée sur la méthode de radiosité a été précédemment explorée. Nous discuterons des limites qui ont motivé l'étude d'autres méthodes de simulation et d'inversion et qui font l'objet de cette thèse.

I.3.1. Problème de la mesure de température dans un milieu métallique et chauffé

L'objectif de la thermographie infrarouge quantitative est d'estimer avec précision la température des surfaces observées par la caméra à partir des images infrarouges capturées. Chaque pixel de la caméra mesure une valeur de radiance $L_{\text{capturée}}$ en ($\mathrm{W}\cdot\mathrm{m}^{-2}\cdot\mathrm{sr}^{-1}$), décrivant la puissance radiative capturée. En supposant que la caméra est correctement calibrée et qu'il n'y a ni absorption ni émission de radiance du plasma dans la chambre à vide, la radiance $L_{\text{capturée}}$ pour un pixel correspond à la radiance sortant de la surface directement observée par la caméra. Bien que cette radiance sortante dépende de la température T de la surface observée, elle n'est pas uniquement déterminée par celle-ci.

Premièrement, la radiance sortante d'une surface est composée d'une radiance émise, qui est liée à la température de la surface, mais aussi d'une part de radiance réfléchie provenant des autres surfaces de la scène. Ce flux additionnel peut donner l'impression qu'une surface est chaude alors qu'elle ne fait que réfléchir la lumière d'une autre surface chaude. Dans le contexte des caméras placées dans un tokamak, ce flux réfléchi est significatif car les surfaces métalliques sont fortement réfléchissantes, ce qui peut entraîner des faux points chauds, comme montré en Figure I.7.

De plus, l'amplitude de la lumière émise et réfléchie dépend des propriétés optiques de la surface. Dans le modèle du corps noir parfait, la radiance émise d'un corps à température T est notée $L^{\circ}(T)$, une fonction croissante en T donnée par la loi de Planck. Pour des surfaces réelles,

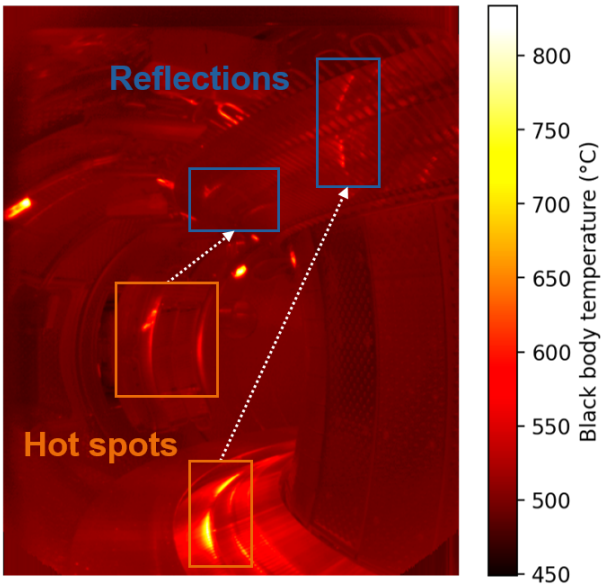


FIGURE I.7. – Exemple d'image capturée par la caméra tangentielle grand angle de WEST. L'image, affichée en température apparente, contient de faux points chauds, qui sont en réalité des motifs de réflexion provenant d'autres surfaces ponctuellement chaudes. L'échelle de température apparente, ou température corps noir, ne correspond pas à la température réelle des surfaces : c'est une conversion de la radiance capturée par la caméra en la température qu'aurait un émetteur parfait pour produire cette radiance. Cette échelle ne prend ainsi pas en compte le flux réfléchi et l'émissivité variable des surfaces.

cette radiance est pondérée par une émissivité, notée ε, comprise entre 0 et 1. De même, la lumière réfléchie par la surface est une proportion de la radiance l'éclairant, appelée réflectivité et notée ρ. Pour un matériau opaque, émissivité et réflectivité sont liées par la relation $\varepsilon + \rho = 1$.

$$\begin{aligned} L_{\text{capturée}} &= L_{\text{émise}} + L_{\text{réfléchie}} \\ &= \varepsilon L^{\circ}(T) + \rho L_{\text{incidente}} \\ &= \varepsilon L^{\circ}(T) + (1-\varepsilon) L_{\text{incidente}} \end{aligned} \tag{I.3}$$

Ainsi, la lumière capturée ne dépend pas seulement de la température de la surface observée, mais aussi de ses paramètres d'émissivité et de la lumière provenant du reste de la scène. La température apparente affichée par les systèmes d'imagerie, ou température corps noir, ne correspond ainsi par directement à la température réelle des surfaces, mais à la température qu'aurait un corps noir, c'est-à-dire un émetteur parfait, pour générer cette radiance. Celle-ci peut être radicalement différente de la température réelle en cas de faible émissivité et de fort flux réfléchi.

$$T_{\text{apparente}} = \left(L^{\circ}\right)^{-1}\left(\varepsilon L^{\circ}(T) + (1-\varepsilon) L_{\text{incidente}}\right) \tag{I.4}$$

Estimer la température T, à partir de la mesure $L_{\text{capturée}}$ ou sa conversion en température

$T_{\text{apparente}}$, est donc une forme de problème inverse (c'est-à-dire cherchant à estimer une cause à partir de la mesure d'un effet) qui nécessite de prendre en compte les propriétés locales de la surface ainsi que celles de l'ensemble de la scène. Dans un tokamak, les surfaces métalliques et chaudes amplifient ce problème. Les surfaces métalliques ont une émissivité faible, de l'ordre de 0,2 à 0,3, (et donc une réflectivité élevée). De plus, leur température élevée en fait des sources significatives de radiance. Ainsi, mesurer la température ponctuelle dans un environnement où toutes les surfaces sont à la fois émissives et réfléchissantes est particulièrement complexe.

De plus, l'émissivité des surfaces est fortement variable spatialement, et au cours du temps. Les phénomènes d'érosion et de dépôt modifient cette émissivité au cours de la vie des composants. Avec le temps, cette variabilité rend obsolètes les mesures d'émissivité faites in situ. La Figure I.8 illustre l'erreur qui peut être commise sur l'estimation de la température en fonction de l'estimation de l'émissivité d'une surface dans des conditions similaires à celles d'un divertor dans un tokamak, en supposant d'abord qu'il n'y a pas de flux réfléchi, puis en ajoutant un flux réfléchi parasite.

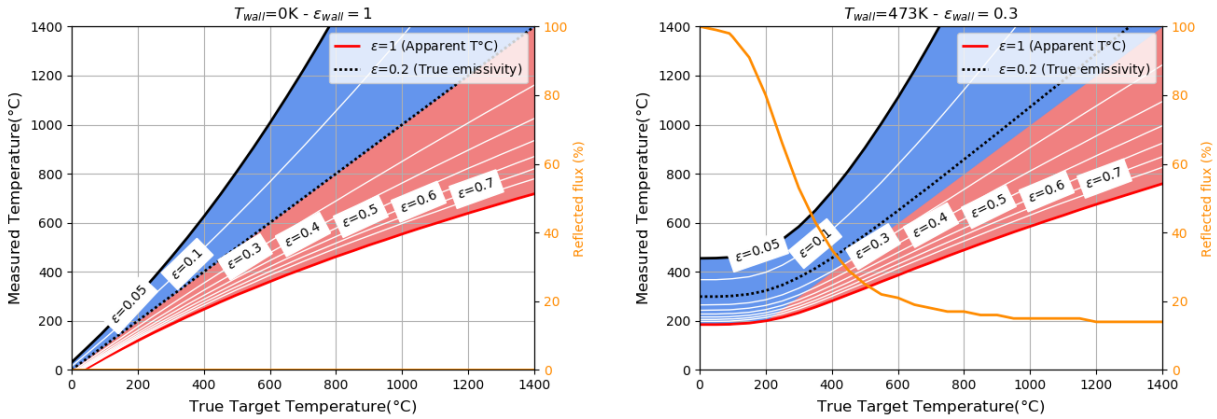


FIGURE I.8. – Température de surface apparente mesurée en fonction de la véritable température de surface d'une cible d'émissivité 0,2 placée au sein d'une enceinte de température T_{wall} et d'émissivité $\varepsilon_{\text{wall}}$. **Gauche –** Sans environnement radiatif et sans flux réfléchi. L'erreur provient uniquement de l'imprécision de l'émissivité de la cible. **Droite –** Avec la contribution du flux réfléchi. Dans ce cas, les erreurs proviennent de l'imprécision de l'émissivité de la cible et de la lumière parasite additionnelle. La courbe orange montre la part du flux réfléchi dans le flux total capturé par la caméra. La zone bleue indique les cas où la température mesurée surestime la véritable température (sans risque pour la sécurité de la machine, mais limitant les scénarios plasma et provoquant de fausses alarmes). La zone rouge montre les cas où la température mesurée sous-estime la véritable température (avec un risque de dommages aux composants de la paroi). La courbe rouge représente la température apparente (ou température corps noir) mesurée par défaut par le système de thermographie infrarouge, c'est-à-dire lorsque l'émissivité est considérée comme égale à 1. La courbe noire en pointillés représente la température mesurée lorsque l'émissivité de la cible est connue et corrigée : dans le premier cas, la véritable température est bien récupérée, tandis que dans le second cas, la température mesurée est surestimée en raison de la lumière parasite additionnelle. Source : [Aum+21].

La présence d'un flux réfléchi parasite peut entraîner une surestimation de la température estimée du composant, ce qui peut provoquer un arrêt injustifié de la machine. Inversement, une surestimation de l'émissivité peut gravement sous-estimer la température réelle du composant, ce qui pourrait causer des dommages potentiels. Il est donc crucial de résoudre ce problème, surtout dans le contexte du prochain réacteur de fusion, ITER, qui nécessitera un système de surveillance extrêmement fiable pour atteindre ses records de puissance. Ce système doit éviter les limitations injustifiées tout en minimisant les risques d'endommagement de la machine. Cette thèse se concentre sur le développement de méthodes numériques visant à corriger ces erreurs et à améliorer la précision des mesures de température infrarouge dans les tokamaks. Un état de l'art des méthodes numériques employées est présenté dans la section suivante.

I.3.2. Modélisation directe par méthode des radiosités

La méthode des radiosités [GCT86] est une technique de modélisation directe utilisée pour calculer les échanges d'énergie entre surfaces d'une scène, permettant ainsi, en incluant un modèle de caméra, de générer une image virtuelle. Elle a été utilisée dans le cadre de la thermographie infrarouge dans les tokamaks des surfaces [Tal+20 ; Tal+21]. Cette méthode suppose des surfaces émettant et réfléchissant de manière diffuse : la radiance dans n'importe quelle direction donnée est simplement la radiosité divisée par π. Dans ce cadre, la scène est discrétisée en un ensemble de n surfaces diffuses S_i, chacune caractérisée par une température T_i et une émissivité ε_i. L'émittance noire de chaque surface M_i° est donnée par $M^\circ(T_i) = \pi L^\circ(T_i)$.

La radiosité J_i de la surface S_i est exprimée par :

$$J_i = \varepsilon_i M_i^\circ + \rho_i \sum_{j=1}^{n} F_{i,j} J_j \tag{I.5}$$

où $\rho_i = 1 - \varepsilon_i$ et $F_{i,j}$ est le facteur de forme, une mesure géométrique de la proportion de flux partant de S_i et incident sur S_j. Le facteur de forme est défini par :

$$F_{i,j} = \frac{1}{|S_i|} \int_{S_i} \int_{S_j} v_{i,j} \frac{\cos(\theta_i)\cos(\theta_j)}{\pi r_{i,j}^2} \, \mathrm{d}S_j \, \mathrm{d}S_i \tag{I.6}$$

où $v_{i,j}$ représente la visibilité entre deux points de surface (1 si les points sont mutuellement visibles, 0 sinon), θ_i et θ_j sont les angles par rapport à la normale, et $r_{i,j}$ est la distance entre les points.

Cela conduit à un système matriciel à résoudre :

$$\begin{bmatrix} 1-\rho_1 F_{1,1} & -\rho_1 F_{1,2} & \cdots & -\rho_1 F_{1,n} \\ -\rho_2 F_{2,1} & 1-\rho_2 F_{2,2} & \cdots & -\rho_2 F_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ -\rho_n F_{n,1} & -\rho_n F_{n,2} & \cdots & 1-\rho_n F_{n,n} \end{bmatrix} \begin{bmatrix} J_1 \\ J_2 \\ \vdots \\ J_n \end{bmatrix} = \begin{bmatrix} \varepsilon_1 M_1^\circ \\ \varepsilon_2 M_2^\circ \\ \vdots \\ \varepsilon_n M_n^\circ \end{bmatrix} \tag{I.7}$$

ou, plus succinctement,

$$\mathbf{Aj} = (\mathbf{I} - \mathbf{RF})\mathbf{j} = \mathbf{Em}^\circ \tag{I.8}$$

où $\mathbf{I} \in \mathbb{R}^{n\times n}$ est la matrice identité, $\mathbf{E}$ est la matrice diagonale des émissivités, $\mathbf{R} \in \mathbb{R}^{n\times n}$ est la matrice diagonale des réflectivités ($= \mathbf{I} - \mathbf{E}$), $\mathbf{F}$ est la matrice des facteurs de forme, $\mathbf{j} \in \mathbb{R}^n$ est le vecteur des radiosités, et $\mathbf{m}^\circ \in \mathbb{R}^n$ est le vecteur des émittances de corps noir.

Une image est ensuite obtenue en capturant, pour chaque pixel de la caméra, certaines radiances transmises de surface en surface. Par exemple, un modèle de caméra *pinhole* (ou sténopé) peut être utilisé pour déterminer les radiances correspondantes à chaque pixel. En supposant que l'image générée $\mathbf{y}$ est un vecteur qui associe à chaque coordonnée la radiance capturée pour un pixel de la caméra, la valeur pour chaque pixel est une combinaison linéaire des radiances échangées entre chaque surface. Il existe donc une matrice $\mathbf{V}$, capturant tous les paramètres de la caméra (position, orientation, focale, réponse fréquentielle et impulsionnelle spatiale), qui lie les radiances provenant des surfaces aux radiances capturées par chaque pixel de la caméra :

$$\mathbf{y} = \frac{1}{\pi}\mathbf{V}\mathbf{j} \tag{I.9}$$

Pour résoudre le système, il est nécessaire de connaître la géométrie, la température, l'émissivité et la réflectivité des surfaces. Une fois $\mathbf{A}$ et $\mathbf{m}^\circ$ déterminés, $\mathbf{j}$ peut être calculé en résolvant l'équation, soit par inversion de $\mathbf{A}$ (ce qui est computationnellement coûteux pour de grands systèmes, avec une complexité en $O(n^3)$), soit par des méthodes itératives qui évitent l'inversion explicite de la matrice.

La détermination de $\mathbf{j}$ permet de calculer la radiance sortante dans n'importe quelle direction de la scène (la radiance étant la radiosité divisée par π). En utilisant un modèle de caméra virtuelle déterminant la matrice de projection $\mathbf{V}$, une image simulée $\mathbf{y}$ peut être générée à partir de ces radiosités.

I.3.3. Méthodes classiques de résolution de problème inverse appliquées à la thermographie infrarouge inverse

I.3.3.1. Estimation des températures par méthode linéaire

Lorsqu'on suppose que la géométrie de la scène, le placement de la caméra et les émissivités des surfaces sont connus, les températures des surfaces peuvent être estimées à partir d'une image capturée $\mathbf{y}$ par des méthodes de moindres carrés de manière linéaire. Le problème est linéaire par rapport aux émittances de corps noir $\mathbf{m}^\circ$, permettant ainsi une résolution linéaire pour déterminer ces émittances, et par la suite, d'en déduire les températures de surface en utilisant la loi de Planck inverse.

En posant $\mathbf{S}$ la matrice telle que $\mathbf{y} = \mathbf{S}\mathbf{m}^\circ$, calculable d'après les équations de radiosité définies précédemment, on peut utiliser la solution de moindres carrés avec régularisation de Tikhonov [Tik63] :

$$\mathbf{m}^\circ = \left(\mathbf{S}^\top\mathbf{S} + \mathbf{\Gamma}^\top\mathbf{\Gamma}\right)^{-1}\mathbf{S}^\top\mathbf{y} \tag{I.10}$$

Cette méthode minimise la fonction de coût des moindres carrés avec un terme de régularisation :

$$\left\|\mathbf{S}\mathbf{m}^\circ - \mathbf{y}\right\|^2 + \left\|\mathbf{\Gamma}\mathbf{m}^\circ\right\|^2 \tag{I.11}$$

Ici, $\mathbf{\Gamma}$ permet d'ajouter un terme de régularisation, particulièrement utile lorsque le problème est mal posé et nécessite l'introduction de contraintes sur les solutions. Typiquement, le cas $\mathbf{\Gamma} = \lambda \mathbf{I}$ correspond à la *ridge regression* [HK70], empêchant d'obtenir des solutions aux valeurs trop grandes. Il est aussi possible d'utiliser un opérateur pénalisant les discontinuités ou les hautes fréquences dans la solution. Le cas $\mathbf{\Gamma} = \mathbf{0}$ correspond à la solution classique des moindres carrés, utilisant la matrice pseudo-inverse de Moore-Penrose [Pen55].

Ainsi, cette approche permet de résoudre de manière robuste les températures des surfaces en combinant les informations de sensibilité et la régularisation pour gérer les cas où les données sont bruitées ou incomplètes.

I.3.3.2. Estimation de l'émissivité par méthode non linéaire

En supposant que la géométrie, le placement de la caméra et les températures des surfaces sont connus, il est possible d'estimer l'émissivité des surfaces à partir d'une image capturée $\mathbf{y}$ en utilisant les mêmes équations de radiosité. Contrairement à l'estimation des températures, où le problème est linéaire, l'image capturée est une fonction non linéaire des émissivités. Par conséquent, une approche directe comme celle utilisée précédemment ne peut pas être appliquée. Une méthode itérative est alors nécessaire, commençant par un vecteur d'émissivités initiales $\boldsymbol{\varepsilon}_0$ et générant itérativement une nouvelle proposition $\boldsymbol{\varepsilon}_t$ à partir de la précédente.

En posant $S_t = \left[\frac{\partial \mathbf{y}}{\partial \boldsymbol{\varepsilon}_t}\right]$, la matrice jacobienne ou matrice de sensibilité de l'image vis-à-vis des émissivités, calculable à partir des équations de radiosité, on peut utiliser la méthode de Gauss-Newton :

$$\boldsymbol{\varepsilon}_{t+1} = \boldsymbol{\varepsilon}_t - \left(\mathbf{S}_t^\top \mathbf{S}_t\right)^{-1} \mathbf{S}_t^\top \left(\mathbf{f}(\boldsymbol{\varepsilon}_t) - \mathbf{y}\right) \tag{I.12}$$

où $\mathbf{f}(\boldsymbol{\varepsilon}_t)$ représente le calcul de l'image obtenue en prenant $\boldsymbol{\varepsilon}_t$ comme valeurs d'émissivité.

Cette méthode nécessite un calcul et une succession d'inversions de la matrice de sensibilité $\mathbf{S}_t$, ce qui peut être très coûteux en temps et en espace. Des méthodes plus simples, comme une descente de gradient, peuvent être utilisées. En définissant la fonction de résidu $r(\boldsymbol{\varepsilon}) = \left\|\mathbf{f}(\boldsymbol{\varepsilon}) - \mathbf{y}\right\|^2$, la descente de gradient cherche à minimiser $r(\boldsymbol{\varepsilon})$ en se déplaçant par pas successifs dans le sens opposé de son gradient :

$$\boldsymbol{\varepsilon}_{t+1} = \boldsymbol{\varepsilon}_t - \eta_t \nabla r(\boldsymbol{\varepsilon}_t) \tag{I.13}$$

où η_t est le taux d'apprentissage.

Il est également possible d'ajouter un terme de régularisation, similaire au cas linéaire. Par exemple, la méthode de régularisation de Levenberg-Marquardt introduit un terme de régularisation dans le calcul de la pseudo-inverse de $\mathbf{S}_t$, permettant d'interpoler entre une méthode de Gauss-Newton classique et une descente de gradient.

I.3.4. Limites de l'approche par radiosité et de ses méthodes d'inversion

L'approche par radiosité, bien que permettant de modéliser les échanges radiatifs et d'inverser une mesure infrarouge, présente plusieurs limitations qui restreignent son application dans un environnement aussi complexe qu'un tokamak.

Premièrement, l'hypothèse de surfaces purement diffuses constitue une simplification qui peut ne pas être représentative des conditions réelles dans un tokamak. Les surfaces des parois peuvent avoir des propriétés de réflexions spéculaires, rendant l'application des modèles de radiosité moins adéquate.

Deuxièmement, la discrétisation de la scène en un nombre fini de surfaces n limite la résolution spatiale du modèle 3D. Un nombre insuffisant de surfaces discrètes peut entraîner une perte de détails importants dans la distribution des températures, ce qui peut empêcher l'application de cette méthode à une scène aussi complète qu'un tokamak, pouvant exhiber des augmentations très localisées de température. Un modèle de tokamak suffisamment détaillé pouvant atteindre plusieurs millions de triangles, la création de matrices non creuses de taille $n \times n$, où n est le nombre de triangles, est inenvisageable. Des méthodes de raffinement du maillage spécifiquement aux endroits observés et soumis à de forts flux sont toutefois possible pour limiter le nombre de surfaces sans impacter trop fortement les zones critiques [LBLS23].

Troisièmement, la résolution des systèmes d'équations linéaires ou non linéaires issus des équations de radiosité est computationnellement intensive. Pour de grands systèmes, la complexité en $O(n^3)$ de l'inversion matricielle ou les coûts des méthodes itératives comme Gauss-Newton peuvent devenir prohibitifs, nécessitant des ressources computationnelles importantes et un temps de calcul élevé.

Pour ces raisons, une nouvelle approche basée sur une simulation plus réaliste et utilisant des méthodes d'apprentissage automatique a été proposée et est étudiée dans cette thèse.

Conclusion I

La fusion thermonucléaire est une source d'énergie prometteuse, mais sa viabilité pose de nombreux défis techniques. Un de ces défis est la protection des composants internes, soumis à de forts flux et dont il est crucial de contrôler la température. Dans cette première partie, nous avons montré l'importance de l'imagerie infrarouge pour la surveillance des parois, ainsi que les défis numériques que pose l'estimation des températures des composants.

La thermographie infrarouge est une technique non intrusive qui permet de contrôler la température de surface des composants du tokamak en temps réel. Elle est essentielle pour la surveillance des composants face aux flux de chaleur intenses, en particulier dans des zones critiques comme le divertor, où les températures peuvent atteindre plusieurs milliers de degrés Celsius.

Actuellement, la surveillance des surfaces dans les tokamaks tels que WEST est effectuée à partir de régions d'intérêt (ROI) définies dans les images infrarouges. Chaque ROI a des seuils prédéfinis sur les valeurs des pixels pour déclencher des actions spécifiques, comme l'asservissement du système ou la coupure du plasma en cas de dépassement des limites de sécurité. Cependant, en raison des effets de réflexion sur les surfaces métalliques, et de la variabilité de l'émissivité des surfaces, ces valeurs de pixels ne correspondent pas à la température réelle des surfaces observées. Il existe un risque significatif d'interprétation erronée des données, qui peut conduire soit à l'arrêt injustifié du plasma, soit à la sous-estimation de la température des composants, provoquant des dommages potentiels. Pour surmonter ces limitations, on cherche à traiter numériquement les images infrarouges afin de résoudre quantitativement la température réelle des surfaces.

Dans ce chapitre, nous avons montré que dans un tokamak métallique, l'estimation à partir d'images infrarouges des températures de parois, à la fois émettrices et réflectrices, constitue un problème inverse complexe. De nombreux paramètres interviennent en effet dans ce problème : les propriétés d'émissivité et de réflectivité des surfaces, qui sont variables spatialement et au cours du temps, la géométrie de la scène et le positionnement de la caméra. De plus, le simple flux collecté par un pixel de la caméra ne suffit pas à déterminer la température de la surface visible, car le flux réfléchi additionnel dépend de toute la scène.

Une première approche présentée dans ce chapitre consiste à prédire les températures à partir des images en se basant sur un modèle direct différentiable, et en utilisant sa matrice de sensibilité pour inverser le modèle. Cependant, cette méthode présente, à ce stade de développement, des limitations significatives, notamment en ne modélisant que des surfaces émettant et réfléchissant de manière isotrope. Une autre difficulté est l'inversion matricielle dont la grande complexité asymptotique en temps et en espace devient limitante pour traiter des scènes de grande taille.

La suite de cette thèse explorera de nouvelles approches numérique pour la thermographie infrarouge quantitative dans les tokamaks, en se concentrant sur l'application de l'apprentissage automatique pour résoudre ce problème inverse.

II. Apprentissage automatique via des données simulées : méthode employée et preuve de concept

Résumé

Dans le chapitre précédent, nous avons vu que les méthodes numériques sont intéressantes pour l'estimation des températures à partir d'images infrarouges dans des environnements complexes commes les machine de fusion nucléaires. Nous allons voir dans ce chapitre une première preuve de concept de l'utilisation des techniques d'apprentissage automatique pour résoudre ce problème inverse. Après avoir présenté l'état de l'art des techniques de rendu inverse utilisant l'intelligence artificielle dans le domaine du visible, une méthode d'inversion d'image infrarouge par réseaux de neurones, utilisant des données d'apprentissage simulées, est présentée. La faisabilité de ce concept est illustrée et démontrée sur un cas simplifié d'un tokamak constitué de miroirs parfaits (réfléxions spéculaires). Ces développements ont fait l'objet d'une publication [Juv+22] et ont contribué à deux autres publications [Aum+22 ; Bru+23].

Sommaire

II.1. Principe de l'inversion par réseau de neurone grâce à des données simulées 31
II.1.1. État de l'art des méthodes d'apprentissage dans le domaine du rendu inverse 31
II.1.2. Apprentissage automatique assisté par simulation pour la mesure de température dans les tokamaks 32
II.2. Démonstration de la méthode sur un cas simplifié 33
II.2.1. Modèle de lancer de rayons par réflexions purement spéculaire 33
II.2.2. Génération du jeu de données synthétique 35
II.2.3. Apprentissage automatique pour l'inversion 36
II.2.4. Résultats sur données synthétiques purement spéculaires 38

II.1. Principe de l'inversion par réseau de neurone grâce à des données simulées

Les méthodes d'apprentissage automatique, plus précisément les réseaux de neurones artificiels, ont montré au cours des dernières années leur efficacité pour le traitement d'images, dans (par exemple, la détection d'objet, reconnaissance de forme, etc.), et ceci même sous une contrainte de traitement en temps réel de flux vidéo. Dans cette thèse, nous proposons d'utiliser cette technique pour résoudre un problème inverse d'estimation de température à partir d'images infrarouge.

L'apprentissage automatique, ou machine learning, est un domaine de l'intelligence artificielle qui permet aux systèmes informatiques d'apprendre et de s'améliorer automatiquement à partir de l'expérience sans être explicitement programmés. Il repose sur le développement d'algorithmes capables d'identifier des motifs dans des ensembles de données et de faire des prédictions ou des décisions basées sur ces motifs.

Dans le cadre de cette thèse, nous utiliserons plus particulièrement les réseaux de neurones artificiels. Ce sont des approximateurs universels de fonctions : ils sont capables de représenter des fonctions hautement non linéaires, ce qui permet d'extraire les relations complexes entre les données d'entrée (ici les images infrarouges) et celles de sortie (ici les paramètre physiques comme les températures ou émissivités).

II.1.1. État de l'art des méthodes d'apprentissage dans le domaine du rendu inverse

Le problème que nous cherchons à résoudre durant cette thèse s'approche fortement d'une thématique explorée dans l'image visible : le rendu inverse (*inverse rendering*). Le rendu inverse vise à décomposer une image en un ensemble de paramètres qui définissent la scène observée : normale et profondeur des surfaces, albédo, éclairage, etc. Cela revient donc à faire l'inverse de ce qu'un algorithme de rendu ferait.

Le rendu inverse a suscité un grand intérêt dans la communauté des *computer graphics*. Par exemple, il permet d'extraire, à partir d'une image réelle, des profils d'albédo et de réflectivité qui peuvent être réappliqués dans une scène pour générer des graphismes informatiques réalistes. Cela permet, par exemple, de réinsérer un objet dans une scène tout en lui appliquant un éclairage approprié, ou de remplacer un visage par un autre.

La faisabilité du rendu inverse a été étudiée théoriquement au début des années 2000 dans le cadre de l'estimation des modèles de réflectivité (BRDF) d'un objet [RH01] ou dans l'estimation des paramètres d'une scène [SMK05].

Les applications récentes utilisent l'apprentissage automatique, et plus particulièrement les réseaux de neurones convolutifs. Le problème de ces méthodes réside dans le manque de données réelles annotées. Pour contourner ce problème, de nombreuses applications [Sen+19 ; Li+20 ; Li+22] se basent sur la création d'un jeu de données synthétiques, créées par rendu physique réaliste (*physically based rendering*), ce qui permet d'avoir à la fois l'image et les informations concernant les paramètres d'entrée de la simulation. Souvent, une phase d'apprentissage supplémentaire, dite auto-supervisée, est réalisée en utilisant des données réelles comme entrée, et un

modèle direct différentiable qui permet de vérifier que les paramètres estimés par le réseau génèrent une image synthétique proche de celle fournie en entrée. Cet apprentissage auto-supervisé peut être réalisé après la première phase d'apprentissage sur les données simulées [Wan+21] ou simultanément [LS18]. Le modèle direct peut être simple [Jan+17], ou utiliser un réseau de neurones conçu pour apprendre le modèle direct de simulation tout en étant différentiable [Li+18].

Nous utiliserons la même approche pour résoudre le problème de thermographie infrarouge : nous nous appuierons sur une simulation réaliste pour constituer une base de données synthétique qui servira d'entraînement au réseau de neurones convolutif.

II.1.2. Apprentissage automatique assisté par simulation pour la mesure de température dans les tokamaks

L'objectif est de réaliser un apprentissage supervisé, ce qui nécessite de fournir au modèle une association entre les données d'entrée et les sorties attendues. Un défi majeur dans le cas des machines de fusion est le manque de données annotées : nous n'avons pas d'accès à la vérité terrain, c'est-à-dire aux températures exactes des surfaces des parois du tokamak à partir des images infrarouges expérimentales.

Nous allons donc utiliser la simulation pour constituer un jeu de données annotées liant l'image infrarouge prise par la caméra aux températures de surface.

La première étape est donc de développer un modèle direct permettant de réaliser des simulations infrarouges aussi proches que possible des données expérimentales. Cela implique de générer suffisamment de données, et cela, rapidement pour constituer une base de données conséquente pour l'apprentissage automatique, et de s'assurer que les images simulées sont suffisamment réalistes pour que le réseau puisse, après un apprentissage basé uniquement sur des données simulées, être directement utilisable sur des données réelles.

L'approche abordée durant la thèse repose alors sur trois points clefs :

1. Le développement d'un modèle direct capable de simuler rapidement des données infrarouges synthétiques les plus réalistes possible.
2. La modélisation des diverses scènes thermiques à fournir en entrée de notre modèle direct afin de créer une base de données simulée qui couvre de manière cohérente et représentative l'ensemble de scénarios possibles.
3. L'utilisation de modèles de réseaux de neurones profonds et leur apprentissage sur les données simulées pour résoudre le problème inverse et fournir une association fiable entre les images infrarouges et les températures des surfaces.

Ces trois étapes sont résumées en Figure II.1.

Dans la prochaine section, nous allons démontrer la démarche dans un cas simple, ce qui permettra une première validation de l'approche proposée dans cette thèse et la mise en place initiale des différentes briques logicielles, qu'il s'agira par la suite de complexifier dans l'objectif de résoudre à terme le problème sur des données expérimentales.

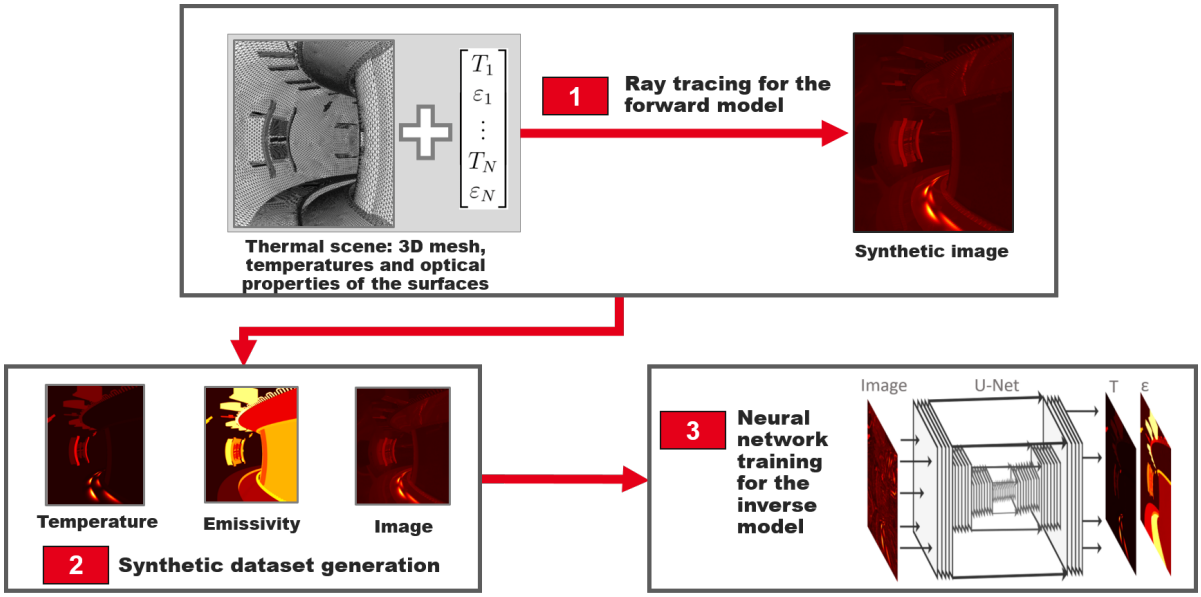


FIGURE II.1. – Principe général de la méthode employée durant la thèse pour fournir une méthode numérique estimant les températures des surfaces à partir d'une image infrarouge. Un modèle direct permet de simuler une image infrarouge synthétique à partir d'un modèle de tokamak virtuel, et de profils de température, d'émissivités et de réflectivités appliqués au modèle. Ce modèle est utilisé pour créer un large jeu de données synthétiques en faisant varier les différents paramètres. Ce jeu de donné est enfin utilisé pour entraîner un réseau de neurones artificiel, afin de lui faire réaliser la tâche inverse de l'étape de simulation.

II.2. Démonstration de la méthode sur un cas simplifié

Nous présentons ici une preuve de concept permettant d'illustrer dans un cas simple la méthode qui sera utilisée durant la thèse. Cette preuve de concept se repose sur la génération de données simulées supposant des surfaces dont les réflexions se comportent comme des miroirs (pures spéculaires). Ces données sont utilisées pour entraîner un réseau de neurone adapté au traitement d'image, afin qu'il puisse, à partir d'une image comportant des réflexions, renvoyer des cartes contenant les paramètres utilisés lors de la simulation.

Les prochains chapitres détailleront en profondeur la manière de générer des données simulées (chapitre III) et le modèle d'apprentissage automatique utilisé (chapitre IV).

II.2.1. Modèle de lancer de rayons par réflexions purement spéculaire

Pour démontrer la faisabilité de la méthode sur un cas simple, nous développons ici un programme de lancer de rayons déterministe. Il propage les rayons dans une scène 3D représentée par un maillage triangulaire surfacique, en supposant que les réflexions des rayons sur les surfaces se font de manière purement spéculaire, c'est-à-dire comme sur un miroir. L'image est générée par *backward ray tracing* : un rayon défini par un point de départ et une direction $\mathbf{u}_i$ est lancé à travers un modèle de caméra virtuelle, dit caméra *pinhole* (voir section III.3.1). L'intersection des rayons avec la géométrie de la scène thermique 3D est calculée à l'aide de l'outil Intel

Embree [Wal+14], qui implémente une primitive d'intersection rayon-triangle parallélisée sur CPU. À chaque intersection, une nouvelle direction $\mathbf{u}_r$ est calculée selon la loi de Snell-Descartes décrite par l'équation (II.1).

$$\mathbf{u}_r = \mathbf{u}_i - 2\mathbf{n}\mathbf{n}^\top \mathbf{u}_i \tag{II.1}$$

Ici, $\mathbf{u}_r$ est le vecteur directionnel des rayons réfléchis, $\mathbf{u}_i$ est le vecteur directionnel des rayons incidents, et $\mathbf{n}$ est la normale de la face intersectée. Ces vecteurs sont représentés en Figure II.2.

Une fois le calcul des chemins de rayons effectué, la radiance sortante à chaque face intersectée est calculée selon l'équation du transport de la lumière suivante, en supposant une surface spéculaire, comme montré dans l'équation (II.2).

$$L_i = \varepsilon_i L^\circ(T_i) + (1-\varepsilon_i) L_{i+1} \tag{II.2}$$

Avec L_i représentant la radiance sortante à l'intersection i, ε_i l'émissivité de la face intersectée, $L^\circ(T_i)$ le flux émis calculé à partir de la loi de Planck (voir section III.4.1) en fonction de la température de la face T_i, et L_{i+1} le flux incident provenant de la face intersectée suivante. Le calcul de la radiance est illustré en Figure II.2.

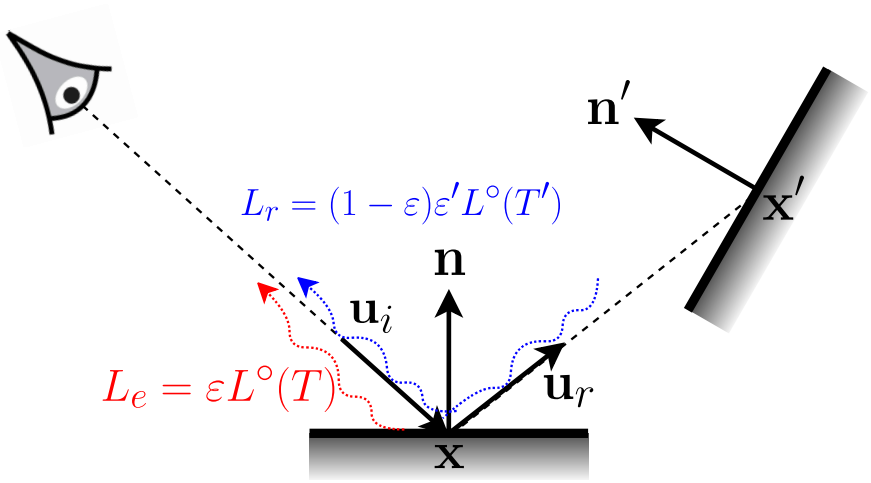


FIGURE II.2. – Représentation de la radiance capturée par un observateur dans le cas de réflexions purement spéculaires, sur une scène simple composée de deux surfaces. La radiance capturée est une somme d'une radiance émise par la surface observée, et d'une radiance réfléchie provenant de l'autre surface. Ces radiances dépendent des températures des surfaces, mais sont pondérées par leurs émissivités et réflectivités.

Pour chaque rayon, la liste des températures et des émissivités $(T_0, \varepsilon_0, T_1, \varepsilon_1, \ldots)$ des différentes faces intersectées est enregistrée et le flux total collecté par le pixel de la caméra est calculé. Enfin, la radiance finale sur chaque pixel de la caméra est fournie en moyennant la radiance calculée à partir de 16 rayons lancés depuis des positions espacées uniformément dans le pixel. Cela permet d'être plus précis en cas de système d'imagerie à basse résolution et de scène thermique non uniforme.

L'algorithme d'intersection de rayons est exécuté par le CPU en parallèle sur plusieurs *threads* (fils d'exécution), nécessitant un temps de calcul de quelques secondes à quelques dizaines

de secondes. Cependant, une fois le calcul géométrique pur des chemins de rayons effectué, il peut être utilisé pour générer autant d'images que souhaité en ne variant que les propriétés des surfaces. Les calculs de radiance des rayons sont parallélisés sur le GPU en utilisant des *kernels* CUDA [Nic+08], ce qui permet de générer un grand nombre d'images rapidement (environ une à cinq images par seconde). Cela permet de créer un grands ensembles de données couvrant une large gamme de scénarios de propriétés de surface.

II.2.2. Génération du jeu de données synthétique

La scène virtuelle est observée depuis une vue grand angle tangente du tokamak WEST. Le modèle 3D est divisé en 36 composants. Chaque composant se voit attribuer une température et une émissivité uniformes, sauf pour le divertor. Le divertor est un composant clé recevant la charge thermique maximale (jusqu'à 10 à 20 $\mathrm{MW \cdot m^{-2}}$ en régime permanent) et présente un profil de température approximativement modélisé à partir de l'équation d'Eich (voir section III.8.4). Une telle scène thermique est décrite avec 75 paramètres (36 émissivités par composants, 35 températures sur les composants à température uniforme et 4 paramètres pour définir la distribution de température sur le divertor). Un modèle sténopé de caméra (ou *pinhole*, voir section III.3.1) est utilisé pour reproduire l'image infrarouge expérimentale, 640 par 512 pixels dans la plage de longueurs d'onde [3,9 – 4,1 µm]. Un exemple d'image générée peut être vu à la Figure II.3 à côté d'un exemple d'image capturée lors des expériences.

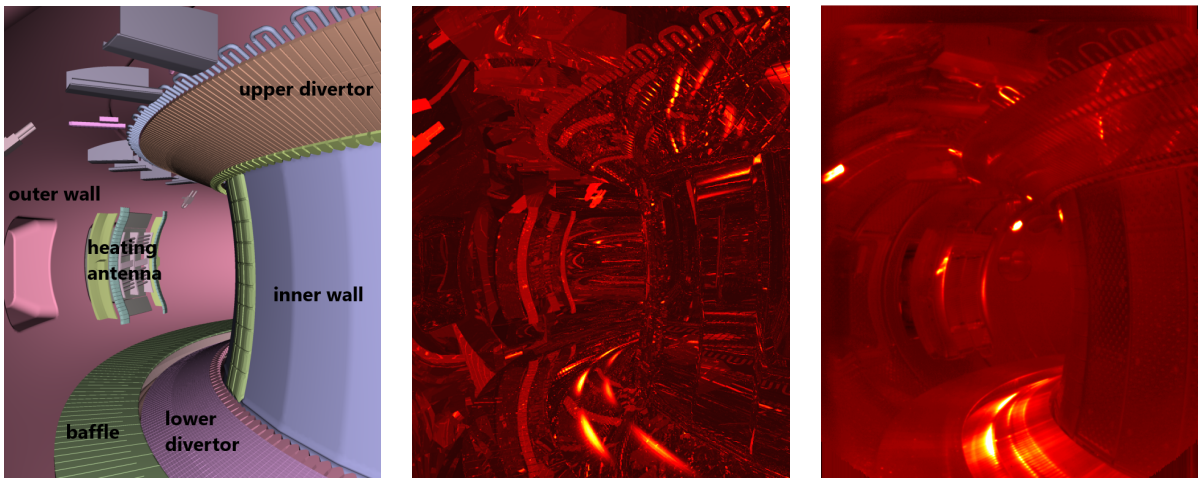


FIGURE II.3. – Gauche – Modèle 3D du tokamak vu du point de vue de la caméra grand angle, utilisant une couleur différente par composant. **Milieu –** Images infrarouges simulées à partir du modèle supposant des réflexions purement spéculaires. **Droite –** Image expérimentale de la caméra grand angle de WEST pour une comparaison qualitative avec la simulation. Pour les deux images, on peut observer une réflexion du motif du divertor bas sur le divertor haut.

Une première base de données de 15 000 images a été réalisée à partir du modèle de simulation en tirant aléatoirement les 75 paramètres d'entrée. Les scénarios couverts par cette base de données simulée sont basés sur les expériences du tokamak WEST : la température de la première

paroi est comprise entre 90 °C et 110 °C, la température des composants les plus proches de l'enceinte (*bumpers*, protections d'antenne) varie entre 300 °C et 500 °C. La température maximale du profil de température du divertor peut atteindre 1800 °C. La plage d'émissivité des matériaux est fixée en fonction des mesures expérimentales : entre 0,05 et 0,5. Enfin, l'échantillonnage en hypercube latin [MBC00] est utilisé pour couvrir de manière optimale la plage des paramètres.

La base de données utilisée pour l'apprentissage machine est composée de triplets d'images :

— l'image de la caméra simulée en température apparente, qui est la température corps noir équivalente, c'est-à-dire en supposant une émissivité égale à 1. Cette image est obtenue à partir de l'image de radiance (en $\mathrm{W}\cdot\mathrm{m}^{-2}\cdot\mathrm{sr}^{-1}$) calculée à partir du modèle direct et en inversant la loi de Planck ;

— l'image de température réelle, donnant la véritable température de surface pour chaque pixel de la caméra ;

— et l'image d'émissivité réelle, donnant l'émissivité des surfaces pour chaque pixel.

Les images de température et d'émissivité réelles (cartes 2D) sont obtenues de la scène thermique 3D utilisée comme entrée du code de lancer de rayons par une opération de projection 3D à 2D. Après conversion des points 3D de la scène thermique en points 2D sur le plan de vue de la caméra, la température réelle (ou l'émissivité) est calculée pour chaque pixel en moyennant la température (ou l'émissivité) associée des points 3D. Un exemple de triplet stocké dans le jeu de donné est montré en Figure II.4.

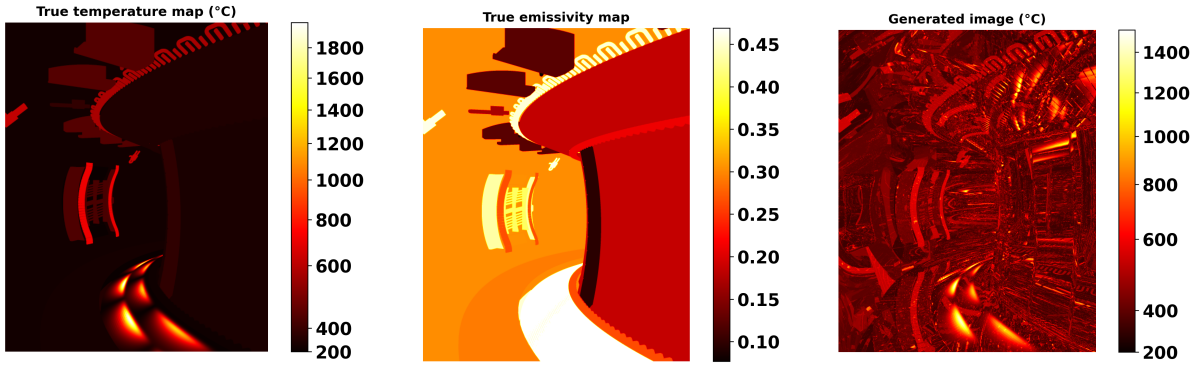


FIGURE II.4. – Exemple de triplets stockés dans le jeu de données : deux cartes de température et d'émissivité, obtenues par projection du champ 3D pour créer une image 2D, ainsi que le résultat de la simulation pour ces paramètres d'entrée. **Gauche –** Carte des températures des surfaces. **Milieu –** Carte des émissivités des surfaces. **Droite –** Image générée à partir de ces températures et émissivités en supposant des réflexions pure spéculaire. La radiance capturée ($\mathrm{W}\cdot\mathrm{m}^{-2}\cdot\mathrm{sr}^{-1}$) est traduite en température apparente (°C), c'est-à-dire la température que devrait avoir un corps noir pour émettre la même radiance que celle capturée.

II.2.3. Apprentissage automatique pour l'inversion

L'apprentissage automatique supervisé est effectué à partir d'un ensemble de données simulées afin de proposer un algorithme inverse qui déduit la carte de température et d'émissivité des surfaces visibles à partir d'une image de scène thermique.

Un réseau de neurones convolutif, inspiré de l'architecture U-Net [RFB15] (voir section IV.2.4.2), est implémenté. Un tel réseau permet d'apprendre à reproduire des fonctions de type "image vers image", et est particulièrement adapté dans notre cas où les images de sortie (cartes de température ou d'émissivité) contiennent des détails similaires et une localisation des objets identique à l'image d'entrée (carte de température apparente). L'architecture du réseau est adaptée à notre contexte. Les tailles d'image correspondent à celles de la caméra infrarouge : 512 pixels de large par 640 de haut, et le nombre d'opérations de *downsampling* (voir section IV.2.3.2) est augmenté (6 au lieu de 4 dans le modèle original) afin de propager une information globale à chaque pixel de sortie, ce qui est essentiel pour traiter la propriété non locale des réflexions. Étant donné que le réseau est utilisé pour une tâche de régression, aucune fonction d'activation n'est utilisée pour la couche de sortie.

