Plasma Instability Identification through Machine Learning

E D D Zapata-Cornejo

To cite this version:

E D D Zapata-Cornejo. Plasma Instability Identification through Machine Learning. Physics [physics]. Aix Marseille University, 2024. English. NNT : 2024AIXM0376 . tel-04904999

Aix Marseille Université

NNT : 2024AIXM0376

THÈSE DE DOCTORAT

Soutenue à Aix-Marseille Université
le 20 Novembre 2024 par

Enrique de Dios ZAPATA CORNEJO

Titre de la thèse:
Plasma Instability Identification Through Machine Learning

Discipline
Physique et Sciences de la Matière

Spécialité
Energie, Rayonnement et Plasma

École doctorale
ED 352 PHYSIQUE ET SCIENCES DE LA MATIERE

Laboratoire/Partenaires de recherche
PIIM
M2P2
ITER

Composition du jury

Jesús VEGA Directeur de recherche,	Rapporteur CIEMAT
Emmanuel FRÉNOD Professeur des universités,	Rapporteur UBS
Mervi MANTSINEN Directrice de recherche,	Examinatrice BSC
Cristina REA Maître de conférences,	Examinatrice MIT
Geert VERDOOLAEGE Maître de conférences,	Examinateur Gent Univ
Virginie GRANDGIRARD Directrice de recherche,	Présidente du jury CEA
David ZARZOSO Chargé de Recherche,	Directeur de thèse CNRS
Simon D. PINCHES Chef de section,	Co-Directeur de thèse ITER

iter
china eu india japan korea russia usa

cnrs

Institut
Sciences de la Fusion et de l'Instrumentation en Environnements Nucléaires
Aix Marseille Université

PIIM

M2P2

Affidavit

Je soussigné, Enrique de Dios ZAPATA CORNEJO, déclare par la présente que le travail présenté dans ce manuscrit est mon propre travail, réalisé sous la direction scientifique de David ZARZOSO FERNÁNDEZ, 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 Aix en Provence le 17/09/2024

List of publications and conference participation

List of publications and/or patents produced as part of the thesis project:

1. Segmentation of MHD modes using Fourier transform, wavelets and computer vision algorithms. E.d.D. Zapata-Cornejo, D. Zarzoso, S.D. Pinches, S. E. Sharapov, M. Fitzgerald. 2024 Plasma Phys. Control. Fusion 66 095016. DOI: 10.1088/1361-6587/ad6a84

2. A novel unsupervised machine learning algorithm for automatic Alfvénic activity detection in the TJ-II stellarator. E.d.D. Zapata-Cornejo, D. Zarzoso, S. D. Pinches, A. Bustos, A. Cappa, E. Ascasibar. 2024 Nucl. Fusion 64 126057. DOI: 10.1088/1741-4326/ac3b3b

Participation in conferences and summer schools during the thesis period:

1. Culham Plasma Physics Summer School 2022. (Attended)

2. Unsupervised learning and feature extraction of Alfvén activity in fusion plasmas. E.d.D. Zapata-Cornejo, D. Zarzoso, S. Pinches, A. Bustos, P. Pons-Villalonga, and S. E. Sharapov. IAEA's Workshop on AI for Accelerating Fusion and Plasma Science. 30 November 2023. Poster presentation.

3. Algorithms for unsupervised learning of MHD modes in TJ-II stellarator and JET tokamak. E.d.D. Zapata-Cornejo, D. Zarzoso, S. Pinches, A. Bustos, A. Cappa, and M.Fitzgerald, JET contributors and WPTE team: P4.040 (2024). 50th EPS Conference on Contr. Fusion and Plasma Phys, 8-12 July 2024. Poster presentation.

Résumé

Cette thèse présente le développement de techniques avancées basées sur les données pour automatiser la détection et la classification des modes plasma dans les expériences de fusion, en se concentrant particulièrement sur les instabilités d'Alfvén et les modes magnétohydrodynamiques (MHD).

La première contribution de ce travail est le développement d'un algorithme de codage parcimonieux capable d'identifier des modes directement à partir des signaux bruts du plasma. Cette méthode, appelée "Elastic Random Mode Decomposition", applique une régression en réseau élastique parallélisée à des dictionnaires aléatoires d'atomes de Gabor. Cet algorithme isole les composantes oscillatoires significatives, même dans des signaux bruités.

De plus, des techniques d'apprentissage non supervisé sont utilisées pour regrouper les modes MHD en se basant sur les signaux plasma et l'information mutuelle, permettant la classification automatique de différents modes oscillatoires sans nécessiter de données étiquetées. Ces étapes de création de caractéristiques et ces méthodes de regroupement offrent une solution évolutive pour traiter de grands ensembles de données issus des expériences de fusion, permettant d'identifier systématiquement les instabilités plasma importantes.

La thèse explore également des méthodes de filtrage en vision par ordinateur pour l'extraction de caractéristiques à partir d'images de spectrogrammes. Ces filtres reposent sur l'analyse spectrale : transformée de Fourier, transformée en ondelettes et transformée de Hough. Ils améliorent la qualité des données de spectrogrammes en réduisant le bruit et les caractéristiques indésirables, améliorant ainsi les structures temporelles et fréquentielles liées aux oscillations du plasma.

En outre, des algorithmes de segmentation couramment utilisés en vision par ordinateur sont adaptés pour identifier les modes dans les images de spectrogrammes, permettant une segmentation précise des motifs oscillatoires. Le flux d'algorithmes de vision par ordinateur pour la segmentation est le suivant : des filtres de bruit, un détecteur de crêtes, un seuillage automatique et l'étiquetage des régions.

Ce résultat pourrait être déterminant pour l'étiquetage systématique des signaux, une étape cruciale vers l'automatisation de l'étiquetage des signaux diagnostiques du plasma. Les méthodes développées ici constituent une étape nécessaire pour la formation future de modèles d'apprentissage profond, qui pourraient encore améliorer la surveillance et le contrôle en temps réel du plasma dans les réacteurs de fusion.

Mots clés : modes MHD, instabilités d'Alfvén, apprentissage automatique, parcimonieux, réseau élastique, atomes de Gabor, analyse de signaux, apprentissage non supervisé, vision par ordinateur, segmentation, étiquetage.

Abstract

This thesis presents the development of advanced data-driven techniques to automate the detection and classification of plasma modes in fusion experiments, focusing particularly on Alfvén instabilities and magnetohydrodynamic (MHD) modes.

The first contribution of this work is the development of a sparse coding algorithm capable of identifying modes directly from raw plasma signals. This method called Elastic Random Mode Decomposition applies parallelized elastic net regression to random dictionaries of Gabor atoms, this algorithm isolates significant oscillatory components, even with noisy signals.

In addition, unsupervised learning techniques are employed to cluster MHD modes using plasma signals and mutual information, enabling the automatic classification of different oscillatory modes without needing labeled data. These feature creation steps and clustering methods offer a scalable solution for processing large datasets from fusion experiments, allowing for systematically identifying important plasma instabilities.