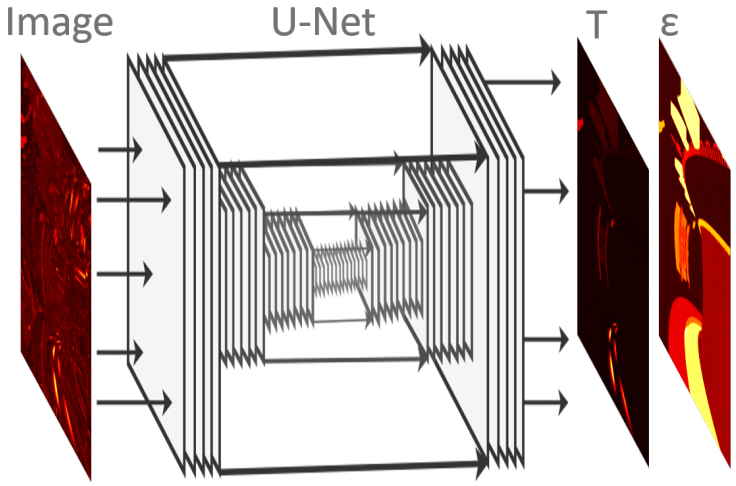


FIGURE II.5. – Représentation du réseau de neurone utilisé pour l'inversion des images simulées avec des réflexions purement spéculaires. L'architecture est issue du U-Net, un réseau de neurone artificiel qui applique une succession de filtres de convolution à une image, pour en renvoyer une autre. Le réseau prend en entrée l'image simulée contenant des réflexions, et renvoie une image à deux canaux : une pour les températures des surfaces, l'autre pour les émissivités.

L'architecture du réseau est schématisée en Figure II.5. Le réseau reçoit en entrée une image à canal unique (carte de température corps noir apparente) et renvoie une image à deux canaux (cartes de température et d'émissivité des surfaces visibles). Ces images sont normalisées dans l'intervalle $[0\,;1]$ en utilisant les valeurs minimales et maximales rencontrées dans l'ensemble d'apprentissage.

L'ensemble de données de 15 000 scènes a été divisé en un ensemble d'apprentissage de 10 000 éléments, un ensemble de validation de 2 000 éléments et un ensemble de test de 3 000 éléments. L'apprentissage a été effectué en utilisant une descente de gradient stochastique (voir section IV.1.2.2) en utilisant l'optimiseur Adam (section IV.1.2.3) sur la perte d'erreur quadratique moyenne pendant 100 *epochs*. Des *batchs* de taille 10 ont été utilisés, et mélangés aléatoirement après chaque époque. Le réseau avec le taux d'erreur le plus bas sur l'ensemble de validation est utilisé pour évaluer ses performances sur l'ensemble de test.

II.2.4. Résultats sur données synthétiques purement spéculaires

Après l'apprentissage, le réseau est évalué sur l'ensemble de test de 3 000 images simulées. L'erreur moyenne de température est calculée à partir des 3 000 images de test sur chaque composant principal de l'enceinte. L'erreur moyenne de température pour les composants de l'enceinte à température uniforme est de 4,8 °C, et l'erreur relative moyenne de 4,3 %. Ces résultats constituent un net progrès par rapport à la méthode classique de température apparente, dont l'erreur de température moyenne est estimée à 41 °C en supposant des corps noirs d'émissivité de 1 et sans corrections des réflexions.

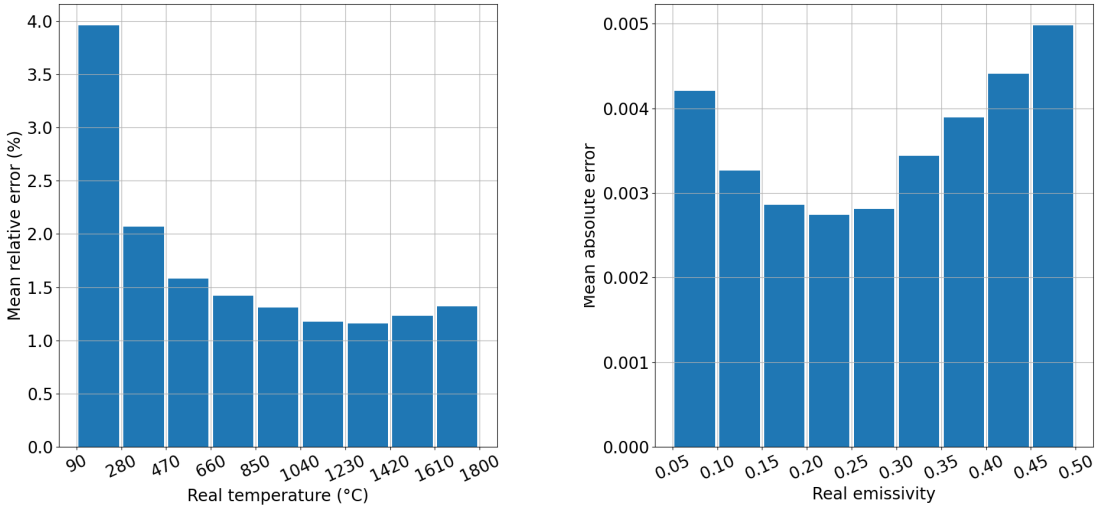


FIGURE II.6. – Erreur pour l'ensemble de test constitué d'images simulées de la caméra grand angle en réflexions purement spéculaires. Les erreurs sont uniquement calculées sur les pixels correspondant au divertor bas. **Gauche –** Erreur relative moyenne (%) pour l'estimation de la température par plage de température. **Droite –** Erreur absolue moyenne pour l'estimation de l'émissivité par plage d'émissivité.

La Figure II.6(Gauche) montre l'erreur de température pour le divertor, caractérisé par une température non uniforme. L'erreur relative moyenne est présentée pour différentes plages de température réelle. Les températures les plus basses sont plus sensibles aux réflexions, tandis que les températures les plus élevées seront sensibles à l'inexactitude de l'émissivité. Les erreurs de température sont plus importantes pour les températures plus basses (<200 °C) bien qu'elles soient dans les spécifications d'ITER (erreur de 20 % requise pour les températures inférieures à 400 °C et erreur de 10 % pour les températures supérieures). Les températures les plus élevées sont très bien estimées avec une erreur inférieure à 1,5 %, ce qui suggère que l'émissivité doit également être bien évaluée par les algorithmes. Cela est confirmé par la Figure II.6(Droite), qui montre l'erreur absolue de l'émissivité en fonction de l'émissivité réelle de la cible : la valeur de l'émissivité est estimée avec une précision meilleure que 0,05.

Conclusion II

Ce chapitre a présenté la méthode générale adoptée pour cette thèse, visant à prédire quantitativement la température de surface à partir d'une image infrarouge en utilisant l'apprentissage automatique. Pour cela, un jeu de données synthétique a été créé à partir d'un modèle simplifié et rapide pour entraîner un réseau de neurones à convolution de type U-Net.

Ce modèle simplifié suppose des surfaces lisses, purement spéculaire, se comportant comme un miroir. La propagation du rayon dans la géométrie complexe des tokamaks en est alors simplifié et se ramène à un calcul déterministe beaucoup plus rapide capable de générer une grande base de données à un temps réduit.

À partir de cette base de données simulées, un réseau de neurones de type U-Net a été entraîné pour estimer les températures et les émissivités à partir d'images infrarouges. Il est démontré que, dans ces conditions, la température et l'émissivité peuvent être estimées avec une grande précision (respectivement mieux que 5 % et 0,05).

Conforté par les résultats de cette première étude, l'étape suivante est désormais de s'approcher des images infrarouges expérimentales en prenant en compte les propriétés de surfaces plus complexes (c'est-à-dire non purement spéculaires). Cela va nécessiter le développement d'un nouveau code de lancer de rayons (non déterministe) capable de simuler des images infrarouges dans des conditions réelles avec des matériaux rugueux et des scènes thermiques plus complexes. Et tout cela, en conservant des temps de calcul raisonnables pour générer une base de donnée suffisamment conséquente pour l'apprentissage du réseau de neurones. C'est l'objet du chapitre III, qui présente la nouvelle simulation des images infrarouges, avant de consacrer les deux autres chapitres suivants (IV et V) à la résolution du problème inverse par réseaux de neurones.

III. Simulation d'images infrarouge par lancer de rayons

Résumé

Dans le chapitre précédent, l'inversion par réseaux de neurones a montré des résultats encourageants sur un modèle de lancer de rayons déterministe purement spéculaire. Cependant, ce modèle simple de réflexions génère des images encore très éloignées des images réelles : il est nécessaire de développer un programme de simulation d'une part réalisant un rendu physiquement réaliste, prenant en compte des propriétés de réflexions complexes, et d'autre part suffisamment rapide pour permettre de générer des images en grande quantité.
Ce chapitre présente les exigences d'un modèle de lancer de rayons en infrarouge par rapport à la modélisation classique de l'imagerie visible. Il détaille le modèle de lancer de rayons développé au cours de la première partie de cette thèse pour répondre aux besoins spécifiques de l'infrarouge, ainsi que les temps de calcul nécessaires pour constituer une base de données représentative des scènes dans les machines de fusion, essentielle pour l'apprentissage des réseaux de neurones qui sera abordé dans le chapitre suivant. La dernière section est consacrée à la modélisation spécifique de la scène thermique dans le tokamak WEST.
Ce modèle a été utilisé pour la simulation des images dans plusieurs publications :[JAM24b ; Aum+24 ; Tic+24].

Sommaire

III.1. État de l'art des modèles directs existants . . . 42
III.2. Les différents éléments du modèle direct . . . 43
III.3. La caméra virtuelle . . . 43
III.3.1. Le modèle de caméra *pinhole* . . . 43
III.3.2. Modèle de distorsion . . . 45
III.3.3. Alignement caméra . . . 46
III.4. Équations de rendu infrarouge et modèle de BRDF . . . 46
III.4.1. La radiance émise : modèle du corps noir et émissivité . . . 47
III.4.2. La radiance réfléchie : modèle de la BRDF . . . 49
III.4.2.1. Définition et propriétés physiques . . . 49
III.4.2.2. Modèles simples de BRDF . . . 51

III.4.2.3. Les BRDFs par micro-facettes 52
III.4.3. L'équation du rendu infrarouge à résoudre 54
III.5. Résolution de l'équation du rendu par intégration de Monte Carlo 54
III.6. Intersection accélérée entre un rayon et le modèle 3D 57
III.6.1. Intersection entre un rayon et le modèle 3D 58
III.6.1.1. Algorithme d'intersection rayon-triangle 58
III.6.1.2. Recherche rapide du premier triangle intersecté 59
III.6.2. Accélération des calculs : parallélisme sur carte graphique 60
III.7. Validation du modèle direct 61
III.7.1. Validation de l'implémentation de la BRDF dans le rendu Monte Carlo . . . 61
III.7.2. Validation sur une scène fermée à température uniforme 63
III.8. Modélisation des scènes thermiques 63
III.8.1. Le modèle 3D du tokamak et les caméras 64
III.8.2. Température uniforme par composants 65
III.8.3. Mélange de BRDF Phong et miroir 66
III.8.4. Profil de température du divertor bas 66
III.8.5. Profil de température des antennes de chauffage 68
III.8.6. Non-uniformité des composants par bruit de gradient et points chauds . . 68
III.8.7. Base de donnée d'images générées 70

III.1. État de l'art des modèles directs existants

Plusieurs modèles directs, permettant se simuler des images infrarouges à partir de températures et de propriétés optiques de surface, sont actuellement utilisés pour simuler des images infrarouges dans les dispositifs de fusion.

La première solution historiquement utilisée pour la simulation d'images infrarouges dans les machines de fusion est le logiciel SPEOS, commercialisé par la société ANSYS. Il s'agit d'un programme de lancer de rayons par Monte Carlo capable de propager les rayons dans une géométrie 3D en tenant compte des propriétés radiatives et optiques complexes des matériaux, en utilisant la Fonction de Distribution Bidirectionnelle de Réflectance (BRDF) pour décrire la dépendance angulaire et spectrale de la réflectance des matériaux. Ces simulations détaillées sont généralement utilisées pour aider à l'interprétation infrarouge (de manière qualitative) dans les dispositifs de fusion actuels tels que WEST et ASDEX-Upgrade [Aum+21], et pour prédire les performances des futurs systèmes d'imageries visible et infrarouge d'ITER [Aum+17]. Cependant, ce programme nécessite un temps de calcul élevé, avec un parallélisme CPU limité à peu de cœurs (typiquement 12 heures sur un CPU à 4 cœurs), ce qui restreint fortement la quantité de données pouvant être générée, limitant ainsi la possibilité de créer un vaste ensemble de données. De plus, ce modèle est limitant du fait de son incapacité à gérer des surfaces d'émissivités variable spatialement, ce qui contraint à utiliser des objets uniformes en termes de propriétés optiques.

À l'opposé, des modèles plus rapides simplifient les propriétés de réflexion, en adoptant des hypothèses de surfaces purement diffuses, comme la méthode des radiosités décrite en section I.3.2 [Tal+20 ; Tal+21]. Cette forte hypothèse de surface diffuse n'est pas tout à fait pertinente pour les matériaux dans les dispositifs de fusion, qui ont souvent une composante spéculaire significative. À l'inverse, les modèles purement spéculaires, tels que vus en section II.2, bien que simplifiés, ne reproduisent pas pour autant le comportement réel des réflexions.

Dans le domaine du rendu physique, des options libres telles que Mitsuba 3 et Raysect existent. Mitsuba [ND+19] utilise notamment le lancer de rayon différentiable pour le rendu inverse, et Raysect a déjà été utilisé, dans le cadre de la fusion, comme modèle direct permettant une inversion [Car+19]. Cependant, ces modèles font l'hypothèse que les objets sont soit des sources, soit des réflecteurs et ne sont donc pas adaptés à notre cas où les objets de la scène thermique peuvent être à la fois source d'émission et réflecteurs. De plus, lorsqu'un objet est une source, il est supposé émettre de manière uniforme sur toute sa surface. Dans notre cas, nous voulons pouvoir, de manière continue, spécifier les propriétés d'émission et de réflexion pour chaque objet, comme un champ 3D.

Le problème avec les algorithmes de rendu utilisés pour le visible réside dans leur distinction entre sources et réflecteurs. Les scènes à rendre sont souvent considérées comme étant constituées d'un ensemble restreint de sources, qui sont de purs émetteurs d'une radiance uniforme, ce qui permet l'utilisation de techniques telles que le *bidirectional path tracing* [LW93] et les algorithmes de Metropolis [VG97], qui estiment la lumière perçue en générant un ensemble de chemins reliant les sources à la caméra. Cependant, dans le cas de l'infrarouge, cette approche est insuffisante. Nous devons modéliser un champ de température sur toute la scène, déterminant un champ de radiance, et les propriétés optiques de surface doivent être modulables à l'échelle du triangle. Nous devons donc simuler la propagation des rayons dans un environnement où tout est à la fois source de lumière et réflecteur, et ce, de manière rapide pour la génération massive de

données. Pour cela, nous voulons bénéficier du parallélisme massif offert par les GPU. De plus, il est crucial de maîtriser les modèles utilisés dans le rendu et de ne pas dépendre d'une boîte noire, car toute erreur ou approximation à ce niveau pourrait biaiser l'apprentissage du réseau.

Pour toutes ces raisons, il est pertinent de développer notre propre modèle direct, adapté aux exigences spécifiques du rendu de scènes de fusion thermonucléaire dans le domaine infrarouge.

III.2. Les différents éléments du modèle direct

Le programme de lancer de rayon (*ray tracer*) développé durant cette thèse reçoit différentes entrées, et se repose sur plusieurs modules principaux pour générer une image infrarouge simulée. L'ensemble de ce processus est résumé par la Figure III.1.

Les entrées requises sont les suivantes :

— Un modèle 3D de la scène, réalisé ici en utilisant un maillage surfacique, où la scène est ainsi décrite par un ensemble de triangles. Dans le cas du tokamak WEST, nous disposons déjà d'un maillage détaillé de ce type.
— Un modèle de caméra, placé dans le modèle 3D. Les paramètres de la caméra détermineront la zone du modèle observé ainsi que la plage de longueur d'onde.
— Une distribution en température, associant une température à chaque point de la surface du modèle.
— Les propriétés optiques des surfaces, déterminant en chaque point la manière dont la lumière interagit avec la surface lors des réflexions, ainsi que lors de l'émission de rayons.

Les différents modules nécessaires pour obtenir une image à partir de ces éléments sont les suivants :

— le modèle de caméra, décrit en section III.3.
— le modèle d'émission et de réflexion, détaillé en section III.4.
— la méthode de propagation aléatoire des rayons pour simuler les différentes réflexions, détaillée en section III.5.
— la primitive fondamentale d'intersection entre un rayon virtuel et le modèle 3D, détaillée en section III.6.

III.3. La caméra virtuelle

La simulation d'images infrarouges à partir d'un modèle 3D nécessite la modélisation d'une caméra virtuelle. Cette section explore les différents aspects de cette modélisation, en commençant par le modèle de caméra *pinhole*, une approche simple pour projeter des points 3D sur un plan image 2D, et inversement. Nous aborderons ensuite l'intégration des distorsions optiques pour rendre les images plus réalistes, et enfin, les méthodes d'alignement qui permettent d'ajuster la caméra virtuelle sur les images obtenues par une caméra réelle.

III.3.1. Le modèle de caméra *pinhole*

Le modèle de caméra *pinhole*, ou sténopé, est un modèle simple de caméra virtuelle permettant de réaliser une projection des points 3D de la scène sur le plan 2D de l'image. Dans ce modèle,

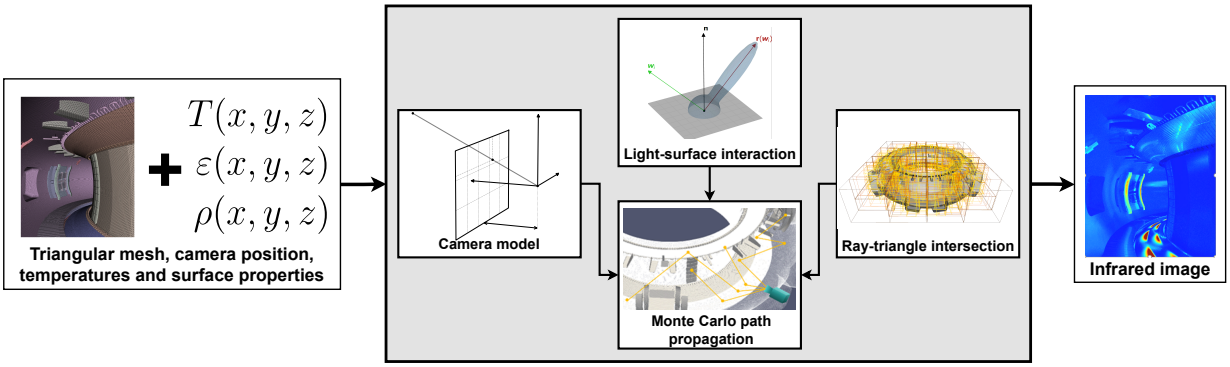


FIGURE III.1. – Schéma général du lancer de rayon. Le *ray tracer* reçoit un maillage triangulaire pour modéliser la scène 3D, le placement de la caméra, un profil de températures et de propriétés optiques des surfaces. Différents modules interagissent entre eux pour propager des rayons dans la scène afin de générer une image infrarouge.

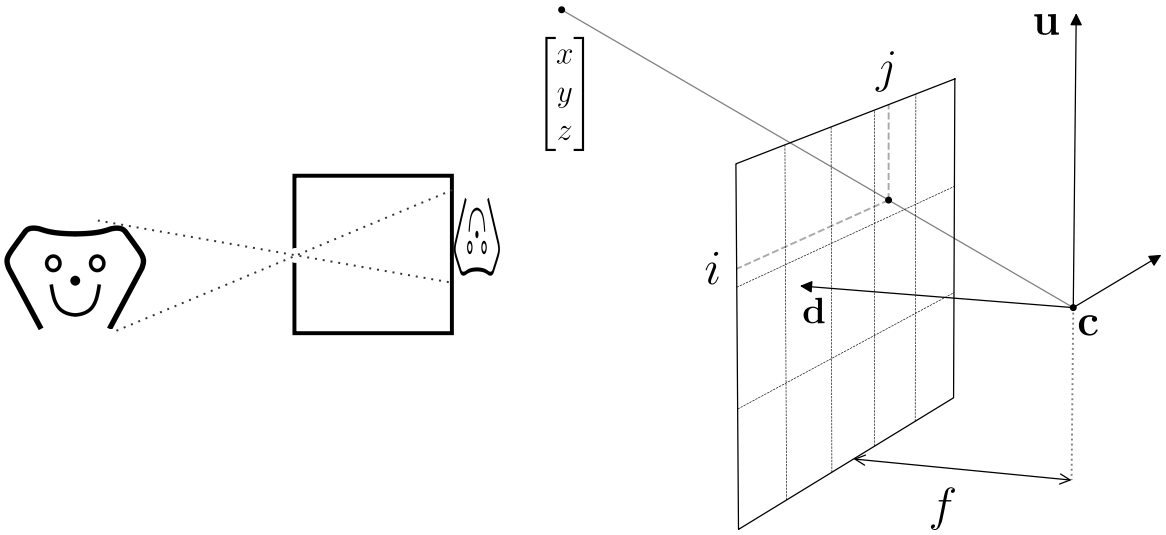


FIGURE III.2. – Représentation du modèle de caméra *pinhole*, modélisant le cas d'une pupille parfaite. Le modèle associe à chaque point (x, y, z) des coordonnées de pixel (i, j) en le projetant sur un plan situé à distance focale f de la pupille. Inversement, chaque pixel (i, j) est associé à une direction observée par la caméra depuis sa pupille.

la pupille de la caméra est un point de projection unique, et le capteur de la caméra est sur un plan de projection situé à une distance focale définie de cette pupille. Ainsi, cette simplification suppose que tous les rayons lumineux frappant un pixel passent par ce point unique : pour un point situé sur le plan focal, une direction unique, reliant ce point à la pupille, détermine le trajet du rayon lumineux. Le modèle *pinhole* fournit ainsi une transformation géométrique qui projette un point 3D (x, y, z) sur un plan image en coordonnées pixel (i, j). Inversement, il permet de déterminer la direction observée à partir des coordonnées (i, j).

En coordonnées homogènes, l'opération de projection 3D vers 2D devient linéaire, comme décrite en équation (III.1).

$$s\begin{bmatrix} i \\ j \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{III.1}$$

$\mathbf{R} \in \mathbb{R}^{3\times3}$ est une matrice de rotation, $\mathbf{t} \in \mathbb{R}^3$ est un vecteur de translation, tous deux dépendant de la position et de l'orientation de la caméra (appelés paramètres extrinsèques), $\mathbf{K} \in \mathbb{R}^{3\times3}$ est une matrice permettant de transformer le système de coordonnées sur le plan focal, prenant en compte la focale et le système de coordonnées en termes de pixels (paramètres intrinsèques), et s est la dernière coordonnée du vecteur obtenue en appliquant la transformation. Cette formulation est pratique pour réaliser rapidement l'opération de projection, ainsi que pour la calibration caméra [Zha00]. Une fois ces matrices déterminées, le calcul des coordonnées (i, j) s'effectue en appliquant ces matrices au vecteur homogène des coordonnées, puis en divisant par s, la dernière coordonnée du vecteur.

III.3.2. Modèle de distorsion

Cependant, les lentilles des caméras réelles introduisent des distorsions optiques qui dévient les rayons lumineux de leur trajectoire idéale. Ces distorsions peuvent être radiales ou tangentielles, chacune ayant un effet spécifique sur la géométrie de l'image projetée. De manière classique, ces distorsions sont prises en compte en appliquant une transformation des coordonnées (i, j).

En posant (i', j') comme les coordonnées obtenues dans le plan de distance focale égal à 1 et centrées par rapport à l'écran, un modèle de distorsion permet d'obtenir les coordonnées distordues (i'', j'') à partir de (i', j'). Ici, en posant la distance radiale r comme $r = \sqrt{i'^2 + j'^2}$, le modèle utilisé est exprimé par l'équation (III.2).

$$\begin{bmatrix} i'' \\ j'' \end{bmatrix} = \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6}\begin{bmatrix} i' \\ j' \end{bmatrix} + \begin{bmatrix} p_1(r^2 + 2i'^2) + 2p_2 i' j' + s_3 r^2 + s_4 r^4 \\ 2p_1 i' j' + p_2(r^2 + 2j'^2) + s_1 r^2 + s_2 r^4 \end{bmatrix} \tag{III.2}$$

Les termes k_i représentent les coefficients de distorsion radiale, tandis que les p_i représentent les coefficients de distorsion tangentielle, et les s_i représentent des coefficients de *prism distortion*. Ce modèle de distorsion, utilisé dans la bibliothèque de traitement d'image OpenCV, permet de choisir entre le modèle de distorsion de Brown-Conrady [Bro66] et le modèle par division [Fit01].

Dans le cadre de la génération d'images par *backward ray tracing*, ce modèle de distorsions ne peut pas être appliqué directement puisque le rayon est lancé depuis le pixel de la caméra. Dans ce contexte, le modèle de distorsion est utilisé dans le sens inverse de celui de la projection habituelle : pour chaque pixel (i, j) de l'image finale, on calcule les coordonnées (i'', j'') à partir desquels le rayon est lancé pour déterminer la surface intersectée. La Figure III.3 est une illustration d'images simulées avec et sans distorsions à partir du modèle décrit dans cette section.



FIGURE III.3. – Exemple d'images générées avec un modèle de caméra *pinhole*. **Gauche –** sans distorsion. **Milieu –** distorsion radiale en barillet ($k_1 > 0$). **Droite –** distorsion radiale en coussinet ($k_1 < 0$).

III.3.3. Alignement caméra

Lorsqu'on dispose d'une caméra réelle et qu'on cherche à la simuler via un modèle de caméra *pinhole*, il est nécessaire de trouver les paramètres du modèle qui correspondent le mieux à l'image obtenue par la caméra réelle avec celle générée par la caméra virtuelle. C'est l'étape de calibration, ou d'alignement. Différentes méthodes existent, notamment lorsqu'il n'y a pas de distorsion et qu'on est seulement intéressé par la matrice de projection du 3D vers le 2D, une simple inversion matricielle peut être réalisée à partir de couples de points.

Dans notre cas, nous cherchons à obtenir les paramètres extrinsèques (position, orientation), intrinsèques (focale, champ de vision, taille des pixels), ainsi que potentiellement des coefficients de distorsion. Nous avons opté pour une simple descente de gradient sur les paramètres de la caméra, en minimisant une erreur mesurée par les distances entre les points de l'image réelle et les points obtenus en projetant les points du maillage 3D dans les coordonnées 2D de la caméra virtuelle.

Une fois la caméra correctement positionnée dans le modèle 3D, on peut désormais simuler la propagation des rayons capturés par chacun de ses pixels.

III.4. Équations de rendu infrarouge et modèle de BRDF

La simulation d'une image infrarouge nécessite une modélisation de la propagation de la lumière infrarouge au sein de la scène thermique. If faut pour cela déterminer la quantité de lumière sortante d'un point $\mathbf{x}$ de la surface du tokamak dans une direction donnée $\boldsymbol{\omega}$. Nous faisons ici l'hypothèse que toutes le surfaces sont opaques, c'est-à-dire qu'il n'y a pas de transmission de la lumière au sein des matériaux. De plus, on suppose qu'il n'y a pas d'absorption ou d'émission du plasma dans les longueurs d'ondes considérées.

La quantité de lumière est formellement exprimée par la radiance, notée L, mesurée en Watt par mètre carré par steradian ($\mathrm{W}\cdot\mathrm{m}^{-2}\cdot\mathrm{sr}^{-1}$). La radiance permet de quantifier le flux énergétique émis ou reçu pour une surface et un angle solide donnés.

Ainsi, on peut définir la fonction L_o qui indique la radiance sortante d'un point de la surface du modèle dans une direction spécifique : $L_o(\mathbf{x}, \boldsymbol{\omega}_o)$. Pour simuler le comportement d'une caméra et générer une image, il faut évaluer L_o pour un ensemble de points $\mathbf{x}$ et de directions $\boldsymbol{\omega}$ faisant face à la caméra.

Dans le domaine infrarouge, la lumière sortante d'un objet résulte de deux phénomènes : l'émission de lumière due à sa température, et la réflexion d'une partie du flux incident à la surface. Ainsi, la radiance sortante $L_o(\mathbf{x}, \boldsymbol{\omega}_o)$ d'un point $\mathbf{x}$ dans une direction $\boldsymbol{\omega}_o$ peut être décomposée en une composante émise $L_e(\mathbf{x}, \boldsymbol{\omega}_o)$ et une composante réfléchie $L_r(\mathbf{x}, \boldsymbol{\omega}_o)$, comme le montre l'équation (III.3).

$$L_o(\mathbf{x}, \boldsymbol{\omega}_o) = L_e(\mathbf{x}, \boldsymbol{\omega}_o) + L_r(\mathbf{x}, \boldsymbol{\omega}_o) \tag{III.3}$$

III.4.1. La radiance émise : modèle du corps noir et émissivité

Pour comprendre la radiance émise $L_e(\mathbf{x}, \boldsymbol{\omega}_o)$ par une surface, on utilise le modèle du corps noir. Le corps noir représente un cas limite où l'objet absorbe intégralement le flux qu'il reçoit, et le réémet totalement. La loi de Planck [Pla01] fournit alors la radiance spectrale émise par ce corps en fonction de sa température T (en Kelvin) et de sa longueur d'onde λ. L'équation (III.4) définit la radiance spectrale, c'est-à-dire la dérivée de la radiance émise par rapport à la longueur d'onde. Cette formule utilise la vitesse de la lumière c, la constante de Planck h, et la constante de Boltzmann k_B.

$$\frac{\partial L^\circ}{\partial \lambda}(T, \lambda) = \frac{2hc^2}{\lambda^5 \left(\exp\left[\frac{hc}{\lambda k_B T} \right] - 1 \right)} \tag{III.4}$$

La Figure III.4 montre la valeur de la radiance spectrale en fonction de la longueur d'onde pour différentes températures de matériaux.

La radiance émise par un corps noir dans un intervalle de longueur d'ondes $[\lambda_{\min}; \lambda_{\max}]$ est obtenue en intégrant la radiance spectrale dans cet intervalle, comme le montre l'équation (III.5).

$$L^\circ(T) = \int_{\lambda_{\min}}^{\lambda_{\max}} \frac{2hc^2}{\lambda^5 \left(\exp\left[\frac{hc}{\lambda k_B T} \right] - 1 \right)} \, \mathrm{d}\lambda \tag{III.5}$$

Théoriquement, en intégrant sur l'intervalle de longueurs d'onde $[0; +\infty]$, on obtient alors la radiance totale émise, qui d'après la loi de Stefan-Boltzmann est proportionnelle à T^4 [BB71]. En pratique, une caméra telle qu'utilisée pour la protection des tokamaks n'est sensible qu'à un certain intervalle étroit de longueur d'onde (d'une largeur de l'ordre de 0.1 µm autour d'une longueur d'onde observée dans l'infraouge moyen, de l'ordre de 3 à 5 µm [Cou+19]), la radiance n'est alors calculée que dans des bandes étroites. La Figure III.5 montre par exemple la radiance corps noir, en fonction de la température, calculée dans la bande passante du filtre utilisé pour la caméra grand angle de WEST [4,3 – 4,4 µm]. Pour la suite du manuscrit, les radiances seront

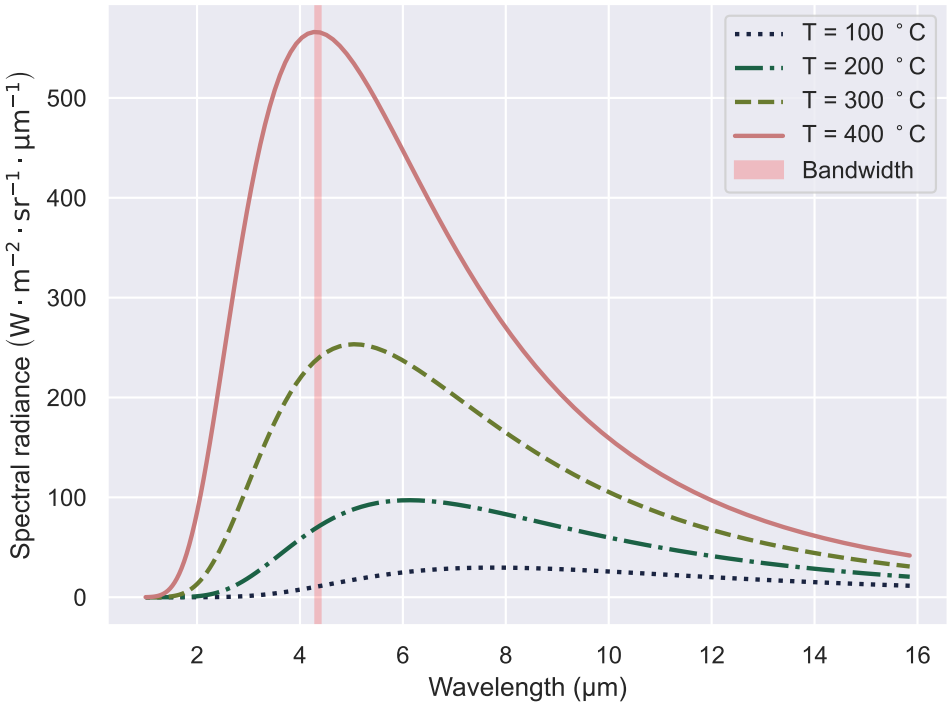


FIGURE III.4. – Courbes représentant la radiance spectrale, due à la loi de Planck, qu'émet un corps noir en fonction la longueur d'onde, et pour différentes températures. La bande étroite de longueur d'onde utilisée par la caméra grand angle de WEST est montrée. La radiance émise par un corps noir dans ces longueurs d'onde est alors obtenue numériquement en intégrant la radiance spectrale dans cet intervalle, par exemple en utilisant la méthode des trapèzes [Tai94].

toujours implicitement exprimées pour une certaine plage de longueur d'ondes.

Néanmoins, les objets de la scène ne peuvent pas être considérés comme des corps noirs, qui représentent un cas idéal d'objets qui absorbent toute la lumière qu'ils reçoivent. En pratique, une surface n'émet qu'une proportion de ce qu'un corps noir de même température émettrait. Pour décrire le profil d'émission d'une surface en un point $\mathbf{x}$, qui dépend de la direction observée $\boldsymbol{\omega}_o$, il faut pondérer l'émission du corps noir L° par une émissivité directionnelle $\varepsilon(\mathbf{x}, \boldsymbol{\omega}_o)$, comprise entre 0 et 1, comme le montre l'équation (III.6).

$$L_e(\mathbf{x}, \boldsymbol{\omega}_o) = \varepsilon(\mathbf{x}, \boldsymbol{\omega}_o) L^\circ(T(\mathbf{x})) \tag{III.6}$$

Si les notations utilisées dans cette thèse considèrent implicitement les grandeurs à une longueur d'onde spécifique, il est important de noter que l'émissivité d'un matériau varie aussi en fonction de la longueur d'onde, et faiblement en fonction de la température de la surface [Gas+19]. Dans nos simulations, nous négligeons la dépendance de l'émissivité à la température, et nous considérons que la bande passante spectrale observée est suffisamment étroite (0,1 µm pour les caméras de WEST) pour supposer l'émissivité constante sur cette bande.

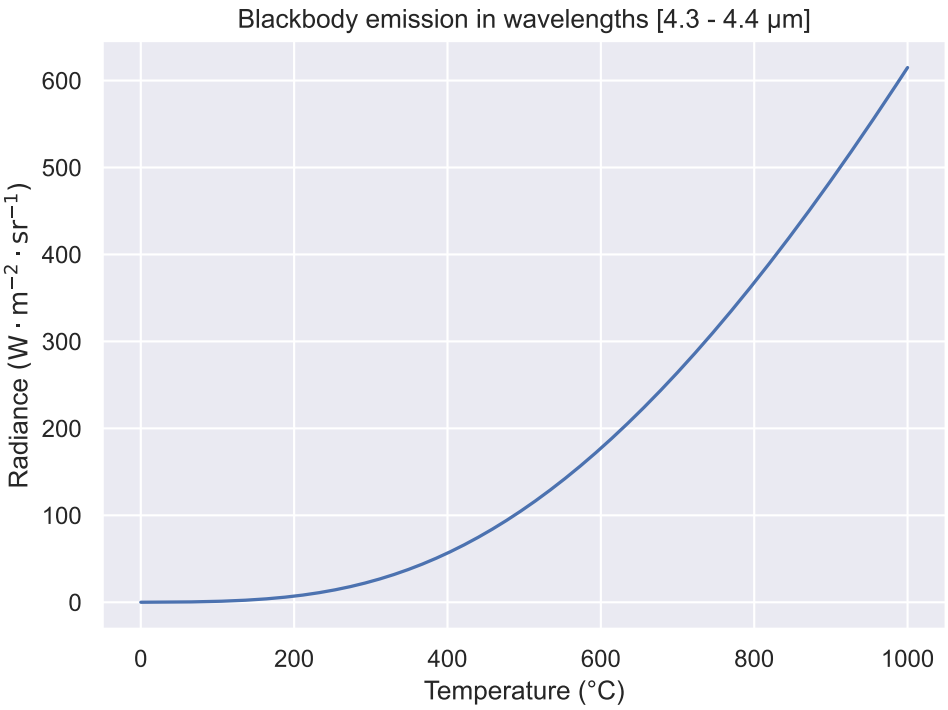


FIGURE III.5. – Radiance émise par un corps noir, en fonction de sa température, et dans la plage de longueurs d'ondes observées par la caméra grand angle de WEST : [4,3 – 4,4 µm]. La radiance présente une croissance approximativement exponentielle pour des faibles températures, avant de tendre vers une fonction affine pour les hautes températures.

Ainsi, la connaissance des températures T et émissivités ε de toutes les surfaces de la scène thermique permet de déterminer le flux émis par rayonnement de Planck en chaque point et chaque direction. Comme mentionné précédemment, cette connaissance n'est pas suffisante pour déterminer la lumière totale sortant d'une surface, car les objets n'étant pas des corps noirs parfaits, une partie de la lumière les éclairant est réfléchie., il faut alors pouvoir fournir une expression de la radiance réfléchie L_r.

III.4.2. La radiance réfléchie : modèle de la BRDF

La description de la radiance réfléchie $L_r(\mathbf{x}, \boldsymbol{\omega}_o)$ dépend de la totalité de la lumière que la surface au point $\mathbf{x}$ perçoit des autres surfaces, ainsi que des propriétés optiques de l'objet. Pour décrire le comportement réfléchissant d'une surface, nous utilisons le modèle de la reflectivité bidirectionelle [Nic65], abrégée en BRDF (*Bidirectional Reflectance Distribution Function*).

III.4.2.1. Définition et propriétés physiques

Le modèle de BRDF définit la fonction notée f_r par l'équation (III.7). Cette fonction lie la radiance infinitésimale $\mathrm{d}L_r(\mathbf{x}, \boldsymbol{\omega}_o)$ réfléchie dans la direction $\boldsymbol{\omega}_o$ lorsque le point $\mathbf{x}$ est éclairé seulement par une radiance incidente $L_i(\mathbf{x}, \boldsymbol{\omega}_i)$ provenant d'un ensemble de directions centrées

autour de la direction $\boldsymbol{\omega}_i$ (d'angle θ_i à la normale de la surface, comme représenté en Figure III.6) et contenues dans un angle solide infinitésimal $\mathrm{d}\omega_i$. Ces différentes grandeurs sont représentées en Figure III.6.

$$f_r(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \stackrel{\text{def}}{=} \frac{\mathrm{d}L_r(\mathbf{x}, \boldsymbol{\omega}_o)}{L_i(\mathbf{x}, \boldsymbol{\omega}_i)\cos(\theta_i)\,\mathrm{d}\omega_i} \tag{III.7}$$

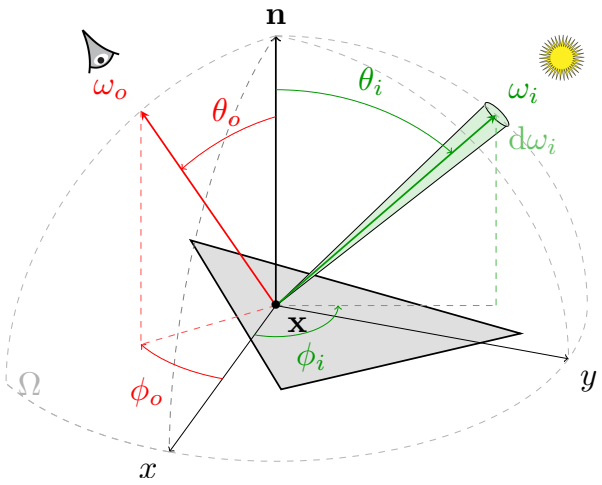


FIGURE III.6. – Représentation des notations utilisées dans les équations de rendu. Si $\mathbf{x}$ est un point sur une surface observée, on cherche à déterminer la radiance sortante dans la direction $\boldsymbol{\omega}_o$. Pour prendre en compte le flux éclairant la surface, il faut comptabiliser la radiance entrante dans tous les angles solides infinitésimaux $\mathrm{d}\omega_i$ centrés sur les directions $\boldsymbol{\omega}_i$ prises dans l'hémisphère qui surplombe $\mathbf{x}$, noté Ω. Les différentes directions peuvent être représentées par deux angles $\theta \in [0; \pi/2[$ et $\phi \in [0; 2\pi[$ en prenant un référentiel centré sur la normale $\mathbf{n}$ de la surface au point $\mathbf{x}$.

Par linéarité de la radiance réfléchie par rapport aux différentes radiances incidentes, ce modèle de BRDF permet de définir la radiance totale réfléchie $L_r(\mathbf{x}, \boldsymbol{\omega}_o)$ en intégrant la radiance entrante $L_i(\mathbf{x}, \boldsymbol{\omega}_i)$ sur $\Omega_{\mathbf{x}}$, l'ensemble des directions possibles dans un hémisphère qui surplombe la surface, comme décrit dans l'équation (III.8).

$$L_r(\mathbf{x}, \boldsymbol{\omega}_o) = \int_{\Omega_{\mathbf{x}}} f_r(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) L_i(\mathbf{x}, \boldsymbol{\omega}_i) \cos(\theta_i)\,\mathrm{d}\omega_i \tag{III.8}$$

Cette modélisation permet d'exprimer une large variété de propriétés de matériaux en choisissant diverses fonctions f_r. Ces fonctions sont soumises à certaines contraintes liées aux comportements physiques de la lumière, telles que la conservation de l'énergie et la réciprocité de Helmholtz, décrites par les équations (III.9) et (III.10), où $\rho^{\cap}(\mathbf{x}, \boldsymbol{\omega}_i)$ désigne la réflectivité hémisphérique.

$$\forall(\mathbf{x},\boldsymbol{\omega}_i),\quad \rho^{\cap}(\mathbf{x},\boldsymbol{\omega}_i) \overset{\text{def}}{=} \int_{\Omega_{\mathbf{x}}} f_r(\mathbf{x},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o)\cos(\theta_o)\,\mathrm{d}\omega_o \leqslant 1 \tag{III.9}$$

$$\forall(\mathbf{x},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o),\quad f_r(\mathbf{x},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o) = f_r(\mathbf{x},\boldsymbol{\omega}_o,\boldsymbol{\omega}_i) \tag{III.10}$$

Il est à noter que la relation (III.10) n'est plus vraie dans le cas de matériaux transparents, modélisés par une BSDF (*Bidirectional Scattering Distribution Function*), qui doit prendre en compte les indices de réflexions de chaque milieu [Vea96].

Enfin, comme le montre l'équation (III.11), la BRDF f_r détermine la fonction d'émissivité ε, car les rayonnements incidents, réfléchis et absorbés sont liés par la loi de Kirchhoff [Nic65 ; Mod13].

$$\begin{aligned}\varepsilon(\mathbf{x},\boldsymbol{\omega}_o) &= 1-\rho^{\cap}(\mathbf{x},\boldsymbol{\omega}_o)\\ &= 1-\int_{\Omega_{\mathbf{x}}} f_r(\mathbf{x},\boldsymbol{\omega}_o,\boldsymbol{\omega}_r)\cos(\theta_r)\,\mathrm{d}\omega_r\end{aligned} \tag{III.11}$$

Ainsi, pour la suite il ne sera question que de fournir aux surfaces des modèles de BRDF appropriés : du moment que le modèle de BRDF fournit un moyen de calculer sa reflectivité hémisphérique (ou du moins un estimateur pour celle-ci), l'emissivité de la surface en sera déduite.

III.4.2.2. Modèles simples de BRDF

Comme représenté en Figure III.7, il est possible de modéliser la diversité des comportements des matériaux en termes de réflexion en utilisant différents modèles de BRDF. Certains de ces modèles sont présentés ici.

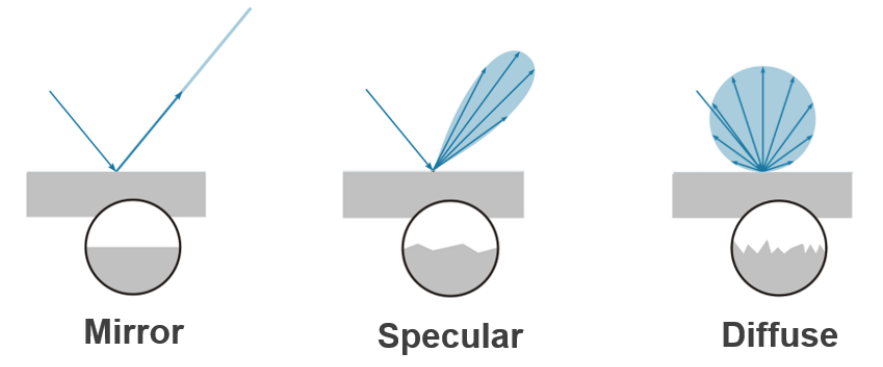


FIGURE III.7. – Exemples de comportements de BRDFs en fonction des différentes rugosités.

Le modèle le plus simple de BRDF correspond à une réflexion purement spéculaire, similaire à celle d'un miroir, comme présenté dans la section II.2. Dans ce cas, toute la lumière est réfléchie dans une direction unique. La BRDF est alors représentée par une impulsion de Dirac, qui permet de modéliser une distribution concentrée en une valeur unique. L'équation (III.12) donne l'expression de la BRDF entre un rayon incident $\boldsymbol{\omega}_i$ et sortant $\boldsymbol{\omega}_o$. Cette expression est paramétrée

par un coefficient $k_m \in [0;1]$ qui contrôle la réflectivité totale de la surface. δ_w représente une impulsion de Dirac dans l'espace mesurable des directions, et $\mathbf{r}(\boldsymbol{\omega}_i)$ représente la direction de réflexion selon la loi de Snell-Descartes.

$$f_r^{\text{mirroir}}(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = k_m \frac{1}{\cos(\theta_i)} \delta_\omega(\boldsymbol{\omega}_o - \mathrm{r}(\boldsymbol{\omega}_i)) \tag{III.12}$$

À l'opposé, le modèle de réflexion diffuse, dans lequel les réflexions sont réparties uniformément dans toutes les directions (comme utilisé dans le modèle des radiosités, voir section I.3.3), est décrit par l'équation (III.13). Dans ce modèle, la BRDF est indépendante des directions de l'incident et du sortant, et est paramétrée par k_d, représentant la réflectivité totale de la surface.

$$f_r^{\text{diffus}}(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = k_d \frac{1}{\pi} \tag{III.13}$$

Ces deux modèles représentent des cas extrêmes de comportements des matériaux. En pratique, les matériaux présentent souvent des comportements intermédiaires, avec une composante spéculaire non totalement concentrée et une certaine proportion de réflexion diffuse.