The thesis also explores computer vision filtering methods for feature extraction from spectrogram images. These filters are based on spectral analysis: Fourier transform, wavelet transform, and Hough transform. They improve the quality of the spectrogram data by reducing noise and undesired features, enhancing time frequency structures related to the plasma oscillations.

Furthermore, segmentation algorithms commonly used in computer vision (CV) are adapted to identify modes in spectrogram images, enabling precise segmentation of oscillatory patterns. The pipeline of CV algorithms for segmentation is the following: noise filters, ridge detector, automatic thresholding, and labeling regions.

This result might be key for systematic signal labeling, a crucial step toward automating the labeling of plasma diagnostic signals. The methods developed here provide a necessary step for future training of deep learning models, which could further enhance real-time plasma monitoring and control in fusion reactors.

Key words: MHD modes, Alfvén instabilities, machine learning, sparse, elastic net, Gabor atoms, signal analysis, unsupervised learning, computer vision, segmentation, labelling.

Résumé de la Thèse

Introduction

Cette thèse explore l'application des techniques d'apprentissage automatique, d'analyse de signaux et de vision par ordinateur à la détection et à la caractérisation non supervisée des modes plasma dans les expériences de fusion nucléaire, en mettant l'accent sur les instabilités d'Alfvén.

Le travail est structuré en trois blocs thématiques. Les deux premiers chapitres sont introductifs et offrent une vue d'ensemble des concepts de base de la fusion et de l'apprentissage automatique. Les chapitres 3 et 4 examinent l'apprentissage non supervisé des motifs MHD à travers le regroupement et le codage parcimonieux. Les chapitres 5 et 6 se concentrent sur l'utilisation de l'analyse spectrale et de la vision par ordinateur pour détecter les modes MHD.

Chapitre 1 : Physique de la fusion

L'énergie de la fusion nucléaire est le produit de la réaction entre deux noyaux isotopiques différents de l'hydrogène : le deutérium (D) et le tritium (T). Le processus nucléaire se termine par la fusion des deux isotopes, ce qui entraîne la formation d'isotopes d'hélium (He) et la libération de neutrons. La réaction nucléaire peut être représentée comme suit :

$$D\,(^{2}_{1}\mathrm{H}) + T\,(^{3}_{1}\mathrm{H}) \rightarrow {}^{4}_{2}\mathrm{He}(3.5MeV) + {}^{1}_{0}\mathrm{n}(14.1MeV). \tag{0.1}$$

La différence de masse entre les réactifs et les produits est convertie en énergie cinétique pure, $E = mc^2$. L'énergie cinétique est répartie entre les particules alpha (noyaux d'He) et les neutrons, 3,5 MeV pour chaque particule alpha et 14,1 MeV pour chaque neutron.

Les densités, températures et temps de confinement nécessaires pour que la réaction de fusion ait lieu sont définis par le critère de Lawson, qui pour la réaction D-T, s'exprime sous la forme d'un produit triple [WC11] :

$$nT\tau_E \geq 3 \times 10^{21}\,\mathrm{m}^{-3}\,\mathrm{keV\,s}, \tag{0.2}$$

où n est la densité du plasma (en particules par mètre cube), T est la température du plasma (en keV) et τ_E est le temps de confinement d'énergie (en secondes). Aux températures et densités requises pour la fusion, le combustible se trouve à l'état de plasma, un état de la matière où les électrons sont séparés des noyaux. Le plasma est confiné par des champs magnétiques dans des dispositifs appelés tokamaks ou stellarators.

Les fréquences caractéristiques des ondes MHD sont comparables à la fréquence de précession des ions confinés. Cela est important car une onde peut gagner de l'énergie grâce à l'interaction avec des particules énergétiques. Si une perturbation dispose d'énergie pour croître, une instabilité peut apparaître et conduire à une perte de confinement. L'une des instabilités les plus courantes dans les plasmas de fusion sont les ondes d'Alfvén, qui sont des ondes MHD se propageant le long des lignes de champ magnétique. La détection de ces ondes est essentielle pour le contrôle du plasma.

La vitesse d'Alfvén est proportionnelle au champ magnétique B, et inversement proportionnelle à la racine carrée de la densité de masse du plasma $\sum n_i m_i$. Lorsque le plasma est composé d'espèces avec le même rapport charge/masse, la densité de masse [Hei08] est proportionnelle à la densité électronique, $V_A \propto B/\sqrt{n_e}$. Dans ces cas, en substituant dans la relation de dispersion, nous obtenons :

$$f \propto \frac{B}{\sqrt{n_e}} \propto \frac{1}{\sqrt{n_e}}. \tag{0.3}$$

Cette équation nous permet de distinguer les modes d'Alfvén des autres ondes plasma, car ils sont fortement corrélés avec le signal de diagnostic de densité du plasma.

Il existe de nombreux types de modes d'Alfvén qui peuvent être excités dans les dispositifs de fusion, tels que les modes d'Alfvén globaux (GAE), les modes d'Alfvén toroïdaux (TAE), les modes d'Alfvén elliptiques (EAE) et les modes d'Alfvén compressibles (CAE).

L'identification des différents types d'instabilités ou leur prévision est essentielle pour le contrôle du plasma. La détection de ces modes se fait généralement en analysant les signaux des bobines magnétiques ou de l'interférométrie.

Chapitre 2 : Science axée sur les données

Apprentissage automatique

Une définition informelle de l'apprentissage automatique (ML) pourrait être : “ Les ordinateurs se programment eux-mêmes avec des données. ” Ou une définition plus détaillée : *Étant donné un ensemble de données, les ordinateurs peuvent apprendre à détecter des motifs (représentation optimale des données), puis effectuer des actions complexes selon certaines contraintes (décisions optimales).* En d'autres termes, nous pouvons modéliser des relations fonctionnelles complexes entre les données disponibles et la solution du problème d'intérêt : en supposant que $\boldsymbol{y} = f(\boldsymbol{x})$ existe pour une fonction inconnue f, l'objectif de l'apprentissage statistique est d'estimer la fonction f en utilisant un ensemble de données d'apprentissage X, afin de faire des prédictions à partir de l'estimation $\hat{y} = \hat{f}(X)$ [Jam+14].

La différence principale entre l'apprentissage supervisé et non supervisé réside dans la présence d'étiquettes dans les données. En apprentissage supervisé, les données possèdent des étiquettes, et l'algorithme apprend à les prédire. En apprentissage non supervisé, les données n'ont pas d'étiquettes, et l'algorithme apprend à trouver des motifs dans les données.

Un problème courant de l'apprentissage automatique est d'éviter le surapprentissage (overfitting). Pour éviter cela, nous pouvons utiliser des techniques de régularisation. La régularisation ajoute un terme de pénalité à la fonction de perte pour empêcher le modèle d'apprendre des motifs trop complexes.