Un modèle simple et communément utilisé pour représenter ces comportements intermédiaires est la BRDF de Phong normalisée de Lafortune [LW98], une amélioration du modèle d'illumination original de Phong [Pho98] qui respecte les propriétés de conservation de l'énergie. Ce modèle combine une part diffuse et une part spéculaire qui privilégie la direction de réflexion selon Snell-Descartes, en calculant un produit scalaire entre le rayon sortant et la direction purement spéculaire. L'équation (III.14) décrit ce modèle. Une représentation de la distribution obtenue pour un rayon incident est montrée en Figure III.8.

$$f_r^{\text{Phong}}(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = k_d \frac{1}{\pi} + k_s \frac{n+2}{2\pi} \max(0, \langle \boldsymbol{\omega}_o, \mathbf{r}(\boldsymbol{\omega}_i) \rangle)^n \tag{III.14}$$

Ce modèle est paramétré par $k_d \in [0;1]$ et $k_s \in [0;1]$, qui contrôlent respectivement la part de réflexion diffuse et spéculaire, ainsi que par $n \in \mathbb{R}^+$, qui contrôle la largeur du lobe spéculaire. La réflectivité hémisphérique $\rho^{\cap}(\mathbf{x}, \boldsymbol{\omega}_i)$ de ce modèle dépend du rayon incident, mais est majorée par $k_d + k_s$ (qui doit donc être inférieur à 1), ce qui correspond à la réflectivité lorsque le rayon incident est perpendiculaire à la surface. Il peut être montré que pour un n suffisamment grand, la réflectivité hémisphérique peut être approximée par $k_d + k_s \cos(\theta_i)$. La réflectivité décroît donc lorsque le rayon incident devient tangent à la surface, ce qui peut être problématique si l'on cherche de comportement inverse qui est courant pour beaucoup de matériaux. Dans le cadre de cette thèse, nous avons choisi ce modèle empirique permettant de caractériser les propriétés de réflectance des matériaux par quelques paramètres intuitifs. Pour un modèle plus proche des mesures, le modèle de micro-facettes présenté en section suivante semble plus adapté. Il pourra être intégré après analyse des mesures expérimentales réalisées sur des matériaux avant et après exposition pour obtenir des comportements plus proches de ceux des composants du tokamak.

III.4.2.3. Les BRDFs par micro-facettes

Le modèle de BRDF par micro-facettes est un cadre général pour modéliser la réflexion de la lumière sur des surfaces. Dans ce modèle, la surface est composée, au niveau microscopique,

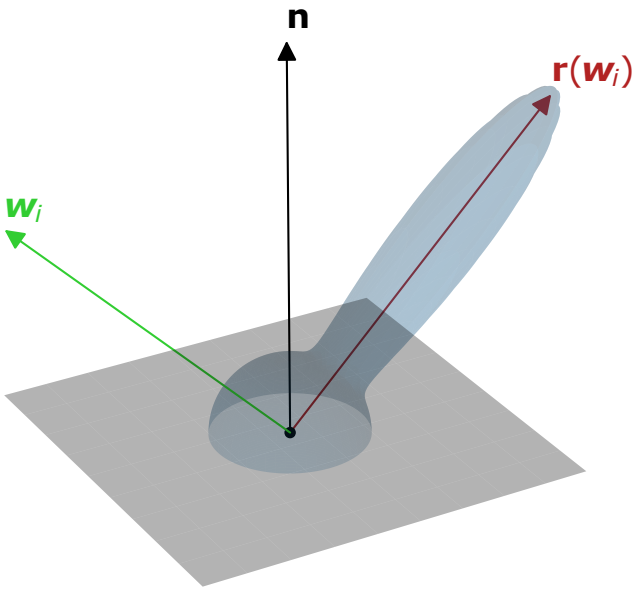


FIGURE III.8. – Représentation de la BRDF de Phong normalisée pour $k_d = 0.6$, $k_s = 0.1$, $n = 50$ et pour un rayon incident arrivant à 45° par rapport à la normale. L'hémisphère représente la composante diffuse qui répartit équitablement l'irradiance incidente dans toutes les directions, tandis que le lobe additionnel centré autour de la direction miroir représente la part de réflexion spéculaire.

de petites facettes, d'orientation différentes, chacune agissant comme un miroir parfait. Au niveau macroscopique, la distribution des orientations de ces micro-facettes détermine les caractéristiques de la réflexion.

Le modèle de Cook-Torrance [CT82] est une implémentation populaire du modèle par micro-facettes. Il est largement utilisé en synthèse d'image en raison de sa capacité à produire des réflexions réalistes tout en respectant les propriétés d'une BRDF physiquement réaliste. Il combine trois composantes : la distribution des normales de micro-facettes, le facteur géométrique de masquage et ombrage, et la fonction de Fresnel. L'expression de la BRDF est donnée par l'équation (III.15).

$$f_r^{\text{Cook–Torrance}}(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{D(\boldsymbol{\omega}_h) F(\boldsymbol{\omega}_i, \boldsymbol{\omega}_h) G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)}{4 \cos(\theta_i) \cos(\theta_o)} \tag{III.15}$$

Ici, $\boldsymbol{\omega}_h$ désigne la direction à mi-chemin entre $\boldsymbol{\omega}_i$ et $\boldsymbol{\omega}_o$: $\boldsymbol{\omega}_h = (\boldsymbol{\omega}_i + \boldsymbol{\omega}_o) / \|\boldsymbol{\omega}_i + \boldsymbol{\omega}_o\|$. Les réflexions de $\boldsymbol{\omega}_i$ vers $\boldsymbol{\omega}_o$ étant purement spéculaires, on cherche la proportion des micro-facettes qui ont pour normale $\boldsymbol{\omega}_h$. Cette proportion est donnée par $D(\boldsymbol{\omega}_h)$. $G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)$ décrit la proportion des micro-surfaces de normale $\boldsymbol{\omega}_h$ qui sont visibles depuis $\boldsymbol{\omega}_i$, $\boldsymbol{\omega}_o$, c'est-à-dire qui ne sont pas masquées par des aspérités de la micro-surface. Enfin, le terme de Fresnel $F(\boldsymbol{\omega}_i, \boldsymbol{\omega}_h)$ permet de simuler la différence de comportement de la lumière en fonction de l'angle d'incidence, et permet notamment une plus forte réflexion spéculaire des rayons rasants.

Ces trois composantes possèdent différentes implémentations paramétrables [Wal+07]. Le modèle de Cook-Torrance permet ainsi de modéliser une large gamme de comportements de matériaux tout en respectant les propriétés nécessaires au rendu physique.

III.4.3. L'équation du rendu infrarouge à résoudre

En supposant qu'il n'y a pas d'émission ou d'absorption de lumière infrarouge par le plasma situé dans la chambre, la radiance entrante $L_i(\mathbf{x}, \boldsymbol{\omega}_i)$ correspond à la lumière sortante de la surface directement visible depuis ce point dans cette direction. En raison de la géométrie de la scène, il existe donc une fonction de "lancer de rayon", $\mathbf{cast} : (\mathbf{x}, \boldsymbol{\omega}_i) \mapsto (\mathbf{x}', \boldsymbol{\omega}'_o)$, qui décrit le point directement visible, de sorte que $L_i(\mathbf{x}, \boldsymbol{\omega}_i) = L_o(\mathbf{cast}(\mathbf{x}, \boldsymbol{\omega}_i))$. La définition de L_o contient donc une autoréférence : son estimation numérique en un point est de fait calculatoirement coûteuse car elle nécessite une prise en compte de toute la scène.

Ainsi, la version infrarouge de l'équation du rendu [Kaj86], fournissant une expression récursive de L_o prenant en compte à la fois les phénomènes d'émission et de réflexions, est donnée par l'équation (III.16). Les termes utilisés sont schématisés en Figure III.9.

$$L_o(\mathbf{x}, \boldsymbol{\omega}_o) = \varepsilon(\mathbf{x}, \boldsymbol{\omega}_o) L^\circ(T(\mathbf{x})) + \int_{\Omega_\mathbf{x}} f_r(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) L_o(\mathbf{cast}(\mathbf{x}, \boldsymbol{\omega}_i)) \cos(\theta_i) \, \mathrm{d}\omega_i \tag{III.16}$$

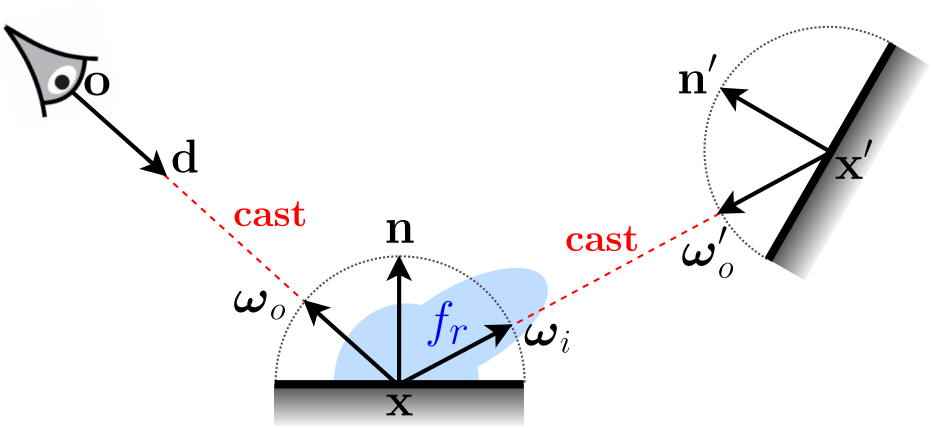


FIGURE III.9. – Formalisme utilisé pour exprimer l'équation de rendu infrarouge. Un capteur reçoit une radiance incidente au point d'origine **o** dans la direction **d**, qui, en lançant un rayon virtuel, est exprimée comme la radiance sortant d'une surface du modèle au point **x** dans la direction $\boldsymbol{\omega}_o$. Pour déterminer la radiance réfléchie dans la direction $\boldsymbol{\omega}_o$, la BRDF f_r au point **x** permet de pondérer la radiance incidente pour chaque direction $\boldsymbol{\omega}_i$. Un autre lancé de rayon permet d'exprimer récursivement cette radiance incidente comme la radiance sortante d'une autre surface.

III.5. Résolution de l'équation du rendu par intégration de Monte Carlo

Pour résoudre l'équation de rendu (III.16), la radiance réfléchie nécessite un calcul infini et récursif d'une intégrale sur l'ensemble de l'hémisphère des directions de réflexion possibles. Ce problème est résolu à l'aide d'une technique appelée intégration de Monte Carlo [MU49], qui est une méthode stochastique permettant d'obtenir une approximation itérative d'une intégrale.

La méthode fonctionne en reliant l'intégration à l'espérance des variables aléatoires. Pour une variable aléatoire X sur un espace mesurable $\mathcal{X}$ et suivant une loi donnée par une fonction de densité de probabilité p_X, l'espérance $\mathbb{E}_X[f(X)]$ pour une fonction f est donnée par une intégrale, exprimée dans l'équation (III.17).

$$\mathbb{E}_X[f(X)] = \int_{\mathcal{X}} p_X(\mathbf{x}) f(\mathbf{x}) \, \mathrm{d}x \tag{III.17}$$

Ainsi, l'intégrale d'une fonction sur un domaine peut être considérée comme l'espérance d'une variable aléatoire (équation (III.18)), qui peut être approchée à partir d'une séquence d'éléments $(\mathbf{x}_i)_{i\in\mathbb{N}^*}$ tous tirés suivant la fonction de densité de probabilité p_X (équation (III.19)).

$$\int_{\mathcal{X}} f(\mathbf{x}) \, \mathrm{d}x = \mathbb{E}_X\left[\frac{f(X)}{p_X(X)}\right] \tag{III.18}$$

$$= \lim_{N\to\infty} \frac{1}{N} \sum_{\substack{i\in[\![1;N]\!] \\ \mathbf{x}_i \sim p_X}} \frac{f(\mathbf{x}_i)}{p_X(\mathbf{x}_i)} \tag{III.19}$$

Cette méthode fonctionne pour toute fonction de densité de probabilité p_X qui permet d'échantillonner l'ensemble $\mathcal{X}$ dans son intégralité, ce qui signifie qu'elle n'est égale à zéro nulle part.

Le choix d'une fonction de distribution p_X appropriée est crucial pour une convergence plus rapide vers la solution. L'échantillonnage préférentiel (*importance sampling*) est une technique qui peut être utilisée pour améliorer l'efficacité de l'intégration Monte Carlo, en choisissant un p_X biaisant l'échantillonnage vers les régions où la fonction à intégrer est importante [Vea98]. Dans le cas particulier où p_X est proportionnel à f, la variance de la somme obtenue est strictement égale à zéro, ainsi un seul échantillon suffit pour calculer l'intégrale. En pratique, plus la distribution utilisée p_X est similaire à la fonction f à intégrer, moins la variance de la somme sera importante, ce qui permet d'approximer l'intégrale avec moins d'échantillons.

L'intérêt principal de l'intégration de Monte Carlo par rapport à l'intégration classique, telle que l'intégration de Riemann, réside dans son efficacité pour traiter des intégrales sur des espaces de grande dimension. Dans de tels cas, les méthodes classiques deviennent inapplicables en raison de la "malédiction de la dimensionnalité", où le nombre de points nécessaires pour obtenir une approximation précise croît exponentiellement avec la dimension de l'espace [Hin+14]. En revanche, par le théorème central limite, l'intégration de Monte Carlo a une convergence en $\mathcal{O}\left(1/\sqrt{N}\right)$, où N est le nombre d'échantillons [Caf98]. Cela signifie que l'erreur de l'approximation diminue proportionnellement à la racine carrée du nombre de points d'échantillonnage, indépendamment de la dimension de l'espace. De plus, cette méthode permet de calculer des intégrales sur des ensembles mesurables ayant des géométries ou des caractéristiques inhabituelles, pour lesquels il serait difficile de définir une subdivision comme dans le cas de l'intégrale de Riemann.

Dans le contexte du rendu, la résolution de l'équation du rendu est souvent effectuée en considérant un espace mesurable constitué de chemins, c'est-à-dire d'une succession de rayons se propageant dans la scène et réalisant une ou plusieurs réflexions. En définissant correctement

cet espace mesurable et une mesure de probabilité pour chaque chemin aléatoire possible, il est alors possible d'exprimer la radiance sortante L_o en un point comme l'intégrale d'une fonction de contribution sur cet espace de chemins lumineux [VG97]. Cette intégrale peut être calculée par Monte Carlo comme une moyenne pondérée des contributions de chaque chemin.

Dans le cadre le thèse, nous pouvons plus simplement considérer que nous réalisons seulement l'intégrale d'une fonction sur un hémisphère de directions possibles. La méthode de Monte Carlo appliquée à l'équation de rendu infrarouge conduit ainsi à l'expression décrite dans l'équation (III.20).

$$\begin{aligned} L_o(\mathbf{x},\boldsymbol{\omega}_o) &= \varepsilon(\mathbf{x},\boldsymbol{\omega}_o)L^\circ(T(\mathbf{x})) + \int_{\Omega_\mathbf{x}} f_r(\mathbf{x},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o)L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i))\cos(\theta_i)\,\mathrm{d}\omega_i \\ &= \lim_{N\to\infty}\frac{1}{N}\sum_{\substack{i\in[\![1;N]\!]\\ \boldsymbol{\omega}_i\sim p_W}} \varepsilon(\mathbf{x},\boldsymbol{\omega}_o)L^\circ(T(\mathbf{x})) + \frac{f_r(\mathbf{x},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o)\cos(\theta_i)}{p_W(\boldsymbol{\omega}_i)}L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i)) \end{aligned} \tag{III.20}$$

Cependant, cette formulation contient une auto référence à la radiance sortante $L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i))$ estimée en un autre point. On peut démontrer que lors d'un calcul d'espérance, un terme au sein d'une somme ou d'un produit de termes peut être remplacé par une variable aléatoire dont l'espérance conditionnelle (sachant la valeur des autres variables aléatoires) est égale à ce terme. Ceci permet de remplacer la valeur de $L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i))$ présente dans l'intégrale par un estimateur de cette valeur, menant ainsi à un algorithme récursif.

Pour estimer la radiance, on tire ainsi N rayons en *backward ray tracing*, c'est-à-dire du capteur vers l'émetteur, remontant ainsi le trajet du rayon lumineux réel. Ces rayons intersectent le point $\mathbf{x}$ dans la direction $\boldsymbol{\omega}_o$. À chaque intersection, la BRDF au point $\mathbf{x}$ est utilisée pour tirer aléatoirement une direction incidente $\boldsymbol{\omega}_i$ selon une distribution de probabilité p_W fournie par la BRDF, qui idéalement favorise les directions de forte contribution. En lançant récursivement un rayon, on calcule alors un estimateur pour $L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i))$, c'est-à-dire une variable aléatoire dont l'espérance est cette valeur, et qui pourra être utilisée dans le calcul d'intégration.

Pour éviter une infinité de lancers récursifs, il faut une condition d'arrêt qui limite les réflexions après un certain nombre d'itérations. On peut choisir arbitrairement un nombre de réflexions à effectuer avant de s'arrêter, mais cela biaise alors le calcul, car il manquera la contribution des réflexions lointaines. Une manière de contourner ce problème est la roulette russe [PJH24] : puisqu'on peut remplacer la radiance récursive $L_o(\mathbf{cast}(\mathbf{x},\boldsymbol{\omega}_i))$ par un estimateur, il suffit de décider aléatoirement de s'arrêter ou de continuer. Si on s'arrête, cela revient à renvoyer une radiance réfléchie de 0. Si on continue, on divise alors la radiance calculée récursivement par la probabilité de réfléchir, ce qui ne biaise pas l'espérance : $(1-p_{\text{refl}})\times 0 + p_{\text{refl}}\times(L_o^{\text{rec}}/p_{\text{refl}}) = L_o^{\text{rec}}$. Le choix de la probabilité de réflexion peut être fixe ou être variable à chaque intersection afin de ne pas stopper en cours de route un rayon de forte contribution.

L'ensemble de ce processus mène alors à l'algorithme décrit en Figure III.10, permettant de définir une fonction `sample_radiance`, qui étant donné un rayon définit par une origine $\mathbf{o}$ et une direction $\mathbf{d}$, renvoie un estimateur de la radiance incidente $L_i(\mathbf{o},\mathbf{d})$.

En moyennant la valeur aléatoire renvoyée par la fonction `sample_radiance`, on converge ainsi vers la radiance incidente dans la direction observée, comme l'exprime l'équation (III.21).

```
def sample_radiance(o, d, p_r = 0.8):
    """
    Returns an estimator for the incomming radiance
        o : ray origin ∈ ℝ^3
        d : unit ray direction ∈ ℝ^3
        p_r : probability of recursive reflection ∈ [0;1]
    """

    x, ω_o = GEOMETRY.cast_ray(o, d)
    ε = 1 - BRDF.hemispherical_reflectivity(x, ω_o)
    T = TEMPERATURE(x)
    L_e = ε * planck(T)

    if sample_uniform(0., 1.) > p_r:
        return L_e

    ω_i, p_i = BRDF.sample_direction(x, ω_o)
    n = GEOMETRY.normal(x)
    cosθ_i = |⟨n, ω_i⟩|
    ρ_weighted = BRDF(x, ω_i, ω_o) * cosθ_i / (p_i * p_r)
    L_i = sample_radiance(x, ω_i, p_r)

    return L_e + ρ_weighted * L_i
```

FIGURE III.10. – Pseudo-code Python d'une fonction renvoyant un estimateur de la radiance incidente au point **o** dans la direction **d**.

$$L_i(\mathbf{o},\mathbf{d}) = \lim_{N\to+\infty} \frac{1}{N} \sum_{n=1}^{N} \texttt{sample_radiance}(\mathbf{o},\mathbf{d}) \tag{III.21}$$

Ainsi, cet algorithme permet de calculer la radiance perçue depuis n'importe quel point du modèle et dans n'importe quelle direction. Ces directions d'observation sont fournies par la caméra virtuelle placée dans la scène pour chacun de ses pixels, ce qui permet de générer une image. Le procédé est schématisé en Figure III.11.

Pour que cette méthode soit complète et permette la génération d'images synthétiques, il est indispensable de disposer d'une implémentation de la fonction de lancer de rayon, **cast**, qui détermine le point d'intersection entre un rayon et le modèle 3D. Cette opération est cruciale, car elle est computationnellement intensive et doit être réalisée un grand nombre de fois. Le chapitre suivant détaillera l'implémentation de cet élément.

III.6. Intersection accélérée entre un rayon et le modèle 3D

Cette section présente comment calculer efficacement l'intersection entre un rayon et un modèle 3D constitué de millions de triangles. Le processus de recherche du triangle intersecté le plus proche est une opération fondamentale dans le lancer de rayons, mais elle peut devenir extrêmement coûteuse en temps de calcul si elle n'est pas optimisée. Nous présentons ici les

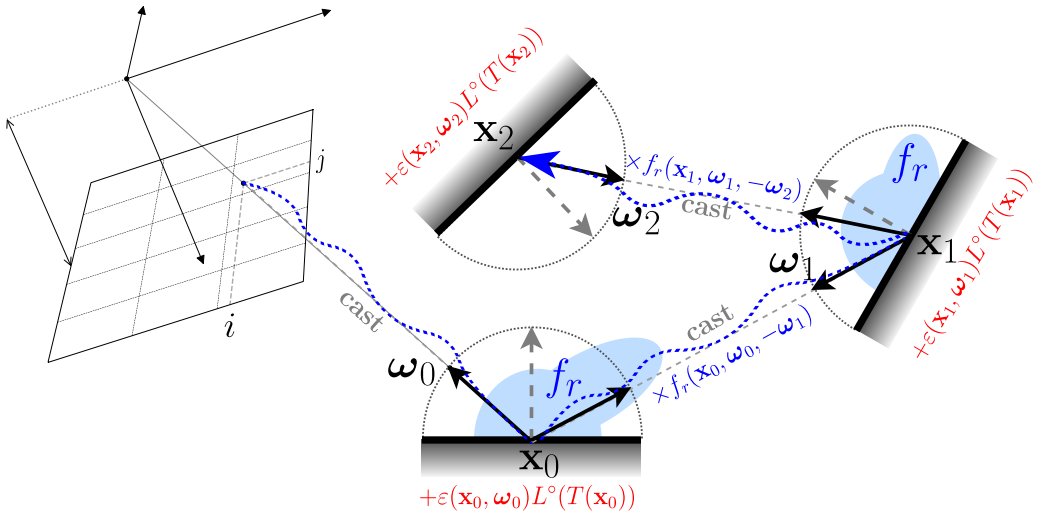


FIGURE III.11. – Génération d'image infrarouge par intégration de Monte Carlo en *backward ray tracing*. Un rayon virtuel est tiré depuis un pixel de la caméra et remonte le trajet d'un rayon réel en se propageant dans la scène par rebonds aléatoires. La contribution de ce rayon est calculée en utilisant les émissivités ε, températures T et BRDF f_r rencontrées à chaque intersection. La moyenne des contributions d'un ensemble de rayons propagés ainsi aléatoirement tend vers la radiance perçue par la caméra.

algorithmes et structures de données permettant de réaliser ces calculs, notamment à travers l'utilisation des hiérarchies de volumes englobants (BVH) et la parallélisation sur GPU.

III.6.1. Intersection entre un rayon et le modèle 3D

L'intersection entre un rayon et le modèle 3D se décompose en deux étapes principales. La première étape repose sur l'utilisation d'une primitive d'intersection entre un rayon et un triangle individuel, qui constitue la base de la détection d'intersections dans la scène. La deuxième étape consiste à utiliser une structure de données accélératrice, permettant de trouver rapidement le triangle le plus proche intersecté par le rayon, sans avoir à tester tous les triangles du modèle.

III.6.1.1. Algorithme d'intersection rayon-triangle

L'opération fondamentale sur laquelle repose la simulation par lancer de rayon est la primitive d'intersection entre un rayon et un triangle. Cette fonction doit prendre en entrée un rayon et un triangle, déterminer si le rayon intersecte le triangle, et si oui, fournir les coordonnées d'intersection. L'algorithme classique pour résoudre ce problème est l'algorithme de Möller-Trumbore [MT05].

Cet algorithme utilise les coordonnées barycentriques : un point $\mathbf{p}$ appartient à un triangle de sommets $(\mathbf{v}_0,\mathbf{v}_1,\mathbf{v}_2)$ si et seulement si il existe $(u,v) \in [0;1]^2$ tels que $u+v \leq 1$ et $\mathbf{p} = (1-u-$

$v)\mathbf{v}_0 + u\mathbf{v}_1 + v\mathbf{v}_2$. De plus, pour un rayon ayant pour origine $\mathbf{o}$ et une direction donnée par un vecteur unitaire $\mathbf{d}$, le rayon intersecte $\mathbf{p}$ si et seulement si il existe $t \in \mathbb{R}^+$ tel que $\mathbf{p} = \mathbf{o} + t\mathbf{d}$. En résolvant l'équation $(1-u-v)\mathbf{v}_0 + u\mathbf{v}_1 + v\mathbf{v}_2 = \mathbf{o} + t\mathbf{d}$ pour (t,u,v), on obtient un système linéaire fournissant ces valeurs. En s'assurant que le système soit inversible et que les valeurs retournées respectent les conditions nécessaires pour que le point d'intersection soit à l'intérieur du triangle et sur le trajet du rayon, on peut ainsi renvoyer ce triplet. Ici, t représente la distance le long du rayon jusqu'au point d'intersection, et (u,v) sont les coordonnées barycentriques du point d'intersection par rapport aux sommets du triangle.

Les coordonnées barycentriques sont particulièrement utiles, car elles permettent d'exprimer n'importe quel point à l'intérieur d'un triangle comme une combinaison pondérée des trois sommets. Une fois les coordonnées barycentriques (u,v) déterminées pour un certain point du triangle, il est possible d'interpoler les valeurs définies aux sommets du triangle en ce point. Par exemple, si chaque sommet $\mathbf{v_0}, \mathbf{v_1}, \mathbf{v_2}$ du triangle a une température respective T_0, T_1, T_2, ces valeurs peuvent être interpolées de manière fluide à travers la surface en renvoyant $(1-u-v)\,T_0 + u\,T_1 + v\,T_2$.

III.6.1.2. Recherche rapide du premier triangle intersecté

Trouver le triangle de la scène réellement intersecté revient à identifier, parmi les triangles de la scène, celui qui est géométriquement intersecté par le rayon et dont la distance (c'est-à-dire la valeur t renvoyée par la primitive d'intersection rayon-triangle) au rayon est minimale.

Un algorithme naïf consisterait à itérer sur tous les triangles de la scène pour trouver le plus proche intersecté. Cela donne lieu à un algorithme avec un temps de calcul linéaire en fonction du nombre de triangles de la scène, ce qui devient rapidement prohibitif pour des scènes de grande taille, par exemple de l'ordre de plusieurs millions de triangles pour celle représentant le tokamak.

Pour améliorer l'efficacité de la recherche d'intersections, des structures de données arborescentes comme les BVH [Klo+98] (Bounding Volume Hierarchies) ou les *k-d trees* [FS05] sont souvent utilisées. Ces structures permettent de réduire drastiquement le nombre de tests nécessaires en organisant les triangles de manière hiérarchique, de sorte que de grandes portions de la scène peuvent être exclues rapidement des tests d'intersection en fonction de la position du rayon.

Dans notre cas, nous avons opté pour l'utilisation d'un BVH, schématisé en Figure III.12. Cette structure hiérarchique permet de réduire le nombre de tests d'intersection nécessaires en organisant les triangles dans des boîtes englobantes alignées sur les axes (AABB, *Axis-Aligned Bounding Boxes*). Ces boîtes sont elles-mêmes englobées dans d'autres boîtes de manière hiérarchique, créant ainsi un arbre binaire dont la racine est une boîte contenant l'intégralité de la scène. Lorsqu'on cherche le premier triangle intersecté par un rayon, le BVH permet d'éliminer rapidement de grandes portions de la scène, au lieu de tester chaque triangle individuellement.

La construction d'un BVH est une étape essentielle pour garantir un coût optimal à l'algorithme de recherche de la plus proche intersection. Un BVH bien équilibré peut réduire la complexité des tests d'intersection de $O(n)$ (pour une recherche linéaire sur n triangles) à $O(\log n)$. La méthode couramment utilisée repose sur la *Surface Area Heuristic* (SAH) [Wal07]. Cette approche *top-down* commence par regrouper tous les triangles dans une grande boîte, puis sépare cette boîte en deux

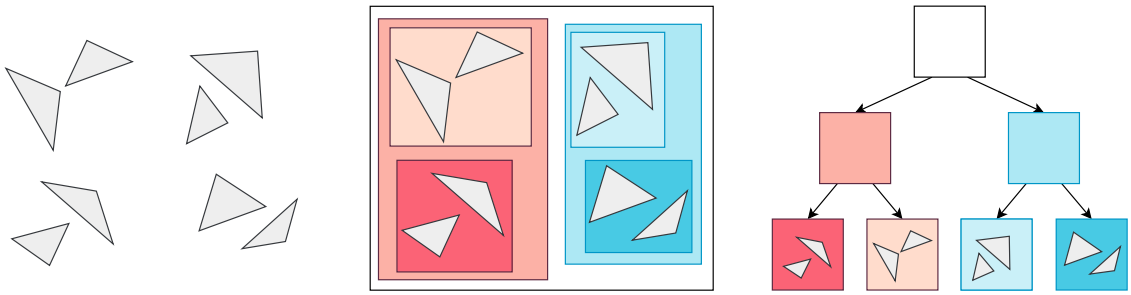


FIGURE III.12. – Principe d'un arbre BVH. Les différents triangles sont organisés au sein de boîtes englobantes de manière hiérarchique, formant un arbre binaire où chaque nœud représente une boîte englobant entièrement ses fils.

sous-boîtes en trouvant un plan de séparation optimal, et ce, de manière récursive. L'heuristique SAH utilise les surfaces des boîtes englobantes pour optimiser la partition des triangles : elle évalue l'espérance du temps de calcul attendu d'un test d'intersection en supposant que la probabilité qu'un rayon intersecte une boîte est proportionnelle à sa surface. La construction de l'arbre commence par tester différents plans de séparation et choisir celui qui minimise le coût estimé, ou de s'arrêter si aucune découpe n'est idéale.

Une fois la phase de construction du BVH achevée, la structure peut être utilisée pour chaque calcul d'intersection rayon-triangle. L'algorithme commence par tester l'intersection du rayon avec l'AABB de la racine du BVH (la plus grande boîte englobante). Si une intersection est trouvée, le test continue de manière récursive avec les nœuds enfants. À cet effet, la méthode par tranche (*slab method*) permet de trouver efficacement l'intersection entre un rayon et un AABB [Maj+18], et de renvoyer la distance parcourue par le rayon. La connaissance de cette distance, ainsi que de la distance au plus proche triangle intersecté jusqu'à présent, permet d'éliminer certaines AABB sans avoir à tester les triangles qu'elles contiennent. Cette descente dans l'arbre se poursuit jusqu'à atteindre les feuilles de l'arbre (qui contiennent les triangles). À ce niveau, des tests d'intersection rayon-triangle individuels sont effectués pour déterminer le point d'intersection exact.

III.6.2. Accélération des calculs : parallélisme sur carte graphique

Le calcul a été implémenté en utilisant le langage de programmation CUDA (*Compute Unified Device Architecture*) [Nic+08]. CUDA permet aux développeurs de tirer parti des capacités de calcul massivement parallèle des cartes graphiques NVIDIA. Ce langage étend le C++ et fournit un modèle de programmation qui permet de lancer des milliers de threads en parallèle, chacun exécutant la même fonction (appelée *kernel*) sur des données distinctes.

L'utilisation de CUDA pour le lancer de rayons permet d'exploiter pleinement l'architecture des GPU, conçue pour traiter un grand nombre de threads simultanément. Contrairement aux CPU, qui possèdent généralement quelques cœurs puissants, les GPU sont dotés de centaines à des milliers de cœurs plus simples, capables de traiter efficacement des tâches en parallèle. Cela est particulièrement avantageux pour le lancer de rayons, où chaque rayon peut être traité indépendamment des autres.

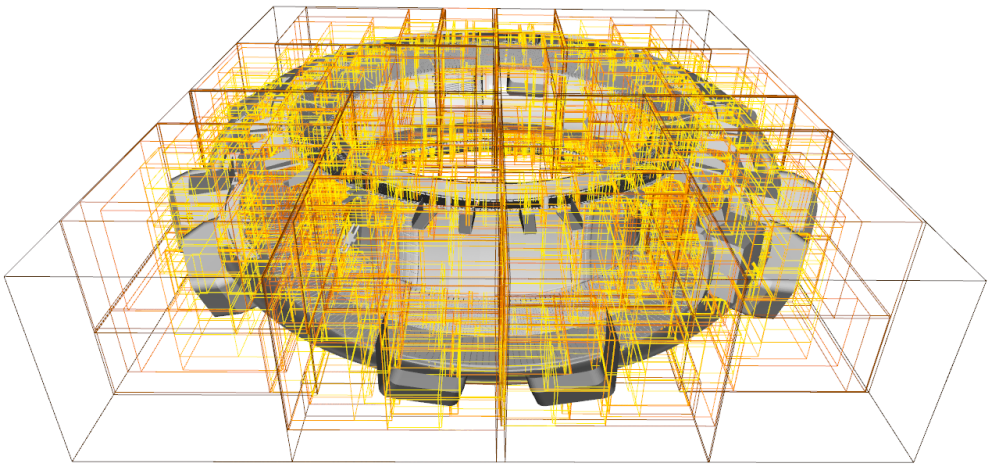


FIGURE III.13. – Représentation des différentes boîtes englobantes (AABB) de l'arbre BVH qui hiérarchise les différents triangles composant le modèle 3D du tokamak WEST. Plus les AABBs sont de couleur vive, plus ils sont profond dans l'arbre.

Les gains de performance obtenus grâce à l'utilisation de CUDA sur GPU sont significatifs. En comparant l'implémentation du lancer de rayon en CUDA avec une implémentation CPU parallélisée via OpenMP [DM98], on trouve un gain de temps d'exécution de l'ordre 40, en comparant une carte graphique NVIDIA A40 à un processeur Intel Xeon Gold 5220R à 96 processeurs logiques, pour le rendu d'une scène de tokamak à plusieurs millions de triangles.

On obtient finalement un temps de génération d'image de l'ordre de la minute par image pour une scène de tokamak et en lançant un millier de rayons par pixels : ce temps de calcul permet la génération de grandes bases de données de simulations dans des temps raisonnables.

III.7. Validation du modèle direct

Différentes validations peuvent être effectuées pour garantir que le programme de lancer de rayons produit les valeurs attendues, conformément aux prédictions théoriques.

III.7.1. Validation de l'implémentation de la BRDF dans le rendu Monte Carlo

La validation de l'implémentation d'un modèle de BRDF et de son intégration dans l'algorithme de rendu Monte Carlo est réalisée via une scène de test spécifique qui permet de mesurer numériquement la BRDF implémentée et de la comparer à sa valeur théorique attendue.

Pour cela, la scène de test est construite en plaçant un plan horizontal sous un hémisphère composé de petits triangles. Chaque triangle est allumé successivement, tandis que les autres restent éteints. Une caméra d'un pixel, observe le point $\mathbf{x}$ du plan situé au centre de l'hémisphère. En effectuant un rendu par *backward ray tracing*, l'image générée permet de déterminer la

proportion de radiance émise par chaque triangle qui est réfléchie en direction de la caméra depuis le point **x**.

Selon la définition de la BRDF donnée par l'équation (III.7), il est possible de déduire la BRDF mesurée $f_r^{\text{mesurée}}(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o)$ à partir de la radiance mesurée L_o^{mesure}. En notant N le nombre de rayons lancés, $\boldsymbol{\omega}_i$ la direction allant de **x** au centre du triangle source, faisant un angle θ_i par rapport à la normale du plan, $\boldsymbol{\omega}_o$ la direction de **x** vers la caméra, L_i la radiance du triangle source, et ΔW_i l'angle solide sous lequel le triangle source est vu depuis **x**, l'équation (III.22) montre le calcul de la BRDF mesurée, et la condition sous laquelle elle est censée tendre vers la BRDF théorique.

$$\frac{L_o^{\text{mesurée}}}{L_i \cos(\theta_i) \Delta W_i} = f_r^{\text{mesurée}}(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \xrightarrow[\Delta W_i \to 0, \quad N \Delta W_i \to +\infty]{} f_r^{\text{théorique}}(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \tag{III.22}$$

Cette scène de test permet donc de valider les implémentations de BRDFs en s'assurant que l'algorithme de rendu parvient à reproduire la valeur théorique lorsque la taille des triangles source tend vers zéro et que le nombre de rayons lancés par triangle tend vers l'infini.

Un exemple de cette validation est donné pour une BRDF de Phong avec les paramètres $k_d = 0{,}4$, $k_s = 0{,}3$, et $n = 40$, que l'on compare à l'expression théorique fournie par l'équation (III.14), sur un hémisphère composé d'environ 10 000 triangles. La Figure III.14(Gauche) illustre l'erreur relative obtenue en fonction du nombre de rayons lancés. Les erreurs sont calculées pour une position de caméra spécifique ($\theta_o = \frac{\pi}{3}$) et agrégées sur tous les triangles $\mathcal{T}_i$ successivement allumés. L'erreur maximale et l'erreur moyenne tendent tous deux vers zéro à mesure que le nombre de rayons augmente, ce qui confirme la convergence vers la BRDF théorique. La Figure III.14(Droite) présente la BRDF mesurée pour chaque triangle de l'hémisphère.

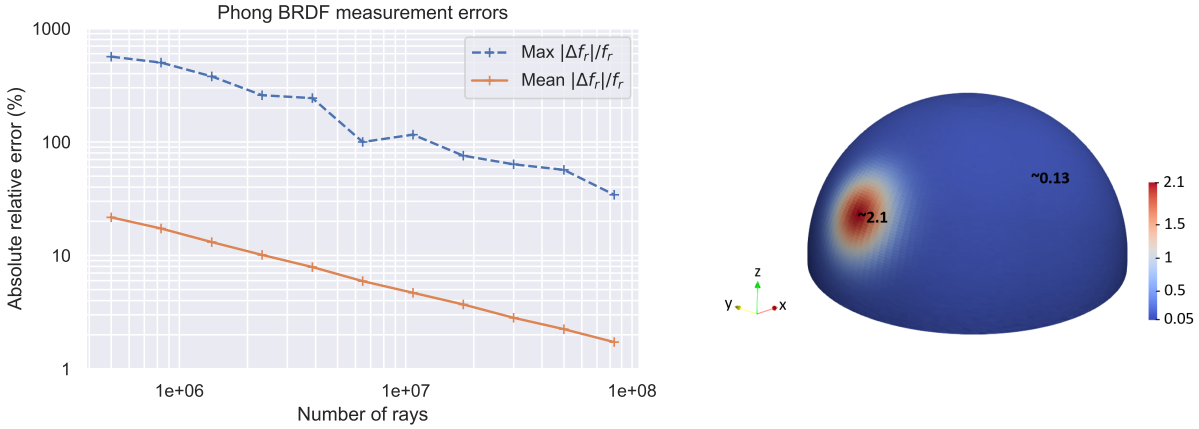


FIGURE III.14. – Mesure de BRDF normalisée de Phong avec $k_d = 0.4$, $k_s = 0.3$, $n = 40$ pour une direction de mesure à $\frac{\pi}{3}$ radians par rapport à la normale. **Gauche –** Erreurs relatives maximales et moyennes, sur tous les triangles successivement allumés, en fonction du nombre de rayons lancés. Les erreurs tendent vers 0, ce qui confirme la convergence vers la bonne valeur. **Droite –** Représentation de la BRDF mesurée pour chaque triangle de l'hémisphère.

III.7.2. Validation sur une scène fermée à température uniforme

La validation de l'implémentation de l'algorithme de lancer de rayons peut également être effectuée en générant des images dans des scénarios où le résultat théorique attendu est connu. Un test pertinent consiste à exploiter le cas de la sphère intégrante : pour une scène fermée où toutes les surfaces ont la même température T, la radiance sortante de n'importe quelle surface doit, quelle que soit la BRDF utilisée, être égale à la radiance corps noir, $L^\circ(T)$. Cette propriété peut être déduite en combinant l'équation du rendu (III.16) dans le cas d'une température uniforme, avec la loi de Kirchhoff liant l'émissivité et la réflectivité hémisphérique dans l'équation (III.11). Ce principe est d'ailleurs utilisé pour créer des sources de corps noirs en laboratoire : une enceinte est chauffée uniformément, et l'émission est observée à travers une petite ouverture.

Pour vérifier l'implémentation du lancer de rayons, une scène fermée et uniformément chauffée peut être créée, en attribuant à chaque surface des BRDFs aléatoires. La radiance capturée par une caméra devrait alors correspondre à la radiance du corps.

Dans cette démonstration, le modèle fermé du tokamak WEST est utilisé, tel que décrit en section III.8.1, observé depuis la caméra grand angle. Chaque triangle du modèle reçoit la même température T de 200 °C, et une BRDF de Phong dont les paramètres sont tirés aléatoirement par triangle. En lançant la simulation, on observe que les images générées deviennent de moins en moins bruitées à mesure que le nombre de rayons augmente, avec une radiance globalement uniforme qui converge vers celle du corps noir.

La Figure III.15 illustre ce processus appliqué au cas de l'image caméra grand angle de WEST. À gauche, les erreurs relatives moyennes et maximales de l'image sont tracées en fonction du nombre de rayons par pixel. On observe que l'erreur moyenne diminue régulièrement, tandis que l'erreur maximale converge vers une valeur limite, indiquant une convergence globale correcte mais avec quelques pixels présentant des anomalies. À droite, l'image obtenue pour 10 000 rayons par pixel montre encore un léger bruit. Un point sombre isolé (indiqué par un cercle vert) est visible vers le bas de l'image, probablement dû à un échec du lancer de rayon dans cette région, bien que l'image soit globalement correcte à l'exception de quelques pixels sous-estimés.

Ces résultats confirment la robustesse de l'implémentation de l'algorithme de lancer de rayons, renforçant ainsi la confiance dans la validité du modèle utilisé pour simuler les scènes complexes. La prochaine section détaille la modélisation qui est faite pour générer un jeu de données synthétique simulant le tokamak WEST en opération.

III.8. Modélisation des scènes thermiques

Cette section décrit les différents modèles et hypothèses utilisés pour la génération d'images synthétiques dans la machine de fusion WEST.

Pour générer un large jeu de données destiné à l'entraînement de l'intelligence artificielle, il est crucial de produire une diversité de scènes représentant ce qui pourrait se passer dans un tokamak lors des chocs. L'objectif est de générer facilement et rapidement des champs de température sur toutes les surfaces du tokamak, ainsi que les propriétés de réflexion et d'émission de ces surfaces.

S'il existe aujourd'hui des codes de physiques capables de prédire la distribution en flux et températures des composants face au plasma, tels que PFCFlux [Fir+13], ces codes, computation-

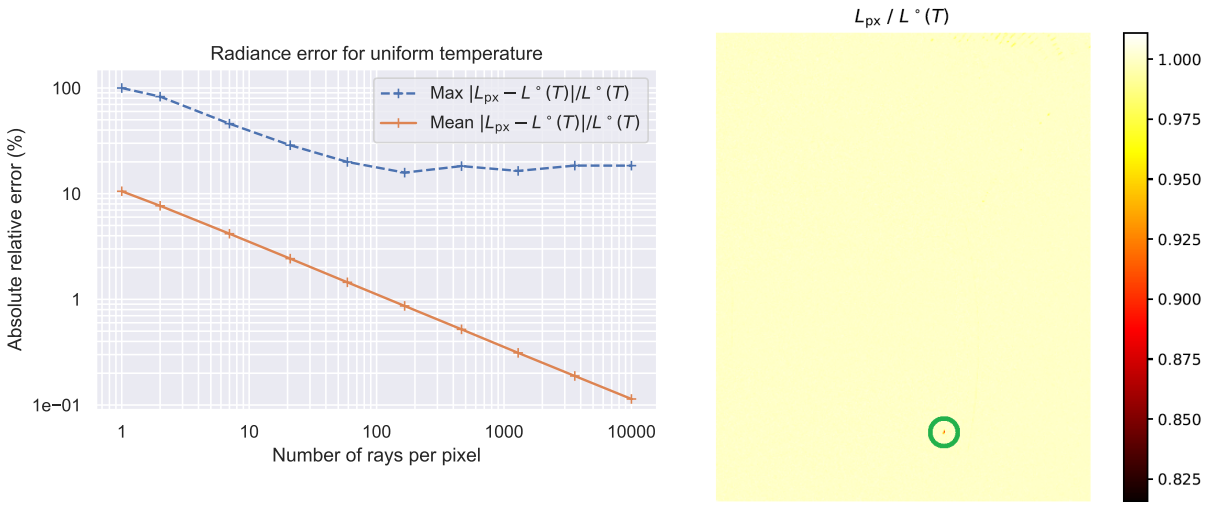


FIGURE III.15. – Validation du lancer de rayons sur une scène de tokamak à température uniforme et avec des BRDFs de Phong dont les paramètres sont tirés aléatoirement pour chaque triangle. **Gauche –** Erreurs relatives moyennes et maximales en fonction du nombre de rayons par pixel. L'erreur moyenne diminue avec l'augmentation des rayons, tandis que l'erreur maximale converge vers une valeur limite. **Droite –** Image obtenue avec 10 000 rayons par pixel montrant un léger bruit résiduel et un point sombre sur le bas de l'image probablement dû à un échec du lancer de rayon par instabilité numérique.

nellement intensifs et non parallélisés sur GPU, ne sont pas encore adaptés pour la génération de données massives. Dans le cadre de cette thèse, nous avons donc choisi d'approximer ces distributions de température à l'aide de modèles simples, paramétrables, et générant des motifs cohérents. Chaque paramètre peut être ainsi tiré aléatoirement pour couvrir un large espace, avec l'espoir qu'avec une modélisation suffisamment exhaustive, les situations pouvant survenir expérimentalement se retrouvent quelque part au sein de cet espace.

III.8.1. Le modèle 3D du tokamak et les caméras

Un modèle détaillé du tokamak WEST est utilisé pour générer des images synthétiques. Celui-ci est un maillage triangulaire surfacique composé d'environ 2,3 millions de points formant les sommets de 4,3 millions de triangles. Il est décomposé en 38 composants principaux, représentés par une couleur différente en Figure III.16.

Deux caméras du tokamak WEST sont modélisées par des caméras *pinhole* : la visée grand angle et une visée divertor (Q4A left). Les paramètres les caractérisant, tels que leur position et orientation, taille et nombre de pixels, champ de vue (FOV), et longueurs d'onde observées sont ajustés aux données réelles. Les paramètres sont décrits dans le Tableau III.1.

Pour tenir compte des incertitudes sur les paramètres réels des caméras et pour maîtriser le non-alignement parfait susceptible de varier au cours du temps (par exemple leur orientation qui peut légèrement varier au cours des chocs), une variation des paramètres des caméras est choisie aléatoirement pour chaque scène.

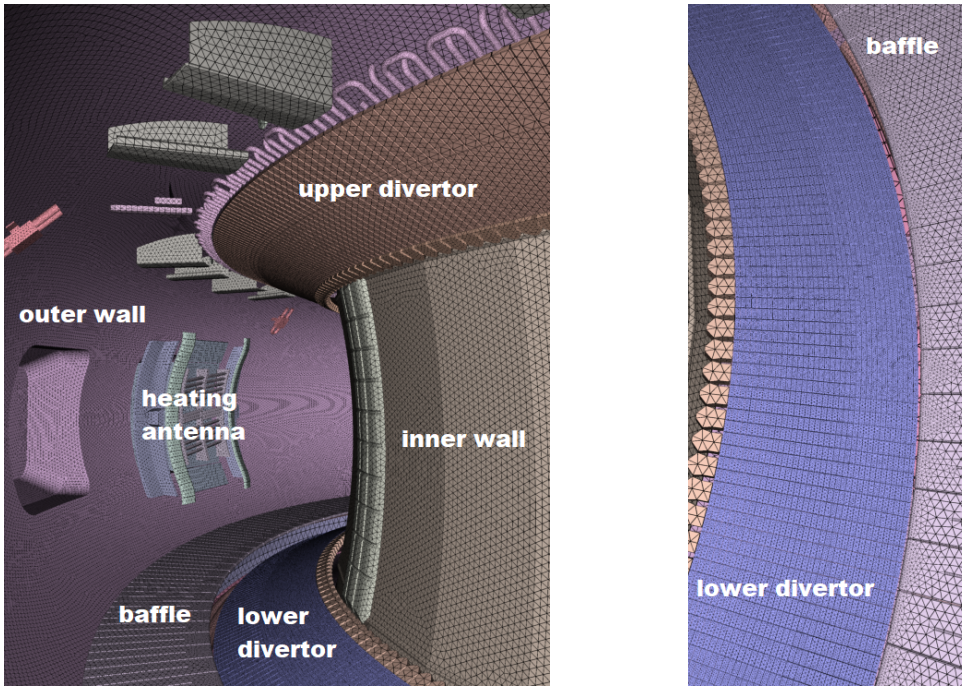


FIGURE III.16. – Modèle triangulaire surfacique du tokamak WEST, vu depuis la caméra grand angle et de la visée divertor Q4A left, utilisant une couleur différente par composant.