Traitement du signal

Les informations contenues dans les signaux temporels et les images sont incroyablement denses ; pour démêler les informations pertinentes, les scientifiques utilisent des techniques de traitement du signal. Essentiellement, le traitement du signal est la recherche de représentations alternatives des données qui révèlent les détails d'intérêt: la transformée de Fourier, l'analyse par ondelettes et la transformée de Hough sont des exemples de techniques de traitement du signal. Un exemple de décomposition d'image utilisant cette DWT est montré dans la Fig. 2.8, et comparé à la Fig. 2.10.

Vision par ordinateur

La vision par ordinateur (CV) étudie les algorithmes et les modèles pour traiter, analyser et interpréter des images et des vidéos. L'objectif ultime est d'imiter la perception visuelle humaine. Pour s'attaquer à la complexité de la compréhension visuelle, la recherche dans ce domaine est divisée en tâches spécialisées ou en sujets.

Dans ce travail de thèse, nous avons exploré des techniques de vision par ordinateur traditionnelles et basées sur l'apprentissage profond pour analyser les données de diagnostic du plasma. Les techniques traditionnelles de vision par ordinateur utilisées incluent la détection de crêtes, le seuillage automatique et l'étiquetage de région. Les techniques de vision par ordinateur basées sur l'apprentissage profond incluent les réseaux de neurones convolutifs profonds (CNN) et l'apprentissage par transfert. Les détecteurs de crêtes mentionnés ont été appliqués à une photographie de JET pour comparer les différents résultats, Fig. 2.13. Une fois les crêtes améliorées, la même technique de seuillage est appliquée. Dans la Fig. 2.14, les méthodes de seuillage automatique sont comparées.

Chapitre 3 : Identification des modes à l'aide de la régression parcimonieuse et des dictionnaires aléatoires de Gabor

Dans ce chapitre, un nouvel algorithme de décomposition de mode appelé “ Elastic Random Mode Decomposition ” (ERMD) est introduit. Il traite les signaux bruts des bobines Mirnov de TJ-II pour trouver des structures temps-fréquence. La phase de codage du signal est détaillée dans la Fig. 3.13.

Le problème de l'analyse du signal peut être compris comme un problème de régression linéaire. Étant donné un signal $\boldsymbol{s}$, et une matrice de formes d'onde $\mathbf{D}$, on cherche les poids $\boldsymbol{c}$, qui représentent le signal $\boldsymbol{s} = \mathbf{D}\boldsymbol{c}$. La matrice $\mathbf{D}$ est connue sous le nom de matrice dictionnaire. Dans le cas de SRMD [RST22], les colonnes de $\mathbf{D}$ sont composées d'atomes de Gabor réels, avec des fréquences, phases et emplacements temporels aléatoires (d'où le terme aléatoire).

L'objectif est de trouver une représentation qui conserve les modes correspondant aux harmoniques MHD tout en éliminant les caractéristiques indésirables comme le bruit. Pour construire une matrice dictionnaire, nous supposons que le signal de la bobine Mirnov peut être reproduit comme une combinaison linéaire d'atomes de Gabor : $s_i = W(t-\tau_i)\cos(2\pi f_i t + \phi_i)$, où $W(t-\tau_i) = \exp\left(-\frac{(\boldsymbol{t}-\tau_i)^2}{2\sigma^2}\right)$ est une fonction gaussienne de largeur σ. Le terme $\cos\left(2\pi f_i t + \phi_i\right)$ reconstruit la structure oscillatoire, et la fonction gaussienne module l'amplitude sur un temps spécifique τ. Une représentation des différents atomes de Gabor est visible dans la Fig. 3.3.

Le signal peut être reconstruit en utilisant la combinaison linéaire de tous les atomes, à condition que nous connaissions les coefficients c_i qui représentent l'amplitude de chaque atome.

$$\boldsymbol{s} \approx \sum_i c_i \cdot s_i = \sum_i c_i \cdot \exp\left(-\frac{(\boldsymbol{t}-\tau_i)^2}{2\sigma^2}\right)\cos(2\pi f_i t + \phi_i). \tag{0.4}$$

Pour calculer les coefficients c_i, il est possible d'utiliser un ajustement par régression linéaire, où la pratique courante consiste à minimiser l'erreur quadratique $||\boldsymbol{s} - \mathbf{D}\boldsymbol{c}||^2$. Cependant, les valeurs des coefficients c_i représenteront également le bruit dans le signal audio.

Nous pouvons utiliser un réseau de neurones à une seule couche avec une fonction d'activation linéaire $\boldsymbol{y} = \mathbf{X} \cdot \boldsymbol{w} + \boldsymbol{b}$, et $\boldsymbol{b} = \mathbf{0}$. Nous entraînons le réseau de manière à ce que l'entraînement soit équivalent à la résolution de notre représentation du signal $\boldsymbol{s} = \mathbf{D} \cdot \boldsymbol{c}$. En d'autres termes, l'architecture du réseau de neurones est équivalente à l'équation du décodeur (ou équation de synthèse)

$$\begin{pmatrix}\text{Décodeur}\\ \text{équation}\end{pmatrix} \quad \boldsymbol{s} = \mathbf{D} \cdot \boldsymbol{c}, \tag{0.5}$$

et l'entraînement du réseau de neurones est équivalent à résoudre le problème d'optimisation de l'équation de l'encodeur (ou équation d'analyse)

$$\begin{pmatrix}\text{Encodeur}\\ \text{équation}\end{pmatrix} \quad \boldsymbol{c} = \arg\min_{\boldsymbol{c}} \left\{||\mathbf{D} \cdot \boldsymbol{c} - \mathbf{s}||_2^2 + \lambda_1 ||\boldsymbol{c}||_1 + \lambda_2 ||\boldsymbol{c}||_2\right\}. \tag{0.6}$$

Il est important de noter la grande asymétrie entre le codage et le décodage. Parce que l'équation du décodeur est linéaire, alors que l'équation de l'encodeur est non linéaire. La Fig. 3.13 illustre schématiquement l'architecture du réseau de neurones utilisé. Même si la fonction d'activation est linéaire, la non-linéarité est présente dans l'atome de Gabor, qui agit comme un noyau non linéaire. La profondeur est remplacée par la grande longueur de la couche d'entrée. Dans ce travail, PyTorch [Pas+19] avec les modules Lightning ont été utilisés pour implémenter ce modèle. En utilisant ce réseau de neurones, nous pouvons bénéficier du calcul parallèle sur GPU.

Il convient de noter que l'équation de l'encodeur est différente de la formulation BPDN, la fonction de perte dans l'équation de l'encodeur est connue sous le nom de *elastic net* [ZH05]. Il y a cependant deux principales différences. Premièrement, la formulation du problème est non contrainte (elle ne nécessite pas de paramètre de bruit σ). Deuxièmement, il y a deux termes de régularisation λ_1 et λ_2 associés aux modules du code $\boldsymbol{c}$ dans la norme ℓ_1 $||\boldsymbol{c}||_1$ et la norme ℓ_2 $||\boldsymbol{c}||_2$ respectivement, tandis que dans BPDN, seule la norme ℓ_1 est utilisée. La norme $||\boldsymbol{c}||_2$ favorise la présence de groupes corrélés dans le code $\boldsymbol{c}$ [ZH05].