Parameter	Wide Angle Camera	Divertor Q4A left Camera
Position (mm)	[658.5, 230.5, 3301.6]	[-2424.7, 837.0, 912.8]
Direction	[-0.76768, 0.05370, -0.63858]	[0.37309, -0.91357, 0.16185]
Up	[0.04398, 0.99855, 0.03111]	[0.45828, 0.33314, 0.82402]
HFOV (°)	46	22.0
Pixel size (mm)	0.015	0.015
Height (px)	640	640
Width (px)	512	256
Filter λ (µm)	[4.3 ; 4.4]	[3.8 ; 4.0]

TABLEAU III.1. – Paramètres par défaut des caméras *pinhole* modélisant la visée grand angle et la vidée divertor Q4A left de WEST.

III.8.2. Température uniforme par composants

Chaque composant principal de la machine, qu'on peut observer dans des couleurs différentes en Figure III.16, se voit attribuer une température uniforme, tirée aléatoirement (de manière uniforme) entre une valeur minimale et maximale qui dépend du composant. Comme il sera vu en section III.8.6, une non-uniformité de la température au sein du composant sera appliquée par la suite via un bruit continu spatialement.

III.8.3. Mélange de BRDF Phong et miroir

Pour modéliser les réflexions sur les surfaces nous utilisons un mélange du modèle de BRDF de Phong normalisé et de mirroir parfait (voir section III.4.2.2). La BRDF résultante est fournie en équation (III.23).

$$f_r^{\text{Mirror}-\text{Phong}}(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = k_d \frac{1}{\pi} + k_s \frac{n+2}{2\pi} \max(0, \langle \boldsymbol{\omega}_o, \mathbf{r}(\boldsymbol{\omega}_i) \rangle)^n + k_m \frac{1}{\cos(\theta_i)} \delta_\omega(\boldsymbol{\omega}_o - \mathrm{r}(\boldsymbol{\omega}_i)) \quad \text{(III.23)}$$

Ainsi, cette BRDF est caractérisée par 4 paramètres : k_d, k_s, k_m et n, qui contrôle respectivement la part de réflectivité diffuse, spéculaire, miroir (spéculaire pure), et la largeur du lobe spéculaire de directions privilégiées. La sélection de ces paramètres permet ainsi de passer de réflexions purement spéculaires à purement diffuses, en passant par un spectre de comportements intermédiaires.

Chaque triangle du maillage se voit attribuer un ensemble de paramètres (k_d, k_s, k_m, n). Si chaque composant se voit tirer aléatoirement des paramètres de BRDF. Pour chaque composant, la réflectivité totale $\rho_\perp$ est tirée aléatoirement entre 0,5 et 0.95. Celle ci est alors aléatoirement répartie entre k_d, k_s et k_m de manière à ce que $k_d + k_s + k_m = \rho_\perp$. Le paramètre n est tiré selon une loi *log-uniform* entre 100 et 1000. Une variation spatiale intra-composant de chaque paramètre est ensuite réalisée (voir section III.8.6), créant ainsi un champ variable de BRDF au sein de la scène.

III.8.4. Profil de température du divertor bas

Pour modéliser des profils de températures cohérents en termes de motifs et de valeurs avec les observations expérimentales, notre approche consiste à développer une modélisation simple permettant de générer rapidement des motifs paramétrés. Cette modélisation s'appuie sur des modèles utilisés pour approximer la forme du flux reçu par le divertor (en $\mathrm{MW \cdot m^{-2}}$) et les utiliser comme première approximation des profils de température.

Pour modéliser le profil de température du divertor bas, le composant recevant le plus de flux, une modélisation du flux développée par T. Eich [Eic+11] a été utilisée. Cette modélisation montre que dans la configuration point X, le flux incident sur chacune des deux *strike line*, lorsqu'observé dans le sens radial, peut être approximé comme une exponentielle décroissante convoluée par une gaussienne. L'expression du flux q en fonction de la distance radiale r est ainsi donnée par l'équation (III.24).

$$q(r) = \frac{q_0}{2} \exp\left[\left(\frac{S}{2\lambda_q}\right)^2 - \frac{r - r_0}{\lambda_q}\right] \operatorname{erfc}\left[\frac{S}{2\lambda_q} - \frac{r - r_0}{S}\right] \quad \text{(III.24)}$$

Où r_0 paramètre la position radiale du pic de flux, λ_q est la longueur de décroissance exponentielle du flux, S est la largeur de la gaussienne, et q_0 contrôle la valeur maximale de flux, pouvant atteindre de 10 à 20 $\mathrm{MW \cdot m^{-2}}$. Ces paramètres permettent de décrire différentes positions, amplitudes, et étalements de pics de flux. La Figure III.17 montre différents profils obtenus en faisant varier λ_q et S pour un flux q_0 de 20 $\mathrm{MW \cdot m^{-2}}$ et pour un même r_0.

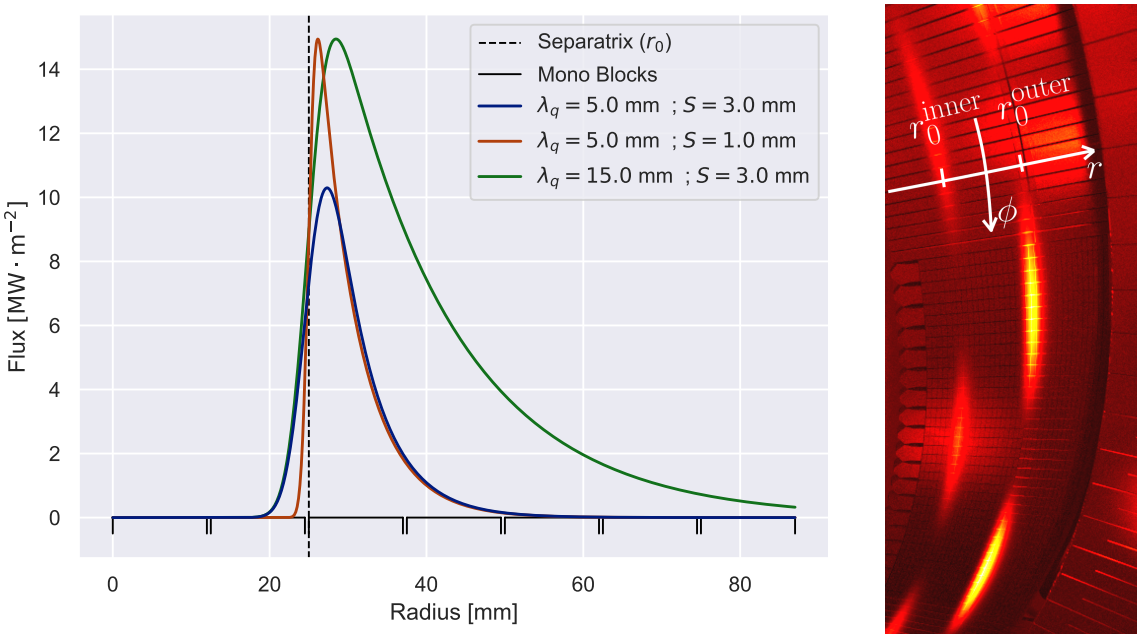


FIGURE III.17. – Gauche – Profils de flux sur le divertor bas pour différentes valeurs de longueur de décroissance (λ_q) et étalement gaussien (S) pour un flux maximum (q_0) (quand $S \to 0$) de 20 MW·m^{-2}. **Droite –** Simulation d'une image infrarouge de visée divertor. Les profils de température de deux *strike lines* reprennent le motif de Eich dans l'axe radial (r) et un effet de perte *ripple* dans l'axe poloïdal (ϕ).

La distance radiale seule ne suffit pas à déterminer le flux reçu par le divertor bas. Il y a également une variation dans la direction poloïdale du flux. En effet, le tokamak WEST contient 18 bobines toroïdales, dont l'espacement crée des variations dans le champ magnétique, résultant en une oscillation périodique du flux tous les dix-huitièmes de tour du tokamak due à des pertes de particules [Gas+21]. Cet effet est connu sous le nom de perte *ripple*. Cette oscillation peut être modélisée ici par une simple sinusoïde.

En combinant le motif radial fourni par la modélisation avec la variation poloïdale due à l'effet *ripple*, il est possible de créer un modèle de profil de température qui associe une valeur à toutes les coordonnées du divertor, comme décrit par l'équation (III.25).

$$T_{\text{Div}}(r,\phi) = T_{\text{off}} + \frac{T_0}{2} \exp\left[\left(\frac{S}{2\lambda_q}\right)^2 - \frac{r-r_0}{\lambda_q}\right] \operatorname{erfc}\left[\frac{S}{2\lambda_q} - \frac{r-r_0}{S}\right] \left(1 + \sin\left[k\left(\phi-\phi_0\right)\right]\right) \tag{III.25}$$

Dans cette équation, T_{off} représente la température minimale, T_0 permet de contrôler le pic de température, r_0 et ϕ_0 contrôlent la position de la *strike line*, et λ_q et S déterminent l'étalement du profil. En faisant varier ces différents paramètres, il est possible de générer des profils de température cohérents couvrant différents cas expérimentaux.

En réalité, il existe deux *strike lines* : une intérieure et une extérieure. Le divertor présente donc

deux motifs de ce genre à deux rayons différents. Les mêmes paramètres sont utilisés pour les deux, sauf la position de la *strike line* et leur intensité maximale respective.

III.8.5. Profil de température des antennes de chauffage

Les antennes de chauffage, dont l'une est visible depuis la caméra grand angle de WEST, sont des composants dont les protections latérales chauffent particulièrement, de manière non uniforme. Ne disposant pas de modèle permettant de générer de tels motifs pour le jeu de données synthétique, des données expérimentales sont utilisées pour les reproduire approximativement.

Les températures des protections de l'antenne visible depuis la caméra grand angle sont obtenues en projetant des images expérimentales (en température apparente de corps noir) sur le maillage 3D. Pour introduire des variations dans les motifs, une moyenne pondérée de 10 distributions (prises à partir de différentes images expérimentales) est sélectionnée aléatoirement pour chaque image générée, puis mise à l'échelle de manière aléatoire.

Des exemples de profils de température générés de cette manière sont présentés en Figure III.18.



FIGURE III.18. – Exemples de profils de températures sur les protections latérales d'antennes. Les profils sont obtenus via une projection 2D vers 3D d'une combinaison linéaire aléatoire d'images expérimentales en température apparente.

III.8.6. Non-uniformité des composants par bruit de gradient et points chauds

L'objectif est de générer facilement une diversité de profils de température ou de paramètres de réflectivité au sein de la scène. Une solution simple pour éviter l'uniformité des composants consiste à utiliser un bruit spatialement cohérent pour moduler les valeurs à l'intérieur de chaque composant. Pour cela, le bruit de Perlin [Per85] a été utilisé.

Le bruit de Perlin est une technique largement utilisée en informatique graphique pour la génération procédurale de textures. Il permet de générer un bruit aléatoire continu spatialement, et peut être défini en n'importe quelle dimension, par exemple : 1D pour créer une séquence temporelle variant aléatoirement, 2D pour des textures, et 3D pour des volumes. Dans ce contexte, le bruit de Perlin en trois dimensions sera utilisé pour introduire des variations dans des champs de température 3D ou d'autres paramètres de la scène.

Cet algorithme définit une grille régulière partitionnant l'espace. Pour chaque nœud de la grille, un vecteur unitaire aléatoire, appelé gradient, est assigné. Pour chaque point **p** de l'espace, les gradients des sommets de la cellule à laquelle appartient le point (4 vecteurs en 2D, 8 vecteurs en 3D) sont utilisés pour attribuer une valeur de bruit à cet endroit. Cette valeur est obtenue en interpolant les différents produits scalaires entre ces gradients et les vecteurs reliant chaque nœud au point **p**.

Concrètement, pour chaque point **p**, les étapes suivantes sont suivies :

1. Identifier les sommets de la cellule de la grille à laquelle appartient **p**.
2. Calculer les vecteurs reliant chaque sommet de la cellule au point **p**.
3. Effectuer le produit scalaire entre ces vecteurs et les gradients assignés aux sommets correspondants.
4. Interpoler les valeurs obtenues en utilisant une fonction d'interpolation douce, généralement la fonction cubique *smoothstep* ou quintique *smootherstep*, pour assurer une transition continue et lisse du bruit dans l'espace.

Généralement, plusieurs bruits de Perlin, définissant des grilles de différentes tailles, sont combinés linéairement afin de générer des textures complexes, présentant des variations à différents niveaux d'échelle. Un exemple de bruit de Perlin en dimension 2 est fourni en Figure III.19.

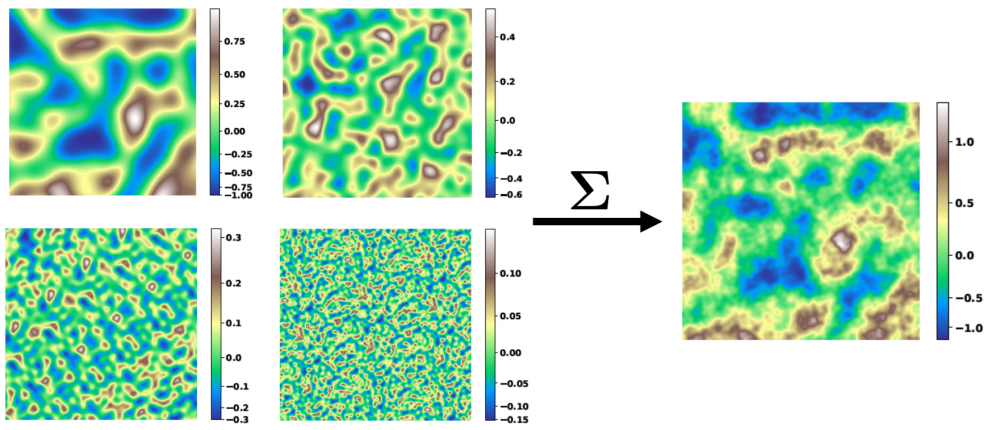


FIGURE III.19. – Exemple de bruits de Perlin en dimension 2, utilisés pour générer des variations aléatoires et cohérentes spatialement. Une somme de bruits de Perlin de différentes tailles de grille et amplitudes peuvent être sommés pour générer des textures complexes.

De la même manière, une somme de bruits de Perlin 3D, générés aléatoirement, est utilisée pour moduler le champ de température ou d'émissivité des composants. Par exemple, ces variations peuvent entraîner des fluctuations de plus ou moins 20 %.

Enfin, des augmentations ponctuelles de température sont également modélisées à l'aide de formes gaussiennes, dont la position, amplitude et largeur sont tirées aléatoirement, ce qui permet d'augmenter localement la température de surface de 20 % à 200 %.

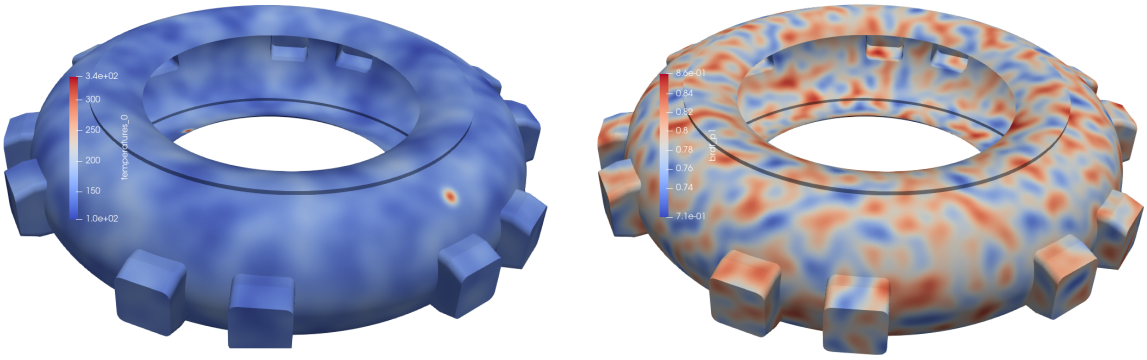


FIGURE III.20. – Exemples de champs non uniformes générés sur la paroi externe du tokamak. **Gauche –** Champ de température, les non-uniformités dues au bruit de Perlin et à des points chauds aléatoires peuvent être observées. **Droite –** Champ de réflectivité spéculaire.

Parameter	Min	Max	Parameter	Min	Max
T_{uniform}	50 °C	200 °C	$\rho_\perp = k_d + k_s + k_m$	0.5	0.95
T_{glow}	50 °C	700 °C	$\log_{10} n$	2	3
$T_{\text{hot pipes}}$	200 °C	700 °C	Antenna scaling	0.5	5.0
r_0 (inner)	2.08 m	2.12 m	Cam displacement	-2 cm	+2 cm
r_0 (outer)	2.25 m	2.29 m	Cam rotation	-2°	+2°
ϕ_0	0	2π	Cam FOV variation	-5°	+5°
T_0 (inner)	300 °C	1000 °C	T variation	±0 %	±40 %
T_0 (outer)	850 °C	1500 °C	k_d, k_s, k_m, n variation	±0 %	±10 %
λ_q	1.5 cm	4.0 cm	Nb hot spots	1	100
S	1.2 cm	2.0 cm	Hot spot size	1 cm	7.5 cm
T_{off}	80 °C	100 °C	Hot spot amplitude	+20 %	+300 %

TABLEAU III.2. – Bornes minimales et maximales des paramètres pour la modélisation des scènes thermiques. Les scènes du jeu de donné synthétique sont générées en tirant uniformément entre ces bornes chacun des paramètres.

III.8.7. Base de donnée d'images générées

En utilisant la modélisation décrite précédemment, une scène thermique observée par une caméra est décrite par 200 paramètres fixes, auxquels viennent s'ajouter les perturbations aléatoires dues au bruit de Perlin et aux points chauds. En tirant aléatoirement chaque paramètre caractérisant la caméra et les profils de température et de BRDF, une base de donnée de scènes aléatoires peut être générée. Les bornes minimales et maximales de variations des paramètres est résumée par le Tableau III.2.

La Figure III.21 montre quelques images générées avec cette modélisation du tokamak, en utilisant la caméra grand angle et en lançant 1000 rayons par pixels.

En sauvegardant les images générées ainsi qu'une carte de température vraie, obtenue par projection 3D vers 2D du champ de température fournie en entrée du *ray tracer*, cette modélisation permet ainsi la génération d'un grand jeu de données de scénarios diverses pour l'apprentissage

automatique, fournissant une association directe entre image infrarouge et carte de température.

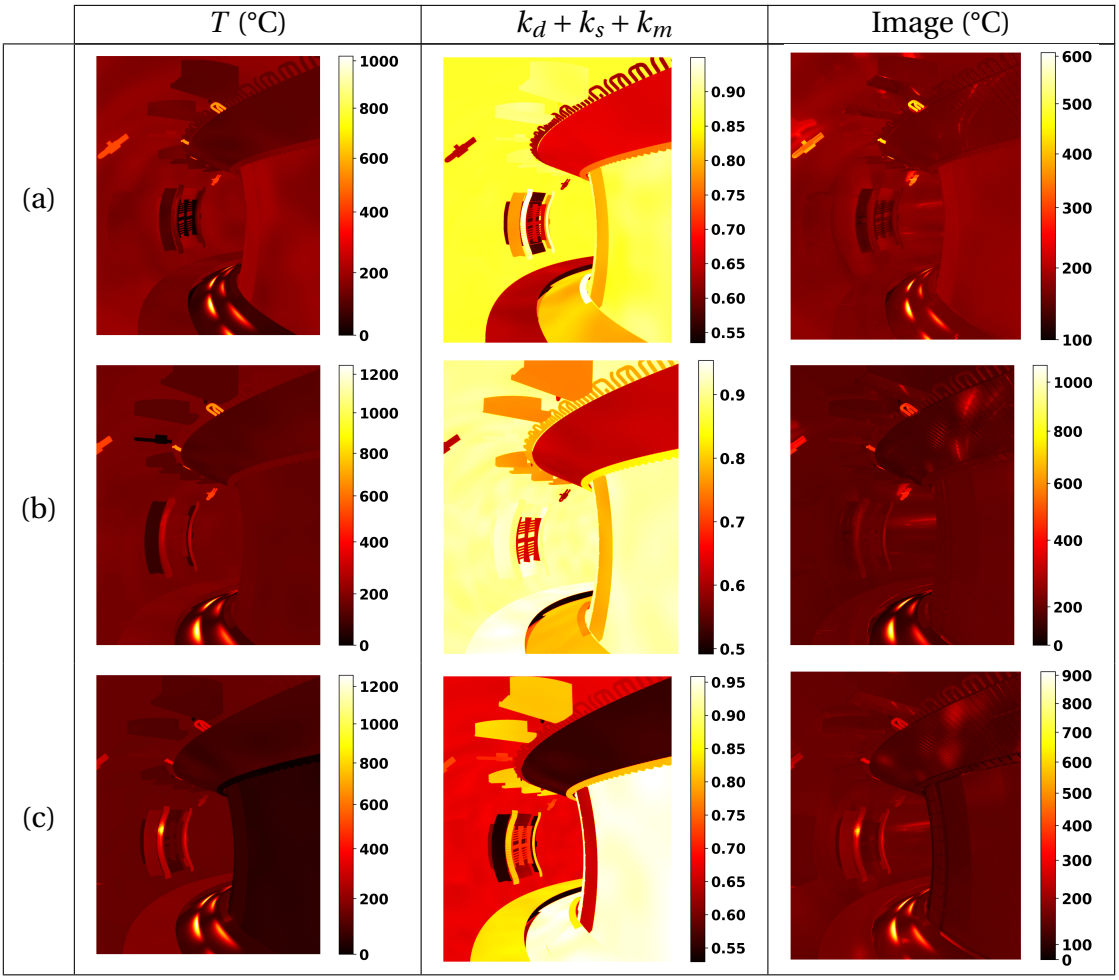


FIGURE III.21. – Exemples de champs de température et de réflectivités générés, ainsi que l'image simulée résultante.

Conclusion III

Dans ce chapitre, il est montré que les logiciels de rendus existants, principalement appliqué pour le rendu visuel dans le visible, étaient inadaptés pour la modélisation des scènes thermiques et pour certains, limités pour en termes de temps de calcul. Un nouveau code de rendu dédié à la modélisation des images infrarouge dans les machines de fusion a donc été développé.

Ce logiciel est basé sur une intégration de Monte Carlo, résolvant les équations de rendu via une propagation aléatoire de rayons depuis le pixel vers la scène observée (*backward ray tracing*). Grâce au modèle de BRDF, il est possible de décrire pour chaque triangle de la géométrie 3D ses propriétés de réflexions (et donc d'émissivité) avec une grande précision et hétérogénéité.

Ce modèle gère efficacement des scènes géométriques de grande taille grâce à une structure hiérarchique des triangles, réduisant ainsi le coût computationnel des calculs d'intersection rayon-triangle. L'utilisation des GPU a considérablement accéléré le temps de calcul des images, de l'ordre d'une minute pour les images que l'on cherche à simuler, rendant possible la création de jeux de données conséquents adaptés pour l'apprentissage automatique.

En modélisant les profils de température et les propriétés optiques par des fonctions paramétrables, menant à un total de 200 paramètres par scène, une diversité de scènes a été générée en tirant aléatoirement ces paramètres. Cette approche a permis de stocker l'association entre les cartes de température et les images générées, créant ainsi un jeu de données qui reproduit les conditions présentes dans un tokamak.

Cette base de données ouvre la voie à l'utilisation de l'apprentissage automatique, et plus spécifiquement des réseaux de neurones artificiels, pour entraîner un modèle inverse capable de retrouver les températures à partir des images infrarouges. La section suivante explorera en détail les résultats obtenus en appliquant cette méthode.

IV. Inversion des images infrarouge par réseau neurones à convolution

Résumé

Dans le chapitre précédent, une méthode rapide de génération de données synthétiques a été présentée, permettant de constituer des bases de données d'apprentissage représentatives des scénarios dans une machine de fusion. Ce chapitre introduit le concept général d'apprentissage profond supervisé appliqué au traitement d'image. Ensuite, une architecture de réseau de neurone artificiel de type U-Net est présentée. Ce réseau est entraîné à estimer les températures de surface à partir d'images infrarouge, et évalué sur un jeu de données simulé de scénarios standards. Enfin, les limites de ce modèle déterministe sont explorées, notamment dans les cas de problèmes mal posés, où fournir une unique sortie peut ne pas être suffisante.
Ces travaux ont fait l'objet de publications : [JAM24b ; Aum+24].

Sommaire

IV.1. Introduction à l'apprentissage automatique . . . 75
IV.1.1. Apprentissage supervisé et fonction de perte . . . 75
IV.1.1.1. Fonction de perte . . . 75
IV.1.1.2. Ensemble d'apprentissage, de validation, et de test . . . 75
IV.1.2. Optimiseurs pour minimiser la fonction de perte . . . 76
IV.1.2.1. La descente de gradient ordinaire . . . 76
IV.1.2.2. L'algorithme de descente de gradient stochastique . . . 77
IV.1.2.3. Variantes de descente de gradient adaptative . . . 78
IV.2. Réseaux de neurones et apprentissage profond pour le traitement d'image . . . 79
IV.2.1. Architecture des réseaux de neurones . . . 79
IV.2.1.1. Le perceptron . . . 79
IV.2.1.2. La couche dense . . . 80
IV.2.1.3. Le perceptron multicouche . . . 81
IV.2.2. L'algorithme de rétropropagation du gradient . . . 82
IV.2.2.1. Algorithme général . . . 82
IV.2.2.2. La couche dense dans l'algorithme de rétropropagation . . . 84
IV.2.3. Couches adaptées au traitement d'image . . . 85
IV.2.3.1. Couches de convolution . . . 85
IV.2.3.2. Downsampling, upsampling . . . 86

IV.2.3.3. Skip connection et couche résiduelle 86
IV.2.4. Architectures de réseaux pour le traitement d'image 87
IV.2.4.1. L'architecture de l'auto-encodeur 87
IV.2.4.2. L'architecture du U-Net 88
IV.3. Estimation de température par U-Net sur des données simulées 89
IV.3.1. Modélisation de la scène thermique 89
IV.3.2. Génération des données 91
IV.3.3. Apprentissage 92
IV.3.4. Pertinence de l'ensemble de données 93
IV.3.5. Résultats sur images simulées 94
IV.3.5.1. Images de test standards 94
IV.3.5.2. Images détériorées 95
IV.3.5.3. Motif de température inattendu 96
IV.3.6. Application à une image expérimentale 97
IV.4. Limite de la méthode : problème mal posé sur des scènes en visée divertor 98

IV.1. Introduction à l'apprentissage automatique

L'apprentissage automatique a révolutionné de nombreux domaines en permettant aux machines d'apprendre à réaliser des tâches à partir de la lecture de données. Cette section introduit les éléments fondamentaux de l'apprentissage automatique, plus précisément l'apprentissage supervisé, qui permet à un modèle de s'améliorer dans l'exécution d'une tâche en utilisant un ensemble d'exemples disponibles. Ces méthodes seront utilisées pour entraîner des modèles de réseaux de neurones artificiels, qui seront présentés en détail dans la section IV.2.

IV.1.1. Apprentissage supervisé et fonction de perte

L'apprentissage supervisé consiste à enseigner à un modèle $\mathcal{M}(\cdot|\boldsymbol{\theta})$, paramétré par un vecteur de paramètres $\boldsymbol{\theta} \in \mathbb{R}^d$, à approximativement associer des entrées $\mathbf{x} \in \mathbb{R}^p$ à des sorties de $\mathbf{y} \in \mathbb{R}^n$ grâce à des exemples à dispositions. Cela se fait en utilisant un jeu de données étiquetées, c'est-à-dire un ensemble $\mathcal{D}$ de couples $(\mathbf{x}, \mathbf{y})$, où l'objectif est de trouver un jeu de paramètres $\boldsymbol{\theta}^*$ tel que $\forall (\mathbf{x}, \mathbf{y}) \in \mathcal{D}, \quad \mathcal{M}(\mathbf{x}|\boldsymbol{\theta}^*) \approx \mathbf{y}$.

IV.1.1.1. Fonction de perte

Pour évaluer la qualité de cette approximation, une fonction de perte $\mathcal{L}$ (aussi appelée erreur ou coût) est définie. Cette fonction de perte est minimale lorsque le modèle fournit une bonne approximation de l'association entre $\mathbf{x}$ et $\mathbf{y}$. Par exemple, l'erreur quadratique moyenne (*Mean Squared Error*, MSE) peut être utilisée comme fonction de perte :

$$\mathcal{L}_{\text{MSE}}(\boldsymbol{\theta}, \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}} \left\| \mathcal{M}(\mathbf{x}|\boldsymbol{\theta}) - \mathbf{y} \right\|^2. \tag{IV.1}$$

L'objectif de la procédure d'apprentissage est alors de trouver l'ensemble de paramètres optimaux $\boldsymbol{\theta}^*$ qui minimise cette fonction de perte, c'est-à-dire :

$$\boldsymbol{\theta}^* = \text{argmin}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \mathcal{D}). \tag{IV.2}$$

Cette approche permet de résoudre des problèmes de régression, où l'objectif est d'approximer une fonction continue, ainsi que des problèmes de classification, où le modèle apprend à associer une entrée à une catégorie spécifique, par exemple en renvoyant 1 pour la composante correspondant à la classe correcte et 0 pour les autres.

IV.1.1.2. Ensemble d'apprentissage, de validation, et de test

L'objectif de l'apprentissage n'est pas strictement de trouver un jeu de paramètres $\boldsymbol{\theta}$ qui minimise l'erreur sur les données utilisées pour l'entraînement. Il est essentiel de développer un modèle capable de fournir des prédictions précises sur des données non vues durant l'apprentissage.

Un modèle qui minimise uniquement la fonction de perte sur l'ensemble d'apprentissage pourrait simplement mémoriser les exemples vus, ce qui conduirait à une mauvaise généralisation sur des données nouvelles. Pour éviter ce phénomène de surapprentissage (*overfitting*),

les procédures d'apprentissage et de test des modèles séparent les données en trois ensembles disjoints : un ensemble d'apprentissage, un ensemble de validation, et un ensemble de test.

L'ensemble d'apprentissage est utilisé pour ajuster les paramètres internes du modèle afin de minimiser la fonction de coût.

L'ensemble de validation sert à évaluer régulièrement la performance du modèle sur des données non utilisées pour l'apprentissage. Pendant l'apprentissage, la fonction de coût est calculée sur cet ensemble de validation. Tant que cette perte diminue, le modèle améliore sa capacité à généraliser. Lorsque la perte sur l'ensemble de validation stagne ou commence à augmenter, cela indique que le modèle commence à surapprendre les données d'entraînement. À ce stade, il est judicieux d'arrêter l'apprentissage et de conserver les paramètres obtenus.

Enfin, l'ensemble de test est utilisé pour évaluer les performances finales du modèle sélectionné. Cet ensemble n'est utilisé qu'une fois l'apprentissage terminé, afin de fournir une estimation impartiale des performances du modèle sur des données complètement nouvelles.

IV.1.2. Optimiseurs pour minimiser la fonction de perte

Différentes méthodes d'optimisation, connues sous le nom d'optimiseurs, peuvent être utilisées pour ajuster les paramètres internes d'un modèle à la minimisation d'une fonction de perte. Cette section présente les principaux algorithmes d'optimisation utilisés en apprentissage automatique, en commençant par la descente de gradient, avant d'explorer ses variantes telles que la descente de gradient stochastique et les méthodes adaptatives.

IV.1.2.1. La descente de gradient ordinaire

Pour trouver un ensemble de paramètres internes du modèle $\boldsymbol{\theta}$ qui minimise la fonction de perte $\mathcal{L}(\boldsymbol{\theta}, \mathcal{D})$, souvent non résoluble analytiquement, la méthode couramment employée est la descente de gradient. Le gradient d'une fonction $f : \mathbb{R}^p \to \mathbb{R}$ différentiable en un point $\mathbf{p}$ est un vecteur défini par

$$\nabla f(\mathbf{p}) = \begin{bmatrix} \partial_1 f(\mathbf{p}) & \cdots & \partial_p f(\mathbf{p}) \end{bmatrix}^\top \tag{IV.3}$$

où $\partial_j f(\mathbf{p})$ est la dérivée partielle de f par rapport à la j-ème coordonnée de $\mathbf{p}$. Le gradient a la propriété de pointer dans la direction de plus forte variation de f au point $\mathbf{p}$. En effet, aux alentours de $\mathbf{p}$, pour un petit déplacement infinitésimal $\mathrm{d}\mathbf{p}$, on a

$$f(\mathbf{p} + \mathrm{d}\mathbf{p}) = f(\mathbf{p}) + \nabla f(\mathbf{p})^\top \mathrm{d}\mathbf{p} + o\left(\left\|\mathrm{d}\mathbf{p}\right\|\right). \tag{IV.4}$$

Autrement dit, l'augmentation de f est maximale lorsque le produit scalaire entre le gradient et le petit déplacement est maximal, c'est-à-dire lorsqu'ils sont dans le même sens.

Inversement, pour minimiser une fonction, il est possible de partir d'un point de départ et de se déplacer dans le sens opposé à son gradient, d'où le nom d'algorithme de descente de gradient. Lorsque la fonction à minimiser est la fonction de perte $\mathcal{L}$, il est possible de commencer avec un ensemble de paramètres initiaux $\boldsymbol{\theta}_0$. Les paramètres sont alors mis à jour itérativement en se déplaçant dans le sens opposé du gradient :

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t, \mathcal{D}), \tag{IV.5}$$

où $\eta_t > 0$ est un pas d'apprentissage, permettant de contrôler l'ampleur des déplacements effectués à chaque itération. Ce pas ne doit pas être trop grand pour ne pas déstabiliser la convergence de l'algorithme, ni trop petit pour éviter un nombre excessif d'itérations. En pratique, ce pas peut être fixe, ou être adaptatif de manière à décroître d'itération en itération.

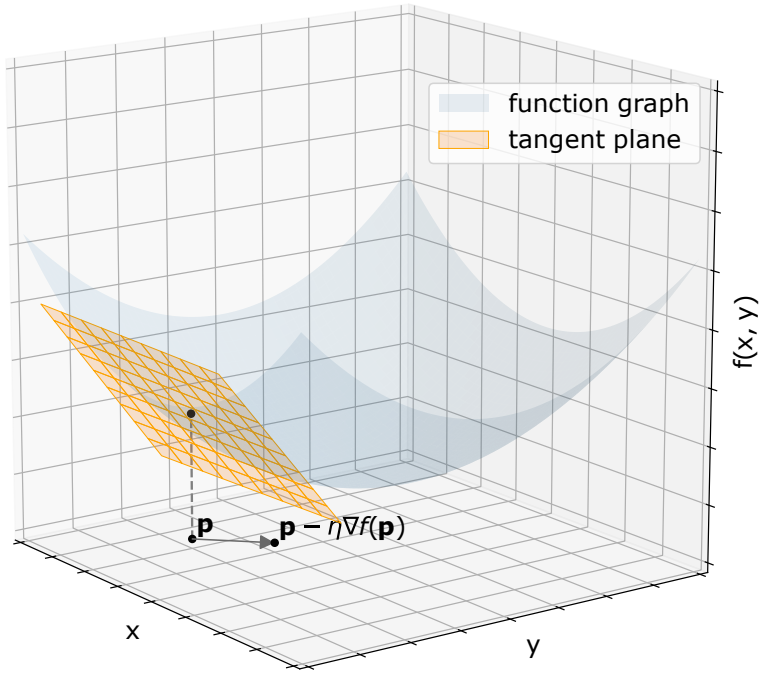


FIGURE IV.1. – Descente de gradient d'une fonction f en un point $\mathbf{p} = (x, y)$. Le gradient $\nabla f(\mathbf{p})$ permet d'approximer f par une fonction affine aux alentours de $\mathbf{p}$. Un petit déplacement de $-\eta \nabla f(\mathbf{p})$ permet d'aller en direction de la plus forte décroissance locale de f.

IV.1.2.2. L'algorithme de descente de gradient stochastique

La procédure simple de descente de gradient décrite précédemment présente un problème computationnel. En définissant la fonction de coût comme étant une erreur moyenne calculée en évaluant le modèle $\mathcal{M}$ sur l'ensemble complet $\mathcal{D}$, l'évaluation devient computationnellement coûteuse si l'ensemble $\mathcal{D}$ est de grande taille (qui typiquement vaut plusieurs dizaines de milliers).

Ce problème de complexité computationnelle est résolu en utilisant un algorithme de descente de gradient stochastique. Plutôt que de calculer l'erreur $\mathcal{L}$ sur l'ensemble $\mathcal{D}$, à chaque itération, un sous-ensemble de $\mathcal{D}$, de taille beaucoup plus petite , $\mathcal{B}_t$, appelé *batch*, est sélectionné aléatoirement. La fonction de perte et son gradient sont alors calculés uniquement sur ce batch, et les paramètres sont mis à jour de la manière suivante :

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t, \mathcal{B}_t). \tag{IV.6}$$

La sélection aléatoire d'un *batch* ne se fait pas par simple tirage avec remise des éléments. L'ensemble d'apprentissage $\mathcal{D}$ est préalablement réordonné en choisissant aléatoirement une permutation. À chaque itération, les prochains éléments consécutifs non utilisés sont choisis pour constituer le *batch* courant. Après avoir utilisé tous les éléments de l'ensemble, une nouvelle permutation de $\mathcal{D}$ est effectuée et le processus recommence. L'ensemble des étapes menant à une utilisation de l'intégralité de l'ensemble d'apprentissage est appelé une *epoch*. L'apprentissage se fait alors en décidant du nombre d'*epoch* à réaliser (typiquement, une centaine) et de la taille des *batch* (typiquement quelques dizaines d'éléments).

IV.1.2.3. Variantes de descente de gradient adaptative

Différentes variantes de la descente de gradient stochastique ont été introduites afin d'accélérer la convergence de l'algorithme et d'éviter les situations où l'algorithme pourrait être bloqué dans une solution sous-optimale (point col ou minimum local de la fonction de coût). L'idée générale est de réaliser une mise à jour en utilisant le gradient à l'instant t, tout en incorporant une information prenant en compte la valeur des précédents gradients. Par exemple, il est possible d'ajouter un moment (similairement à la notion de moment en physique), qui agit comme une inertie servant de mémoire des gradients précédents :

$$\Delta\boldsymbol{\theta}_{t+1} = \alpha\Delta\boldsymbol{\theta}_t - \eta_t\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_t, \mathcal{B}_t) \quad ; \quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{t+1} \tag{IV.7}$$

où $\alpha \in [0;1[$ est un facteur de décroissance permettant de garder en mémoire la valeur des précédents gradients. Cette méthode améliore la convergence en limitant les oscillations du gradient introduites par la nature stochastique de la sélection des batchs.

De nombreuses autres variantes de cette méthode ont été développées et sont utilisées. Ces variantes incluent des méthodes de gradient et de moments adaptatifs [DHS11], qui prennent en compte l'historique des gradients et leurs valeurs pour ajuster la taille du pas d'apprentissage à chaque itération pour chaque composante.

Enfin, l'algorithme d'optimisation le plus souvent utilisé est Adam (*Adaptive Moment Estimation*) [KB14], qui combine les différentes idées de ces optimisateurs et a montré empiriquement d'excellentes performances. Adam utilise à la fois les moyennes mobiles (lissage exponentiel) des gradients et des carrés des gradients pour adapter le pas d'apprentissage. L'algorithme Adam est décrit par les équations suivantes :

$$\begin{aligned}
&\mathbf{m}_{t+1} = \beta_1\mathbf{m}_t + (1-\beta_1)\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_t, \mathcal{B}_t) \quad ; \quad \hat{\mathbf{m}}_{t+1} = \frac{\mathbf{m}_{t+1}}{1-\beta_1^{t+1}} \\
&\mathbf{v}_{t+1} = \beta_2\mathbf{v}_t + (1-\beta_2)\left(\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_t, \mathcal{B}_t)\right)^2 \quad ; \quad \hat{\mathbf{v}}_{t+1} = \frac{\mathbf{v}_{t+1}}{1-\beta_2^{t+1}} \\
&\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t\frac{\hat{\mathbf{m}}_{t+1}}{\sqrt{\hat{\mathbf{v}}_{t+1}} + \epsilon}
\end{aligned} \tag{IV.8}$$

où $\mathbf{m}_{t+1}$ est la moyenne mobile des gradients, $\mathbf{v}_{t+1}$ est la moyenne mobile des carrés des gradients, $\hat{\mathbf{m}}_{t+1}$ et $\hat{\mathbf{v}}_{t+1}$ sont les versions corrigées des biais de $\mathbf{m}_{t+1}$ et $\mathbf{v}_{t+1}$ respectivement, β_1 et β_2 sont les coefficients de décroissance pour les moyennes mobiles (typiquement $\beta_1 = 0.9$ et

$\beta_2 = 0.999$), η_t est le taux d'apprentissage (souvent laissé constant, avec une valeur typique de 10^{-4}), ϵ est un petit terme de stabilisation pour éviter la division par zéro (typiquement $\epsilon = 10^{-8}$).

IV.2. Réseaux de neurones et apprentissage profond pour le traitement d'image

Nous avons vu comment réaliser l'entraînement d'un modèle $\mathcal{M}(\cdot|\boldsymbol{\theta})$ de manière supervisée, en définissant une fonction de perte $\mathcal{L}$ qui peut être minimisée à condition de pouvoir calculer son gradient $\nabla_{\boldsymbol{\theta}}\mathcal{L}$. Nous présentons ici les types de modèle les plus courament utilisés dans le cadre cet algorithme de minimisation : les réseaux de neurones profonds.

Les réseaux de neurones artificiels, des modèles inspirés du fonctionnement du cerveau humain, sont aujourd'hui omniprésents dans notre quotidien. Ils permettent la reconnaissance vocale dans les assistants virtuels, la reconnaissance ou la génération d'images pour des applications allant de la santé à la robotique autonome, ou encore le traitement du langage naturel pour des tâches telles que la traduction automatique et la création d'agents conversationels.

La puissance des réseaux de neurones réside dans leur capacité à approximer des fonctions hautement non linéaires et à s'améliorer avec l'acquisition de nouvelles données, rendant possible des tâches autrefois considérées comme insurmontables.

Cette section présente leur fonctionnement et les modèles pertinents à utiliser dans le cadre du traitement d'images infrarouges.

IV.2.1. Architecture des réseaux de neurones

L'architecture d'un réseau de neurones définit la manière dont ses unités de base, les neurones artificiels, sont organisées et interconnectées pour traiter les informations. Cette section explore les principales composantes architecturales des réseaux de neurones, en commençant par le perceptron, l'unité de calcul élémentaire, pour ensuite aborder les couches denses et les perceptron multicouche, qui permettent de représenter des fonctions plus complexes.

IV.2.1.1. Le perceptron

Le perceptron est l'unité de calcul élémentaire des réseaux de neurones artificiels. Il prend une entrée $\mathbf{x} \in \mathbb{R}^p$ et renvoie une sortie scalaire $y \in \mathbb{R}$. Les paramètres internes du perceptron sont les poids $\mathbf{w}$ et le biais b, formant le vecteur $\boldsymbol{\theta} = \begin{bmatrix} b & w_1 & \cdots & w_p \end{bmatrix}^\top$. La sortie du perceptron est obtenue par une combinaison linéaire de ses entrées suivie d'une fonction d'activation non linéaire σ, comme exprimé en équation (IV.9).

$$y = \sigma\left(\sum_{j=1}^{p} w_j x_j + b\right) = \sigma\left(\mathbf{w}^\top \mathbf{x} + b\right) \tag{IV.9}$$

La fonction d'activation σ est non linéaire et différentiable. Historiquement, la fonction sigmoïde était utilisée, mais les fonctions ReLU (*Rectified Linear Unit*) et ses variantes comme le LeakyReLU sont désormais plus courantes. Ces fonctions sont illustrées en Figure IV.3.

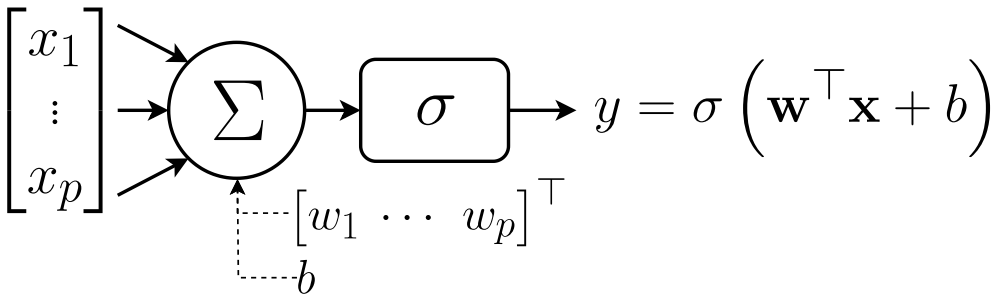
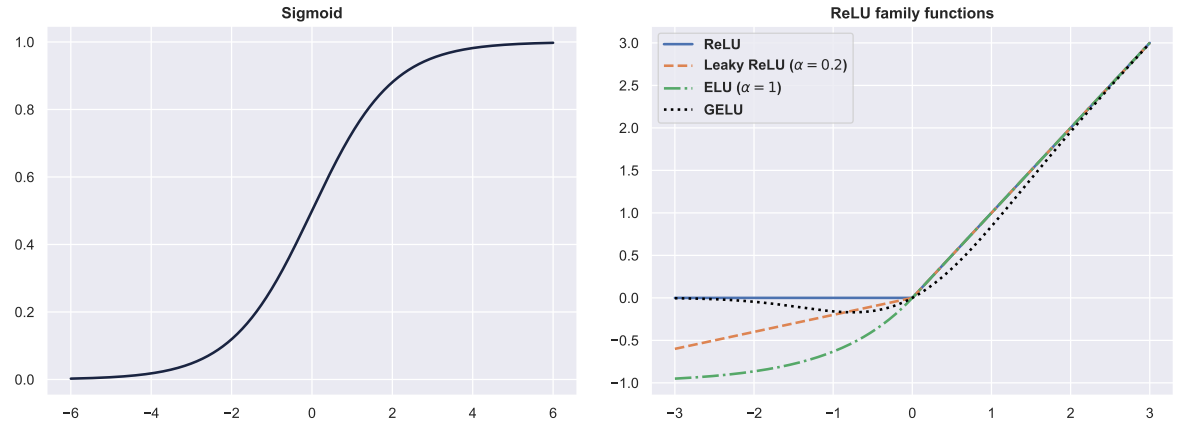


FIGURE IV.2. – Représentation d'un perceptron à p entrées. Le perceptron utilise ses poids w_j, biais b et fonction d'activation σ pour former une sortie scalaire.



(a) Sigmoïde

(b) ReLU et ses variantes

FIGURE IV.3. – Fonctions d'activations courantes permettant d'introduire des non-linéarités dans les réseaux de neurones

Le perceptron établit un parallèle intéressant avec les neurones biologiques. Dans le cerveau, un neurone biologique reçoit des signaux à travers ses dendrites, effectue une intégration de ces signaux et produit un potentiel d'action transmis le long de son axone. De même, le perceptron reçoit des entrées pondérées, effectue une combinaison linéaire suivie d'une activation non linéaire pour produire une sortie.