Une fois le signal encodé, nous pouvons créer des groupes d'atomes proches dans le temps et la fréquence. Les algorithmes de regroupement basés sur la densité sont adaptés à notre application, car les atomes encodés tendent à se concentrer sur les structures de modes temps-fréquence, et ils sont équipés pour gérer les points de bruit et les regroupements non linéairement séparables de formes et tailles arbitraires. Nous proposons l'utilisation de DBSCAN hiérarchique (HDBSCAN) [Cam+15] qui n'assume pas une densité constante et est capable de déterminer la structure de densité variable. Un exemple du résultat de HDBSCAN est montré dans la Fig. 3.20.

Pour chaque mode identifié, à l'exception des regroupements de bruit (-1), il existe deux options pour déterminer leur caractère potentiellement Alfvénique : (1) Calculer la corrélation de Pearson entre les fréquences des modes et $1/\sqrt{n}$. Si la corrélation est supérieure au seuil de corrélation ρ_{thr}, étiqueter le mode comme 1 (Alfvénique), sinon 0 (Non Alfvénique). (2) Utiliser l'information mutuelle (MI) à la place, ce qui peut aider à récupérer plus de modes. (Les détails peuvent être trouvés dans le chapitre 4). Le nouvel algorithme appelé Elastic Random Mode Decomposition (ERMD) est résumé dans l'algorithme 2.

Chapitre 4 : Reconnaissance de motifs de haut niveau

Une fois les atomes représentant le signal de la bobine Mirnov convenablement extraits et regroupés en modes, un second regroupement peut être appliqué pour extraire des motifs de plus haut niveau. Les signaux du plasma utilisés en plus de la fréquence des modes sont le courant I_p, l'énergie magnétique W, et la densité $1/\sqrt{n}$. Un exemple de ces signaux est tracé dans la Fig. 3.21. Le profilage de chaque regroupement aidera à comprendre le résultat de l'exécution de l'ERMD, et la nature des modes collectés. La méthode de regroupement choisie est K-means.

Le résultat du (t-distributed Stochastic Neighbor Embedding) T-SNE est montré dans la Fig. 4.14, tous les modes découverts sont représentés dans un espace bidimensionnel. Les numéros des étiquettes sont positionnés sur le tracé en prenant les valeurs moyennes des dimensions 1 et 2 dans chaque groupe. Enfin, l'étiquette de classe résultant du regroupement K-means est utilisée pour assigner une couleur aux points de projection et interpréter le résultat de la projection. L'interprétation de l'analyse des regroupements est résumée dans le tableau 4.4. Pour plus de clarté, nous fournissons des exemples de regroupements dans la Fig. 4.15.

Une autre façon d'analyser les relations entre les modes et les signaux du plasma est la suivante : pour chaque mode, nous aurons des catégories binaires : La fréquence du mode est-elle corrélée à la densité $1/\sqrt{n}$? (Oui ou Non) ; La fréquence du mode est-elle corrélée au courant I_p ? (Oui ou Non) ; La fréquence du mode est-elle corrélée à l'énergie W ? (Oui ou Non). Le tableau de contingence résultant (Tableau 4.2) pour les caractéristiques des modes montre la distribution de fréquence multivariée de la classe de mode Alfvénique, quantifiant la dépendance à ces nouvelles variables catégorielles, et interprétant les fréquences relatives comme une probabilité si le tableau est normalisé.

Chapitre 5 : Extraction de caractéristiques spectrales par analyse du signal : Transformée de Fourier, ondelette et transformée de Hough

Suppression du bruit large bande à l'aide des ondelettes discrètes

Les ondelettes discrètes sont des réseaux de filtres multiscale-multidirectionnels. Un exemple de décomposition spectrogramme est montré dans la Fig. 5.2. Le caractère du bruit dans les spectrogrammes est fortement anisotrope, en raison de la fuite spectrale du STFT dans chaque fenêtre temporelle. Ce bruit large bande se situe principalement dans la sous-image des détails verticaux. Ainsi, en supprimant les détails verticaux de l'image, nous pouvons réduire considérablement le bruit dans les spectrogrammes. La Fig. 5.3 montre le résultat, y compris l'effet des événements d'injection de pellets dans cet exemple particulier. Les étapes sont résumées dans l'algorithme 3. Le résultat est une image propre ne contenant que les structures de mode.

Détection de motifs de lignes droites

Certains signaux observés dans les spectrogrammes apparaissent comme des lignes droites avec des orientations spécifiques. Dans de nombreux cas, ces signaux ne sont pas causés par des instabilités du plasma. Par exemple, l'antenne TAE crée un motif de balayage de fréquence, visible dans la Fig. 5.4. Nous devons développer une méthode pour détecter et extraire ces lignes droites des images, algorithme 4. Dans la première étape de la détection, nous avons utilisé la transformée de Hough [DH72; GMK99] pour trouver les lignes droites $y = mx + b$ dans les images des spectrogrammes.

Le résultat de la détection de lignes droites est montré dans la Fig. 5.6. La transformée de Hough probabiliste [GMK99] peut être utilisée pour intégrer des priorités telles que la longueur de ligne, l'orientation angulaire ou la séparation entre les lignes. Les résultats de l'application de cette technique sont présentés dans la Fig. 5.7.

Suppression des motifs orientés avec la FFT2D

Pour procéder à la suppression des motifs orientés, nous commençons par les lignes droites détectées à l'aide de la transformée de Hough. Dans l'espace de Fourier, les informations de l'image sont exprimées sous forme de fréquences spatiales dans deux directions. Les étapes sont résumées dans l'algorithme 5. Nous pouvons alors concevoir un masque sur le spectre d'amplitude pour supprimer les fréquences dans les directions souhaitées.

Si les algorithmes 3 et 5 sont combinés, l'image résultante offre un contraste amélioré, mettant en évidence les caractéristiques MHD du signal. Dans la Fig. 5.9, la majeure partie du bruit large bande et le signal complet de l'antenne TAE sont supprimés.

Correspondance de motifs multiscale-multidirectionnelle avec les curvelets

La transformée de curvelet [Can+06] est une généralisation multidirectionnelle des ondelettes discrètes (DWT), mieux adaptée aux images avec des bords courbes et des caractéristiques directionnelles multiples, comme les modes MHD dans nos spectrogrammes. Pour éliminer le bruit et les composants indésirables dans une direction angulaire sélectionnée, nous fixons les sous-images correspondantes à 0, puis appliquons la transformée inverse des curvelets.

Le résultat de l'utilisation de l'algorithme 6 est montré dans la Fig. 5.11. Les curvelets éliminent efficacement le bruit anisotrope et améliorent le contraste des spectrogrammes. Le signal de l'antenne TAE et les sous-harmoniques ICRH sont également supprimés avec succès du spectrogramme, ainsi que l'élimination des dents de scie.

Chapitre 6 : Segmentation des modes avec la vision par ordinateur

La segmentation d'une image $I(x, y)$ peut être définie (page 131 de la Ref. [AR05]) comme l'identification d'un ensemble fini de régions, ou segments $(R_1, R_2, R_j, \ldots, R_N)$ tels que où la propriété de définition P détermine le résultat de la segmentation.

La fluctuation et la structure de l'intensité des pixels déterminent les caractéristiques d'une image. La valeur de l'intensité des pixels peut être vue comme une carte de paysage avec des vallées, des plateaux, des montagnes avec des crêtes et des changements brusques comme des arêtes, ou des pics causés par du bruit. Dans la Fig. 2.11, nous représentons une coupe simplifiée en une dimension de l'intensité des pixels de l'image, avec les arêtes et crêtes mises en évidence.

Les filtres de crêtes utilisés dans cette thèse sont les détecteurs de crêtes de Sato [Sat+98], de Meijering [Mei+04], de Frangi [Fra+98] et de Hessian [Ng+15].

Une fois que les arêtes des spectrogrammes sont améliorées, la binarisation d'une image $I(x,y)$ peut être effectuée par la sélection d'un seuil global T. Le paramètre seuil T peut être ajusté manuellement. Cependant, cela n'est pas pratique pour analyser différentes expériences dans de grandes bases de données. Les algorithmes de seuillage automatique global utilisés dans cette thèse sont ceux d'Otsu [Ots79] et de Li [LL93; LT98]. À des fins de comparaison, nous montrons également le résultat de la méthode de Yen [YCC95; San04], la méthode ISODATA [RC78; San04], la méthode du triangle [ZRL77], le seuil minimal [Gla93; PM66], et le seuil moyen.

Les structures des modes binarisées sont clairement identifiables après l'amélioration des caractéristiques et l'application du seuillage automatique. Cela souligne la nécessité d'utiliser la détection des arêtes ou des crêtes, avant la binarisation pour garantir que les modes sont correctement séparés de l'arrière-plan. L'exemple 6.8 est utilisé pour démontrer les méthodes de seuillage.

Il peut y avoir des cas où il n'y a pas de signaux indésirables, peut-être parce que l'antenne TAE ou l'ICRH ne sont pas en fonctionnement dans ces cas. L'algorithme présenté 7 peut être appliqué directement. Un exemple complet de pipeline de vision par ordinateur est donné par l'algorithme 8. Il est à noter que ces pipelines sont flexibles et peuvent être adaptés à différents signaux et différents spectrogrammes.

Nous pouvons utiliser le résultat de la Fig. 5.11 pour comparer les différentes méthodes de détection des crêtes. Nous pouvons observer dans la Fig. 6.9, la Fig. 6.10, et la Fig. 6.11, que des résultats équivalents pour séparer les modes de l'arrière-plan sont obtenus avec les méthodes DoG, Meijering, Sato et Frangi à plusieurs échelles $\sigma = [1,2,3,4]$. Dans ce cas, seuls les seuils calculés par la méthode de Yen fonctionnent. Nous pouvons voir que les valeurs de seuil de la première rangée de la Fig. 6.11 sont significativement plus élevées que les autres valeurs calculées par Otsu et la méthode moyenne. Des preuves supplémentaires de l'efficacité de la méthode de Yen pour les images traitées avec des curvelets peuvent être trouvées dans la Fig. 6.12.

Après avoir amélioré le contraste et supprimé d'autres composantes indésirables du signal, il est raisonnable de supposer que les zones connectées de pixels d'image représentent des modes MHD, qui peuvent être détectés en utilisant un filtre de connectivité [WOS05; FG96]. L'application de l'algorithme 8 à un signal de bobine Mirnov de l'expérience JET 92416 donne le résultat présenté dans la Fig. 6.15. Nous pouvons conclure que les modes individuels sont étiquetés de manière satisfaisante.

Application aux spectrogrammes d'interférométrie

Les interféromètres laser sont des outils très utiles pour le diagnostic des plasmas. Selon la conception de la machine, plus ou moins de composants optiques sont montés autour de la chambre à vide. Les oscillations mécaniques peuvent créer des motifs parasites dans le spectrogramme. En outre, il peut y avoir d'autres sources de bruit non identifiées ou d'interférences électriques qui peuvent affecter le spectrogramme résultant. Dans la Fig. 6.17 (premier panneau), nous pouvons voir un spectrogramme de l'interféromètre rapide FIR de JET (KG1F) canal 3.

Dans le chapitre précédent, nous avons introduit comment créer des masques dans l'espace de Fourier. Nous pouvons utiliser un masque très fin dans l'espace de Fourier pour supprimer les lignes horizontales dans le spectrogramme.

Dans la deuxième rangée de la Fig. 6.17, le résultat du filtrage du spectrogramme est visible, ainsi que le bruit supprimé de l'image. Il est maintenant possible de distinguer certaines structures de modes présentes dans le noyau du plasma, juste en dessous de 200 et 400 kHz. Dans la troisième rangée de la Fig. 6.17, nous montrons le résultat du seuillage du spectrogramme. Enfin, l'algorithme d'étiquetage a été appliqué à l'image binaire pour identifier les modes dans la Fig. 6.18.

Segmentation d'un spectrogramme d'un décharge avec des fishbones

Dans les algorithmes introduits au chapitre 5, nous supposions que toutes les informations contenues dans la direction verticale étaient du bruit ou des artefacts indésirables. Ce n'est pas le cas lorsque des fréquences balayées rapidement d'origine physique, comme les modes " fishbones " ou les modes de particules énergétiques, sont présents.

Une approche plus simple, si la segmentation est appliquée en premier (algorithme 7), consiste à utiliser les propriétés des régions *après segmentation* pour séparer et filtrer les structures de mode du résultat. Le spectrogramme de la décharge JET 54300 illustré dans la Fig. 6.19 est un exemple d'un décharge avec des fishbones. Dans la Fig. 6.20, l'orientation des régions est utilisée pour séparer le résultat de la segmentation en structures orientées horizontalement et verticalement.

Utilisation de l'apprentissage par transfert pour la détection des modes

Enfin, nous avons revisité l'analyse des données TJ-II avec des algorithmes de vision par ordinateur. Nous avons démontré l'utilisation de l'apprentissage par transfert pour faire face à un faible volume de données. L'apprentissage par transfert est le processus dans lequel des modèles pré-entraînés sont ajustés pour fonctionner avec un ensemble de données spécifique. Les modèles pré-entraînés utilisés dans ce travail sont SAM [Kir+23] (Segment Anything Model), Detectron2 [Wu+19] (un framework open-source de Facebook pour la vision par ordinateur), et YOLOv8 [Ult] (un framework open-source de vision par ordinateur de la société Ultralytics). Ces frameworks incluent des modèles de pointe dans la segmentation et la segmentation d'instance.