IV.2.1.2. La couche dense

Pour représenter une fonction de $\mathbb{R}^p$ vers $\mathbb{R}^n$, on peut regrouper n perceptrons simples en une couche de neurones partageant la même entrée, appelée couche dense. En regroupant les poids $\omega_{i,j}$ et les biais b_i de chaque perceptron, on obtient une matrice de poids $\mathbf{W}$ et un vecteur de biais $\mathbf{b}$. La sortie d'une couche dense est alors donnée par l'équation (IV.10).

$$\mathbf{y} = \boldsymbol{\sigma}(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{IV.10}$$

La fonction d'activation $\boldsymbol{\sigma}$ peut être une application composante à composante, ou une fonc-

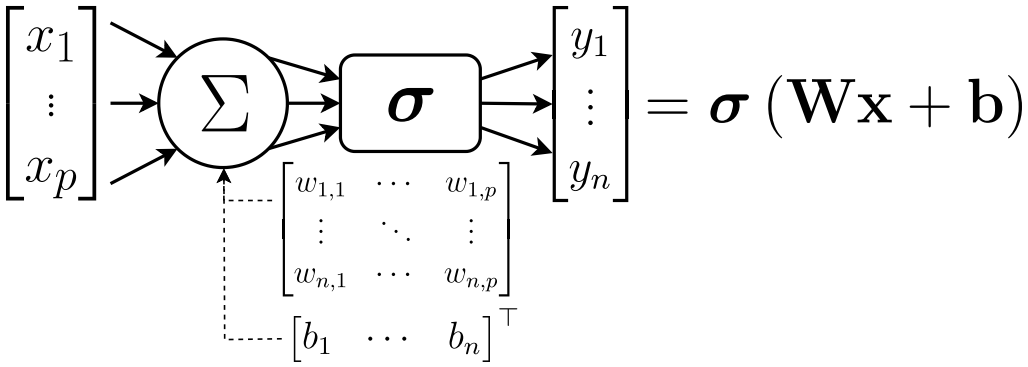


FIGURE IV.4. – Représentation d'une couche simple à p entrées et n sorties.

tion combinant chaque entrée, comme la fonction *softmax* utilisée pour la classification, transformant les sorties en une forme de distribution de probabilité, comme exprimé en équation (IV.11).

$$\mathbf{softmax}(\mathbf{y}|\beta)_i = \frac{\exp\left[\beta y_i\right]}{\sum_{j=1}^{n} \exp\left[\beta y_j\right]} \tag{IV.11}$$

Les perceptrons simples et les couches denses ont cependant des capacités limitées, ne permettant de séparer l'espace d'entrée que par des hyperplans affines, limitant leur application à des problèmes linéairement séparables. Pour des fonctions plus complexes, il est nécessaire de connecter différentes couches denses entre elles, ce qui est fait dans un perceptron multicouche.

IV.2.1.3. Le perceptron multicouche

Le perceptron multicouche est l'architecture la plus connue de réseaux de neurones, capable d'approximer des fonctions non linéaires de $\mathbb{R}^p$ vers $\mathbb{R}^n$. Les couches sont organisées séquentiellement (*feedforward*). Chaque couche dense $\mathbf{f}_k(\cdot|\boldsymbol{\theta}_k)$ contient d_k neurones, prenant une entrée $\mathbf{x}_k \in \mathbb{R}^{d_k}$ et produisant une sortie $\mathbf{y}_k \in \mathbb{R}^{d_{k+1}}$. Les couches sont organisées de manière à injecter la sortie d'une couche dans la suivante, comme exprimé en équation (IV.12).

$$\mathbf{x}_{k+1} = \mathbf{y}_k = \mathbf{f}_k(\mathbf{x}_k|\mathbf{W}_k, \mathbf{b}_k) = \boldsymbol{\sigma}_k(\mathbf{W}_k\mathbf{x}_k + \mathbf{b}_k) \tag{IV.12}$$

Les fonctions d'activation non linéaires $\boldsymbol{\sigma}_k$ sont essentielles pour permettre au réseau d'approximer une grande variété de fonctions. Sans ces non-linéarités, le réseau se réduirait à une simple composition d'applications affines, qui résulterait en une simple application affine. L'architecture d'un perceptron multicouche est illustrée en Figure IV.5.

Grâce à ce modèle permettant la création de réseaux profonds, il est possible de créer des structures complexes de neurones interconnectés, paramétrées par un vecteur $\boldsymbol{\theta}$ constitué de l'ensemble des poids et de biais de chaque neurone. Comme vu précédemment, trouver le bon jeu de paramètres $\boldsymbol{\theta}$ pour approximer une fonction cible nécessite le calcul du gradient de la fonction de perte $\nabla_{\boldsymbol{\theta}}\mathcal{L}$. Cela est permis par l'algorithme de rétropropagation du gradient.

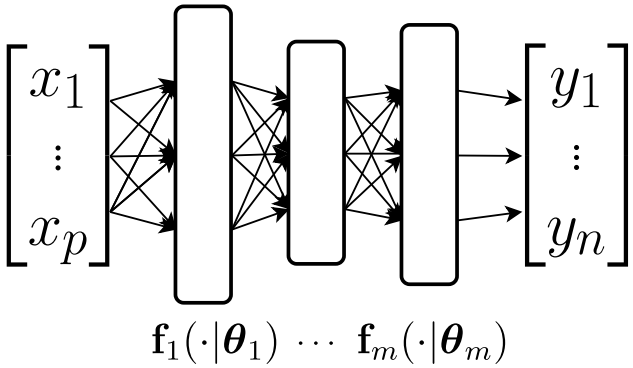


FIGURE IV.5. – Architecture d'un perceptron multicouche à p entrées, n sorties, et m couches denses.

IV.2.2. L'algorithme de rétropropagation du gradient

L'entraînement des réseaux de neurones profonds repose sur l'algorithme de rétropropagation du gradient. Ce processus itératif utilise la règle de dérivation de fonctions composées pour calculer les gradients nécessaires à l'optimisation, en propageant les gradients intermédiaires à travers le réseau. Dans cette section, nous détaillons le fonctionnement général de cet algorithme, en montrant comment il est appliqué aux architectures de réseaux de neurones classiques utilisant des couches denses.

IV.2.2.1. Algorithme général

L'algorithme de rétropropagation du gradient [RHW86] permet d'entraîner les réseaux de neurones profonds en ajustant les paramètres internes des couches de manière à minimiser une fonction de perte $\mathcal{L}$. Cet algorithme fonctionne sur n'importe quel réseau de neurones dont l'architecture forme un graphe de calcul acyclique orienté (DAG). Chaque couche est représentée par une fonction $\mathbf{f}_k(\cdot|\boldsymbol{\theta}_k)$, prenant en entrée un vecteur $\mathbf{x}_k$ et renvoyant un vecteur $\mathbf{y}_k$. Le vecteur $\mathbf{y}_k$ est ensuite transmis en entrée aux couches suivantes, regroupées dans un ensemble que nous notons $\mathcal{S}(k)$. Dans un perceptron multicouche classique, où le résultat d'une couche est envoyé directement à la suivante, on a simplement $\mathcal{S}(k) = \{k+1\}$. Comme illustré dans la Figure IV.6, une connectivité entre couche plus complexe peut être utilisée.

Pour permettre l'apprentissage, l'objectif est de calculer le gradient de la fonction de perte $\mathcal{L}$ par rapport aux paramètres internes de chaque couche : $\nabla_{\boldsymbol{\theta}_k}\mathcal{L}$. Pour cela, chaque couche k doit être capable de calculer la matrice jacobienne (c'est-à-dire la matrice des dérivées partielles) de sa sortie par rapport à son entrée, $\left[\frac{\partial \mathbf{y}_k}{\partial \mathbf{x}_k}\right]$, et par rapport à ses paramètres internes, $\left[\frac{\partial \mathbf{y}_k}{\partial \boldsymbol{\theta}_k}\right]$. L'idée centrale de l'algorithme de rétropropagation du gradient est d'utiliser la dérivation de fonctions composées (règle de la chaîne) pour exprimer $\nabla_{\boldsymbol{\theta}_k}\mathcal{L}$ en fonction des gradients des couches connectées en sortie de la couche k. Premièrement, on peut écrire :

$$\nabla_{\boldsymbol{\theta}_k}\mathcal{L} = \left[\frac{\partial \mathbf{y}_k}{\partial \boldsymbol{\theta}_k}\right]^\top \nabla_{\mathbf{y}_k}\mathcal{L} \tag{IV.13}$$

Ici, $\nabla_{\mathbf{y}_k}\mathcal{L}$ est le gradient de la fonction de perte par rapport à la sortie de la couche k, et $\left[\frac{\partial \mathbf{y}_k}{\partial \boldsymbol{\theta}_k}\right]$

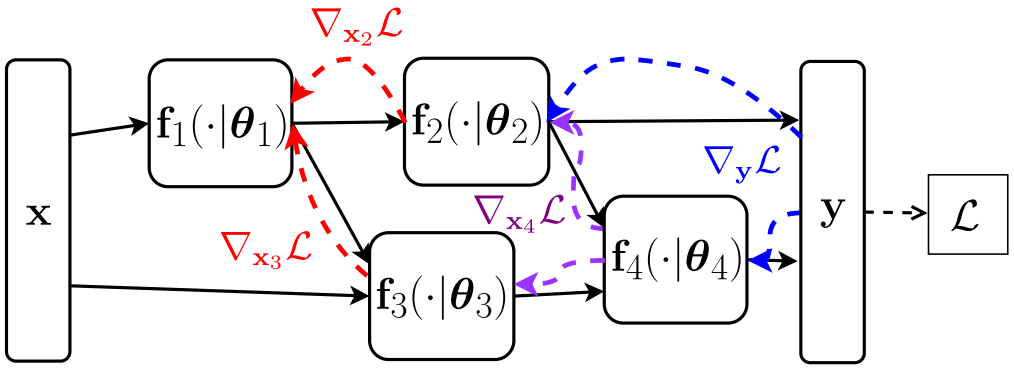


FIGURE IV.6. – Exemple de réseau de neurones profond représenté sous forme de graphe de calcul acyclique orienté (DAG), illustrant le flux des données et la propagation des gradients lors de l'application de l'algorithme de rétropropagation.

est la jacobienne de la sortie de la couche par rapport à ses paramètres internes. De plus, $\nabla_{\mathbf{y}_k}\mathcal{L}$ dépend des couches suivantes, ce qui nécessite un calcul récurrent :

$$\nabla_{\mathbf{y}_k}\mathcal{L} = \sum_{n\in\mathcal{S}(k)} \left[\frac{\partial \mathbf{x}_n}{\partial \mathbf{y}_k}\right]^\top \nabla_{\mathbf{x}_n}\mathcal{L} \tag{IV.14}$$

Ici, $\nabla_{\mathbf{x}_n}\mathcal{L}$ est le gradient de la fonction de perte par rapport à l'entrée de la couche n qui reçoit notamment $\mathbf{y}_k$ comme entrée. $\left[\frac{\partial \mathbf{x}_n}{\partial \mathbf{y}_k}\right]$ est la jacobienne de l'entrée de la couche n par rapport à la sortie de la couche k, et ne dépend que de la connectivité des différentes couches au sein du réseau.

Enfin, pour calculer le gradient de la perte par rapport à l'entrée de la couche k, qui sera propagée en arrière, on utilise :

$$\nabla_{\mathbf{x}_k}\mathcal{L} = \left[\frac{\partial \mathbf{y}_k}{\partial \mathbf{x}_k}\right]^\top \nabla_{\mathbf{y}_k}\mathcal{L} \tag{IV.15}$$

L'algorithme de rétropropagation se décompose ainsi en plusieurs étapes :

1. **Propagation avant** : Chaque entrée $\mathbf{x}$ du *batch* courant est fournie au réseau, et chaque couche $\mathbf{f}_k$ produit une sortie $\mathbf{y}_k$ à partir de son entrée $\mathbf{x}_k$ calculée à partir des sorties des couches précédentes, jusqu'à produire la sortie finale du réseau, $\mathbf{y}$.
2. **Calcul de la perte** : La fonction de perte $\mathcal{L}$ est évaluée sur le *batch* courant en comparant les sorties finales $\mathbf{y}$ du réseau à la valeur attendue. Pour chaque $\mathbf{y}$ produit par le réseau, le gradient de la fonction de perte par rapport à cette prédiction, $\nabla_{\mathbf{y}}\mathcal{L}$, est calculé.
3. **Propagation arrière** : On utilise les équations ci-dessus pour calculer les gradients de la fonction de perte par rapport aux paramètres de chaque couche, en commençant par la sortie du réseau et en remontant jusqu'aux couches d'entrée. Comme illustré en Figure IV.6, chaque couche k utilise les différents $\nabla_{\mathbf{x}_n}\mathcal{L}$ des couches suivantes pour déterminer son propre $\nabla_{\boldsymbol{\theta}_k}\mathcal{L}$ ainsi que le $\nabla_{\mathbf{x}_k}\mathcal{L}$ à propager aux couches en amont dans le réseau.
4. **Ajustement des poids** : Les différents $\boldsymbol{\theta}_k$ sont ajustés par l'optimiseur à partir des gradients de perte calculés $\nabla_{\boldsymbol{\theta}_k}\mathcal{L}$.

Ce processus permet d'ajuster les paramètres internes $\boldsymbol{\theta}_k$ de chaque couche en fonction des gradients calculés. L'algorithme est itératif et continue à ajuster les poids jusqu'à ce que la fonction de perte soit minimisée ou qu'un critère d'arrêt soit atteint.

L'élégance de cet algorithme réside dans le fait qu'il est applicable à n'importe quel réseau représentable comme un DAG, du moment que chaque couche peut calculer les dérivées nécessaires. Cela permet une grande flexibilité dans la conception des architectures de réseaux de neurones.

Des bibliothèques comme TensorFlow [Aba+16] et PyTorch [Pas+19] utilisent cette approche pour créer dynamiquement des graphes de calcul, les différencier automatiquement [Pas+17], et tirer parti de la puissance des GPU pour accélérer l'entraînement des modèles.

IV.2.2.2. La couche dense dans l'algorithme de rétropropagation

Nous avons vu que pour entraîner une couche au sein d'une architecture de réseau de neurone grâce à la rétropropagation du gradient, il suffit qu'elle fournisse, pour une entrée $\mathbf{x}$ donnée et après avoir produit la sortie $\mathbf{y}$, la matrice jacobienne de sa sortie par rapport à son entrée $\left[\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\right]$ ainsi que de sa sortie par rapport à ses paramètres internes $\left[\frac{\partial \mathbf{y}}{\partial \boldsymbol{\theta}}\right]$.

Dans le cas de la couche dense présentée en section IV.2.1.2, les paramètres internes sont constitués de la matrice de poids $\mathbf{W}$ et du vecteur de biais $\mathbf{b}$, les différentes variables sont reliées par la relation $\mathbf{y} = \boldsymbol{\sigma}(\mathbf{W}\mathbf{x} + \mathbf{b})$. En posant :

$$\mathbf{s} = \mathbf{W}\mathbf{x} + \mathbf{b} \quad , \quad \mathbf{W} = \begin{bmatrix} \mathbf{w}_1 & \cdots & \mathbf{w}_n \end{bmatrix}^\top \quad \text{et} \quad \boldsymbol{\theta} = \begin{bmatrix} \mathbf{w}_1^\top & \cdots & \mathbf{w}_n^\top & \mathbf{b}^\top \end{bmatrix}^\top$$

la jacobienne $\left[\frac{\partial \mathbf{y}}{\partial \boldsymbol{\theta}}\right]$ est alors la concaténation des différentes jacobiennes $\left[\frac{\partial \mathbf{y}}{\partial \mathbf{w}_i}\right]$ et de $\left[\frac{\partial \mathbf{y}}{\partial \mathbf{b}}\right]$. En utilisant la dérivation de fonctions composées, faisant apparaitre la dérivée de la fonction d'activation, on obtient ces différentes valeurs en équation (IV.16).

$$\left[\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\right] = \left[\frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{s}}\right] \mathbf{W} \quad ; \quad \left[\frac{\partial \mathbf{y}}{\partial \mathbf{w}_i}\right] = \left[\frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{s}}\right] \begin{bmatrix} \mathbf{0} & \cdots & \overset{(i)}{\mathbf{x}} & \cdots & \mathbf{0} \end{bmatrix}^\top \quad ; \quad \left[\frac{\partial \mathbf{y}}{\partial \mathbf{b}}\right] = \left[\frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{s}}\right] \tag{IV.16}$$

Ainsi, l'algorithme de rétropropagation du gradient peut s'appliquer à un réseau composé de ces couches, permettant de trouver les poids et biais minimisant la fonction de perte à optimiser durant l'apprentissage.

Pour les autres types de couches présentées dans les prochaines sections, la méthode de calcul des dérivées des sorties par rapport aux entrées ou aux paramètres internes ne sera pas détaillée, mais le principe restera le même : à partir du moment où cela est possible, toute couche peut être intégrée dans le graphe de calcul, et l'algorithme de rétropropagation du gradient pourra alors optimiser ses paramètres internes pour minimiser la fonction de perte globale du réseau.

IV.2.3. Couches adaptées au traitement d'image

Cette section présente le principe des réseaux de neurones à convolution, qui seront utilisés pour le traitement des images infrarouge. Depuis les travaux fondateurs [Lec+98] et les premiers modèles profonds qui ont popularisé leur utilisation [KSH12], ces réseaux font référence dans les tâches de traitement d'image [Li+21].

IV.2.3.1. Couches de convolution

Les couches denses des perceptrons multicouches ne sont pas adaptées au traitement d'image en raison du nombre important de paramètres qu'elles nécessitent. Une couche avec n neurones et p entrées comporte $n \times (p+1)$ paramètres internes (poids et biais). Par conséquent, pour traiter des images contenant des centaines de milliers, voire des millions de pixels, le nombre de paramètres deviendrait rapidement trop important, à la fois en terme d'espace mémoire que de temps d'apprentissage.

Les couches de convolution offrent une solution à ce problème en réduisant drastiquement le nombre de paramètres nécessaires tout en permettant des transformations complexes de l'image. L'image de sortie est calculée en balayant sur l'image d'entrée un même filtre de convolution, comme représenté en Figure IV.7. Les poids du filtre de convolution sont alors partagés entre tous les pixels de l'image de sortie. L'opération de convolution d'un tenseur $\mathbf{F}$ par un noyau de convolution $\mathbf{K}$ est décrite en équation (IV.17).

$$(\mathbf{F} * \mathbf{K})_{i,j,c} = \sum_{k_i,k_j} \mathbf{F}_{i+k_i,j+k_j} \mathbf{K}_{c,k_i,k_j} \tag{IV.17}$$

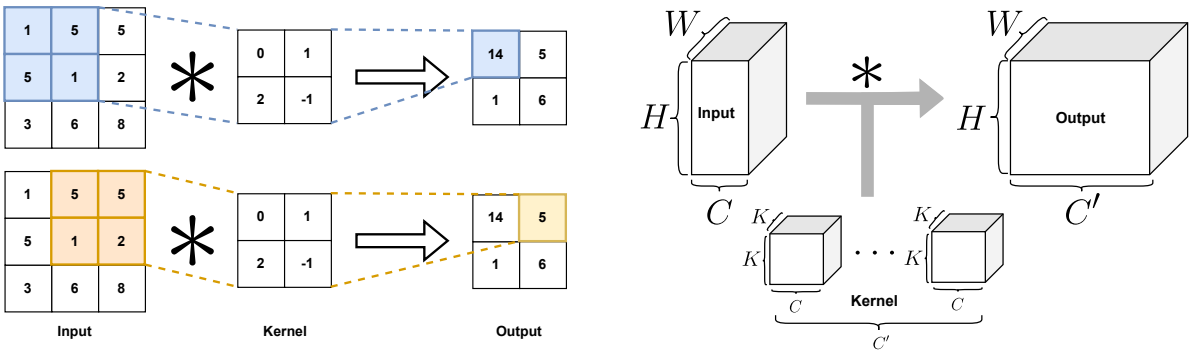


FIGURE IV.7. – Gauche – Exemple de convolution d'une image 3x3 en balayant un filtre 2x2. L'image de sortie résultante est ici plus petite, mais peut être de la même taille que celle d'entrée en ajoutant des zéros sur les bords de l'image d'entrée. **Droite –** Représentation des tenseurs lors de la convolution d'une image à C canaux pour obtenir une image de sortie à C' canaux.

Une couche de couche de convolution peut être appliqué à une image à plusieurs canaux (par exemple, les canaux de couleur rouge, vert et bleu pour une image RGB), représentée par un tenseur de taille (H, W, C), où H et W représentent respectivement la hauteur et la largeur de

l'image, et C le nombre de canaux. Une couche de convolution avec C' canaux de sortie produit un tenseur de sortie de taille (H, W, C'). Pour ce faire, elle applique C' filtres de convolution distincts sur l'entrée. Chaque filtre est de taille (K, K, C), où K est la dimension du noyau de convolution (typiquement 3 ou 5). Ainsi, en prenant aussi un biais additionné au produit de convolution, le nombre total de paramètres internes de la couche est de $K^2C(C'+1)$. Ce nombre réduit de poids, indépendant de la largeur et de la hauteur de l'image, permet ainsi de traiter des images de grandes dimensions avec un nombre raisonnable de paramètres à apprendre.

IV.2.3.2. Downsampling, upsampling

Les filtres de convolution agissent de manière locale, typiquement en utilisant des fenêtres de taille 3x3 ou 5x5 autour de chaque pixel. Cette approche permet de capturer des caractéristiques locales de l'image. Cependant, pour agréger l'information de l'ensemble de l'image ou pour étendre une même information à toute l'image, des opérations supplémentaires telles que le *downsampling* et l'*upsampling* sont nécessaires.

Le *downsampling* consiste à réduire la résolution de l'image pour obtenir une sortie, de plus faible dimensionnalité, qui agrège l'information contenue dans l'entrée. Cette opération peut être faite par une couche de convolution qui balaye son filtre avec un pas (*stride*) plus grand que 1, ou par des opérations de *pooling* : le *max-pooling* partitionne l'image en fenêtre, par exemples de taille 2x2, et sélectionne la plus grande valeur parmi chaque région. Il permet ainsi de sélectionner les caractéristiques les plus importantes de l'image.

À l'inverse, l'*upsampling* vise à reconstituer une image de plus grande dimension, ce qui permet de propager une information globale. Cette opération peut être faite en dupliquant les pixels ou en réalisant des interpolations des valeurs.

IV.2.3.3. Skip connection et couche résiduelle

L'un des défis majeurs dans l'apprentissage des réseaux de neurones profonds est la difficulté de propager efficacement le gradient à travers de nombreuses couches. Ce phénomène limite la profondeur maximale des réseaux que l'on peut entraîner de manière efficace.

Pour surmonter ce problème, les *skip connections* ont été introduites, notamment dans l'architecture U-Net [RFB15], ainsi que dans les réseaux ResNets [He+16], qui utilisent des couches résiduelles.

Une *skip connection* permet de créer un chemin alternatif pour le flux d'informations et les gradients, en sautant une ou plusieurs couches intermédiaires. Dans une couche résiduelle, l'entrée de la couche est additionnée à la sortie des couches intermédiaires avant d'être passée à la couche suivante.

Les couches résiduelles ont prouvé leur efficacité dans de nombreuses tâches de vision par ordinateur, notamment la classification d'images, la détection d'objets et la segmentation. Elles sont devenues un composant standard dans les architectures modernes de réseaux de neurones profonds.

IV.2.4. Architectures de réseaux pour le traitement d'image

Le traitement d'image par des réseaux de neurones requiert des architectures spécifiques. Cette section explore deux types d'architectures couramment utilisées dans ce domaine : l'auto-encodeur et le U-Net.

IV.2.4.1. L'architecture de l'auto-encodeur

Un auto-encodeur est une architecture de réseau de neurones qui vise à encoder une image en une représentation de plus faible dimension puis à la décoder pour reconstruire l'image originale. L'auto-encodeur est ainsi constitué de deux parties branchées en série : l'encodeur et le décodeur.

L'encodeur applique une succession de filtres de convolution avec des opérations de *downsampling*, ce qui réduit progressivement la taille de l'image tout en augmentant le nombre de canaux. Cela permet de compresser l'information tout en conservant les caractéristiques importantes.

Le décodeur, quant à lui, utilise des opérations de convolution et d'*upsampling* pour augmenter la taille de l'image et réduire le nombre de canaux, recréant ainsi l'image d'origine.

L'objectif est de réduire la dimension totale du tenseur au centre du réseau par rapport à l'entrée ou à la sortie, réalisant ainsi une compression de l'information. L'espace des images encodées de plus petite dimension est appelé espace latent ou espace sémantique. Par exemple, en apprenant la fonction identité de manière non supervisée, le réseau extrait le contenu sémantique des images. Cette même architecture peut être utilisée de manière supervisée dans des tâches de traitement "image vers image".

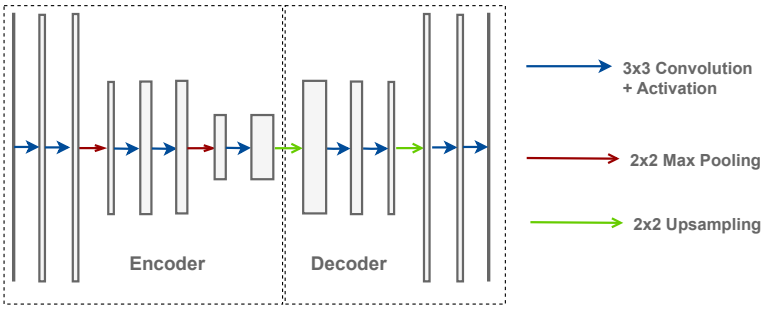


FIGURE IV.8. – Architecture d'un auto-encodeur. L'encodeur comprend des couches de convolution suivies d'activations et de sous-échantillonnage (*max pooling*), réduisant la dimension spatiale et augmentant le nombre de canaux. Le décodeur utilise des couches de convolution et de suréchantillonnage (*upsampling*) pour restaurer la dimension d'origine, recréant ainsi l'image. L'objectif est de compresser l'information en un espace latent de plus faible dimension, capturant les caractéristiques sémantiques essentielles.

Cependant, les auto-encodeurs présentent certaines limitations. La réduction progressive de la taille des images dans l'encodeur et leur reconstruction dans le décodeur peuvent entraîner une perte d'informations fines et de détails importants, particulièrement problématique dans des tâches où la précision des détails est cruciale. Pour régler ce problème, l'introduction de *skip connection* est nécessaire.

IV.2.4.2. L'architecture du U-Net

Le U-Net [RFB15] est une architecture de réseau de neurones conçue initialement pour les tâches de segmentation d'image. À la manière des architectures de type auto-encodeur, il est constitué d'une succession de convolutions avec des opérations de *downsampling* et d'*upsampling*. Sa particularité réside dans l'addition de *skip connections* transférant les informations de haute résolution des premières couches aux dernières. Les informations traversent ainsi le U-Net via deux chemins : une voie qui extrait les caractéristiques sémantiques à l'échelle de l'image entière, et une voie qui préserve les informations locales. Une représentation de l'architecture est présente en Figure IV.9.

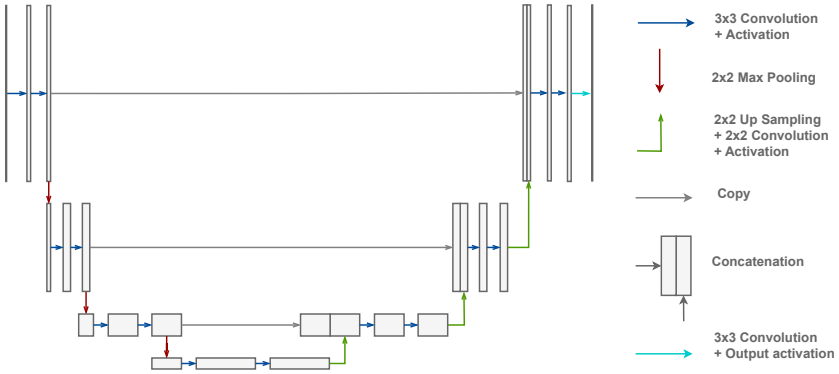


FIGURE IV.9. – Architecture d'un U-Net. L'information traverse le réseau via un chemin constitué de *downsampling* et d'*upsampling* qui extraient les caractéristiques globales de l'image et les propagent à chaque pixel, et via des *skip connections* qui préservent les détails et informations locales de l'image.

Les couches de *downsampling* réduisent progressivement la taille des images tout en augmentant le nombre de filtres. Cette partie du réseau extrait les caractéristiques importantes de l'image en combinant des couches de convolution, de normalisation et de *pooling*. En réduisant la résolution spatiale, le réseau apprend des représentations de plus en plus abstraites des données d'entrée. Les couches d'*upsampling*, quant à elles, reconstruisent l'image à sa taille d'origine, en propageant une l'information globale extraite.

Le U-Net utilise des connexions de type *skip connection* qui relient les couches de *downsampling* aux couches correspondantes d'*upsampling*. Ces connexions permettent de transférer des informations de haute résolution de la partie descendante à la partie ascendante, préservant ainsi les détails tout au long du réseau. Cette préservation des détails est cruciale pour les tâches nécessitant une reconstruction précise, comme la segmentation d'image, ou dans notre cas, la génération de cartes de températures dont les contours sont similaires à l'image d'origine.

L'architecture du U-Net est toujours massivement utilisée, non seulement pour la segmentation d'images, mais aussi pour des tâches de génération d'image dans des architectures telles que les GANs [Iso+17] et les modèles de diffusion [Sah+22]. Ces modèles bénéficient de la capacité du U-Net à maintenir des détails fins tout en apprenant des représentations complexes et de haut niveau. Pour ces raisons, cette architecture sera utilisée dans notre application.

IV.3. Estimation de température par U-Net sur des données simulées

Cette section présente l'entraînement d'un réseau de neurone à convolution de type U-Net sur un jeu de données simulées reproduisant le tokamak WEST en opération, dont la modélisation a été décrite en détail en section III.8. Contrairement à l'expérience de la section II.2, les modèles de réflexions sur les surfaces sont plus complexes, en intégrant un part de réflexion diffuse et spéculaire.

IV.3.1. Modélisation de la scène thermique

Les scènes thermiques synthétiques sont générées à l'aide d'un modèle 3D détaillé du tokamak WEST, composé de 38 composants principaux. Pour chaque scène, les températures et les propriétés optiques de surface sont choisies aléatoirement dans des plages cohérentes par rapport aux expériences du WEST. La modélisation de la scène est résumée ici, pour plus de détails, se référer à la section III.8.

Pour chacune des *strike line* sur le divertor bas, la distribution de température est calculée à l'aide d'une fonction paramétrée qui adapte la fonction Eich en l'utilisant comme motif de température, couplée à une fonction sinus pour inclure l'effet de *ripple* dans la direction toroïdale (voir section III.8.4). Les paramètres libres T_{off}, T_0, λ_q et S sont échantillonnés aléatoirement dans des plages cohérentes.

Les températures des protections de l'antenne visible depuis la caméra grand angle sont obtenues en projetant les images expérimentales 2D (en température apparente de corps noir) sur le maillage 3D, afin de reproduire approximativement les motifs de température en surface. Pour introduire des variations dans les motifs, une moyenne pondérée de plusieurs distributions (prises à partir de différentes images expérimentales) est sélectionnée aléatoirement pour chaque image générée, puis mise à l'échelle de manière aléatoire.

À l'exception du divertor bas et des protections d'antenne, les autres composants se voient attribuer uniformément des températures dans une plage étendue de valeurs (de 50 à 200 °C pour les composants froids et de 200 à 700 °C pour les composants chauds).

Un mélange de BRDF Phong et purement spéculaire (c'est-à-dire semblable à un miroir) est utilisé, afin de simuler des comportements de réflexion intermédiaires, allant de diffus pur à spéculaire pur. Le modèle est défini avec quatre paramètres (k_d, k_s, k_m et n) qui varient selon les triangles dans le maillage. Les paramètres k_d, k_s et k_m contrôlent respectivement la proportion de lumière qui est diffusée, subit une réflexion spéculaire gaussienne ou une réflexion de type miroir. Le quatrième paramètre, n, détermine la plage de directions possibles pour les réflexions spéculaires gaussiennes autour de la direction de réflexion de miroir pur. Les propriétés optiques de surface sont uniformes par composant, avec une réflectivité totale variant entre 0,5 et 0,95 et avec des valeurs aléatoires pour la proportion de réflexions diffuses, spéculaires ou similaires à un miroir.

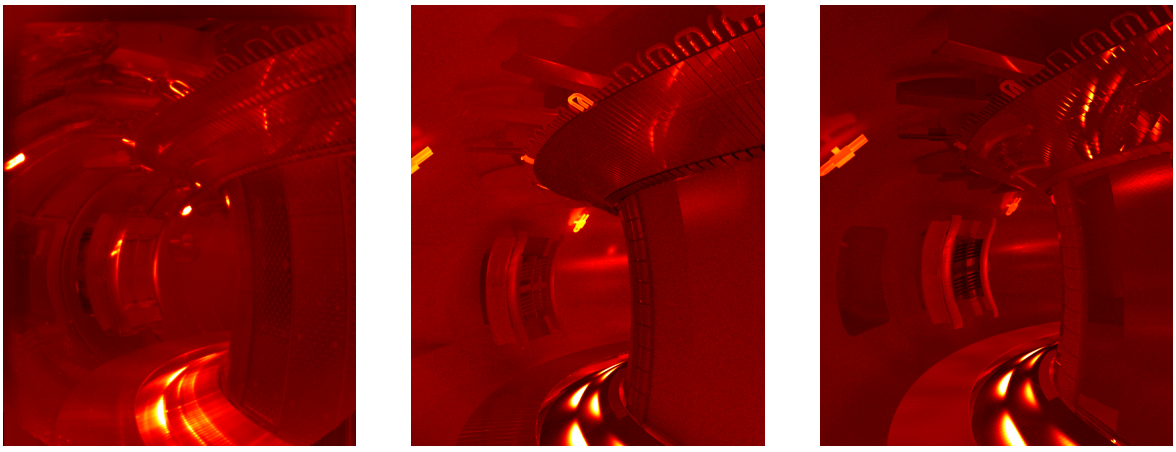
Pour tenir compte des mouvements de la caméra et des vibrations du dispositif pendant le fonctionnement du plasma, la caméra *pinhole*, qui reproduit la visée grand angle de WEST, subit des mouvements allant jusqu'à 2 cm dans toutes les directions, une rotation allant jusqu'à 2

degrés autour de chaque axe, et des changements de champ de vision allant jusqu'à 5 degrés entre chaque image.

De plus, pour inclure la non-uniformité des parois, du bruit de Perlin a été utilisé pour introduire des variations significatives à petite et grande échelle (voir section III.8.6). Pour chaque composant, une somme de trois bruits de Perlin avec des amplitudes et tailles de variation aléatoires est appliquée, ce qui peut entraîner des variations aléatoires continues à l'intérieur de chaque composant principal (jusqu'à 40 % de variation intra-composant pour la température, exprimée en Celsius, et 10 % pour les paramètres BRDF). De plus, des augmentations ponctuelles de température sont également modélisées à l'aide de formes gaussiennes, ce qui permet d'augmenter localement la température de surface de 20 % à 300 % à des endroits aléatoires, afin de reproduire des évènements thermiques ponctuels inattendus.

Au total, chaque scène est décrite avec environ 200 paramètres qui définissent les propriétés déterministes.

La Figure IV.10(b–c) montre deux images synthétiques de la vue grand angle de WEST, générées avec cette méthode. Les simulations sont capables de reproduire les motifs de réflexion observés dans les images expérimentales de la Figure IV.10(a). En variant les paramètres d'entrée du modèle (paramètres de la caméra, propriétés optiques des matériaux et distribution de température), on peut couvrir une large gamme de scénarios possibles pour la phase d'apprentissage du réseau.



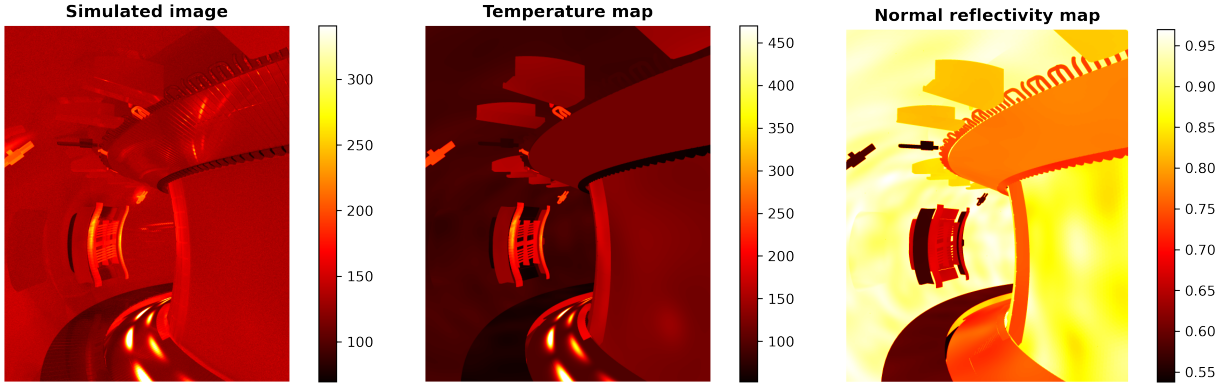
(a) (b) (c)

FIGURE IV.10. – (a) Image expérimentale capturée par la caméra grand angle de WEST (choc 55567). **(b–c)** Exemples d'images synthétiques générées par lancer de rayons en tenant compte de différents champs de vision de la caméra, des propriétés optiques de surface et des distributions de température. En variant les paramètres d'entrée du modèle direct, nous obtenons une certaine similarité avec l'image expérimentale (a), ce qui nous donne confiance dans le fait de traiter, dans les étapes ultérieures, les images expérimentales à partir d'un réseau entraîné à partir d'images synthétiques.

IV.3.2. Génération des données

Un ensemble de données de 160 000 images infrarouges simulées est généré en faisant varier les paramètres d'entrée (paramètres de la caméra, températures de surface et propriétés optiques) dans la plage décrite dans la section précédente. Il est à noter que pour une distribution donnée de BRDF, le rendu devient linéaire par rapport à la distribution de températures (plus précisément, par rapport aux émissions corps noir). Ainsi, plusieurs images avec différentes distributions de température peuvent être générées en parallèle pour une même distribution de BRDF, ce qui accélère le temps de calcul. De cette manière, seuls 16 000 profils BRDF différents ont été générés avec 10 distributions de température différentes pour chacun.

L'ensemble de données comprend des paires d'images : l'image synthétique, telle que mesurée par le système d'imagerie infrarouge, et la carte de température réelle. L'image synthétique est exprimée en température apparente, obtenue en inversant la loi de Planck à partir des cartes de radiance calculées par le lancer de rayons. Les cartes de température vraie sont obtenues par une projection 3D vers 2D du champ de température utilisé en entrée du lancer de rayons. La Figure IV.11(a) montre un exemple d'image générée, la Figure IV.11(b), la carte de température de sortie attendue stockée dans l'ensemble de données. En comparant l'image simulée (a) et la carte de température d'entrée (b), on remarque l'apparition des motifs de réflexion, comme sur le divertor haut, qui est une réflexion du divertor bas.



(a) (b) (c)

FIGURE IV.11. – Exemple d'une scène générée dans le jeu de données. **(a)** Image simulée (radiance convertie en température apparente corps noir) obtenue par lancer de rayons et tenant compte des propriétés de réflexion complexes. **(b)** Carte de température réelle des surfaces observées obtenue par une projection 3D vers 2D de la distribution de température utilisée. Des augmentations de température localisées ou des variations continues des composants internes peuvent être observées. **(c)** Réflectivité normale des surfaces fournies en entrée de la simulation ($\rho_\perp = k_d + k_s + k_m$).

Un réseau de neurones est ainsi entraîné à prédire la carte de température à partir de l'image simulée sans être informé des propriétés de réflectivité utilisées dans la simulation, telle que la carte montrée en Figure IV.11(c).

IV.3.3. Apprentissage

Un réseau de neurones est entraîné, à partir des données simulées, à inférer la carte de température des surfaces visibles sur la base d'une image infrarouge capturée. Ses performances ont été évaluées sur des données synthétiques.

Un réseau basé sur l'architecture U-Net (voir section IV.2.4.2) est utilisé pour prédire des cartes de température à partir d'images infrarouges complexes de scènes thermiques de tokamak. L'architecture du réseau est montrée en Figure IV.12. Ce type de réseau convient parfaitement à notre tâche, car l'image de sortie (carte de température) contient des détails similaires et une localisation d'objet similaire à l'image d'entrée (carte de température apparente). Le nombre d'opérations *downsampling* est augmenté à sept pour propager les informations globales à chaque pixel de sortie, ce qui est crucial lorsque l'on traite la propriété non locale des réflexions. Une fonction d'activation LeakyReLU (avec pente 0.3) est utilisée, et aucune fonction d'activation n'est appliquée à la couche de sortie.

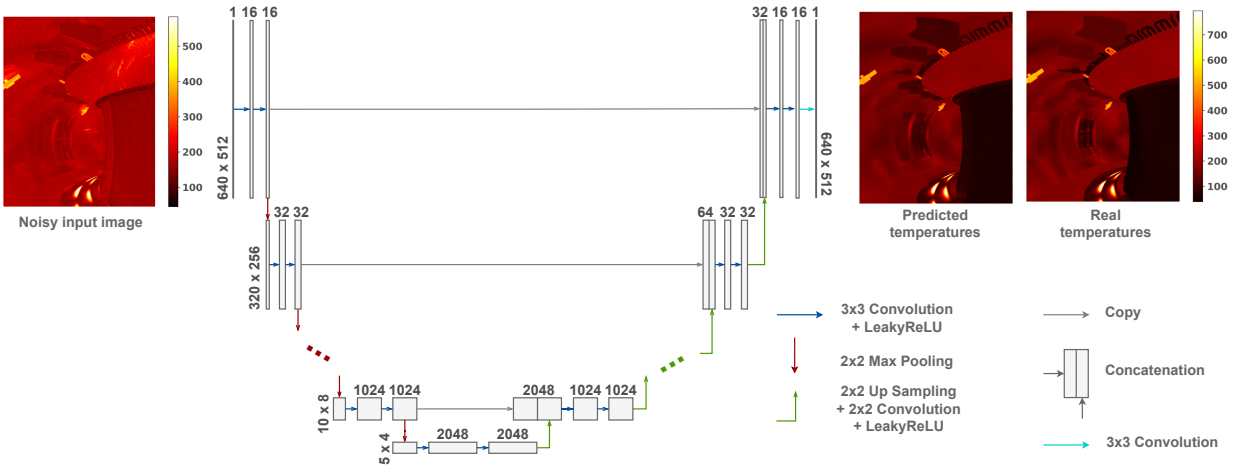


FIGURE IV.12. – Architecture U-Net, la taille de l'image est adaptée à celle de la caméra infrarouge (512 pixels de large sur 640 pixels de haut), ainsi que le nombre de *downsampling*. Le réseau est entraîné pour renvoyer la carte de température des surfaces observées à partir de l'image infrarouge. Un exemple de prédiction de la température de surface à partir d'une image de test bruitée est présenté.

L'ensemble de données est divisé en trois sous-ensembles : 50 000 images pour l'ensemble d'entraînement, 10 000 pour la validation et 100 000 pour l'ensemble de test. La séparation de l'ensemble de données est conçue pour garantir que toutes les 10 images générées avec un même profil de BRDF appartiennent au même ensemble. Cela garantit que les ensembles de validation et de test ne contiennent pas d'images générées avec les mêmes profils BRDF que l'ensemble d'entraînement.

Le réseau de neurones est conçu pour prendre en entrée une image à un canal, représentant une carte de température apparente de corps noir (l'image simulée complexe contenant des réflexions), et pour produire en sortie une image à un canal qui est la carte de température des cibles dans le champ de vision de la caméra. Les images sont représentées en utilisant des

nombres en virgule flottante à simple précision (c'est-à-dire 32 bits).

Pendant le processus d'entraînement, les images sont normalisées dans la plage de $[0;1]$ en utilisant les valeurs minimales et maximales de l'ensemble d'entraînement. L'optimiseur Adam (avec un taux d'apprentissage de 10^{-4}, $\beta_1 = 0.9$ et $\beta_2 = 0.999$) est utilisé pour minimiser l'erreur quadratique moyenne entre la carte de température prédite et la carte de température réelle correspondante. L'entraînement est exécuté pendant 100 *epochs* en utilisant une taille de *batch* de 32. Le modèle avec l'erreur la plus faible pour l'ensemble de validation est sélectionné pour les tests.

Afin d'améliorer les performances de généralisation et de prendre en compte le bruit parasite qui peut survenir sur les données expérimentales, un du bruit gaussien est ajouté aux images d'entrée du réseau pendant le processus d'entraînement, avec une amplitude choisie uniformément entre 0 et 20 °C.

IV.3.4. Pertinence de l'ensemble de données

Comme le même processus de génération de données est utilisé pour les ensembles d'entraînement, de validation et de test, déterminer le nombre approprié d'images à inclure dans l'ensemble d'entraînement peut être une tâche difficile. Si trop peu d'images sont générées, le modèle peut ne pas se généraliser correctement. À l'inverse, si trop d'images sont générées, l'ensemble d'entraînement peut couvrir tous les cas possibles, entraînant un surapprentissage tout en obtenant de bonnes erreurs de validation et de test. Pour éviter de saturer l'espace des images possibles dans l'ensemble d'entraînement, ce qui impliquerait que chaque image des ensembles de validation et de test a une contrepartie très similaire dans l'ensemble d'entraînement, on effectue deux vérifications.

Premièrement, on évalue la complexité des images simulées en utilisant la dimension intrinsèque de l'ensemble de données. La dimension intrinsèque représente la dimension minimale d'une variété (*manifold*, c'est-à-dire un sous-ensemble localement euclidien) qui englobe l'ensemble de données sans perte d'information. Une dimension faible implique que les images infrarouges peuvent être décrites facilement avec quelques paramètres. En revanche, une dimension élevée indique qu'un plus grand nombre d'images est nécessaire dans l'ensemble de données pour que le réseau fonctionne correctement. On évalue cette dimension en utilisant l'estimation par maximum de vraisemblance [LB04] et on la compare à des ensembles de données de référence connus. La dimension intrinsèque de l'ensemble de données synthétiques généré de 50 000 images (avant d'être altérées par le bruit pour l'entraînement) est évaluée à environ 22. Cette valeur est similaire aux dimensions intrinsèques d'ensembles de données de référence tels que CelebA et CIFAR10, qui contiennent respectivement 200 000 et 60 000 images, comme le montre [Pop+21]. Étant donné que le nombre d'éléments dans notre ensemble de données est similaire à ceux des ensembles de données qui ne saturent clairement pas l'espace des éléments possibles (c'est-à-dire, des personnes et des objets de 10 catégories), cela suggère que notre ensemble de données est suffisamment épars. La dimension de 22 peut être mise en perspective par rapport à l'utilisation de plus de 200 paramètres pour la génération des images. Comme ces paramètres n'affectent pas tous de manière égale l'image finale (par exemple, la température d'un petit composant de l'autre côté du tokamak), il est attendu qu'un ensemble d'images de dimension inférieure en résulte. De plus, cette observation suggère également qu'une image

seule ne contient pas suffisamment d'informations pour reconstruire entièrement les 200 paramètres utilisés pour sa génération. Heureusement, notre approche vise à prédire uniquement les températures des surfaces directement visibles, qui sont donc attribuables à des paramètres influents.

Deuxièmement, on utilise une intelligence artificielle factice pour estimer dans quelle mesure l'espace des images de température de surface possibles est couvert par l'ensemble de données d'entraînement. Cette approche implique la création d'un "mémorisateur" qui prédit la température en renvoyant la température associée à l'image la plus proche dans l'ensemble d'entraînement, en utilisant la distance euclidienne entre deux images. Si l'ensemble d'entraînement est suffisant pour couvrir l'espace de toutes les images possibles, nous nous attendons à ce que pour toute image de test donnée, il existe une image d'entraînement proche permettant au mémorisateur de prédire une température proche de la valeur réelle. En créant un tel algorithme et en le testant sur un ensemble d'images de test, on trouve une erreur relative moyenne de prédiction absolue de 34,5 % pour les basses températures de surface inférieures à 400 °C, et de 39,7 % pour les températures élevées supérieures à 400 °C. Cela suggère également que l'ensemble d'entraînement ne couvre pas toutes les situations possibles. Par conséquent, on s'attend à ce qu'un réseau de neurones doté de capacités de généralisation fasse bien mieux que ces erreurs.

IV.3.5. Résultats sur images simulées

Cette section présente les performances d'estimation de la température par le U-Net sur des images simulées. Trois conditions de test différentes ont été utilisées : un ensemble de test composé d'éléments tirés de la même distribution que l'ensemble d'apprentissage, le même ensemble de test avec un bruit parasite d'une amplitude de 20 °C ajouté aux entrées, et un ensemble de test avec une "distribution de température aberrante" qui est généré en utilisant des motifs de chaleur du divertor inattendus.

Pour chacun de ces ensembles de test, deux métriques de performance sont utilisées. La première calcule la moyenne (notée μ) et le $95^{\text{ème}}$ percentile (c'est-à-dire la valeur en dessous de laquelle 95 % des erreurs se situent, notée p_{95}) des erreurs relatives absolues : $|\Delta T|/T$ avec T exprimé en degrés Celsius. La deuxième métrique est la proportion (pixel par pixel) des estimations conformes aux spécifications ITER : pour des surfaces avec des températures inférieures à 400 °C, l'erreur relative absolue ne doit pas dépasser 20 %, et pour des surfaces avec des températures supérieures à 400 °C, elle ne doit pas dépasser 10 %.

Les résultats pour les différents ensembles de test et métriques sont résumés dans le Tableau IV.1 et le Tableau IV.2, et la Figure IV.13(a) montre l'erreur en fonction des températures de surface réelles pour ces trois cas.

IV.3.5.1. Images de test standards

La performance du réseau est d'abord évaluée sur un ensemble de test de 100 000 images simulées. Les images sont générées avec le même modèle et le même processus aléatoire que celles d'entraînement.

Comme attendu, les surfaces à basse température sont plus sensibles aux flux parasites (c'est-à-dire les réflexions) et l'estimation de la température de surface est plus délicate, comme le

Erreur pour	Image standard	Image bruitée	Motif inattendu
$T < 400\,^{\circ}\mathrm{C}$	$\mu = 6,5\%$; $p_{95} = 23,1\%$	$\mu = 9,3\%$; $p_{95} = 32\%$	$\mu = 22\%$; $p_{95} = 89.1\%$
$T \geq 400\,^{\circ}\mathrm{C}$	$\mu = 3,5\%$; $p_{95} = 11,4\%$	$\mu = 4,4\%$; $p_{95} = 13,8\%$	$\mu = 13\%$; $p_{95} = 32,3\%$

TABLEAU IV.1. – Moyenne et 95ème percentile des erreurs relatives absolues sur les températures prédites. Les erreurs sont montrées pour des températures réelles inférieures à 400 °C (où ITER exige une erreur inférieure à 20 %) et supérieures à 400 °C (exigence d'une erreur inférieure à 10 %), et pour les trois conditions de test différentes.

Exigences sur l'erreur	Image standard	Image bruitée	Motif inattendu
Moins de 20 % pour T < 400 °C	93,5 %	88,6 %	69,7 %
Moins de 10 % pour T ≥ 400 °C	93,6 %	90,2 %	45,2 %

TABLEAU IV.2. – Proportion des prédictions du réseau respectant les spécifications d'ITER (erreur relative absolue inférieure à 20% pour les basses températures et inférieure à 10% pour les hautes températures) sur trois ensembles de tests différents.

montre la courbe d'erreur rouge dans la Fig. IV.13(a). Pour les surfaces froides (< 400°C), l'erreur moyenne de prédiction est de 6,5 %, avec un total de 93,5 % des prédictions sous une erreur de 20 %. Pour les surfaces chaudes (≥ 400°C), l'erreur moyenne est de 3,5 %, pour un total de 93,6 % d'erreurs de prédiction sous le seuil de 10 %.

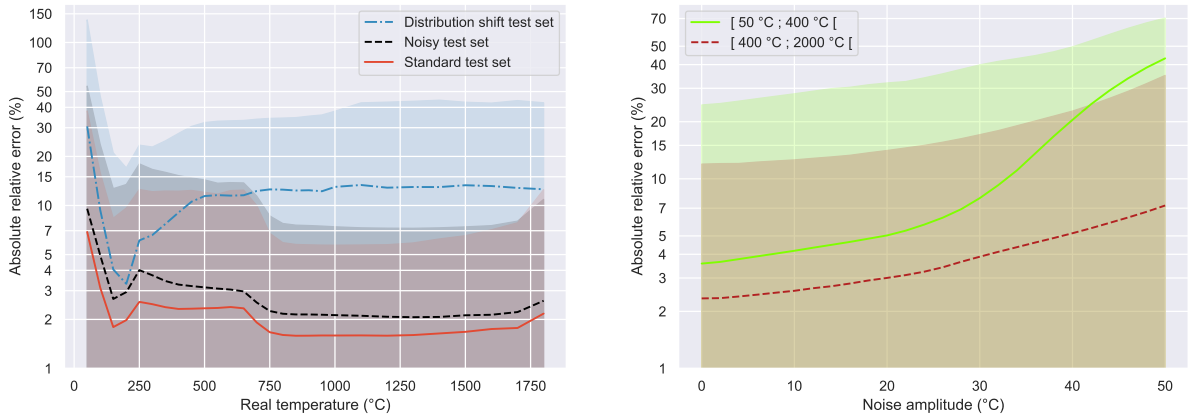
Ces résultats encourageants montrent la grande capacité du réseau à estimer la température à partir d'images simulées proches de celles vues pendant l'entraînement. Étant donné qu'il existe encore un écart entre l'ensemble d'entraînement et les images expérimentales réelles, la performance de généralisation du réseau sur des images légèrement différentes doit également être évaluée.

Pas à pas, nous modifions la façon dont nous dégradons les images de l'ensemble de test, dans le but de nous rapprocher des conditions expérimentales (avec des paramètres non contrôlés). La performance du U-Net est ensuite évaluée sur l'ensemble de test avec des images bruitées, ainsi que sur des images simulées avec une distribution de température différente.

IV.3.5.2. Images détériorées

La robustesse du réseau au bruit parasite est d'abord évaluée. La Fig. IV.13(b) montre l'erreur relative absolue moyenne prédite par le U-Net en fonction de l'amplitude du bruit dans les images de l'ensemble de données. Comme prévu, l'erreur sur la température prédite à basse température est plus élevée, de 8 % à 40 % pour une amplitude de bruit maximale de 50 °C. L'erreur à haute température reste néanmoins assez faible même avec des images bruitées (inférieure à 10 %).

La courbe noire dans la Figure IV.13(a) montre l'erreur sur un ensemble de test constitué en détériorant l'ensemble de test standard avec un bruit gaussien d'une amplitude de 20 °C. Les erreurs restent proches de celles obtenues avec l'ensemble de test standard utilisant des images



(a) (b)

FIGURE IV.13. – Évaluation des erreurs relatives absolues pour les estimations de température dans différentes conditions de test. Les lignes pleines ou en pointillés représentent l'erreur moyenne (moyennée sur toutes les images de test) tandis que les régions colorées représentent le 95ème percentile. (a) Erreur sur les trois ensembles de test différents en fonction de la température réelle de surface. (b) Erreur en fonction de la quantité de bruit ajoutée aux images d'entrée de l'ensemble de test standard pour les températures de surface réelles basses et élevées.

non détériorées, montrant que le réseau est résistant à de petites perturbations.

IV.3.5.3. Motif de température inattendu

Un autre élément à tester est la robustesse du U-Net à des images résultant d'une distribution de température différente de celle utilisée dans l'ensemble d'entraînement. Un deuxième ensemble de test de 20 000 images est alors généré en utilisant une nouvelle fonction paramétrée pour la distribution de température sur le divertor inférieur, en utilisant un simple motif sinusoïdal, afin de se placer dans un scénario de pire cas où il y a une divergence exagérée entre les données d'entraînement et de test. Comme illustré dans la Figure IV.14, la distribution de température testée est inattendue, car le motif du divertor diffère grandement de ceux standard, et affecte également la quantité de lumière réfléchie par le reste de la scène. Le réseau est toujours capable de fournir une réponse cohérente reproduisant le motif, même si les températures estimées tendent à être plus élevées que celles attendues.

La courbe bleue dans la Figure IV.13(a) montre l'erreur dans la température prédite en fonction de la température réelle, obtenue dans ces conditions. Sans surprise, l'erreur est plus élevée que sur les images testées avec une distribution de température similaire à celles de l'ensemble de données d'entraînement, mais reste raisonnable avec une erreur moyenne plus élevée, entre 10 % et 50 % (avec une valeur moyenne de 22 % pour les basses températures et de 13 % pour les hautes températures). Considérant les spécifications ITER, 69,7 % des estimations pour les températures faibles satisfont au seuil d'erreur de 20 %, tandis que 45,2 % satisfont au seuil de 10 % pour les hautes températures.

Si on observe sans surprise une dégradation des performances du réseau dans les cas de distributions de température aberrantes, ces résultats sont encourageants quant à la capacité de généralisation des réseaux de neurones pour traiter des images qui différents du jeu d'entraînement. Pour améliorer les résultats sur ces cas, il faut néanmoins construire un jeu de donnée d'entraînement le plus exhaustif et représentatif des scénarios qui seront rencontrés pendant l'exploitation du réseau.

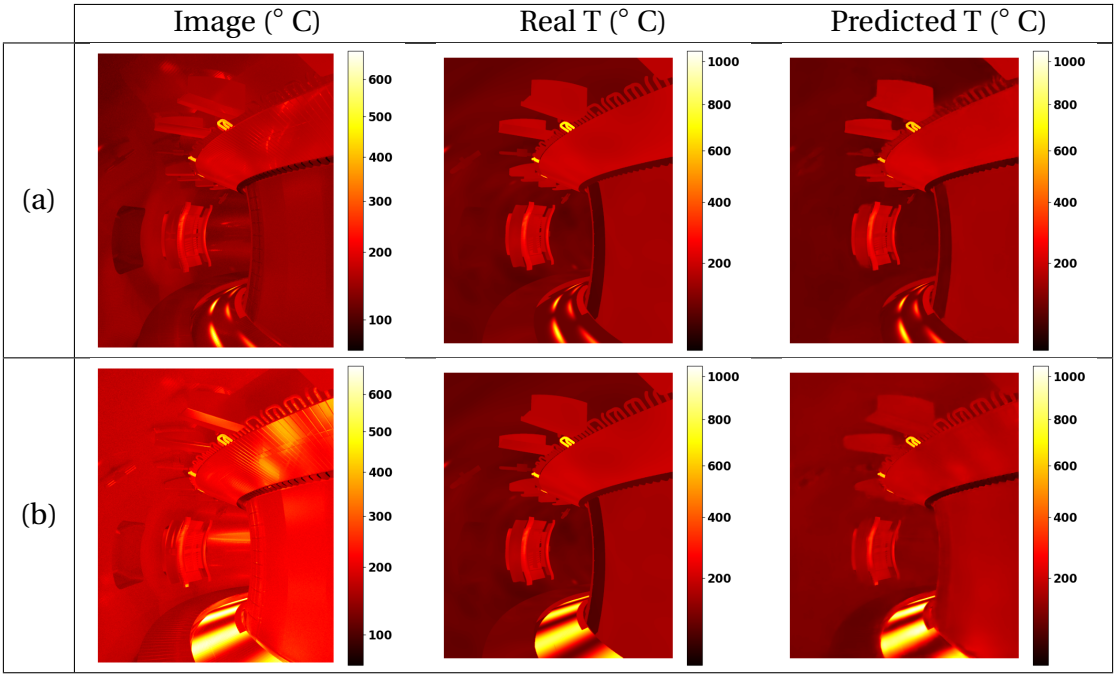


FIGURE IV.14. – Exemples de prédictions de température lors de la variation de la température du divertor inférieur d'un motif similaire à ceux observés lors de l'entraînement à un motif totalement inconnu. **(a)** Motif standard similaire à l'ensemble d'entraînement et de test. **(b)** Motif similaire à l'ensemble de test de décalage de distribution. Le U-Net est capable de reproduire le motif thermique mais tend à prédire des températures plus élevées à la fois pour le divertor et les surfaces froides qui reflètent davantage la lumière du divertor.

IV.3.6. Application à une image expérimentale

Le U-Net entraîné est enfin appliqué aux images expérimentales du système de visualisation angulaire de WEST, comme montré en Figure IV.15. Le réseau prédit la température des surfaces à partir de l'image en température apparente mesurée par la caméra sans connaître l'émissivité.

La température apparente, fournie par la caméra, suppose une émissivité de 1, ce qui conduit à sous-estimer la température des cibles les plus chaudes, tandis que le réseau prédit une température plus élevée, ce qui est en bon accord avec les cibles métalliques de faible émissivité. De plus, on peut observer qualitativement une forte diminution des motifs de réflexions.

À ce stade, sans vérité terrain, il est difficile de quantifier la qualité de la prédiction du U-Net à partir des images expérimentales. La méthode pourrait néanmoins être testée dans un

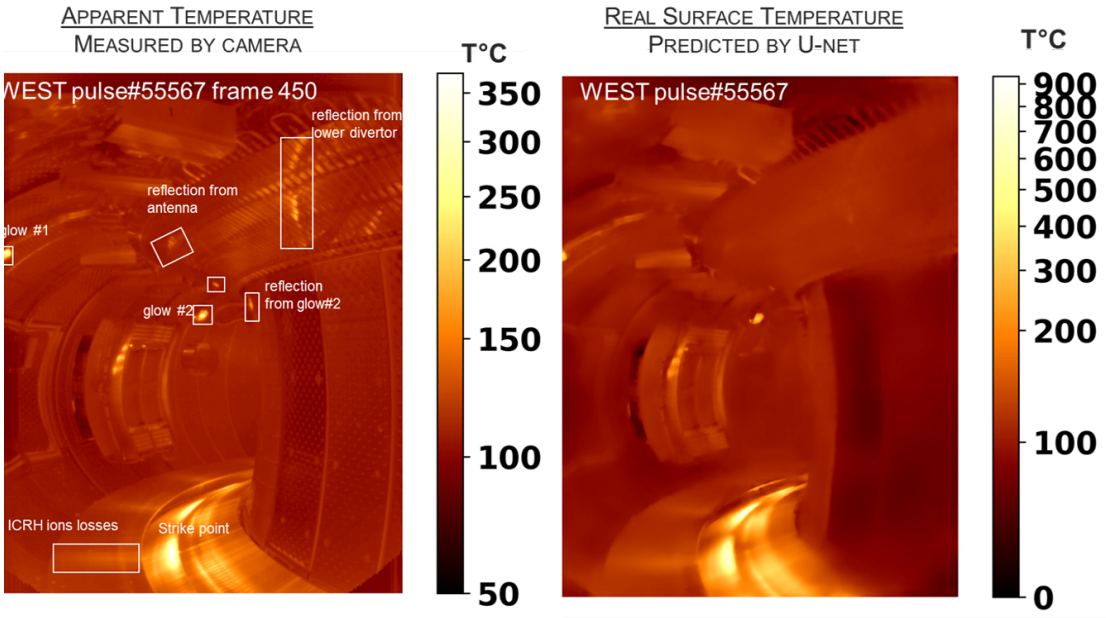


FIGURE IV.15. – Application du réseau sur une image expérimentale de la visée grand angle de WEST. **Gauche –** Température apparente (ou température de corps noir) fournie par la caméra de WEST pendant le choc 55567. **Droite –** Température de surface de la scène observée prédite par le U-Net. La plupart des motifs de réflexion (provenant du divertor inférieur et de l'antenne sur le divertor supérieur) sont effacés par le U-Net.

environnement tokamak complexe tel que WEST en utilisant des diagnostics embarqués (thermocouples, fibre de Bragg) pour vérifier les mesures de température en quelques points de mesure, et en tenant compte des grandes incertitudes inhérentes à ces diagnostics embarqués. De plus, une prochaine étape consistera aussi à valider les méthodes d'estimation d'émissivité ou de température sur des images obtenues à partir d'expériences bien contrôlées et dédiées en laboratoire.

IV.4. Limite de la méthode : problème mal posé sur des scènes en visée divertor

Pour mettre en évidence les limites de la méthode employée dans cette étude, cette section aborde la nature mal posée de la mesure de la température de surface à partir de systèmes d'imagerie infrarouge. Le problème est illustré ici en considérant une scène thermique simplifiée où toutes les surfaces sont purement diffuses, et un champ de vue de la caméra focalisé sur un point unique. Le flux collecté est alors décrit par l'équation (IV.18) :

$$L_o = \varepsilon L^\circ(T) + (1-\varepsilon) L_i^\Omega \tag{IV.18}$$

avec L_o la radiance sortante pour une direction donnée, T la température locale, $\varepsilon \in [0;1]$

l'émissivité, et L_i^{Ω} l'irradiance incidente agrégée de toutes les directions.

Sans connaissance préalable supplémentaire sur les valeurs de ε et L_i^{Ω}, plusieurs solutions sont possibles pour T, car il existe un continuum de combinaisons (T, ε) qui satisfont cette relation.

Pour cette analyse, on s'intéresse à une visée se focalisant sur le divertor inférieur. Ce composant est particulièrement sensible aux changements d'état de surface, dus aux phénomènes d'érosion-dépôt ou les dommages qui peuvent survenir pendant les opérations. Ces phénomènes peuvent entraîner des changements locaux d'émissivité.

La Figure IV.16 illustre le cas d'un problème mal posé en raison de ces phénomènes, en générant deux images identiques avec une émissivité et une température différentes sur un petit point du divertor.

Pour évaluer la performance du réseau pour ce cas particulier, nous avons construit un nouvel ensemble de données d'entraînement contenant des paires d'images identiques, mais générées à partir de différents couples (T, ε) en un petit point. Pour des fins de visualisation et de discussion, nous avons ajouté un canal de sortie : le modèle U-Net est alors entraîné pour produire à la fois la carte de température et la carte d'émissivité. Après le processus d'entraînement, les prédictions du réseau sont évaluées pour la température et l'émissivité au centre du point mal posé sur un ensemble de paires d'images de test.

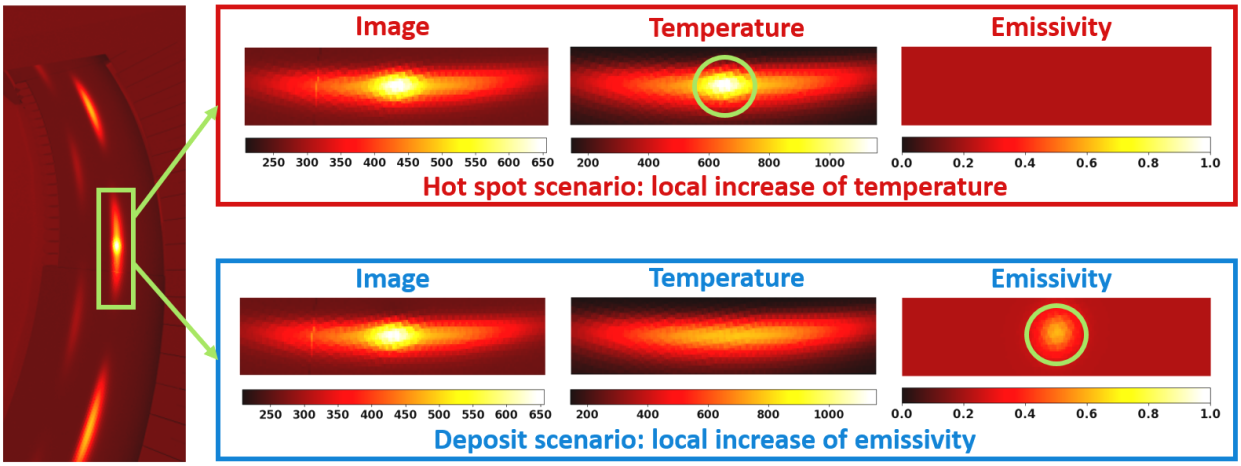


FIGURE IV.16. – Génération de paires d'images ambiguës en utilisant une visée divertor et des réflexions diffuses pures : en augmentant localement soit la température, soit l'émissivité sur un petit point, les mêmes images sont générées. Un problème inverse tel que l'estimation de la température ou de l'émissivité devient mal posé sur ce petit point si l'ensemble des réponses possibles contient les deux situations.

La Figure IV.17 illustre les prédictions du réseau pour des paires d'images de test à haute température ($\geqslant 1200\,^{\circ}\mathrm{C}$). Le réseau a du mal à distinguer entre les deux scénarios, conduisant à une solution qui se situe entre les résultats possibles.

Ce résultat est attendu, il peut être montré [Has+09] qu'en apprentissage supervisé, le modèle $\mathcal{M}^*$ minimisant la perte quadratique moyenne (MSE), est celui qui renvoie la moyenne des solutions possibles, comme décrite dans l'équation (IV.19).

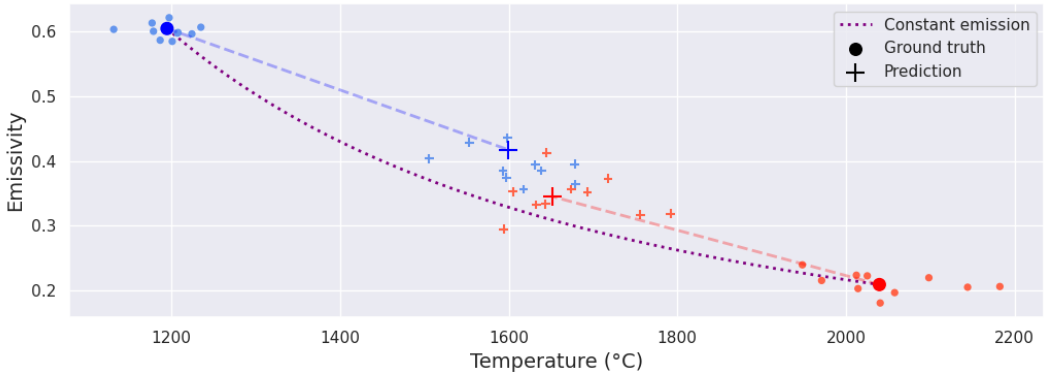


FIGURE IV.17. – Prédictions locales de température (T) et d'émissivité (ε) par U-Net dans le cas d'un "point chaud" ambigu où le problème inverse est mal posé. Les deux grands cercles rouges et bleus représentent la T locale réelle et ε pour deux images, représentant respectivement un point chaud et un dépôt. Ces deux images sont presque identiques, car les deux cercles se trouvent sur la courbe en pointillés violette, qui représente l'ensemble des couples (T, ε) produisant la même radiance émise. Les grandes croix bleues et rouges sont les T et ε prédits par U-Net pour ces deux cas. Seul un petit flux réfléchi cause une légère différence entre les deux images, ce qui explique la différence entre la prédiction bleue et rouge. De petites variations des mêmes images autour des deux grands cercles sont également affichées avec des cercles et des croix plus petits. Les prédictions globales du réseau ont tendance à être centrées autour du couple (T,ε) moyen, ce qui est attendu pour un modèle qui minimise l'erreur quadratique moyenne globale sans avoir la capacité de discriminer un cas d'un autre.

$$\begin{aligned} &\mathcal{M}^* = \operatorname{argmin}_{\mathcal{M}} \mathcal{L}_{\mathrm{MSE}}(\mathcal{M}) \\ &\implies \forall \mathbf{y} \in \mathcal{Y}, \quad \mathcal{M}^*(\mathbf{y}) = \mathbb{E}_{X|Y}\left[X \middle| \mathbf{y}\right] \end{aligned} \tag{IV.19}$$

Où $\mathbb{E}_{X|Y}$ désigne l'espérance conditionnelle de la variable aléatoire X (le paramètre à estimer, ici le couple (T,ε)) étant donné une mesure $\mathbf{y}$ (ici, l'image). Les variables mesurées et celles à déterminer ne sont ainsi pas liées par des relations purement déterministes : puisqu'il y a d'autres phénomènes influents et une perte d'information lors de la mesure, il en résulte plutôt une probabilité qu'un paramètre à estimer ait une certaine valeur étant donné la mesure qui a été faite.

Cet exemple met en évidence la limitation d'avoir une méthode déterministe capable de fournir uniquement une réponse moyenne lorsqu'il n'y a pas suffisamment d'informations pour distinguer plusieurs solutions. En fonction des applications, cette réponse moyenne peut ne pas être satisfaisante, car une moyenne de solutions n'est pas forcément une solution. Par exemple, en Figure IV.17, la prédiction de couple (T,ε) fourni par le réseau ne se trouve pas sur l'ensemble en pointillés de solutions "valides", c'est-à-dire générant la même radiance émise.

Une solution à ce problème est l'utilisation d'une phase d'apprentissage auto-supervisé, comme est souvent fait dans le domaine de l'*inverse rendering* (voir section II.1.1). Après un apprentissage supervisé sur des données annotées, le réseau passe par une phase de *fine tuning*

avec une fonction de perte utilisant un modèle direct différentiable qui permet de vérifier que les paramètres estimés par le réseau génèrent une image synthétique proche de celle fournie en entrée. La fonction de perte est alors une erreur quadratique entre l'image fournie en entrée du réseau, et celle reconstruite par le modèle direct différentiable à partir des prédictions du réseau. La différentiabilité du modèle direct permet la rétropropagation du gradient de l'erreur jusqu'aux poids du réseau. Lorsque la fonction de perte utilisée est l'erreur quadratique moyenne, chaque solution "valide" est un minimum de la fonction de perte : le réseau après apprentissage tend alors à renvoyer une solution valide, par exemple dans notre cas un couple (T, ε) sur la ligne en pointillés de la Figure IV.17. Cette approche, choisissant arbitrairement une solution valide, présente le désavantage de ne pas fournir d'information sur l'incertitude concernant la valeur réelle des paramètres à estimer. Si cette approche est satisfaisante dans un cas de rendu inverse où l'importance est de générer des images plausibles, elle n'est pas satisfaisante dans une optique de sécurité de la machine.

En revanche, une approche probabiliste devrait être explorée pour traiter les situations mal posées : contrairement aux modèles déterministes, les modèles génératifs, déjà utilisés pour certains problèmes d'inversion en physique, peuvent être utilisés pour générer un ensemble de solutions, permettant d'attacher des incertitudes aux prédictions. C'est cette approche qui sera explorée dans le prochain chapitre.

Conclusion IV

Dans ce chapitre, nous proposons une approche numérique basée sur des simulations de lancer de rayons et l'apprentissage automatique pour inférer la température de surface des composants d'un tokamak à partir d'images infrarouges. La scène thermique à résoudre est particulièrement complexe et multiparamétrique : nous devons gérer un état de surface changeant des matériaux métalliques à faible émissivité dans un environnement chaud et réfléchissant.

Des résultats prometteurs sont obtenus avec un modèle de réseau de neurone de type U-Net appliqué sur un prototype numérique de type WEST. Sans fournir en entrée les propriétés réfléchissantes de la surface, le réseau estime la température à partir des images simulées avec une erreur globale évaluée à 6,5 % (± 10 %), et 9,3 % (± 14%) pour des images détériorées. Le réseau est également assez robuste à la présence de distributions de température inattendues, avec une erreur de température évaluée à 22 % (± 35 %), démontrant la capacité de généralisation des réseaux de neurones sur des scénarios non rencontrés durant l'apprentissage. L'amélioration de ces résultats nécessite la création d'un ensemble d'entraînement exhaustif et représentatif des scénarios rencontrés lors de l'opération des tokamaks.

Ces résultats encouragent à développer cette approche afin de traiter les données expérimentales. Bien qu'il ne soit pas possible de quantifier les performances du réseau sur des données réelles, du fait d'absence de données annotées, le réseau est capable de traiter une image expérimentale, et de renvoyer qualitativement une réponse cohérente.

Enfin, une limite du modèle a été mise en lumière dans le cas de problèmes mal posés, c'est-à-dire les cas où l'information disponible ne permet pas de discriminer de manière unique une solution. Dans ces cas-là, le modèle déterministe tend à renvoyer une moyenne des possibilités. Pour assurer une protection optimale de la paroi, il est crucial de pouvoir quantifier l'incertitude résultante d'une situation ne contenant pas assez d'information. Ces observations mettent en évidence la nécessité de développer des approches probabilistes capables de gérer ces incertitudes. Le prochain chapitre explorera cette approche.

V. Inversion des images infrarouge par réseaux de neurones non déterministes

Résumé

Dans le chapitre précédent, les limites de l'inversion par réseaux de neurones déterministes ont été mises en évidence, notamment dans le cas de problèmes mal posés où plusieurs solutions sont envisageables.
Ce chapitre introduit des modèles de réseau de neurones non déterministes, les modèles génératifs, pour traiter ces cas problématiques. Ces modèles fournissent des distributions de probabilité sur leurs sorties, ce qui permet de d'associer des incertitudes aux valeurs prédites.
Ces modèles sont testés dans deux expériences numériques : un cas simplifié permettant de contrôler avec précision tous les paramètres, et la simulation de visée divertor illustrant un problème fortement mal posé. Les résultats montrent la capacité de ces modèles à estimer la température ainsi qu'un intervalle de confiance sur leurs prédictions, ce qui peut être essentiel dans une perpective de protection de la machine.
Ces travaux font l'objet d'une publication : [JAM24a].

Sommaire

V.1. L'inférence bayésienne dans le cadre de la résolution de problème inverse 105
V.1.1. Formulation des probabilités et théorème de Bayes 105
V.1.2. Exemple d'application de l'inférence bayésienne : le problème de Laplace . 106
V.1.3. Formulation du problème inverse dans le paradigme bayésien 107
V.2. Méthodes de résolution de problèmes inverses dans le paradigme bayésien 108
V.2.1. Les méthodes de Monte-Carlo par chaîne de Markov 108
V.2.1.1. Algorithme de Metropolis-Hastings 108
V.2.1.2. Application de Metropolis-Hastings pour la résolution de problème inverse 109
V.2.2. Utilisation de réseaux de neurones non déterministes 110
V.2.2.1. Les GANs : Generative Adversarial Networks 110
V.2.2.2. Le cWGAN-GP pour la résolution de problèmes inverses 112
V.3. Validation du cWGAN-GP dans un environnement contrôlé 113
V.3.1. Génération de l'ensemble de données 113
V.3.2. Architecture du réseau et apprentissage 113
V.3.3. Résultats 115

V.4. Application du cWGAN-GP à un problème mal posé sur une visée divertor 117
V.4.1. Description de l'ensemble de données 117
V.4.2. Architectures et apprentissage 118
V.4.3. Résultats 119

V.1. L'inférence bayésienne dans le cadre de la résolution de problème inverse

L'inférence bayésienne constitue une méthode puissante pour aborder les problèmes inverses, particulièrement dans les situations où les mesures présentent une incertitude inhérente et les observations disponibles ne suffisent pas à déterminer une solution unique. Elle est ainsi utilisée pour estimer l'incertitude de paramètres estimés à partir de mesures physiques [VT11]. Elle repose sur l'utilisation de distributions de probabilité, et l'application du théorème de Bayes pour réviser et mettre à jour les croyances antérieures en fonction des nouvelles données acquises.

Dans ce contexte, l'incertitude sur les paramètres estimés n'est pas vue comme une caractéristique des objets eux-mêmes, mais plutôt comme une expression du degré de certitude ou de croyance de l'observateur. Cette incertitude peut être réévaluée et ajustée à mesure que de nouvelles informations deviennent disponibles, permettant ainsi une mise à jour continue des croyances de l'observateur sur l'état du système étudié.

V.1.1. Formulation des probabilités et théorème de Bayes

Nous introduisons ici les notations et concepts fondamentaux utilisés pour formaliser le problème en termes de probabilités et de variables aléatoires. Pour alléger le propos, certaines simplifications sont faites en comparaison à une formalisation rigoureuse de théorie de la mesure.

Les variables aléatoires sont représentées en majuscules, par exemple, X (respectivement Y), prenant valeur dans un ensemble mesurable $\mathcal{X}$ (respectivement $\mathcal{Y}$). Ici X est une variable aléatoire représentant le paramètre que l'on cherche à estimer à partir d'une observation Y. Ces variables aléatoires sont liées à des fonctions de distribution de probabilités, notées p_X et p_Y. Par exemple, si $\mathcal{E}$ est un sous ensemble mesurable de $\mathcal{X}$, la probabilité que le paramètre X appartienne à $\mathcal{E}$ est donnée par l'équation (V.1).

$$\mathbb{P}[X \in \mathcal{E}] = \int_{\mathcal{E}} p_X(\mathbf{x})\, \mathrm{d}x \tag{V.1}$$

La notation se prolonge à la paire (X, Y), elle-même une variable aléatoire à valeurs dans $\mathcal{X} \times \mathcal{Y}$. On peut caractériser cette variable par sa fonction de distribution $p_{X,Y}$, qui capture la probabilité que X et Y prennent tous deux une valeur spécifique. Cette distribution jointe peut être *a priori* de forme quelconque, on en déduit les probabilités marginales p_X et p_Y en intégrant la fonction de probabilité comme montré en équation (V.2).

$$p_X(\mathbf{x}) = \int_{\mathcal{Y}} p_{X,Y}(\mathbf{x},\mathbf{y})\, \mathrm{d}y \quad ; \quad p_Y(\mathbf{y}) = \int_{\mathcal{X}} p_{X,Y}(\mathbf{x},\mathbf{y})\, \mathrm{d}x \tag{V.2}$$

De plus on définit la probabilité conditionnelle $p_{X|Y}$ (à lire "probabilité de X sachant Y") par l'équation (V.3).

$$p_{X|Y}(\mathbf{x}|\mathbf{y}) = \frac{p_{X,Y}(\mathbf{x},\mathbf{y})}{p_Y(\mathbf{y})} \tag{V.3}$$

Cette distribution $p_{X|Y}$ est à interpréter comme une nouvelle distribution de probabilité sur X, qui est mise à jour après l'observation de la variable Y. Dans ce cadre, alors que p_X est

dénommée la distribution *a priori* (c'est-à-dire capturant les croyances de l'observateur sur X avant la connaissance de la valeur de Y), inversement, $p_{X|Y}$ est appelée la distribution *a posteriori*, puisqu'elle correspond à une mise à jour des croyances de l'observateur après l'acquisition d'une nouvelle information.

Par la définition des probabilités conditionnelles et leur lien avec les distributions jointes et marginales, le théorème de Bayes, en équation (V.4) lie alors les probabilités conditionnelles $p_{X|Y}$ et $p_{Y|X}$.

$$p_{X|Y}(\mathbf{x}|\mathbf{y}) = \frac{p_{Y|X}(\mathbf{y}|\mathbf{x})}{p_Y(\mathbf{y})} p_X(\mathbf{x}) \tag{V.4}$$

Enfin, comme décrit en équation (V.5), ce théorème permet d'exprimer la distribution *a posteriori* $p_{X|Y}$ uniquement en fonction de la distribution *a priori* p_X et de la distribution $p_{Y|X}$, appelée vraisemblance, qui capture la probabilité de réaliser une observation $\mathbf{y}$ dans le cas où la valeur réelle du paramètre à estimer X serait $\mathbf{x}$.

$$p_{X|Y}(\mathbf{x}|\mathbf{y}) = \frac{p_{Y|X}(\mathbf{y}|\mathbf{x})}{\int_{\mathcal{X}} p_{Y|X}(\mathbf{y}|\mathbf{x}') p_X(\mathbf{x}') \, \mathrm{d}x'} p_X(\mathbf{x}) \tag{V.5}$$

Ainsi, la connaissance de la vraisemblance conditionnelle $p_{Y|X}$ de l'observation et l'introduction d'une distribution *a priori* p_X sur le paramètre à estimer suffit en théorie à pouvoir calculer pour X une nouvelle distribution *a posteriori* $p_{X|Y}(\cdot|\mathbf{y})$ à la suite d'une observation $\mathbf{y}$. Cette nouvelle distribution correspond à une mise à jour de l'incertitude de l'observateur concernant la valeur réelle de X.

V.1.2. Exemple d'application de l'inférence bayésienne : le problème de Laplace

Un exemple classique d'inférence bayésienne est illustré par Pierre-Simon de Laplace, qui chercha à répondre à la question suivante : quelle est la probabilité que le soleil ne se lève pas demain, sachant qu'il s'est levé tous les jours durant les 5000 dernières années ?

Pour reformuler et généraliser ce problème, considérons une pièce de monnaie biaisée dont la probabilité X de tomber sur pile, et $1 - X$ de tomber sur face, est inconnue. Afin de déterminer la valeur probable de X, supposons que nous effectuons n lancers indépendants de cette pièce et observons qu'elle tombe k fois sur pile et $n - k$ fois sur face. Notre observation est donc ici "$Y = k$". Partant d'un *a priori* uniforme, c'est-à-dire sans connaissance supplémentaire rendant chaque probabilité x équiprobable, nous avons pour tout $x \in [0;1]$, $p_X(x) = 1$. La vraisemblance de l'observation "$Y = k$" sachant "$X = x$" est ensuite fournie par la loi binomiale :

$$p_{Y|X}(k|x) = \binom{n}{k} x^k (1-x)^{n-k}.$$

À partir de cette vraisemblance, nous pouvons déduire la distribution *a posteriori* conditionnelle correspondant à chaque valeur de x après avoir observé k fois pile pour $n - k$ fois face :

$$p_{X|Y}(x|k) = \frac{x^k(1-x)^{n-k}}{\int_0^1 t^k(1-t)^{n-k}\,\mathrm{d}t} = \frac{(n+1)!}{k!(n-k)!}x^k(1-x)^{n-k}.$$

La probabilité conditionnelle ainsi calculée permet par exemple de déterminer l'espérance conditionnelle de X :

$$\mathbb{E}_{X|Y}[X|k] = \int_0^1 x p_{X|Y}(x|k)\,\mathrm{d}x = \frac{k+1}{n+2}.$$

Ainsi, après k observations de pile et $n-k$ de face, la meilleure estimation de la probabilité d'obtenir un pile au prochain lancer est $(k+1)/(n+2)$.

Appliquant ce raisonnement au problème posé par Laplace, en considérant $k = 0$ et $n = 5000 \times 365.25$, on obtient une probabilité extrêmement faible, de l'ordre de 5×10^{-7}, que le soleil ne se lève pas demain.

V.1.3. Formulation du problème inverse dans le paradigme bayésien

Dans le cadre bayésien, la formulation classique de la résolution de problème inverse [Stu10] introduit un opérateur déterministe direct $\mathcal{F}$, qui lie les variables X et Y. De plus, une source de perturbation supplémentaire N est introduite, indépendante des autres variables, afin de représenter un bruit de mesure aléatoire qui vient perturber l'observation. Les trois variables aléatoires X, Y, et N sont alors liées par l'équation (V.6).

$$Y = \mathcal{F}(X) + N \tag{V.6}$$

Après la mesure d'une valeur spécifique $\mathbf{y}$, la distribution *a posteriori* $p_{X|Y}$ peut être mise à jour en utilisant le théorème de Bayes, comme montré dans l'équation (V.7). Dans cette formulation, la vraisemblance $p_{Y|X}$ est directement donnée par la distribution du bruit p_N.

$$p_{X|Y}(\mathbf{x}|\mathbf{y}) = \frac{p_N(\mathbf{y}-\mathcal{F}(\mathbf{x}))}{\int_{\mathcal{X}} p_N(\mathbf{y}-\mathcal{F}(\mathbf{x}'))\,p_X(\mathbf{x}')\,\mathrm{d}x'}\,p_X(\mathbf{x}) \tag{V.7}$$

Dans ce cas là, le calcul théorique de la probabilité *a posteriori* peut se faire si l'on suppose la connaissance d'un apriori p_X, d'un opérateur déterministe calculable $\mathcal{F}$, et d'un bruit de mesure de distribution connue p_N. Toutefois, cette formulation ne couvre pas tous les scénarios possibles dans la résolution de problèmes inverses. En particulier, elle présume que le bruit est une variable indépendante ajoutée à une mesure déterministe par rapport au paramètre estimé. Lorsque le paramètre X n'est pas suffisant pour déterminer de manière unique la mesure (à l'erreur de bruit près), cette formulation devient incomplète. Une approche plus générale peut être envisagée en introduisant un opérateur $\mathcal{F}$ déterministe à deux entrées tel que $Y = \mathcal{F}(X,N)$, où X et N ne sont pas nécessairement indépendants. Dans ce cadre, N pourrait encapsuler à la fois les effets d'incertitude de mesure ainsi que d'autres paramètres non pris en compte dans X. Bien que cette formulation soit plus complexe, elle permet une description plus fidèle de nombreux problèmes inverses où les interactions entre variables sont non linéaires ou interdépendantes.

La formulation classique présentée ici sera utilisée en section V.3 pour créer un jeu de données dans un cadre contrôlé permettant de valider la méthode de résolution de problème inverse proposée. La méthode sera ensuite appliquée à un cas plus complexe de données de simulation de tokamak en section V.4, où la relation entre les paramètres à estimer et les autres variables influant la mesure correspondent au cas générique.

V.2. Méthodes de résolution de problèmes inverses dans le paradigme bayésien

Pour certaines applications de résolution de problème inverse, il peut être nécessaire de trouver le paramètre $\mathbf{x}^*$ qui maximise la probabilité *a posteriori*, une méthode connue sous le nom d'estimation du maximum *a posteriori* (MAP). Toutefois, nous nous intéressons davantage à une méthode d'échantillonnage qui reproduit fidèlement la distribution de probabilité *a posteriori* $p_{X|Y}$: cette approche permet d'estimer une plage d'incertitude pour les paramètres estimés en calculant les moyennes et écarts types empiriques des échantillons générés.

Calculer précisément l'expression de la distribution *a posteriori*, telle que présentée en équation (V.7), est souvent prohibitif en termes de quantité de calculs, notamment à cause du calcul de l'évidence p_Y, qui nécessite une intégration sur tous les scénarios possibles de couples (X, N). Bien que cela soit réalisable analytiquement sur des exemples simples, une distribution sur des ensembles de grande dimension est difficilement évaluable en pratique. De plus, calculer la valeur d'une distribution de probabilité en un point donné ne fournit pas directement une méthode pour générer des échantillons suivant cette distribution, ce qui est notre principal but.

Plusieurs méthodes numériques sont à notre disposition pour résoudre ce problème. La section V.2.1 présente une méthode de Monte-Carlo par chaîne de Markov, qui sera utilisée pour valider la méthode par réseau de neurones non déterministes, présentée en section V.2.2.

V.2.1. Les méthodes de Monte-Carlo par chaîne de Markov

Les méthodes de Monte-Carlo par chaînes de Markov (MCMC) sont des techniques d'échantillonnage puissantes qui permettent de générer des échantillons $\mathbf{x}_1, \ldots, \mathbf{x}_n$, à partir d'une distribution de probabilité p difficile à traiter analytiquement. L'idée principale de ces méthodes repose sur la construction d'une chaîne de Markov dont la distribution stationnaire est la distribution cible p. En simulant cette chaîne sur un grand nombre d'itérations, on obtient une suite d'échantillons qui convergent théoriquement vers la distribution souhaitée.

V.2.1.1. Algorithme de Metropolis-Hastings

L'algorithme de Metropolis-Hastings [Met+53], une méthode MCMC, ne nécessite que la connaissance de la distribution p à un facteur de normalisation près. On dispose d'une fonction f que l'on peut calculer et qui est proportionnelle à la fonction de distribution p.

Cette méthode utilise l'échantillonnage par rejet pour générer des échantillons à partir de la distribution cible. Elle débute par la mutation de l'état courant $\mathbf{x}_t$ pour proposer un nouvel état $\mathbf{x}_{t+1}$, généralement réalisée par une distribution de proposition $g(\cdot|\mathbf{x}_t)$ (considérée ici comme

symétrique, c'est à dire $g(\mathbf{a}|\mathbf{b}) = g(\mathbf{b}|\mathbf{a})$). Une fois le nouvel état proposé, l'algorithme évalue le ratio de vraisemblance $\alpha = f(\mathbf{x}_{t+1})/f(\mathbf{x}_t)$. Si α est supérieur à une valeur aléatoire tirée uniformément entre 0 et 1, le nouvel état est accepté, sinon, l'état courant est conservé pour la prochaine itération. Cette procédure garantit la convergence vers la distribution cible p, même si elle introduit une corrélation entre échantillons successifs. Cette corrélation n'altère pas le calcul de statistiques comme la moyenne ou l'écart type sur un grand nombre d'échantillons, car les échantillons générés convergent statistiquement vers la distribution de probabilité cible, permettant des estimations fiables des caractéristiques de cette distribution.

L'algorithme est décrit en Figure V.1. Dans cette fonction, x est l'état actuel, f évalue la probabilité *a posteriori*, et g est la fonction de proposition qui génère un nouvel état candidat. L'algorithme décide d'accepter ou de rejeter le nouvel état basé sur le ratio de vraisemblance α.

```
def metropolis_sampling(x, f, g):
    x' ~ g(·|x)
    α = f(x')/f(x)
    u ~ U(0,1)
    if u ≤ α:
        return x'
    return x
```

FIGURE V.1. – Pseudo-code Python de l'algorithme d'échantillonnage de Metropolis-Hastings, renvoyant un nouvel échantillon $\mathbf{x}'$ à partir d'un ancien échantillon $\mathbf{x}$, une fonction f proportionnelle à la fonction de distribution à échantillonner, et une distribution échantillonnable g de mutation d'état.

V.2.1.2. Application de Metropolis-Hastings pour la résolution de problème inverse

Dans notre contexte, l'algorithme de Metropolis-Hastings peut être utilisé pour échantillonner des éléments $\mathbf{x}_1, ..., \mathbf{x}_n$ selon la distribution *a posteriori* $p_{X|Y}(\cdot|\mathbf{y})$. L'un des principaux avantages de cette méthode est qu'elle ne requiert la connaissance de la distribution cible qu'à un facteur multiplicatif près, ce qui élimine la nécessité de calculer l'evidence $p_Y(\mathbf{y})$, une intégrale sur toutes les possibilités qui est souvent difficile à évaluer. De même, la vraisemblance $p_{Y|X}$ peut être approximée par une fonction $\mathcal{L}_{Y|X}$ proportionnelle à $p_{Y|X}$, et la distribution *a priori* p_X peut être représentée à un facteur près par une fonction f_X. Ces simplifications permettent d'utiliser l'algorithme de Metropolis sur la fonction $f = \mathcal{L}_{Y|X}(\mathbf{y}|\cdot)f_X$ pour échantillonner des éléments qui suivent la distribution $p_{X|Y}(\cdot|\mathbf{y})$.

Dans le cadre de la formulation classique du problème inverse présentée précédemment, avec la possibilité de calculer $\mathcal{F}$, p_N et p_X, on utilise alors $f(\mathbf{x}) = p_N(\mathbf{y} - \mathcal{F}(\mathbf{x}))p_X(\mathbf{x})$. Cependant, il est important de noter que cela nécessite de nombreuses évaluations de l'opérateur direct, ce qui peut s'avérer impossible si cette fonction nécessite une quantité trop importante de calculs. De plus, cela requiert des expressions explicites des distributions p_X et p_N, qui peuvent ne pas être disponibles dans des applications pratiques. Pour ces raisons, cette méthode sera principalement utilisée en validation de la méthode présentée dans les prochaines sections, qui vise à utiliser un réseau de neurones pour approximer la distribution visée.

V.2.2. Utilisation de réseaux de neurones non déterministes

Les applications contemporaines utilisent l'apprentissage automatique, en particulier des modèles génératifs profonds, pour approximer efficacement la distribution *a posteriori* et fournir un processus d'échantillonnage pratique. Cette méthode offre une solution viable pour gérer les distributions complexes rencontrées dans la résolution de problèmes inverses génériques.

Nous présentons ici l'utilisation de réseaux de neurones, plus précisément des modèles génératifs, comme méthode d'échantillonnage à partir de la distribution *a posteriori* $p_{X|Y}$. L'idée générale est d'entraîner un modèle $\mathcal{M}$ à transformer une variable aléatoire Z, qui suit une distribution simple et connue, en une prédiction $\mathcal{M}(Y,Z)$ suivant la distribution *a posteriori* $p_{X|Y}$ comme exprimé dans l'équation (V.8).

$$p_{\mathcal{M}(Y,Z)|Y}(\mathbf{x}|\mathbf{y}) \simeq p_{X|Y}(\mathbf{x}|\mathbf{y}) \tag{V.8}$$

Pour une mesure donnée $\mathbf{y}$, le tirage d'un ensemble d'échantillons selon la distribution de p_Z et leur introduction dans le réseau fournissent des échantillons de $p_{X|Y}$.