Les résultats montrent que l'apprentissage par transfert en utilisant Detectron2 produit la meilleure segmentation de l'activité MHD : un indice de Jaccard de 0.491 ± 0.130 sur l'ensemble de test (l'indice de Jaccard le plus élevé précédent était de 0.427 [Bus+21]). De plus, très peu de données étiquetées sont nécessaires (≈20 décharges pour Detectron2) pour obtenir des résultats avec un indice de Jaccard = 0.438 ± 0.13.

Conclusions

Dans le chapitre 3, un nouvel algorithme, ERMD, a été proposé, qui surpasse son prédécesseur, SRMD, en travaillant efficacement dans le domaine temporel avec des signaux bruts. Malgré certaines limitations liées aux coûts computationnels et à l'ajustement des hyperparamètres, ERMD montre des résultats prometteurs, en particulier pour l'identification des motifs Alfvéniques dans les signaux du plasma. L'puissance de l'algorithme ERMD est remarquable, car il peut étiqueter plus de 700 décharges en une semaine, ce qui est difficilement possible lorsque l'étiquetage se réalise manuellement. Bien que la mise en œuvre actuelle soit orientée vers l'analyse inter-décharge, des travaux futurs pourraient inclure le développement d'un modèle de substitution d'apprentissage automatique pour permettre la détection en temps réel de l'activité Alfvénique à partir des signaux bruts des bobines Mirnov. L'algorithme développé pour ce travail est hautement adaptable et peut être appliqué à une gamme de signaux temporels non stationnaires et à haute résolution au-delà de la fusion nucléiare. Les applications potentielles incluent l'analyse d'électrocardiogrammes, l'activité sismique, et même l'analyse musicale, démontrant ainsi la large applicabilité des techniques d'apprentissage automatique et de traitement du signal présentées.

Dans le chapitre 4, nous avons étudié des méthodes innovantes pour détecter des motifs dans les données de fusion. L'une des innovations majeures de cette recherche est l'utilisation de l'information mutuelle (MI) pour capturer à la fois les relations linéaires et non linéaires entre les variables. Cette amélioration augmente la robustesse de la détection des modes, la rendant plus résistante au bruit et aux valeurs aberrantes. De plus, une technique de regroupement appliquée aux modes récupérés par ERMD s'est avérée efficace pour résumer l'activité des modes et filtrer les données bruyantes.

Dans les chapitres 5 et 6, nous avons démontré le potentiel de l'analyse du signal et des algorithmes de vision par ordinateur pour filtrer le bruit et identifier les caractéristiques clés dans les spectrogrammes. Cette thèse introduit des approches innovantes pour le filtrage du bruit dans les spectrogrammes à l'aide de techniques classiques de vision par ordinateur telles que les ondelettes et les curvelets. Les résultats obtenus à l'aide d'un pipeline de segmentation en vision par ordinateur classique (extraction de caractéristiques - détection de crêtes - seuillage - étiquetage) offrent une méthode flexible capable d'étiqueter systématiquement les données des spectrogrammes. De plus, un avantage majeur de ces algorithmes est qu'ils ne nécessitent pas de GPU et que le temps de calcul est faible. Enfin, l'application de l'apprentissage par transfert avec des modèles de pointe comme YOLOv8 a montré une amélioration de la précision et de la vitesse de détection des modes en temps réel. Cela ouvre la voie à l'application de telles techniques aux nouveaux dispositifs de fusion, où les données sont limitées, contribuant ainsi à l'automatisation et à la surveillance en temps réel des diagnostics de fusion.

Acknowledgements

I would like to thank first my supervisors David and Simon for allowing me to live this adventure, they trusted in me and gave me intellectual freedom to learn and research. I could not be more thankful.

During these years I met wonderful people from different countries and in different situations, all of them have contributed in some way to this journey.

Thanks to Sergei Sharapov, Michael Fitzgerald and Oliver from UKAEA for teaching me how to run Helena and Mishka codes and many other Tokamak physics. The 3 months I could spend living in UK and learning directly from them were memorable.

I also need to thank the people at TJ-II. Andrés Bustos for helping me in numerous occasions and sharing his own labeled data. Álvaro Cappa especially helped us to understand much better stellarators, and Pedro Pons-Villalonga helped me in the understanding of TJ-II Mirnov coils system.

I feel very fortunate to have met so many colleagues at AMU and ITER. The only regret that I have of these three years is not having spent more time with them. I would like to thank all of them for their support and friendship.

At last but not least I need to thank my family, my parents for nurturing me in any way they could, also my brothers and sisters whose company I really enjoy. And the person who most sacrifices made and supported me every day was my girlfriend Sonia, she teaches me every day what really is important.

Contents

List of Figures

List of Tables

Introduction

This thesis explores the application of machine learning, signal analysis, and computer vision techniques to the detection and unsupervised characterization of plasma modes in nuclear fusion experiments, with a particular focus on Alfvén instabilities.

The work is structured into three thematic blocks, with each chapter addressing a key aspect. In addition, a certain level of redundancy is included to allow each chapter to be read independently. The first two chapters are introductory, providing an overview of basic concepts in fusion and machine learning. Readers may choose to skip one of these chapters, depending on their expertise or interest. Chapters 3 and 4 examine the unsupervised learning of MHD patterns through clustering and sparse coding. Chapters 5 and 6 focus on the use of spectral analysis and computer vision for detecting MHD modes.

Chapter 1 introduces the physics and engineering principles of fusion, offering a basic understanding of nuclear fusion and MHD instabilities. It discusses key concepts such as fusion reactions, plasma confinement, and Alfvén eigenmodes, while giving a context of fusion research. The goal is to give the reader a foundational understanding of the physics of Alfvén eigenmodes behind this thesis work.

Chapter 2 shifts the focus to data-driven science and the role of machine learning (ML) in analyzing fusion plasma data. This chapter introduces the three key disciplines necessary for the development of the results in this work: machine learning, spectral signal analysis, and computer vision. ML is presented as the culmination of a data analysis process that begins with data extraction, processing, and statistical analysis. Moreover, the spectral analysis introduced is essential for detecting periodic, multiscale patterns in images before applying computer vision techniques to identify MHD activity.

The next block begins with Chapter 3, which develops sparse regression and the use of Gabor random dictionaries for mode identification in plasma signals. It starts with a musical analogy, which recurs throughout the text. This material was part of the seminar “From Music to Fusion: How Artificial Intelligence Helps Interpret Complex Signals”, presented at AMU in May 2024. This chapter introduces sparse regression as a key tool for identifying specific oscillations in time series data, demonstrating its relevance in decomposing signals into meaningful modes.

Chapter 4 builds on these ideas by focusing on high-level pattern recognition. Expanding on the signal decomposition techniques introduced in the previous chapter, this chapter explores the application of unsupervised learning methods to identify more complex, higher-level patterns in plasma signals. It covers techniques for discovering relationships between variables—both continuous and categorical—and introduces the concept of Alfvénic modes, a critical type of plasma instability. The chapter demonstrates how large datasets from fusion experiments can be mined to uncover hidden patterns and relationships, highlighting the potential of unsupervised learning in plasma diagnostics. The algorithms and results discussed in chapters 3 and 4 were published in Nuclear Fusion [Zap+24b] just before the defense of this thesis.

Chapter 5 shifts to signal analysis and feature extraction, aiming to filter noise and identify key features in spectrograms. The chapter introduces Fourier, wavelet, and Hough transforms as tools for spectral feature extraction, emphasizing their role in improving data quality by removing noise and highlighting critical patterns. These techniques enhance the performance of the computer vision (CV) algorithms developed in the following chapter. Some of the material in these two chapters has been published in Plasma Phys. Control. Fusion [Zap+24a].

Chapter 6 explains how CV methods can be used to analyze complex modal structures in spectrogram images. Most of this chapter focuses on classic CV algorithms that do not require GPUs, and are often used in other disciplines, such as microscopy imaging, to label data. Key techniques such as ridge detection, automatic thresholding, and region labeling are highlighted, demonstrating their effectiveness in breaking down spectrograms into mode structures. The chapter also introduces the use of transfer learning in deep learning models to enable fast inference of MHD modes. State-of-the-art models like Detectron2 and YOLOv8 are used to improve the accuracy of segmentation and pattern recognition in plasma data. These computer vision methods allow for the automated identification of time-frequency structures, paving the way for future real-time diagnostics using machine learning.

Finally, a brief conclusion to this thesis is presented. In addition, some appendices provide supplementary material: metrics of ML, ML feature selection methods, and the images of processed spectrograms in Chapter 5 are presented in Fourier space. The thesis ends with the bibliographic references cited throughout the text.

1. Fusion Physics

"The sun and its light,
the ocean and the wave,
the singer and his song.
Not one. Not two."

— Anthony de Mello s.j.

Summary

1.1. Introduction

Nuclear fusion is an extremely rich field that connects with almost all disciplines of physics and engineering. Although the history of fusion research involves many characters and institutions, our intention is to provide a brief perspective of this rich context, including some of the people and places historically involved, and the associated time scales. Many references provide more detailed accounts of fusion history [Mor18; Che11; Wag07; Mir19; Azi12; MSC21; Wag07].

What fuels the Sun? Behind this simple question lie some of the most important discoveries in physics. Determining the energy source of the Sun is essential to estimate its age. Curiously, this was of particular interest to both supporters and detractors of Darwin's theory of evolution [BS02], because early calculations based on energy conservation and experimental measurements showed that the Earth was orders of magnitude older than the Sun.

In 1920, Sir Arthur Eddington proposed that only 5% of hydrogen in the Sun could account for its energy production if the transmutation of hydrogen into helium converted the mass difference into free energy [Edd20]. Eddington's proposal directly cited Rutherford's work on the transmutation of elements and F.W. Aston's work on mass spectroscopy, which showed that the mass of helium is slightly less than the mass of four hydrogen nuclei. Of course, he was also influenced by Einstein's work on relativity and Marie Curie's work on radioactivity.

At the time, it was believed that the Sun was composed of the same matter as Earth's crust, and the neutron had not yet been discovered (the neutron was discovered later by Chadwick in 1932). The hypothesis was that every atomic nucleus was made of hydrogen ions (as explained by Rutherford's early atomic model). Regarding the state of matter in the Sun, it was believed that at least a large portion of stars should be in a gaseous state where electrons are free particles, resembling an ideal gas. The term "plasma" to describe this state was introduced in 1928 by Langmuir. Initially, the ideal gas theory and the Stefan-Boltzmann law were used to estimate the temperature of stars. Remarkably, the kinetic theory developed by Boltzmann to describe ideal gases is still used today to describe plasmas in fusion devices.

But how can you measure something you cannot touch? The history goes back to 1666 when Newton separated the light of the Sun into a continuous "spectrum" of colors, thus initiating the field of spectroscopy. Later, in 1814, Fraunhofer systematically studied the existence of black lines in the light spectrum, and Kirchhoff identified these lines with different atomic compositions years later. Today, we know, thanks to Cecilia Payne-Gaposchkin's thesis (1925), that the Sun's most abundant element is hydrogen ($\approx 3/4$), followed by helium ($\approx 1/4$). Once the composition of the Sun became better understood, Hans Bethe proposed a detailed theory of nucleosynthesis involving several fusion reactions in 1938. Remarkably, the development of spectroscopy was also instrumental in the development of modern quantum theory, which is necessary to describe collisions between atomic nuclei. Specifically, quantum tunneling is needed to account for the correct probability of fusion reactions occurring during collisions.

How can we reproduce fusion reactions on Earth to obtain energy in a controlled way? We need to confine a plasma under controlled and sustained conditions so that the fusion reaction can occur, i.e., a fusion reactor. The conditions for plasma density, temperature, and confinement time were established by Lawson [Law57], and they are detailed later in this chapter. The first man-made controlled fusion reactions were achieved in particle accelerators led by Mark Oliphant at Cavendish Laboratory in 1932. However, in this approach, the energy input is much greater than the energy output.

The first experiments aimed at developing a fusion reactor were variants of pinch machines, in which a current $\mathbf{j}$ is discharged through a gas, ionizing it and creating a magnetic field $\mathbf{B}$ that pulls the ions together through a compressing force $\mathbf{j} \times \mathbf{B}$. However, the first plasma instabilities (particularly kink instabilities) were observed in these devices.

The Stellarator machine was introduced by Spitzer in 1958 [Spi58]. The twisted shape of the magnetic coils in the machine theoretically compensates for the particle drifts by adding a rotational transform, "a twist" to the magnetic field. Other approaches considered various magnetic mirror configurations to trap the particles.

Tokamak is an acronym derived from Russian, meaning "**to**roidal **cha**mber with

magnetic **c**oils.” The “twist” in the magnetic field is produced by a current induced in the plasma by a central solenoid. The first proper tokamak was constructed at the Kurchatov Institute of Atomic Energy and named T3. In 1965, Lev Artsimovich presented the tokamak at an IAEA conference in Culham. Soviet scientists had been working with this approach for a decade in isolation, as nuclear research was classified until the Atoms for Peace conference in 1958. However, their plasma temperature measurements were not reliable. A team from the UK visited the USSR to use their Thomson scattering technique to measure the plasma temperature. It was a success, and the results were published in 1968 [Pea+69], marking a leap forward in plasma confinement. The tokamak subsequently became the leading device in fusion research. Its axisymmetry favored both ease of construction and the theoretical development of plasma physics.

During the next decades, many experiments followed. The next generation of large tokamaks was constructed in the 1980s: TFTR in Princeton, JET and ASDEX in Europe, and DIII-D in California. A significant event during this decade was the discovery of the high-confinement mode in ASDEX (1982) [Wag+82; Wag07], a plasma configuration that improves the achievable density and temperatures.