Diverses architectures de modèles génératifs ont été explorées, comme les autoencodeurs variationnels (VAEs) [Goh+22 ; Pro+23], les réseaux antagonistes génératifs (GANs) et, plus récemment, les modèles de diffusion [Sah+22 ; Chu+22 ; Sah+23]. Dans notre cas, nous avons opté pour une variante de GAN conditionnel, présentée dans les prochaines sections.

V.2.2.1. Les GANs : Generative Adversarial Networks

Les Generative Adversarial Networks (GANs) [Goo+14] sont un type de modèles génératifs qui se compose de deux réseaux de neurones distincts : un générateur $\mathcal{G}$ et un discriminateur $\mathcal{D}$. Le but d'un GAN est d'entraîner son générateur, de manière non supervisée, à générer des éléments semblables à ceux présents dans le jeu de donné utilisé pour son entraînement. Un exemple connu d'application est la génération d'image de visages de personnes fictives à partir d'un jeu de donnée constitué de photos de visages. Plus précisément, le générateur apprend à échantillonner des éléments selon la distribution de probabilité p_X qui a généré l'ensemble de données que l'on cherche à reproduire.

Le générateur prend une variable latente $Z \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ comme entrée, et produit une donnée synthétique $\mathcal{G}(Z)$. Le but du générateur est de générer des échantillons qui sont indiscernables des échantillons réels pour le discriminateur. Le discriminateur, quant à lui, reçoit soit une donnée réelle X provenant de la distribution des données réelles p_X, soit une donnée générée $\mathcal{G}(Z)$, et doit distinguer entre les deux. Le rôle du discriminateur est de maximiser la probabilité d'assigner la bonne étiquette (réelle ou générée) aux échantillons qu'il reçoit.

L'entraînement des GANs est un processus itératif alterné entre les deux réseaux. D'abord, le discriminateur est entraîné pour maximiser sa capacité à distinguer les données réelles des données générées. Cette phase d'entraînement utilise la fonction de perte fournie en équation (V.9).

$$\mathcal{L}_D = -\mathbb{E}_{X\sim p_X}[\log \mathcal{D}(X)] - \mathbb{E}_{Z\sim p_Z}[\log(1-\mathcal{D}(\mathcal{G}(Z)))] \tag{V.9}$$

Ensuite, le générateur est entraîné pour maximiser la probabilité que le discriminateur classifie les données générées comme réelles. Cette phase utilise la fonction de perte définie en équation (V.10).

$$\mathcal{L}_G = -\mathbb{E}_{Z\sim p_Z}[\log \mathcal{D}(\mathcal{G}(Z))] \tag{V.10}$$

Pour cela, le générateur utilise le gradient de la fonction de perte du discriminateur pour ajuster ses propres poids.

Un des principaux avantages des GANs est leur capacité à effectuer une association "*one-to-many*". Contrairement aux méthodes supervisées traditionnelles, qui apprennent une correspondance déterministe entre les entrées et les sorties, les GANs exploitent l'aléatoire introduit par la variable latente Z pour générer une diversité d'échantillons.

En théorie, les GANs minimisent la divergence de Jensen-Shannon (JS) entre la distribution réelle p_X et la distribution générée par le générateur $\mathcal{G}(Z)$. La divergence de Jensen-Shannon est une mesure de similarité entre deux distributions de probabilité et est définie en équation (V.11).

$$\mathrm{JS}(p_X \| p_G) = \frac{1}{2}\left(\mathrm{KL}(p_X \| p_M) + \mathrm{KL}(p_G \| p_M)\right) \tag{V.11}$$

où $p_M = \frac{1}{2}(p_X + p_G)$ et KL est la divergence de Kullback-Leibler. Théoriquement, en minimisant cette divergence, on obtient un générateur dont la distribution de sortie p_G est une reproduction exacte de la distribution réelle p_X.

Cependant, en pratique, les GANs rencontrent plusieurs problèmes d'entraînement. Parmi ces problèmes, on trouve l'instabilité de l'entraînement, où les gradients peuvent devenir trop faibles ou trop forts, et le phénomène de "mode collapse", où le générateur produit une variété limitée d'échantillons, ignorant certaines modes de la distribution réelle. Ces problèmes rendent difficile la convergence des GANs vers une solution optimale.

Pour surmonter ces défis, les Wasserstein GANs (WGANs) ont été introduits [ACB17]. Les WGANs minimisent la distance Wasserstein-1 (ou distance de transport optimal), au lieu de la divergence de Jensen-Shannon. La distance Wasserstein-1 est définie par l'équation (V.12).

$$W_1(p_X, p_G) = \inf\left\{\mathbb{E}_{(X,G)\sim p_{(X,G)}}\left[\|X - G\|_1\right] \,\middle|\, p_{(X,G)} \in \Pi(p_X, p_G)\right\} \tag{V.12}$$

Ici, $\Pi(p_X, p_G)$ est l'ensemble des distributions jointes sur $\mathcal{X} \times \mathcal{X}$ ayant pour distributions marginales respectives p_X et p_G. La distance Wasserstein-1 fournit un gradient utile même lorsque les distributions p_X et p_G ne se chevauchent pas, contribuant ainsi à stabiliser l'entraînement [Wen19]. Bien que l'expression de la distance Wasserstein-1 soit compliquée et son calcul exact relève d'une optimisation, cette distance se trouve minimisée par le générateur en utilisant simplement la fonction de perte en équation (V.13).

$$\mathcal{L}_{\mathrm{WGAN}} = \mathbb{E}_{X\sim p_X}[\mathcal{D}(X)] - \mathbb{E}_{Z\sim p_Z}[\mathcal{D}(\mathcal{G}(Z))] \tag{V.13}$$

Cependant, cela n'est vrai que lorsque l'espace de recherche du discriminateur est restreint aux fonctions 1-Lipschitziennes. En pratique, pour garantir cette contrainte, les WGANs à pénalité de gradient (WGAN-GP) [Gul+17] ajoutent un terme de pénalité de gradient à la fonction de perte du discriminateur pendant l'apprentissage pour le contraindre à avoir un gradient de norme 1. Cette contrainte est formulée par l'équation (V.14), où $\widehat{X}$ est obtenu par interpolation entre les échantillons générés et les échantillons réels, comme exprimé en équation (V.15).

$$\mathcal{L}_{\text{GP}} = \lambda_{\text{GP}} \mathbb{E}_{\widehat{X}\sim p_{\widehat{X}}} \left[\left(\left\| \nabla_{\widehat{X}} \mathcal{D}\left(\widehat{X}\right) \right\|_2 - 1 \right)^2 \right] \tag{V.14}$$

$$\alpha \sim \mathcal{U}(0,1) \quad ; \quad \widehat{X} = \alpha \mathcal{G}(Z) + (1-\alpha) X \tag{V.15}$$

En utilisant cette approche, les WGAN-GP parviennent à stabiliser l'entraînement et à s'approcher des exigences théoriques garantissant la convergence vers la distribution ciblée.

L'entraînement est généralement réalisé par phases alternées : le discriminateur est entraîné à minimiser $\mathcal{L}_{\text{WGAN}} + \mathcal{L}_{\text{GP}}$ sur plusieurs *batchs*, suivi d'un seul *batch* où le générateur est entraîné à maximiser $\mathcal{L}_{\text{WGAN}}$.

V.2.2.2. Le cWGAN-GP pour la résolution de problèmes inverses

Les modèles génératifs tels que les WGAN-GP permettent d'apprendre à générer des éléments provenant d'une distribution quelconque p_X. Or, dans le cadre de la résolution de problème inverse, on veut conditionner cette distribution par une mesure $\mathbf{y}$, de manière à échantillonner des éléments de la distribution *a posteriori* $p_{X|Y}(\cdot|\mathbf{y})$. Pour cela, on ajoute une entrée supplémentaire au générateur et au discriminateur, pour créer un modèle conditionné : le *Conditional Wasserstein Generative Adversarial Network with Gradient Penalty* (cWGAN-GP). L'architecture du modèle est décrite en Figure V.2.

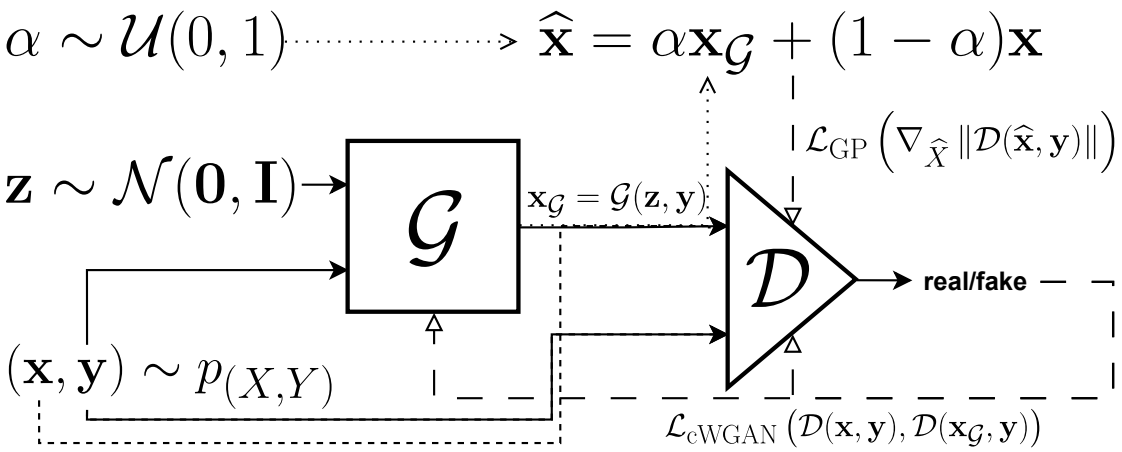


FIGURE V.2. – Représentation de la phase d'apprentissage d'un cWGAN-GP, pour résoudre un problème inverse. Le générateur, lorsque fourni une mesure $\mathbf{y}$ et un vecteur latent $\mathbf{z}$ tiré aléatoirement, produit une prédiction $\mathbf{x}_{\mathcal{G}}$. Le discriminateur reçoit successivement des couples réels $(\mathbf{x}, \mathbf{y})$ et prédits $(\mathbf{x}_{\mathcal{G}}, \mathbf{y})$ et apprend à distinguer les deux, tandis que le générateur append à tromper le discriminateur. De plus, le discriminateur a une perte supplémentaire le contraignant à avoir un gradient de norme 1, en le pénalisant en une valeur $\widehat{\mathbf{x}}$ obtenue par interpolation entre $\mathbf{x}$ et $\mathbf{x}_{\mathcal{G}}$. Sous ces conditions, le générateur apprend à échantillonner selon la distribution $p_{X|Y}(\cdot|\mathbf{y})$.

Ce modèle a largement été utilisé pour la résolution de problèmes inverses [AO18 ; Ray+22 ; YMS22 ; Ray+23] ainsi que pour de la génération d'image en problème direct [Yu+19 ; WTM20]. En

adaptant ce type de modèle à notre problématique, le cWGAN-GP sera utilisé pour approximer de manière autonome la distribution de probabilité des températures et/ou d'émissivité, étant donné une image infrarouge fournie en entrée.

V.3. Validation du cWGAN-GP dans un environnement contrôlé

Dans cette expérience initiale, notre objectif est de valider l'efficacité du cWGAN-GP à reproduire correctement la distribution de probabilité *a posteriori* bayésienne dans le cadre de données bruitées et d'un problème inverse mal posé. Pour ce faire, nous avons construit un ensemble de données contrôlé qui permet de calculer la vraie distribution, afin de la comparer à la prédiction.

V.3.1. Génération de l'ensemble de données

L'ensemble de données créé consiste en des images infrarouges synthétiques modélisant une caméra focalisée sur une petite surface caractérisée par une température T et une émissivité ε inconnues. La radiance incidente sur la surface est négligée, ce qui permet d'exprimer la radiance sortante L_o comme une fonction directe de T et ε. L'image capturée est représentée par une matrice de pixels 3×3, perturbée par un bruit gaussien N. Ceci résulte en une mesure bruitée Y, décrite par l'équation (V.16).

$$Y = \left[\varepsilon L^{\circ}(T)\right]_{3\times 3} + N \tag{V.16}$$

Cet exemple simple met en évidence deux facteurs importants des problèmes inverses : premièrement, la mesure est affectée par du bruit, et deuxièmement, même avec une mesure idéale, plusieurs paires de valeurs de température et d'émissivité peuvent produire la même radiance capturée.

L'ensemble de données de 60 000 éléments est généré en échantillonnant le paramètre $X = (T, \varepsilon)$ avec une distribution *a priori* p_X modélisée comme la combinaison linéaire de deux distributions gaussiennes multidimensionnelles, comme formulé par l'équation (V.17). La fonction de distribution p_N pour le bruit gaussien $N \sim \mathcal{N}(\mathbf{0}_{3\times 3}, \sigma_N^2)$ est fournie par l'équation (V.18). Les paramètres utilisés pour la génération sont résumés dans le Tableau V.1, et la Figure V.3 montre les distributions jointes et marginales résultantes.

$$p_X(T,\varepsilon) \propto \sum_{i=1}^{2} A_i \exp\left[-\frac{(T-\mu_{T,i})^2}{2\sigma_{T,i}^2} - \frac{(\varepsilon-\mu_{\varepsilon,i})^2}{2\sigma_{\varepsilon,i}^2}\right] \tag{V.17}$$

$$p_N(\mathbf{n}) \propto \exp\left[-\frac{1}{2\sigma_N^2}\|\mathbf{n}\|_2^2\right] \tag{V.18}$$

V.3.2. Architecture du réseau et apprentissage

Des perceptrons multicouches sont utilisés pour le générateur et le discriminateur. Chacun a 4 couches cachées avec 15 neurones dans chaque couche et utilise une activation LeakyReLU

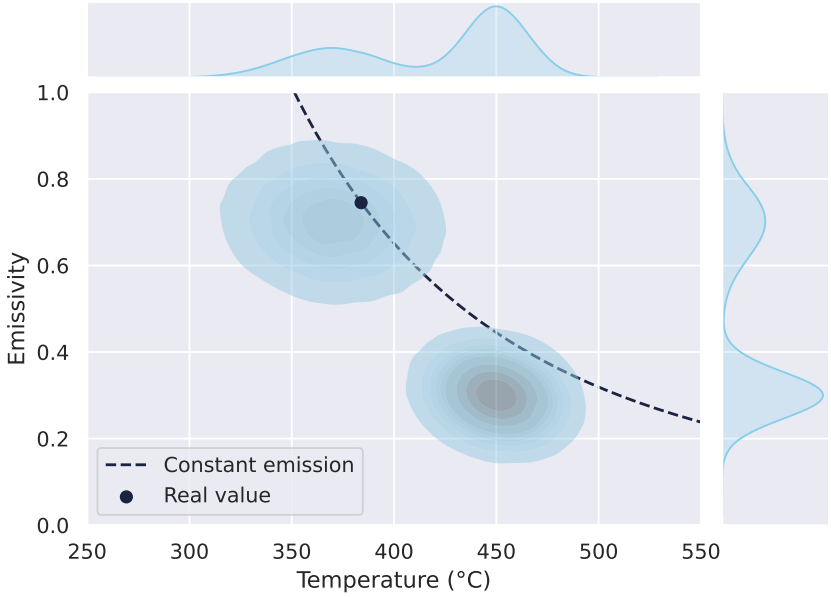


FIGURE V.3. – Distribution *a priori* utilisée pour générer l'ensemble de données en environnement contrôlé : graphiques des distributions jointes de probabilité de température (T) et d'émissivité (ε), avec leur probabilité marginale respectivement tracée en haut et à droite. Pour un couple donné (T^*, ε^*) représenté par le point noir, la ligne en pointillés montre l'ensemble des combinaisons possibles (T, ε) qui mènent à la même radiance émise.

Nom	nb	σ_N	A_1	A_2	$\mu_{T,1}$	$\mu_{\varepsilon,1}$
Valeur	60k	2 W/m^2/sr	1	4	370 °C	0.7
Nom	$\mu_{T,2}$	$\mu_{\varepsilon,2}$	$\sigma_{T,1}$	$\sigma_{T,2}$	$\sigma_{\varepsilon,1}$	$\sigma_{\varepsilon,2}$
Valeur	450 °C	0.3	24 °C	15 °C	0.08	0.05

TABLEAU V.1. – Paramètres utilisés pour générer l'ensemble de données en condition contrôlée à partir d'une distribution *a priori* constituée de deux gaussiennes multidimensionnelles.

avec une pente de 0,2. Le générateur reçoit un vecteur de dimension 13 (un vecteur d'image 3x3 combiné à un vecteur latent de dimension 4) et produit deux valeurs pour la température et l'émissivité. Le discriminateur prend une entrée de dimension 11 (le vecteur d'image de dimension 9 plus un vecteur de dimension 2 pour la température et l'émissivité) et renvoie une valeur scalaire unique.

Le cWGAN-GP est entraîné pendant 200 *epochs* avec une taille de *batch* de 64, en utilisant un ensemble d'entraînement de 50 000 images générées avec la distribution p_X. À chaque étape d'entraînement, le discriminateur est entraîné sur 4 *batchs* consécutifs, tandis que le générateur sur un seul. Le poids de la pénalité de gradient du discriminateur, λ_{GP}, est fixée à 10. L'optimiseur Adam est utilisé pour mettre à jour les poids des deux modèles, avec des paramètres fixés à 0.5 pour β_1 et 0.9 pour β_2, avec un taux d'apprentissage de 10^{-4}.

Après l'apprentissage, le réseau est testé sur un autre ensemble de 10 000 images, générées avec la même distribution p_X.

V.3.3. Résultats

Notre objectif est de déterminer si le modèle entraîné approxime la distribution *a posteriori* réelle $p_{X|Y}(\cdot|\mathbf{y})$ étant donné une observation bruitée $\mathbf{y}$, pour le paramètre $X = (T, \varepsilon)$. Dans nos conditions spécifiques, étant connus p_X (équation V.17) et p_N (équation V.18), on peut échantillonner la distribution *a posteriori* $p_{X|Y}$ en appliquant l'algorithme de Metropolis-Hastings (voir section V.2.1) à l'équation (V.19).

$$p_{X|Y}((T,\varepsilon)\,|\,\mathbf{y}) \propto\ p_N(\mathbf{y} - \varepsilon L^\circ(T))\, p_X(T,\varepsilon) \tag{V.19}$$

Pour valider le cWGAN-GP, les distributions *a posteriori* fournies par la méthode de Metropolis-Hastings et par le réseau de neurones sont comparées. La Figure V.4 montre une comparaison entre les deux distributions pour une mesure donnée du jeu de test. Qualitativement, les distributions semblent correspondre étroitement.

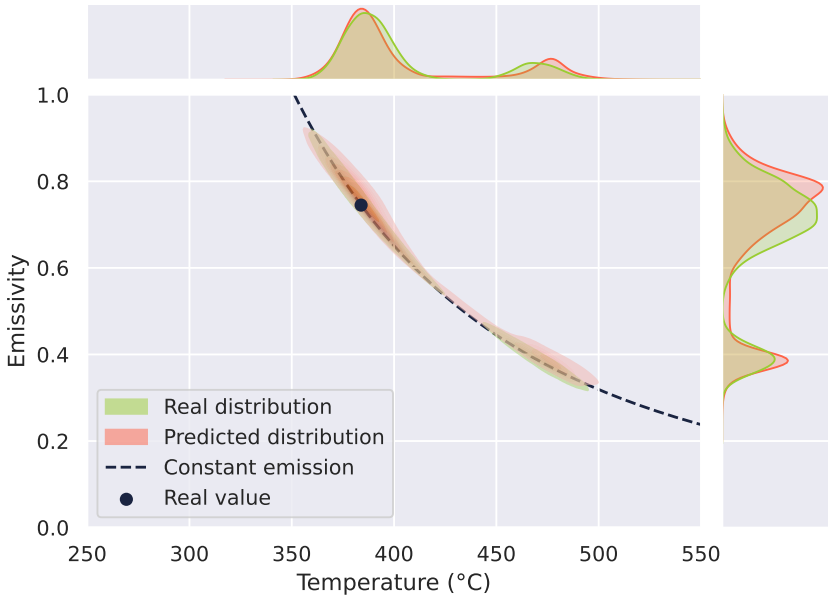


FIGURE V.4. – Distributions *a posteriori* réelles (vertes) et prédites par le cWGAN-GP (rouges) de température (T) et d'émissivité (ε) après la mesure d'une image bruitée générée à partir d'un couple (T^*, ε^*). Les distributions correspondent étroitement et sont toutes deux concentrées autour de la ligne pointillée représentant l'ensemble des combinaisons possibles (T, ε) qui mènent à la même radiance émise.

Pour réaliser une estimation quantitative avec des résultats intuitifs, on utilise un modèle de mixture de gaussiennes [Rey+09]. Ce modèle permet d'approximer une distribution sur un espace de dimension d par une distribution p_{GMM}. Cette distribution est une somme pondérée

Métrique	$\left\|w^{\mathrm{P}} - w^{\mathrm{R}}\right\| / w^{\mathrm{R}}$	$\left\|\mu_T^{\mathrm{P}} - \mu_T^{\mathrm{R}}\right\|$	$\left\|\mu_\varepsilon^{\mathrm{P}} - \mu_\varepsilon^{\mathrm{R}}\right\|$
Valeur moyenne	10.0%	2.7 °C	0.010
Métrique	$\left\|\sigma_T^{\mathrm{P}} - \sigma_T^{\mathrm{R}}\right\| / \sigma_T^{\mathrm{R}}$	$\left\|\sigma_\varepsilon^{\mathrm{P}} - \sigma_\varepsilon^{\mathrm{R}}\right\| / \sigma_\varepsilon^{\mathrm{R}}$	$\left\|r^{\mathrm{P}} - r^{\mathrm{R}}\right\|$
Valeur moyenne	22.0%	19.4%	0.017

TABLEAU V.2. – Erreurs moyennes sur l'ensemble de test, obtenues en comparant les mélanges gaussiens approximant les distributions réels (R) et prédits (P).

de n composantes, chacune étant une loi normale multidimensionnelle, comme montré en équation (V.20).

$$p_{GMM}(\mathbf{x}) = \sum_{k=1}^{n} w_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{V.20}$$

Les w_k sont des poids se sommant à 1, $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ est une loi normale multidimensionnelle, décrite en équation (V.21), de moyenne $\boldsymbol{\mu}_k$ et de matrice de covariance $\boldsymbol{\Sigma}_k$, permettant de fournir les variances et covariances de chaque variable.

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{d/2}\det(\boldsymbol{\Sigma}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right) \tag{V.21}$$

Pour un ensemble d'échantillons provenant d'une distribution quelconque, il est alors possible de trouver les poids $w_1, \ldots, w_n$, moyennes $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_n$, et matrices de covariance $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_n$ approximant le mieux la distribution. Ceci est fait itérativement par l'algorithme d'*Expectation-Maximization*.

L'intérêt d'ajuster un tel modèle à une distribution est de pouvoir extraire un ensemble restreint de valeurs quantifiant de manière pertinente la distribution. Cela permet de résumer la distribution en un ensemble de modes ayant une certaine taille, moyenne, écart type et covariances.

Dans notre cas, les distributions réelles et prédites ont été approximées par un modèle de mélange gaussien à deux composantes, puisque du fait de la distribution *a priori* utilisée, on s'attend à obtenir des distributions *a posteriori* à deux modes. Cela permet ainsi d'extraire et de comparer des paramètres significatifs, tels que les moyennes (μ_T, μ_ε), leurs écarts-types (σ_T, σ_ε), le coefficient de corrélation (r) entre T et ε, et les poids (w) de chaque composante gaussienne.

La connaissance de ces valeurs pour la distribution calculée par la méthode de Metropolis (considérée comme la vérité terrain) et celle prédite par le cWGAN-GP permet ainsi de les comparer. Les différences absolues ou relatives en termes de poids des gaussiennes, de moyenne de température ou d'émissivité prédite, d'écart type ou de corrélations sont ainsi calculables. Les erreurs moyennes absolues ou relatives ont été calculées sur un ensemble de test de 10 000 éléments, et les résultats sont résumés dans le Tableau V.2. Ces résultats indiquent une correspondance étroite entre les valeurs prédites et réelles : une erreur moyenne de 2.7 °C (respectivement 0.01) en température (respectivement en émissivité) lorsqu'on compare les modes des gaussiennes.

Les résultats sont encourageants : le réseau a automatiquement reproduit la distribution

a posteriori telles que prédites par la solution théorique. Cette première étape nous permet de valider l'algorithme d'apprentissage, qui sera appliqué à scénarios plus complexes où une expression précise de la distribution n'est pas connue, comme dans le cas de données infrarouge issues de simulations de tokamak.

V.4. Application du cWGAN-GP à un problème mal posé sur une visée divertor

Pour cette deuxième expérience, notre objectif est de nous rapprocher des situations pratiques rencontrées lors de l'exploitation des images fournies par les caméras infrarouges d'un tokamak.

Dans ce cas, nous utilisons des images issues de simulations d'une visée divertor du tokamak WEST. Contrairement à la dernière expérience, les informations ne sont pas suffisantes pour permettre le calcul d'une distribution bayésienne *a posteriori* réelle. On attend du cWGAN-GP qu'il approxime automatiquement cette distribution, ce qui permettra d'attacher des incertitudes aux prédictions.

V.4.1. Description de l'ensemble de données

Le même ensemble de données que celui de la section IV.4 est utilisé. Celui-ci contient des simulations d'une caméra du tokamak WEST se focalisant sur le divertor inférieur. Ce composant, exposé à un flux de chaleur maximal, est sujet à des changements de surface dus à des dommages ou phénomènes d'érosion-déposition, affectant son émissivité.

L'ensemble de données a été généré par lancer de rayons en utilisant la géométrie du tokamak WEST et une de ses visées divertor (voir section III.8.1), et en supposant des réflexions purement diffuses. Cette situation est connue comme entraînant la plus grande perte d'information, rendant la tâche d'inversion d'autant plus difficile [RH01]. Il suppose des distributions de température uniformes pour chaque composant principal de la machine, à l'exception du divertor inférieur, qui a un motif de chaleur qui reprend les profils de température décrits en section III.8.4. L'émissivité est uniforme pour chaque composant, sauf pour un point unique sur le divertor : il a soit une température plus élevée, soit une émissivité accrue, rendant la lumière émise par ce point identique dans les deux situations. L'ensemble de données contient des paires d'images similaires : une avec le point ayant une émissivité plus élevée, l'autre avec une température accrue, comme le montre la Figure V.5.

Chaque paramètre a été sélectionné aléatoirement pour chaque image afin de générer l'ensemble de données : la position et l'orientation de la caméra varient aléatoirement, l'émissivité varie de 0,05 à 0,5 par composant, les températures de 50 à 500 °C , sauf pour le divertor bas, dont les paramètres du profil de Eich sont aussi tirés aléatoirement, et pour lequel la température maximale peut potentiellement atteindre plus de 2 000 °C .

Dans cette expérience, les images originales, d'une résolution de 640x256 pixels, ont été recadrées pour créer des images plus petites de résolution 128x128 pixels. L'ensemble de données est composé de ces images 128x128, qui ont été converties de mesures de radiance en température corps noir apparente, associées à la carte de température vraie, qui représente une projection 3D vers 2D des températures initialement entrées dans le programme de lancer de rayons.

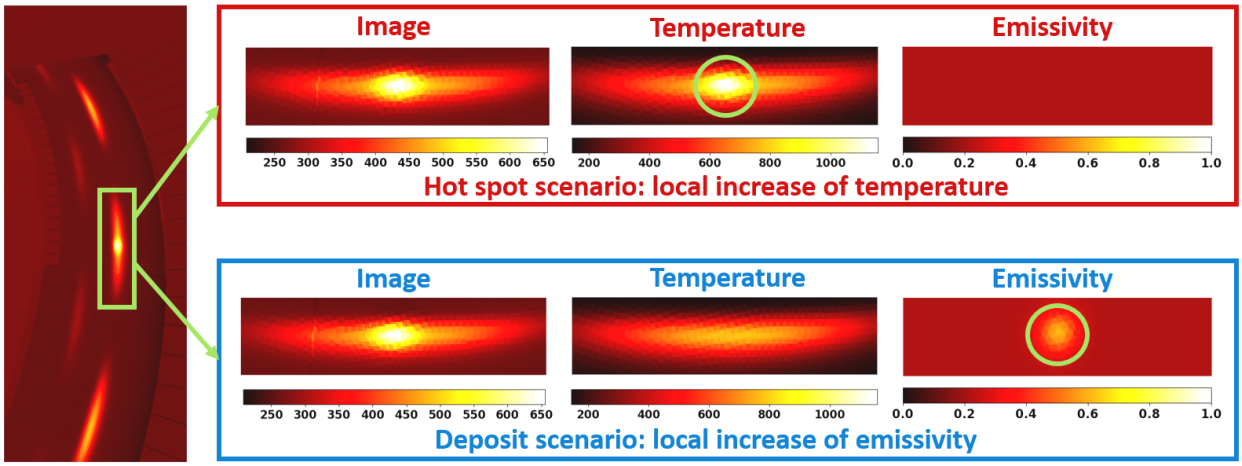


FIGURE V.5. – Exemple d'image présente dans l'ensemble de données simulées. Les images (montrées en température de corps noir apparente) simulent une caméra se focalisant sur le divertor inférieur du tokamak WEST. Un point spécifique sur le divertor peut présenter une augmentation localisée de température ou d'émissivité, les deux conduisant à une radiance émise identique. L'image ne contient ainsi pas assez d'informations pour déterminer ce qui cause une radiance émise plus élevée.

L'objectif est d'entraîner le générateur du cWGAN-GP à produire des cartes de température prédites lorsqu'on lui fournit une image d'entrée, suivant la bonne distribution de probabilité.

V.4.2. Architectures et apprentissage

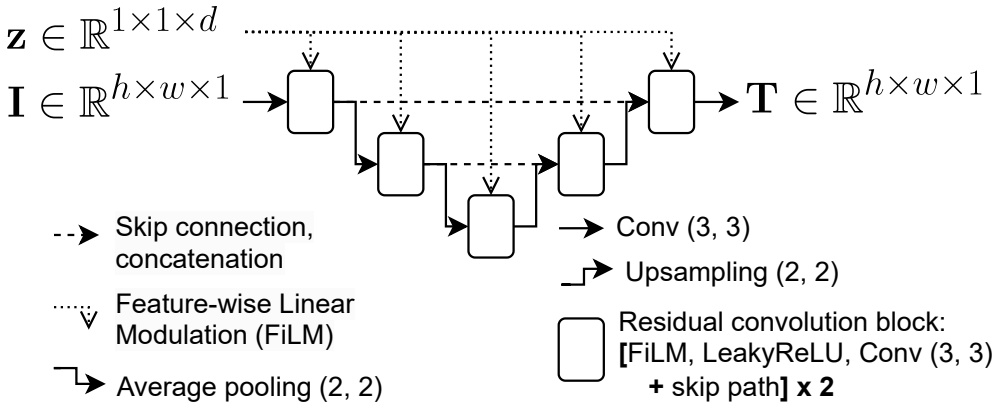


FIGURE V.6. – Représentation schématique du U-Net résiduel conditionnel utilisé comme générateur. Un vecteur latent est injecté à chaque étape de convolution en modifiant les canaux du tenseur.

Pour le modèle du générateur, une architecture de U-Net conditionnel, inspirée de [Ray+22], a été implémentée. La Figure V.6 fournit une représentation visuelle de l'architecture. Cette architecture reprend le U-Net standard, et ajoute une entrée supplémentaire, ici de dimension

4 : le vecteur latent $\mathbf{z}$. Ce vecteur latent est intégré dans le réseau en utilisant une couche de *Feature-wise Linear Modulation* (FiLM) [Per+18]. Comme montré en équation (V.22), la couche FiLM modifie son tenseur d'entrée $\mathbf{F}$ en effectuant une transformation affine canal par canal, déterminée par des couches entraînables $\boldsymbol{\alpha}$ et $\boldsymbol{\beta}$ prenant le vecteur latent $\mathbf{z}$ en entrée.

$$FiLM(\mathbf{F},\mathbf{z}|\boldsymbol{\alpha},\boldsymbol{\beta})_{i,j,c} = \boldsymbol{\alpha}(\mathbf{z})_c \times \mathbf{F}_{i,j,c} + \boldsymbol{\beta}(\mathbf{z})_c \tag{V.22}$$

La couche FiLM est utilisée à chaque étape du réseau, c'est-à-dire après chaque convolution, garantissant que le vecteur latent influence à la fois les détails sémantiques et spatiaux de la sortie. Cette approche contraste avec la normalisation d'instance conditionnelle (CIN) utilisée par [Ray+22], qui, par normalisation de son entrée, perd l'information concernant l'échelle de ses valeurs, un aspect essentiel pour les prédictions quantitatives.

La structure du réseau consiste en des couches de convolution successives organisées en blocs résiduels. Chaque bloc contient deux couches FiLM, une couche de convolution et une couche d'activation, parallélisées par une *skip connection* constituée d'une simple convolution (1, 1). Le *downsampling* est effectué par *average pooling* de fenêtre (2, 2), et un *upsampling* (2, 2) avec interpolation du plus proche voisin est utilisé. Les fonctions d'activation sont des LeakyRELU avec une pente de 0,3.

En commençant avec 32 canaux dans la première couche de convolution, le nombre de canaux est doublé après deux étapes de *downsampling*, avec un total de sept étapes de *downsampling*. Cette architecture résulte en un réseau contenant environ 11,8 millions de paramètres entraînables.

L'entrée du discriminateur est une image de deux canaux, un pour l'image et l'autre pour la carte de température. Une succession de blocs de convolution résiduels et d'opérations de *max pooling* (2, 2) sont effectuées : chaque bloc résiduel est une succession de deux activations LeakyReLU et de convolution (3, 3), et d'une *skip connection* composée d'une simple convolution (1, 1). Le nombre de filtres commence à 16 pour les premières couches et est doublé après chaque *max pooling*. Après 7 opérations de *downsampling*, une couche dense de 128 neurones et une activation LeakyReLU sont utilisées avant une couche dense d'une seule unité sans activation pour obtenir une sortie scalaire non bornée. Cela conduit à un total de 19,7 millions de poids entraînables.

Le générateur et le discriminateur sont entraînés pendant 1 000 *epochs* en utilisant un ensemble d'entraînement de 30 000 images, avec une taille de *batchs* de 48. À chaque étape d'entraînement, le discriminateur est entraîné sur 4 *batchs*, tandis que le générateur sur un seul. Le poids de la perte de pénalité de gradient du discriminateur, λ_{GP}, est fixée à 10. L'optimiseur Adam est utilisé pour mettre à jour les poids des deux modèles, avec des paramètres fixés à 0,5 pour β_1 et 0,9 pour β_2, avec un taux d'apprentissage de 10^{-4}.

V.4.3. Résultats

La performance du modèle est évaluée sur un ensemble de test composé de 10 000 éléments. Pour chaque image de test, le générateur entraîné est alimenté par 128 vecteurs latents différents pour produire un ensemble de 128 prédictions. Ces cartes de température prédites sont agrégées pour former une carte moyenne ($\boldsymbol{\mu}_T$) et une carte d'écart type ($\boldsymbol{\sigma}_T$). Cette section montre que

chaque σ_T prédit par pixel peut être utilisé pour construire un intervalle de confiance fiable pour la température réelle (T_{real}).

Un exemple de prédictions $\boldsymbol{\mu}_T$ et $\boldsymbol{\sigma}_T$ est montré dans la Figure V.7 pour une image de l'ensemble de test. Le $\boldsymbol{\sigma}_T$ prédit peut identifier le point local de plus grande incertitude, et approximer (à un certain facteur d'échelle) l'erreur absolue faite lors de l'utilisation de $\boldsymbol{\mu}_T$ comme prédiction de température.

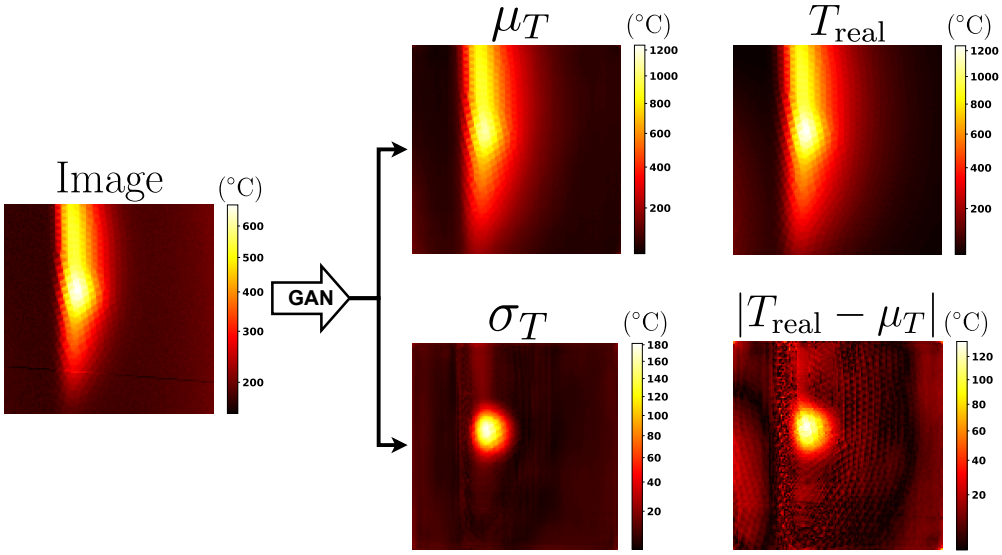


FIGURE V.7. – Exemple de prédiction faite par le cWGAN-GP sur une image de l'ensemble de test, comparée à sa carte de température réelle associée ($\mathbf{T}_{\text{real}}$). Un ensemble de 128 prédictions est généré et agrégé pour calculer une carte de température moyenne ($\boldsymbol{\mu}_T$) et une carte d'écart type ($\boldsymbol{\sigma}_T$). Le $\boldsymbol{\sigma}_T$ prédit peut identifier le point local de plus grande incertitude et approximer (à un certain facteur d'échelle) l'erreur absolue faite lors de l'utilisation de $\boldsymbol{\mu}_T$ comme prédiction de température.

Pour évaluer les résultats quantitatifs, les prédictions ont été séparées en deux catégories : celles qui sont à l'intérieur ou à l'extérieur du point le plus incertain (nommé "point mal posé"), qui peut avoir une augmentation de température ou d'émissivité. Ceci est réalisé en suivant l'emplacement du point mal posé dans chaque image et en utilisant une fenêtre de pixels de 15x15 centrée autour de celui-ci pour séparer les prédictions.

La Figure V.8 confirme que le modèle prédit une plus grande incertitude au niveau du point mal posé par rapport aux autres zones, en montrant l'écart type relatif moyen prédit (μ_T/σ_T) en fonction de la température moyenne prédite (μ_T). Le Tableau V.3 compare les erreurs globales au σ_T prédit, qui semble être une bonne estimation de l'erreur.

De plus, on montre l'applicabilité de la construction d'intervalles de confiance pour la température réelle (T_{real}) en utilisant la température moyenne prédite (μ_T) et l'écart type (σ_T). La Figure V.9 affiche la probabilité que T_{real} soit à moins de $k\sigma_T$ de μ_T, pour les prédictions à l'intérieur et à l'extérieur du point mal posé. La correspondance étroite des deux courbes suggère que σ_T, quelle que soit son ampleur, peut être utilisé de manière fiable pour créer des intervalles de confiance. Spécifiquement pour notre tâche, les courbes indiquent une probabilité de 95 % que la température réelle soit dans les 3 σ_T de μ_T, plus de 80 % pour 2 σ_T et plus de 50 % pour 1 σ_T.

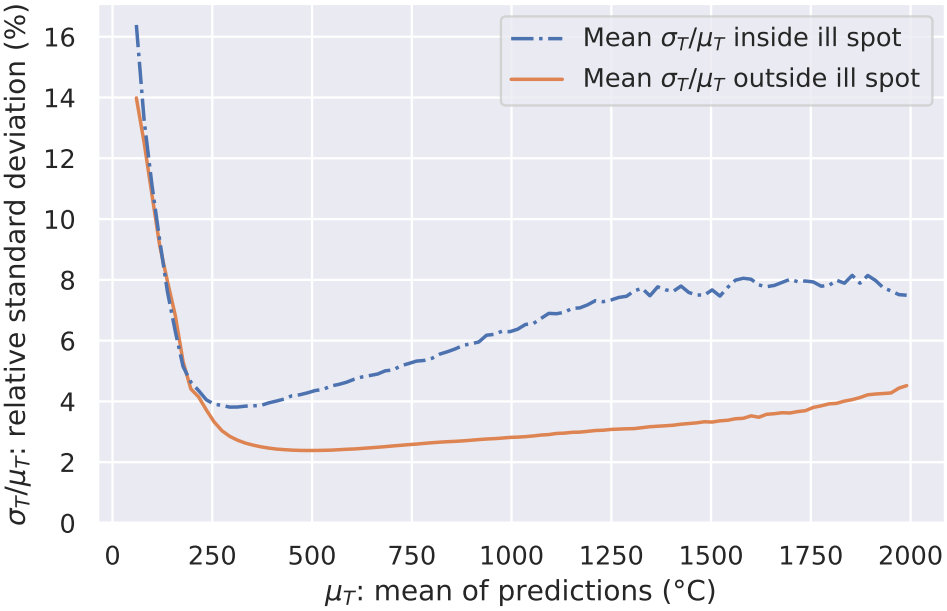


FIGURE V.8. – Écarts types relatifs moyens des prédictions du cWGAN-GP en fonction de la température prédite sur l'ensemble de test. Les prédictions sont séparées en celles appartenant au point mal posé ou en dehors du point. Le réseau a un plus grand écart type dans ses prédictions pour le point mal posé, indiquant qu'il est capable de repérer la zone de plus grande incertitude.

Métrique	Sur le point mal posé	Hors du point mal posé
Moyenne de $\left\|\mu_T - T_{\text{real}}\right\|$	29,6 °C	12,0 °C
Moyenne de σ_T prédit	30,7 °C	10,9 °C
Moyenne de $\left\|\mu_T - T_{\text{real}}\right\| / T_{\text{real}}$	5,9 %	6,8 %
Moyenne de σ_T / μ_T	5,8 %	6,5 %

TABLEAU V.3. – Comparaison de deux métriques d'erreur sur l'ensemble de test avec les écarts moyens prédits.

Ces résultats démontrent la capacité du modèle à générer des incertitudes pertinentes après entraînement avec un ensemble de données synthétiques approprié. Les intervalles de confiance calculés à partir des μ_T et σ_T prédits, pourraient potentiellement informer les processus de prise de décision dans un cadre de protection de la machine.

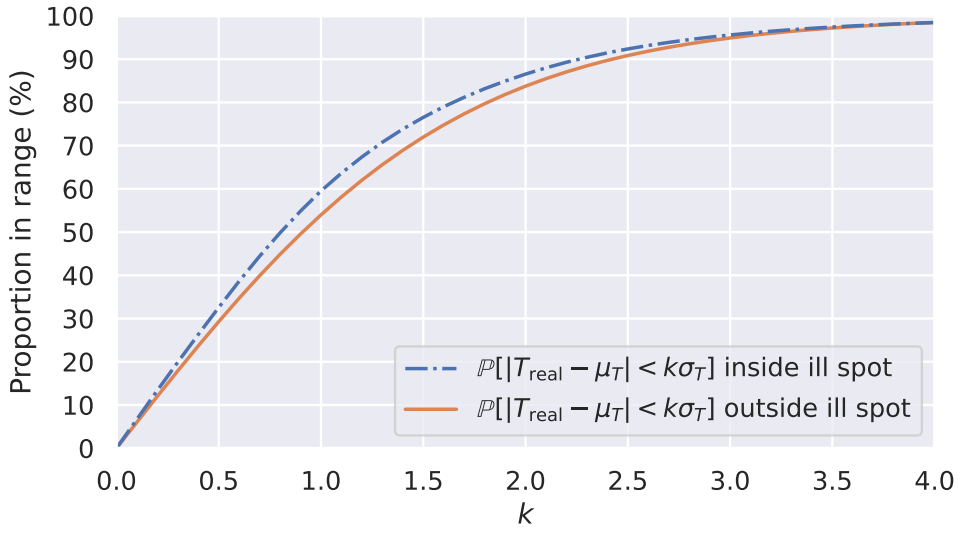


FIGURE V.9. – Proportion des températures réelles de l'ensemble de test qui sont à moins de $k\sigma_T$ de μ_T, en fonction de k, avec μ_T et σ_T étant la moyenne et l'écart type des prédictions. Les prédictions sont séparées en celles appartenant au point mal posé ou en dehors du point. La similitude des courbes indique que l'écart type prédit peut être utilisé comme un bon estimateur d'intervalle de confiance, quelle que soit sa valeur. Par exemple, environ 95 % des valeurs réelles sont à moins de 3 écarts types des prédictions moyennes.

Conclusion V

Dans ce chapitre, un paradigme probabiliste pour la résolution de problèmes inverses a été exploré. Contrairement aux approches déterministes, cette méthode permet de gérer les incertitudes de mesure et les situations où les données disponibles ne contiennent pas suffisamment d'informations pour fournir une solution unique. Cela peut se produire, par exemple, lors de l'estimation des températures où une augmentation locale de la température ou de l'émissivité peut être indiscernable.

L'approche adoptée repose sur un modèle probabiliste capable de reproduire la distribution des données d'entraînement et de fournir une plage d'incertitudes sur la solution (émissivité et/ou température). Pour cela, un modèle génératif de type *Conditional Wasserstein GAN with Gradient Penalty* (cWGAN-GP) a été développé, entraîné et évalué sur des images infrarouges simulées. Ce modèle a démontré sa capacité à générer une distribution de probabilité de température fiable, même dans des scénarios où la distribution exacte est inconnue ou en cas d'incertitude extrême. Dans le cadre étudié, la température moyenne de surface est prédite avec une erreur moyenne de 6 % sur 10 000 images de test simulées, avec des températures de surface comprises entre 50 et 2 000 °C. En moyenne, l'intervalle de confiance à 95 % sur la prédiction de température représente 20 % de la valeur estimée. Ces résultats, offrant un bon intervalle de confiance autour de la prédiction de température, pourraient constituer une information cruciale pour la prise de décision et l'amélioration de la sécurité des opérations des tokamaks.

Un aspect significatif de cette méthode est son adaptabilité automatique à la complexité sous-jacente du problème inverse à résoudre. Bien que cette flexibilité soit avantageuse, elle dépend également de la représentativité du jeu de données d'entraînement, à partir duquel la distribution *a posteriori* bayésienne de la solution au problème inverse est formée. En l'absence de vérité terrain dans les scénarios réels, le réalisme et la représentativité des données simulées utilisées pour la phase d'apprentissage est crucial. Cela nécessite la génération d'un vaste ensemble de données d'entraînement couvrant les scénarios possibles rencontrés lors des opérations des tokamaks, tout en limitant la gamme de variation des paramètres d'entrée pour éviter d'introduire trop d'incertitude. Cette étape peut inclure la caractérisation des propriétés optiques des matériaux en laboratoire et la construction d'un jeu de données synthétique dont la fréquence d'apparition des différents événements imite au mieux la base de données expérimentale.

En conclusion, les résultats obtenus avec le modèle probabiliste ont démontré son potentiel pour l'estimation des températures de surface et de son incertitude associée, dans le cadre de scénarios simulés. Les perspectives suivantes détailleront les développements nécessaires et les futures directions de recherche pour étendre cette approche au traitement des caméras infrarouges expérimentales.

Conclusion générale

Bilan

L'inversion par réseau de neurones pour la thermographie infrarouge

La fusion thermonucléaire se présente comme une source d'énergie prometteuse pour le futur, mais sa viabilité repose sur la résolution de nombreux défis techniques. Parmi ceux-ci, la protection des composants internes des tokamaks, soumis à des flux de chaleur intenses, est cruciale. Pour cela, la surveillance en temps réel des parois est réalisée par imagerie infrarouge.

La thermographie infrarouge, technique non intrusive permettant de contrôler la température de surface des composants du tokamak, est essentielle pour assurer la sureté de fonctionnement, notamment dans des zones critiques comme le divertor où les températures peuvent atteindre plusieurs milliers de degrés Celsius. Actuellement, la protection temps réel des surfaces dans les tokamaks tels que WEST est effectuée via des seuils prédéfinis sur les valeurs des pixels pour déclencher l'asservissement du système ou l'arrêt du plasma en cas de dépassement des limites de sécurité. Cependant, en raison des effets de réflexion sur les surfaces métalliques et de la variabilité de l'émissivité des surfaces, il existe un risque significatif d'interprétation erronée des données, qui peut conduire soit à l'arrêt injustifié du plasma, soit à la sous-estimation de la température des composants, provoquant des dommages potentiels.

Pour surmonter ces limitations, le travail de cette thèse a cherché à traiter numériquement les images infrarouges afin d'estimer quantitativement la température réelle des surfaces. Cependant, dans un tokamak fait de parois métalliques chaudes, l'estimation à partir d'images infrarouges des températures de parois, à la fois émettrices et réflectrices, constitue un problème inverse complexe. Ce problème nécessite la prise en compte de nombreux facteurs tels que les propriétés d'émissivité et de réflectivité des surfaces, variables spatialement et au cours du temps, ainsi que la géométrie de la scène et le placement de la caméra. De plus, le simple flux récolté par un pixel de la caméra ne suffit pas à déterminer la température de la surface visible, car le flux réfléchi additionnel dépend de l'intégralité de la scène. Pour résoudre ce problème, une approche mêlant simulation numérique et apprentissage automatique a été explorée : un jeu de données simulées est généré par lancer de rayons afin de réaliser l'apprentissage de réseaux de neurones avec des données synthétiques annotées.

Une première contribution majeure a été le développement d'un modèle de lancer de rayons infrarouge par Monte Carlo accéléré par GPU, capable de générer des images synthétiques réalistes prenant en compte les propriétés complexes de réflexion des matériaux. Ce modèle a permis de reproduire les phénomènes observés par les caméras infrarouges dans les tokamaks. Le logiciel développé, tirant parti de la parallélisation massive qu'offrent les GPU modernes, a permis de simuler efficacement des scènes détaillées de grande taille, comprenant plusieurs millions de triangles, et avec des réductions drastiques du temps de calcul en comparaison au

parallélisme CPU. Ce temps de calcul réduit a permis de créer une large base de données pour l'entraînement des algorithmes d'apprentissage automatique.

Une seconde contribution a été l'application de réseaux de neurones convolutifs pour l'estimation des températures de surface à partir des images infrarouges simulées. Des résultats prometteurs sont obtenus avec un U-Net appliqué sur un prototype numérique du tokamak WEST. Sans fournir en entrée les propriétés réfléchissantes de la surface, le réseau estime la température à partir des images simulées avec une erreur globale, de l'ordre de 6 % pour des images de test standard et de 10 % pour des images fortement bruitées dans les spécifications d'ITER. Dans le cas où sont simulés et testés des profils de températures non appris, le réseau se montre encore relativement robuste même si, comme attendu, les erreurs sont plus importantes. Pour améliorer la précision, il sera aussi important de disposer d'une base de données simulées couvrant le maximum de scénarios plausibles.

Les modèles déterministes restent toutefois limités dans le cas de problème inverse mal posé, comme c'est le cas dans l'estimation des températures, où une augmentation de la radiance collectée par la caméra peut être due dans certains cas à une modification de l'état de surface (émissivité) ou une augmentation de la température. Dans ce cas, sans information suffisante pour discriminer une unique solution, un modèle déterministe tendra à renvoyer une moyenne des possibilités. Cette limite a alors motivé l'exploration de modèles non déterministes, afin de déterminer l'incertitude sur chaque prédiction.

Une troisième contribution concerne donc l'utilisation de modèles génératifs bayésiens, tels que le cWGAN-GP. Cette approche probabiliste permet de gérer les incertitudes inhérentes aux problèmes inverses mal posés, fournissant des prédictions de température avec un intervalle de confiance associé. Les résultats obtenus ont montré que la température moyenne de surface pouvait être prédite avec une erreur moyenne de 6 % sur des images de test simulées. De plus, l'intervalle de confiance à 95 % sur la prédiction de température représentait 20 % de la valeur estimée en moyenne. Ces résultats, offrant un intervalle de confiance autour de la prédiction de température, sont importants pour aider à une prise de décision éclairée dans une optique de sécurité des opérations des tokamaks. Un aspect significatif de cette méthode est son adaptabilité automatique à la complexité sous-jacente du problème inverse à résoudre. Bien que cette flexibilité soit avantageuse, elle dépend de la représentativité du jeu de données d'entraînement, à partir duquel la distribution *a posteriori* bayésienne de la solution au problème inverse est apprise. En l'absence de vérité terrain dans les scénarios réels, le réalisme et la représentativité des données simulées utilisées pour la phase d'apprentissage sont cruciaux. Cela nécessite la génération d'un vaste ensemble de données d'entraînement couvrant les scénarios possibles rencontrés lors des opérations des tokamaks, tout en limitant la gamme de variation des paramètres d'entrée pour éviter d'introduire trop d'incertitude.

En conclusion, les contributions de cette thèse ouvrent de nouvelles perspectives pour la surveillance thermique des tokamaks, avec des applications potentielles pour ITER et les futures générations de réacteurs de fusion. Les développements réalisés en modélisation et en apprentissage automatique et les résultats préliminaires sur données simulées sont un pas important en vue d'une mesure de température fiable dans les environnements extrêmes des tokamaks. Les prochaines étapes consisteront à valider expérimentalement ces méthodes et à ajuster les modèles aux conditions réelles.

Autres bénéfices de ces travaux

Les outils développés durant cette thèse, en particulier autour de l'accélération du lancer de rayons pour la simulation d'images infrarouges, ont également contribué à l'avancée de travaux visant à améliorer l'interprétation et la compréhension des données expérimentales sur des machines en opération telles que WEST [Tic+24]. Ces développements ont également contribué à la prédiction des performances des systèmes en cours de conception, tels que le système grand angle d'imagerie infrarouge d'ITER ou de DTT (*Divertor Tokamak Test*) [Aum+24].

Enfin, ces travaux sont valorisés en intégrant une partie des outils développés dans des plateformes logicielles existantes : CIVA [Cal+06], un logiciel dédié contrôle non destructif, ainsi que dans IMAS [Imb+15], la plateforme de modélisation et d'analyse d'ITER.

Perspectives

Validation expérimentale par le banc de test MAGRYT

Pour pallier le manque de données annotées disponibles pour le tokamak WEST, l'équipe du GP3 de l'IRFM met en place le banc de test MAGRYT. Ce banc de test a pour objectif de recréer les conditions d'observation des surfaces par une caméra infrarouge dans un tokamak, mais au sein d'un environnement où les différentes températures et émissivités des composants sont contrôlées. Un tel environnement permettra de valider la méthode en comparant les prédictions de températures faites par le modèle à celles mesurées.

L'environnement du tokamak est reproduit en plaçant un échantillon métallique dans une chambre à vide. Cet échantillon est chauffé jusqu'à 1000 °C grâce à un laser. L'échantillon est observé par une caméra infrarouge à travers un hublot et un tube métallique, lui-même chauffé jusqu'à 200 °C, ce qui génère une source parasite à la fois émissive et réfléchissante la lumière de l'échantillon chaud. Des thermocouples sont placés à différents endroits afin de connaître la température de tous les composants. Les émissivités et BRDF des composants sont mesurées en amont par des installations dédiées (goniomètres, sphère intégrante, spectromètre).

Ce dispositif permet de reproduire le cœur du problème : estimer la température de surfaces métalliques chaudes au sein d'un environnement métallique lui-même émissif et réflectif.

En appliquant la même démarche que celle présentée dans cette thèse, il sera possible de construire un jeu de données synthétiques reproduisant l'environnement, de réaliser un apprentissage sur données simulées et de comparer les températures prédites par le réseau de neurones avec celles fournies par les mesures embarquées comme les thermocouples. Cette démarche expérimentale en laboratoire est une étape importante pour valider les modèles développés avant tout déploiement sur les machines de fusion en opération.

Perspectives d'améliorations

À ce jour, trois principaux axes d'amélioration sont discutés et détaillés dans les prochaines sections : la création d'une nouvelle base de données étendue à différents scénarios plasmas, l'exploration de nouvelles architectures plus adaptés au traitement de données expérimentales

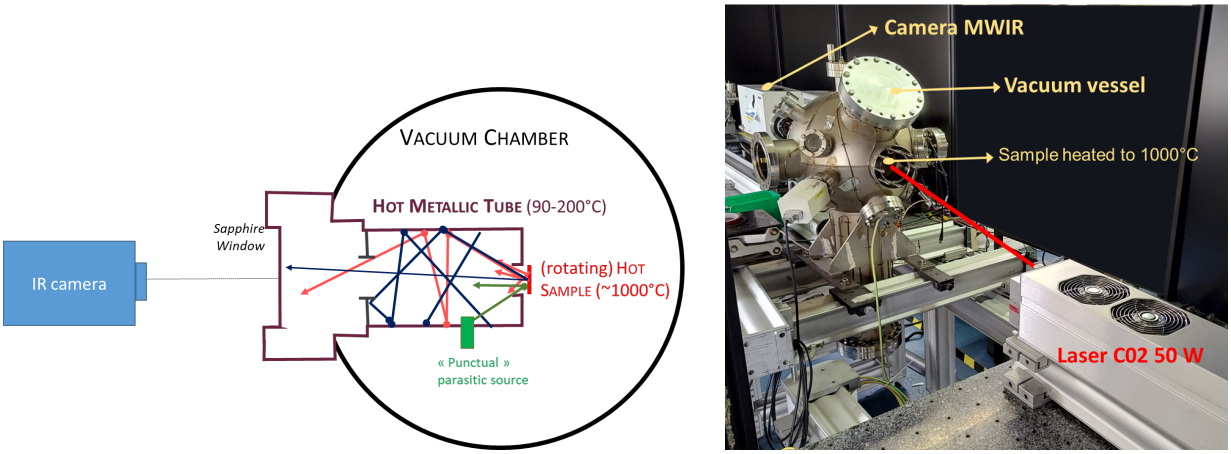


FIGURE V.10. – Banc de test MAGRYT, visant à reproduire les conditions d'observation d'une cible dans un environnement émissif et réflectif. **Gauche –** Schéma montrant les différentes sources observées par une caméra dans la chambre à vide. **Droite –** Photo du banc expérimental.

(avec également l'optique du temps réel), et l'intégration de données multicapteurs dans la phase d'apprentissage.

Une base de données étendue à différents scénarios plasmas

Dans l'optique de traiter des données expérimentales, il est essentiel de créer une base de données réaliste et représentative pour obtenir des estimations de température fiables via les méthodes d'inversion.

Le jeu de données synthétiques doit certes être suffisamment vaste pour couvrir les scénarios possibles, mais être suffisamment restreint pour réduire l'incertitude sur les paramètres estimés. L'incertitude de certains paramètres de la scène, tels que les propriétés de BRDF des matériaux, peut notamment être réduite en réalisant des mesures précises en laboratoire.

De plus, le réalisme des données générées peut être accru de par plusieurs moyens. Par exemple, l'utilisation de modèles BRDF par micro-facettes, détaillées en section III.4.2.3, permettrait d'obtenir des comportements de réflexion plus proches des surfaces métalliques réelles. Le réalisme des simulations pourrait également être amélioré en utilisant des profils de température issus d'une chaîne complète de simulation, allant de l'équilibre magnétique, à la détermination des flux de chaleur sur les surfaces, jusqu'au calcul des températures de surface. Cette approche permettrait d'accéder à des paramètres de plasma déjà enregistrés dans la base de données des expériences de WEST, facilitant la création de profils de température avec des probabilités d'apparition réalistes, correspondant aux différents scénarios expérimentaux précédents.

Enfin, une accélération conséquente du modèle direct est possible. Une amélioration du modèle, exploitant le lancer de rayon matériel des cartes Nvidia RTX, est en cours de développement au sein de l'équipe. Cette avancée permet de réaliser électroniquement les calculs d'intersection rayon-triangle via la bibliothèque OptiX, réduisant drastiquement le temps de calcul et permettant un temps de génération d'image de l'ordre de la seconde. Cette optimisation ouvre la voie à d'autres applications du modèle direct, telle que la génération de vidéos, ou l'utilisa-

tion du modèle lors des phases d'apprentissage des réseaux de neurones, dans un paradigme d'apprentissage autosupervisé.

Exploration de nouveaux modèles d'apprentissage automatique

Suivant le besoin de traitement des données en temps réel ou non, différentes améliorations des modèles d'apprentissage sont envisageables :

— Optimisation et réduction du nombre d'hyper-paramètres du réseau pour diminuer le nombre de poids et ainsi accélérer le temps d'inférence. Des techniques telles que la réduction du nombre de bits ou le *pruning* pourraient être explorées.
— Utilisation des réseaux de type *Visual Transformers* [Liu+24] permettant un traitement de l'information non local et établissant des corrélations entre différents *patchs* de l'image. C'est une propriété qui pourrait s'avérer pertinente pour les images infrarouges, présentant des motifs de réflexions éloignés de la source de chaleur.
— Utilisation de modèles de diffusion [Chu+22 ; Sah+23] très efficace en génération d'images et en résolution de problème inverse. L'inférence d'une image avec ce type de modèle reste cependant coûteuse en temps de calcul à cause de la succession d'étapes de bruitage et de débruitage lors de l'inférence.
— Utilisation de modèles directs différentiables pour un apprentissage autosupervisé. Bien que cette méthode soit classiquement appliquée sur des modèles déterministes, elle offre l'avantage de pouvoir entraîner le réseau sur des images réelles non annotées, constituant ainsi une étape finale de *fine-tuning*. Il serait ainsi envisageable d'appliquer cette méthode d'apprentissage autosupervisé au modèle génératif pour qu'il soit capable de reproduire correctement la distribution probabiliste des températures, tout en ayant légèrement ajusté ses poids au traitement d'images expérimentales.

Utilisation des données multicapteurs

Une autre amélioration envisageable consisterait à utiliser les données des autres caméras ou diagnostics présents dans la machine.

L'utilisation de données multivues, provenant de plusieurs caméras infrarouges, permettrait de réduire les incertitudes et les cas de problèmes mal posés en combinant différentes perspectives sur les mêmes régions d'intérêt.

Il serait également intéressant d'inclure des données multimodales, telles que des paramètres de commande du plasma (par exemple, les puissances appliquées). Cela impliquerait de réaliser des simulations intégrant ces paramètres.

Enfin, l'intégration de la temporalité pourrait réduire les problèmes mal posés. En utilisant une architecture de réseau capable de traiter des vidéos, c'est-à-dire plusieurs images consécutives, le réseau pourrait extraire des corrélations temporelles pour mieux identifier les réflexions et les évolutions des zones chaudes. Par exemple, l'évolution d'un point chaud et de son motif de réflexion seraient fortement corrélées, tandis que des zones chaudes provenant de composants différents pourraient avoir des caractéristiques temporelles distinctes. Pour cela, il serait nécessaire de simuler des vidéos en créant des profils spatio-temporels de température réalistes, ce qui pourrait être réalisé via une chaîne de simulation complexe, intégrant les flux calculés

depuis l'équilibre du plasma et simulant la diffusion de la chaleur au cours du temps dans les matériaux.

Bibliographie

[Aba+16] Martín ABADI, Paul BARHAM, Jianmin CHEN, Zhifeng CHEN, Andy DAVIS, Jeffrey DEAN, Matthieu DEVIN, Sanjay GHEMAWAT, Geoffrey IRVING, Michael ISARD et al. « TensorFlow : a system for Large-Scale machine learning ». *12th USENIX symposium on operating systems design and implementation (OSDI 16)*. 2016, p. 265-283 (cf. p. 84).

[AO18] Jonas ADLER et Ozan OKTEM. « Deep bayesian inversion ». *arXiv preprint arXiv :1811.05910* (2018) (cf. p. 112).

[ACB17] Martin ARJOVSKY, Soumith CHINTALA et Léon BOTTOU. « Wasserstein Generative Adversarial Networks ». *Proceedings of the 34th International Conference on Machine Learning*. ISSN : 2640-3498. PMLR, juill. 2017, p. 214-223 (cf. p. 111).

[Aum+21] M-H. AUMEUNIER, J. GERARDIN, C. TALATIZI, M. LE BOHEC, M. BEN YAALA, L. MAROT, T. LOARER, R. MITTEAU, J. GASPAR, F. RIGOLLET, X. COURTOIS, M. HOURY, A. HERRMANN et M. FAITSCH. « Infrared thermography in metallic environments of WEST and ASDEX Upgrade ». *Nuclear Materials and Energy* 26 (mars 2021), p. 100879. ISSN : 2352-1791. DOI : 10.1016/j.nme.2020.100879 (cf. p. 24, 42).

[Aum+24] M.-H. AUMEUNIER, A. JUVEN, J. GERARDIN, C-M. B. CISSE, S. PAMELA, R. MIORELLI, C. REBOUD, F. RETAILLEAU, J. MAROT, F. RIGOLLET et L. MAROT. « Surface temperature measurement from infrared synthetic diagnostic in preparation for ITER operations ». *Nuclear Fusion* 64.8 (2024), p. 086044. DOI : 10.1088/1741-4326/ad5a1f (cf. p. 3, 40, 73, 126).

[Aum+17] M. H. AUMEUNIER, M. KOČAN, R. REICHLE et E. GAUTHIER. « Impact of reflections on the divertor and first wall temperature measurements from the ITER infrared imaging system ». *Nuclear Materials and Energy*. Proceedings of the 22nd International Conference on Plasma Surface Interactions 2016, 22nd PSI 12 (août 2017), p. 1265-1269. ISSN : 2352-1791. DOI : 10.1016/j.nme.2017.02.014 (cf. p. 42).

[Aum+22] Marie-Hélène AUMEUNIER, Mickaël LE BOHEC, Romain BRUNET, A. JUVEN, M. ADEL, X. ARTUSI, R. MIORELLI, C. REBOUD et F. RIGOLLET. « Development of inverse methods for infrared thermography in fusion devices ». *Nuclear Materials and Energy* 33 (oct. 2022), p. 101231. ISSN : 2352-1791. DOI : 10.1016/j.nme.2022.101231 (cf. p. 3, 30).

[BB71] WR BLEVIN et WJ BROWN. « A precise measurement of the Stefan-Boltzmann constant ». *Metrologia* 7.1 (1971), p. 15 (cf. p. 47).

[Bro66] D. BROWN. « Decentering Distortion of Lenses ». *Photogrammetric Engineering* (1966), p. 19 (cf. p. 45).

[Bru+23] Romain BRUNET, Marie-Hélène AUMEUNIER, Roberto MIORELLI, Alexis JUVEN, C. REBOUD, X. ARTUSI et M. Le BOHEC. « Infrared measurement synthetic database for inverse thermography model based on deep learning ». *Fusion Engineering and Design* 192 (juill. 2023), p. 113598. ISSN : 0920-3796. DOI : 10.1016/j.fusengdes.2023.113598 (cf. p. 3, 30).

[Buc+22] J. BUCALOSSI, J. ACHARD, O. AGULLO, T. ALARCON, L. ALLEGRETTI, H. ANCHER, G. ANTAR, S. ANTUSCH, V. ANZALLO, C. ARNAS, D. ARRANGER, J.F. ARTAUD, M.H. AUMEUNIER, S.G. BAEK, X. BAI, J. BALBIN, C. BALORIN, T. BARBUI, A. BARBUTI, J. BARLERIN et al. « Operating a full tungsten actively cooled tokamak : overview of WEST first phase of operation ». *Nuclear Fusion* 62.4 (avr. 2022), p. 042007. ISSN : 0029-5515, 1741-4326. DOI : 10.1088/1741-4326/ac2525 (cf. p. 16).

[Buc+14] J. BUCALOSSI, M. MISSIRLIAN, P. MOREAU, F. SAMAILLE, E. TSITRONE, D. VAN HOUTTE, T. BATAL, C. BOURDELLE, M. CHANTANT, Y. CORRE, X. COURTOIS, L. DELPECH, L. DOCEUL, D. DOUAI, H. DOUGNAC, F. FAÏSSE, C. FENZI, F. FERLAY, M. FIRDAOUSS, L. GARGIULO et al. « The WEST project : Testing ITER divertor high heat flux component technology in a steady state tokamak environment ». *Fusion Engineering and Design* 89.7 (2014). Proceedings of the 11th International Symposium on Fusion Nuclear Technology-11 (ISFNT-11) Barcelona, Spain, 15-20 September, 2013, p. 907-912. ISSN : 0920-3796. DOI : https://doi.org/10.1016/j.fusengdes.2014.01.062 (cf. p. 16).

[Caf98] Russel E. CAFLISCH. « Monte Carlo and quasi-Monte Carlo methods ». *Acta Numerica* 7 (1998), 1–49. DOI : 10.1017/S0962492900002804 (cf. p. 55).

[Cal+06] Pierre CALMON, Steve MAHAUT, Sylvain CHATILLON et R RAILLON. « CIVA : An expertise platform for simulation and processing NDT data ». *Ultrasonics* 44 (2006), e975-e979 (cf. p. 126).

[Car+19] M. CARR, A. MEAKINS, S. A. SILBURN, J. KARHUNEN, M. BERNERT, C. BOWMAN, A. CALLARELLI, P. CARVALHO, C. GIROUD, J. R. HARRISON, S. S. HENDERSON, A. HUBER, B. LIPSCHULTZ, T. LUNT, D. MOULTON, F. REIMOLD, ASDEX UPGRADE TEAM, JET CONTRIBUTORS, MAST-UPGRADE TEAM et EUROFUSION MST1 TEAM. « Physically principled reflection models applied to filtered camera imaging inversions in metal walled fusion machines ». *Review of Scientific Instruments* 90.4 (avr. 2019), p. 043504. ISSN : 0034-6748, 1089-7623. DOI : 10.1063/1.5092781 (cf. p. 42).

[Chu+22] Hyungjin CHUNG, Jeongsol KIM, Michael T MCCANN, Marc L KLASKY et Jong Chul YE. « Diffusion posterior sampling for general noisy inverse problems ». *arXiv preprint arXiv :2209.14687* (2022) (cf. p. 110, 128).

[CT82] Robert L COOK et Kenneth E. TORRANCE. « A reflectance model for computer graphics ». *ACM Transactions on Graphics (ToG)* 1.1 (1982), p. 7-24 (cf. p. 53).

[Cou+19] X. COURTOIS, MH. AUMEUNIER, C. BALORIN, J. B. MIGOZZI, M. HOURY, K. BLANCKAERT, Y. MOUDDEN, C. POCHEAU, A. SAILLE, E. HUGOT, M. MARCOS et S. VIVES. « Full coverage infrared thermography diagnostic for WEST machine protec-

tion ». *Fusion Engineering and Design*. SI :SOFT-30 146 (sept. 2019), p. 2015-2020. ISSN : 0920-3796. DOI : 10.1016/j.fusengdes.2019.03.090 (cf. p. 47).

[DM98] Leonardo DAGUM et Ramesh MENON. « OpenMP : an industry standard API for shared-memory programming ». *IEEE computational science and engineering* 5.1 (1998), p. 46-55 (cf. p. 61).

[DHS11] John DUCHI, Elad HAZAN et Yoram SINGER. « Adaptive subgradient methods for online learning and stochastic optimization. » *Journal of machine learning research* 12.7 (2011) (cf. p. 78).

[Eic+11] T. EICH, B. SIEGLIN, A. SCARABOSIO, W. FUNDAMENSKI, R. J. GOLDSTON, A. HERRMANN et ASDEX UPGRADE TEAM. « Inter-ELM Power Decay Length for JET and ASDEX Upgrade : Measurement and Comparison with Heuristic Drift-Based Model ». *Physical Review Letters* 107.21 (nov. 2011), p. 215001. ISSN : 0031-9007, 1079-7114. DOI : 10.1103/PhysRevLett.107.215001 (cf. p. 66).

[Fir+13] M FIRDAOUSS, V RICCARDO, V MARTIN, G ARNOUX, C REUX, Jet efda CONTRIBUTORS et al. « Modelling of power deposition on the JET ITER like wall using the code PFCFLux ». *Journal of Nuclear Materials* 438 (2013), S536-S539 (cf. p. 63).

[Fit01] Andrew W FITZGIBBON. « Simultaneous linear estimation of multiple view geometry and lens distortion ». *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. T. 1. IEEE. 2001, p. I-I (cf. p. 45).

[FS05] Tim FOLEY et Jeremy SUGERMAN. « KD-tree acceleration structures for a GPU raytracer ». *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*. 2005, p. 15-22 (cf. p. 59).

[Gas+21] J GASPAR, Yann CORRE, Nicolas FEDORCZAK, JP GUNN, Clarisse BOURDELLE, S BREZINSEK, Jérôme BUCALOSSI, Nicolas CHANET, R DEJARNAC, M FIRDAOUSS et al. « Divertor power loads and scrape off layer width in the large aspect ratio full tungsten tokamak WEST ». *Nuclear fusion* 61.9 (2021), p. 096027 (cf. p. 67).

[Gas+19] J GASPAR, C POCHEAU, Y CORRE, N EHRET, D GUILHEM, M HOURY, T LOARER, Th LOEWENHOFF, C MARTIN, C PARDANAUD et al. « Emissivity measurement of tungsten plasma facing components of the WEST tokamak ». *Fusion Engineering and Design* 149 (2019), p. 111328 (cf. p. 48).

[Goh+22] Hwan GOH, Sheroze SHERIFFDEEN, Jonathan WITTMER et Tan BUI-THANH. « Solving Bayesian Inverse Problems via Variational Autoencoders ». *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*. ISSN : 2640-3498. PMLR, avr. 2022, p. 386-425 (cf. p. 110).

[Goo+14] Ian GOODFELLOW, Jean POUGET-ABADIE, Mehdi MIRZA, Bing XU, David WARDE-FARLEY, Sherjil OZAIR, Aaron COURVILLE et Yoshua BENGIO. « Generative adversarial nets ». *Advances in neural information processing systems* 27 (2014) (cf. p. 110).

[GCT86] Donald P GREENBERG, Michael F COHEN et Kenneth E TORRANCE. « Radiosity : A method for computing global illumination ». *The Visual Computer* 2.5 (1986), p. 291-297 (cf. p. 25).

[Gul+17] Ishaan GULRAJANI, Faruk AHMED, Martin ARJOVSKY, Vincent DUMOULIN et Aaron C COURVILLE. « Improved Training of Wasserstein GANs ». *Advances in Neural Information Processing Systems*. T. 30. Curran Associates, Inc., 2017 (cf. p. 111).

[Has+09] Trevor HASTIE, Robert TIBSHIRANI, Jerome H FRIEDMAN et Jerome H FRIEDMAN. *The elements of statistical learning : data mining, inference, and prediction*. T. 2. Springer, 2009 (cf. p. 99).

[He+16] Kaiming HE, Xiangyu ZHANG, Shaoqing REN et Jian SUN. « Deep residual learning for image recognition ». *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770-778 (cf. p. 86).

[Hin+14] Aicke HINRICHS, Erich NOVAK, Mario ULLRICH et H WOŹNIAKOWSKI. « The curse of dimensionality for numerical integration of smooth functions ». *Mathematics of Computation* 83.290 (2014), p. 2853-2863 (cf. p. 55).

[HK70] Arthur E HOERL et Robert W KENNARD. « Ridge regression : Biased estimation for nonorthogonal problems ». *Technometrics* 12.1 (1970), p. 55-67 (cf. p. 27).

[Imb+15] Frédéric IMBEAUX, SD PINCHES, JB LISTER, Y BURAVAND, T CASPER, Basil DUVAL, B GUILLERMINET, Masanari HOSOKAWA, Wayne HOULBERG, Philippe HUYNH et al. « Design and first applications of the ITER integrated modelling & analysis suite ». *Nuclear Fusion* 55.12 (2015), p. 123006 (cf. p. 126).

[Iso+17] Phillip ISOLA, Jun-Yan ZHU, Tinghui ZHOU et Alexei A. EFROS. « Image-To-Image Translation With Conditional Adversarial Networks ». *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cf. p. 88).

[Jan+17] Michael JANNER, Jiajun WU, Tejas D KULKARNI, Ilker YILDIRIM et Josh TENENBAUM. « Self-Supervised Intrinsic Image Decomposition ». *Advances in Neural Information Processing Systems*. T. 30. Curran Associates, Inc., 2017 (cf. p. 32).

[Juv+22] Alexis JUVEN, Marie-Hélène AUMEUNIER, Romain BRUNET, Mickaël Le BOHEC, Moulloud ADEL, Roberto MIORELLI, Xavier ARTUSI et Christophe REBOUD. « Temperature Estimation in Fusion Devices using Machine Learning techniques on Infrared Specular Synthetic Data ». *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. Juin 2022, p. 1-5. DOI : 10.1109/IVMSP54334.2022.9816270 (cf. p. 3, 30).

[JAM24a] Alexis JUVEN, Marie-Hélène AUMEUNIER et Julien MAROT. « Generative models and simulation to assess uncertainties for tokamak infrared thermography ». *2024 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. 2024. DOI : 10.1109/MLSP58920.2024.10734728 (cf. p. 3, 103).

[JAM24b] Alexis JUVEN, Marie-Hélène AUMEUNIER et Julien MAROT. « U-Net for temperature estimation from simulated infrared images in tokamaks ». *Nuclear Materials and Energy* 38 (mars 2024), p. 101562. ISSN : 2352-1791. DOI : 10.1016/j.nme.2023.101562 (cf. p. 3, 40, 73).

[Kaj86] James T KAJIYA. « THE RENDERING EQUATION ». 20.4 (1986) (cf. p. 54).

[KB14] Diederik P KINGMA et Jimmy BA. « Adam : A method for stochastic optimization ». *arXiv preprint arXiv :1412.6980* (2014) (cf. p. 78).

[Klo+98] James T KLOSOWSKI, Martin HELD, Joseph SB MITCHELL, Henry SOWIZRAL et Karel ZIKAN. « Efficient collision detection using bounding volume hierarchies of k-DOPs ». *IEEE transactions on Visualization and Computer Graphics* 4.1 (1998), p. 21-36 (cf. p. 59).

[KSH12] Alex KRIZHEVSKY, Ilya SUTSKEVER et Geoffrey E HINTON. « ImageNet Classification with Deep Convolutional Neural Networks ». *Advances in Neural Information Processing Systems*. Sous la dir. de F. PEREIRA, C.J. BURGES, L. BOTTOU et K.Q. WEINBERGER. T. 25. Curran Associates, Inc., 2012 (cf. p. 85).

[LW98] Eric LAFORTUNE et Yves WILLEMS. « Using the Modied Phong Reflectance Model for Physically Based Rendering » (jan. 1998) (cf. p. 52).

[LW93] Eric P LAFORTUNE et Yves D WILLEMS. « Bi-directional path tracing » (1993) (cf. p. 42).

[LBLS23] Mickaël LE BOHEC, Denis LEMONNIER et Didier SAURY. « Model reduction of radiative heat transfer by a hierarchical radiosity method ». *Numerical Heat Transfer, Part A : Applications* 83.11 (2023), p. 1175-1194. DOI : 10.1080/10407782.2022.2102397 (cf. p. 28).

[Lec+98] Y. LECUN, L. BOTTOU, Y. BENGIO et P. HAFFNER. « Gradient-based learning applied to document recognition ». *Proceedings of the IEEE* 86.11 (1998), p. 2278-2324. DOI : 10.1109/5.726791 (cf. p. 85).

[LB04] Elizaveta LEVINA et Peter BICKEL. « Maximum likelihood estimation of intrinsic dimension ». *Advances in neural information processing systems* 17 (2004) (cf. p. 93).

[Li+21] Zewen LI, Fan LIU, Wenjie YANG, Shouheng PENG et Jun ZHOU. « A survey of convolutional neural networks : analysis, applications, and prospects ». *IEEE transactions on neural networks and learning systems* 33.12 (2021), p. 6999-7019 (cf. p. 85).

[Li+22] Zhen LI, Lingli WANG, Xiang HUANG, Cihui PAN et Jiaqi YANG. « PhyIR : Physics-based Inverse Rendering for Panoramic Indoor Images ». *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA : IEEE, juin 2022, p. 12703-12713. ISBN : 978-1-66546-946-3. DOI : 10.1109/CVPR52688.2022.01238 (cf. p. 31).

[LS18] Zhengqi LI et Noah SNAVELY. « Cgintrinsics : Better intrinsic image decomposition through physically-based rendering ». *Proceedings of the European conference on computer vision (ECCV)*. 2018, p. 371-387 (cf. p. 32).

[Li+20] Zhengqin LI, Mohammad SHAFIEI, Ravi RAMAMOORTHI, Kalyan SUNKAVALLI et Manmohan CHANDRAKER. « Inverse Rendering for Complex Indoor Scenes : Shape, Spatially-Varying Lighting and SVBRDF From a Single Image ». *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA : IEEE, juin 2020, p. 2472-2481. ISBN : 978-1-72817-168-5. DOI : 10.1109/CVPR42600.2020.00255 (cf. p. 31).

[Li+18] Zhengqin LI, Zexiang XU, Ravi RAMAMOORTHI, Kalyan SUNKAVALLI et Manmohan CHANDRAKER. « Learning to reconstruct shape and spatially-varying reflectance from a single image ». *ACM Trans. Graph.* 37.6 (2018). ISSN : 0730-0301. DOI : 10.1145/3272127.3275055 (cf. p. 32).

[Liu+24] Yang LIU, Yao ZHANG, Yixin WANG, Feng HOU, Jin YUAN, Jiang TIAN, Yang ZHANG, Zhongchao SHI, Jianping FAN et Zhiqiang HE. « A Survey of Visual Transformers ». *IEEE Transactions on Neural Networks and Learning Systems* 35.6 (2024), p. 7478-7498. DOI : 10.1109/TNNLS.2022.3227717 (cf. p. 128).

[Maj+18] Alexander MAJERCIK, Cyril CRASSIN, Peter SHIRLEY et Morgan MCGUIRE. « A ray-box intersection algorithm and efficient dynamic voxel rendering ». *Journal of Computer Graphics Techniques Vol* 7.3 (2018), p. 66-81 (cf. p. 60).

[MBC00] Michael D MCKAY, Richard J BECKMAN et William J CONOVER. « A comparison of three methods for selecting values of input variables in the analysis of output from a computer code ». *Technometrics* 42.1 (2000), p. 55-61 (cf. p. 36).

[Met+53] Nicholas METROPOLIS, Arianna W. ROSENBLUTH, Marshall N. ROSENBLUTH, Augusta H. TELLER et Edward TELLER. « Equation of State Calculations by Fast Computing Machines ». *The Journal of Chemical Physics* 21.6 (juin 1953). Publisher : American Institute of Physics, p. 1087-1092. ISSN : 0021-9606. DOI : 10.1063/1.1699114 (cf. p. 108).

[MU49] Nicholas METROPOLIS et S. ULAM. « The Monte Carlo Method ». *Journal of the American Statistical Association* 44.247 (1949). PMID : 18139350, p. 335-341. DOI : 10.1080/01621459.1949.10483310 (cf. p. 54).

[Mod13] M. F. MODEST. *Radiative heat transfer*. Third Edition. OCLC : ocn813855549. New York : Academic Press, 2013. ISBN : 978-0-12-386944-9 (cf. p. 51).

[MT05] Tomas MÖLLER et Ben TRUMBORE. « Fast, minimum storage ray/triangle intersection ». *ACM SIGGRAPH 2005 Courses*. 2005, 7-es (cf. p. 58).

[Mon+23] V. MONCADA, X. COURTOIS, L. DUBUS, E. GRELIER, M. HOURY, A. Puig SITJES et B. ZHANG. « Software platform for imaging diagnostic exploitation applied to edge plasma physics and real-time PFC monitoring ». *Fusion Engineering and Design* 190 (2023), p. 113528. ISSN : 0920-3796. DOI : https://doi.org/10.1016/j.fusengdes.2023.113528 (cf. p. 21).

[Nic+08] John NICKOLLS, Ian BUCK, Michael GARLAND et Kevin SKADRON. « Scalable Parallel Programming with CUDA : Is CUDA the parallel programming model that application developers have been waiting for ? » *Queue* 6.2 (mars 2008), p. 40-53. ISSN : 1542-7730. DOI : 10.1145/1365490.1365500 (cf. p. 35, 60).

[Nic65] Fred E. NICODEMUS. « Directional Reflectance and Emissivity of an Opaque Surface ». *Applied Optics* 4.7 (juill. 1965). Publisher : Optica Publishing Group, p. 767-775. ISSN : 2155-3165. DOI : 10.1364/AO.4.000767 (cf. p. 49, 51).

[ND+19] Merlin NIMIER-DAVID, Delio VICINI, Tizian ZELTNER et Wenzel JAKOB. « Mitsuba 2 : a retargetable forward and inverse renderer ». *ACM Transactions on Graphics* 38.6 (déc. 2019), p. 1-17. ISSN : 0730-0301, 1557-7368. DOI : 10.1145/3355089.3356498 (cf. p. 42).

[Pas+17] Adam PASZKE, Sam GROSS, Soumith CHINTALA, Gregory CHANAN, Edward YANG, Zachary DEVITO, Zeming LIN, Alban DESMAISON, Luca ANTIGA et Adam LERER. « Automatic differentiation in PyTorch » (oct. 2017) (cf. p. 84).

[Pas+19] Adam PASZKE, Sam GROSS, Francisco MASSA, Adam LERER, James BRADBURY, Gregory CHANAN, Trevor KILLEEN, Zeming LIN, Natalia GIMELSHEIN, Luca ANTIGA et al. « Pytorch : An imperative style, high-performance deep learning library ». *Advances in neural information processing systems* 32 (2019) (cf. p. 84).

[Pen55] Roger PENROSE. « A generalized inverse for matrices ». *Mathematical proceedings of the Cambridge philosophical society*. T. 51. 3. Cambridge University Press. 1955, p. 406-413 (cf. p. 27).

[Per+18] Ethan PEREZ, Florian STRUB, Harm de VRIES, Vincent DUMOULIN et Aaron COURVILLE. « FiLM : Visual Reasoning with a General Conditioning Layer ». *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1 (avr. 2018). Number : 1. ISSN : 2374-3468. DOI : 10.1609/aaai.v32i1.11671 (cf. p. 119).

[Per85] Ken PERLIN. « An image synthesizer ». 19.3 (1985) (cf. p. 68).

[PJH24] Matt PHARR, Wenzel JAKOB et Greg HUMPHREYS. *Physically Based Rendering : From Theory to Implementation*. T. 4. 2024 (cf. p. 56).

[Pho98] Bui Tuong PHONG. « Illumination for computer generated pictures ». *Seminal graphics : pioneering efforts that shaped the field*. 1998, p. 95-101 (cf. p. 52).

[Pla01] Max PLANCK. « On the law of distribution of energy in the normal spectrum ». *Annalen der physik* 4.553 (1901), p. 1 (cf. p. 47).

[Pop+21] Phillip POPE, Chen ZHU, Ahmed ABDELKADER, Micah GOLDBLUM et Tom GOLDSTEIN. *The Intrinsic Dimension of Images and Its Impact on Learning*. arXiv :2104.08894 [cs, stat]. Avr. 2021 (cf. p. 93).

[Pro+23] Jean PROST, Antoine HOUDARD, Andrés ALMANSA et Nicolas PAPADAKIS. « Inverse problem regularization with hierarchical variational autoencoders ». *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, p. 22894-22905 (cf. p. 110).

[RH01] Ravi RAMAMOORTHI et Pat HANRAHAN. « A signal-processing framework for inverse rendering ». *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '01. New York, NY, USA : Association for Computing Machinery, août 2001, p. 117-128. ISBN : 978-1-58113-374-5. DOI : 10.1145/383259.383271 (cf. p. 31, 117).

[Ray+23] Deep RAY, Javier MURGOITIO-ESANDI, Agnimitra DASGUPTA et Assad A. OBERAI. « Solution of physics-based inverse problems using conditional generative adversarial networks with full gradient penalty ». *Computer Methods in Applied Mechanics and Engineering* 417 (2023). A Special Issue in Honor of the Lifetime Achievements of T. J. R. Hughes, p. 116338. ISSN : 0045-7825. DOI : https://doi.org/10.1016/j.cma.2023.116338 (cf. p. 112).

[Ray+22] Deep RAY, Harisankar RAMASWAMY, Dhruv V. PATEL et Assad A. OBERAI. « The efficacy and generalizability of conditional GANs for posterior inference in physics-based inverse problems ». *Numerical Algebra, Control and Optimization* (déc. 2022). Publisher : Numerical Algebra, Control and Optimization, p. 0-0. ISSN : 2155-3289. DOI : 10.3934/naco.2022038 (cf. p. 112, 118, 119).

[Rey+09] Douglas A REYNOLDS et al. « Gaussian mixture models. » *Encyclopedia of biometrics* 741.659-663 (2009) (cf. p. 115).

[RFB15] Olaf RONNEBERGER, Philipp FISCHER et Thomas BROX. « U-Net : Convolutional Networks for Biomedical Image Segmentation ». *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Sous la dir. de Nassir NAVAB, Joachim HORNEGGER, William M. WELLS et Alejandro F. FRANGI. Cham : Springer International Publishing, 2015, p. 234-241. ISBN : 978-3-319-24574-4 (cf. p. 37, 86, 88).

[RHW86] David E RUMELHART, Geoffrey E HINTON et Ronald J WILLIAMS. « Learning representations by back-propagating errors ». *nature* 323.6088 (1986), p. 533-536 (cf. p. 82).

[Sah+22] Chitwan SAHARIA, William CHAN, Huiwen CHANG, Chris A. LEE, Jonathan HO, Tim SALIMANS, David J. FLEET et Mohammad NOROUZI. *Palette : Image-to-Image Diffusion Models*. arXiv :2111.05826 [cs]. Mai 2022. DOI : 10.48550/arXiv.2111.05826 (cf. p. 88, 110).

[Sah+23] Chitwan SAHARIA, Jonathan HO, William CHAN, Tim SALIMANS, David J. FLEET et Mohammad NOROUZI. « Image Super-Resolution via Iterative Refinement ». *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.4 (avr. 2023). Conference Name : IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 4713-4726. ISSN : 1939-3539. DOI : 10.1109/TPAMI.2022.3204461 (cf. p. 110, 128).

[SMK05] S.M. SEITZ, Y. MATSUSHITA et K.N. KUTULAKOS. « A theory of inverse light transport ». *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. T. 2. ISSN : 2380-7504. Oct. 2005, 1440-1447 Vol. 2. DOI : 10.1109/ICCV.2005.25 (cf. p. 31).

[Sen+19] Soumyadip SENGUPTA, Jinwei GU, Kihwan KIM, Guilin LIU, David JACOBS et Jan KAUTZ. « Neural Inverse Rendering of an Indoor Scene From a Single Image ». *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South) : IEEE, oct. 2019, p. 8597-8606. ISBN : 978-1-72814-803-8. DOI : 10.1109/ICCV.2019.00869 (cf. p. 31).

[Stu10] A. M. STUART. « Inverse problems : A Bayesian perspective ». *Acta Numerica* 19 (mai 2010). Publisher : Cambridge University Press, p. 451-559. ISSN : 1474-0508, 0962-4929. DOI : 10.1017/S0962492910000061 (cf. p. 107).

[Tai94] Mary M TAI. « A Mathematical Model for the Determination of Total Area Under Glucose Tolerance and Other Metabolic Curves ». *Diabetes Care* 17.2 (fév. 1994), p. 152-154. ISSN : 0149-5992. DOI : 10.2337/diacare.17.2.152 (cf. p. 48).

[Tal+20] Charly TALATIZI, Marie-Hélène AUMEUNIER, Fabrice RIGOLLET, Mickael LE BOHEC, Jonathan GÉRARDIN, Jonathan GASPAR, Christophe LE NILIOT et Albrecht HERRMANN. « Inverse radiation problem with infrared images to monitor plasma-facing components temperature in metallic fusion devices ». *Fusion Engineering and Design* 159 (2020), p. 111867 (cf. p. 25, 42).

[Tal+21] Charly TALATIZI, Marie-Hélène AUMEUNIER, Fabrice RIGOLLET, Mickael Le BOHEC, Christophe Le NILIOT et Albrecht HERRMANN. « Solving the infrared reflections contribution by inversion of synthetic diagnostics : First results on WEST ». *Fusion Engineering and Design* 171 (oct. 2021), p. 112570. ISSN : 0920-3796. DOI : 10.1016/j.fusengdes.2021.112570 (cf. p. 25, 42).

[Tic+24] Q. TICHIT, M.-H. AUMEUNIER, Y. CORRE, J. GERARDIN, L. DUBUS, J. GASPAR, J.P. GUNN, A. GROSJEAN, R. DEJARNAC, A. PODOLNÍK, A. JUVEN, K. KRIEGER et E. TSITRONE. « Plasma heat load in the toroidal gaps of the ITER-grade plasma facing units in WEST tokamak ». *Nuclear Materials and Energy* (2024) (cf. p. 3, 40, 126).

[Tik63] Andrei N TIKHONOV. « Solution of incorrectly formulated problems and the regularization method. » *Sov Dok* 4 (1963), p. 1035-1038 (cf. p. 26).

[Vea96] Eric VEACH. « Non-symmetric scattering in light transport algorithms ». *Rendering Techniques' 96 : Proceedings of the Eurographics Workshop in Porto, Portugal, June 17–19, 1996 7*. Springer. 1996, p. 81-90 (cf. p. 51).

[Vea98] Eric VEACH. « Robust Monte Carlo methods for light transport simulation » (1998) (cf. p. 55).

[VG97] Eric VEACH et Leonidas J. GUIBAS. « Metropolis light transport ». *Proceedings of the 24th annual conference on Computer graphics and interactive techniques - SIGGRAPH '97*. Not Known : ACM Press, 1997, p. 65-76. ISBN : 978-0-89791-896-1. DOI : 10.1145/258734.258775 (cf. p. 42, 56).

[VT11] Udo VON TOUSSAINT. « Bayesian inference in physics ». *Reviews of Modern Physics* 83.3 (2011), p. 943-999 (cf. p. 105).

[Wal07] Ingo WALD. « On fast Construction of SAH-based Bounding Volume Hierarchies ». *2007 IEEE Symposium on Interactive Ray Tracing*. Ulm : IEEE, sept. 2007, p. 33-40. ISBN : 978-1-4244-1629-5. DOI : 10.1109/RT.2007.4342588 (cf. p. 59).

[Wal+14] Ingo WALD, Sven WOOP, Carsten BENTHIN, Gregory S JOHNSON et Manfred ERNST. « Embree : a kernel framework for efficient CPU ray tracing ». *ACM Transactions on Graphics (TOG)* 33.4 (2014), p. 1-8 (cf. p. 34).

[WTM20] Manhar WALIA, Brendan TIERNEY et Susan MCKEEVER. *Synthesising Tabular Data using Wasserstein Conditional GANs with Gradient Penalty (WCGAN-GP)*. Déc. 2020 (cf. p. 112).

[Wal+07] Bruce WALTER, Stephen R MARSCHNER, Hongsong LI et Kenneth E TORRANCE. « Microfacet Models for Refraction through Rough Surfaces. » *Rendering techniques* 2007 (2007), 18th (cf. p. 53).

[Wan+21] Zian WANG, Jonah PHILION, Sanja FIDLER et Jan KAUTZ. « Learning Indoor Inverse Rendering with 3D Spatially-Varying Lighting ». *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada : IEEE, oct. 2021, p. 12518-12527. ISBN : 978-1-66542-812-5. DOI : 10.1109/ICCV48922.2021.01231 (cf. p. 32).

[Wen19] Lilian WENG. *From GAN to WGAN*. arXiv :1904.08994 [cs, stat]. Avr. 2019. DOI : 10.48550/arXiv.1904.08994 (cf. p. 111).

[YMS22] Kazuo YONEKURA, Nozomu MIYAMOTO et Katsuyuki SUZUKI. « Inverse airfoil design method for generating varieties of smooth airfoils using conditional WGAN-gp ». *Structural and Multidisciplinary Optimization* 65.6 (2022), p. 173 (cf. p. 112).

[Yu+19] Ying YU, Bingying TANG, Ronglai LIN, Shufa HAN, Tang TANG et Ming CHEN. « CW-GAN : Conditional Wasserstein Generative Adversarial Nets for Fault Data Generation ». *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Déc. 2019, p. 2713-2718. DOI : 10.1109/ROBIO49542.2019.8961501 (cf. p. 112).

[Zha00] Zhengyou ZHANG. « A flexible new technique for camera calibration ». *IEEE Transactions on pattern analysis and machine intelligence* 22.11 (2000), p. 1330-1334 (cf. p. 45).