With the turn of the millennium, the next generation of superconducting tokamaks began operations: EAST in China (2006), KSTAR in Korea (2007), and JT-60SA in Japan (2023). Recently, fusion ignition was achieved using inertial confinement at NIF [Abu+22; Zyl+22]. Moreover, private investment has started to be attracted to nuclear fusion, and the number of start-ups is growing. Overall, there are good reasons for optimism at this time.

The ITER project began its technical design phase in the 1990s and is now under construction in the south of France. ITER represents a critical milestone in demonstrating the feasibility of nuclear fusion as a large-scale, carbon-free energy source. Its aim is to achieve self-sustaining ’burning plasmas,’ where fusion reactions primarily heat the fuel, by operating plasmas with currents up to 15, MA, magnetic fields up to 5.3, T, and the highest plasma volume ever produced, with almost 1000 m^3. Moreover, it serves as a technological demonstrator for overcoming construction and operational challenges. Until it is commissioned, research on fusion physics, materials that can withstand plasma conditions, and improving diagnostic methods, to name a few, continues all over the world.

The Stellarator continued to be studied in some laboratories. The lack of axisymmetry made its construction and design more difficult. With the development of computational power in recent decades, it became possible to optimize the design of coils and fields. In 1998, the Large Helical Device (LHD) began operations in Japan, the same year as TJ-II in Spain. The Helically Symmetric Experiment (HSX) was commissioned in 1999 in the US. Recently, Wendelstein 7-X began operations, validating the optimization models and construction process [Ped+16]. Because of the absence of an externally induced plasma current that drives many of the instabilities and disruptions in tokamaks, among other advantages [Boo21], interest in Stellarators has been renewed.

Not only is it necessary to achieve high density, temperature, and confinement time, but it is also essential to maintain steady operation with controlled instabilities. An instability is a plasma perturbation that can grow over time, disrupting confinement. As mentioned earlier, the first type of instabilities discovered were macroscopic (kink), which were understood using magnetohydrodynamic (MHD) theory. Modern MHD theory was born in 1942 [Alf42] with Alfvén's realization of a new kind of coupled fluid-field wave. This was another gift from the Sun, as the first use of Alfvén waves was to explain the velocity displacement of sunspots [Alf42]. Once macroscopic instabilities were reasonably understood, microscopic instabilities (neoclassical transport, turbulence, and other anomalous transport) became new lines of research. The frontier is now to understand and predict the behavior of burning plasmas, which are predominantly heated by fusion reactions and will occur in future fusion devices.

In this chapter, we introduce the basic physics of fusion to understand the experimental data involved in this thesis work. We begin with fusion reactions and basic magnetic confinement theory. Next, a description of MHD waves is necessary to understand the types of instabilities studied in this thesis. The chapter concludes with the problem statement of this thesis work.

1.2. Fusion reactions

Nuclear fusion energy results from the reaction between two atomic nuclei. The most easily achievable fusion reaction on Earth is between two different isotopes of hydrogen: Deuterium (D) and Tritium (T). The nuclear process concludes with the fusion of the two isotopes, resulting in the formation of Helium (He) isotopes and the release of neutrons.

The reaction can be represented by the following equation:

$$D\,({}^{2}_{1}\mathrm{H}) + T\,({}^{3}_{1}\mathrm{H}) \rightarrow {}^{4}_{2}\mathrm{He}(3.5MeV) + {}^{1}_{0}\mathrm{n}(14.1MeV). \tag{1.1}$$

The difference in masses between the reactants and the products is converted into pure kinetic energy, as follows from the famous Einstein equation:

$$E = mc^2 \tag{1.2}$$

The kinetic energy is distributed between the alphas (He nuclei) and the neutrons, 3.5 MeV for each alpha particle and 14.1 MeV for each neutron.

The main advantage of fusion fuel is its high energy gain per unit of fuel mass. It is the densest energy source accessible to humankind. The energy gain per unit of mass is greater in fusion reactions than in fission reactions or chemical reactions.

$$D - T \text{ fusion}: \ 17.6 \text{ MeV} \rightarrow 3.5 \text{ MeV/amu} \tag{1.3}$$

$$U \text{ fission}: \sim 200 \text{ MeV} \rightarrow 0.85 \text{ MeV/amu} \tag{1.4}$$

$$\text{Carbon combustion}: \sim 1 \text{ eV} \rightarrow 0.08 \text{ eV/amu} \tag{1.5}$$

As we can see, nuclear reactions release roughly 10 million times more energy than chemical reactions, and since the combustion of hydrocarbons is the basis of our energy source, their substitution by nuclear fuels would not only result in a more efficient energy source but could also help mitigate the problems of massive burning of hydrocarbons.

For the fusion reaction to take place, Deuterium and Tritium must collide; in other words, they should approach each other at a necessary distance and speed. Both isotopes have positive charges, so they repel each other with the electric force, which has a very long action distance. Only when the short-range attractive nuclear force can overcome the electrical force and keep the nuclei together, can nuclear reactions take place. The nuclear particles naturally seek a state of minimum energy, emitting a neutron and redistributing the difference of mass into kinetic energy.

The densities, temperatures, and confinement time necessary for the fusion reaction to occur are given by Lawson's criteria, which for D-T reaction is expressed as a triple product [WC11]:

$$n T \tau_E \geq 3 \times 10^{21}\,\mathrm{m}^{-3}\,\mathrm{keV\,s}, \tag{1.6}$$

where n is the plasma density (in particles per cubic meter), T is the plasma temperature (in keV), and τ_E is the energy confinement time (in seconds). When the reactants are in thermal equilibrium and follow a Maxwellian distribution, the process is referred to as thermonuclear fusion. These reactions are said to be thermonuclear. When density plays the dominant role, the reactants undergo inertial fusion. In this thesis, we focus on data from magnetic confinement devices that aim to achieve thermonuclear fusion.

1.3. What is plasma

There are four states of matter: solid, liquid, gas, and plasma. At low enough temperatures, most substances are solid, and atoms or molecules pack together in an ordered structure. When kinetic energy is given to the particles, their velocity increases. First, they start vibrating relative to their minimum energy positions, and progressively the ordered structure begins to lose periodicity and form. The particles are then free to move while occupying a similar volume, colliding with one another. At this point, the liquid state is reached. Following the liquid state, if the temperature is increased further, particles travel longer distances and collide more frequently. If they occupy the maximum volume possible, the matter becomes a gas. During the processes of melting and evaporation, atoms remain neutrally charged.

Finally, if more energy is given to gas particles through collisions, two phenomena can occur: first, if the energy is below a certain threshold, electrons excite to higher atomic orbits, emitting electromagnetic radiation when they return to lower energy states. Second, if the energy exceeds a threshold known as ionization energy, which depends on the atomic structure, collisions between atoms remove electrons from their orbits, allowing nuclei and electrons to move separately. This gas, composed of charged particles that act together in a collective manner, is known as plasma. Due to the long-range action of electric forces and the high speed of particles, very complex collective behavior emerges, as plasma particles try to compensate for every small volumetric charge difference. This process is illustrated in the following (classic) cartoon.

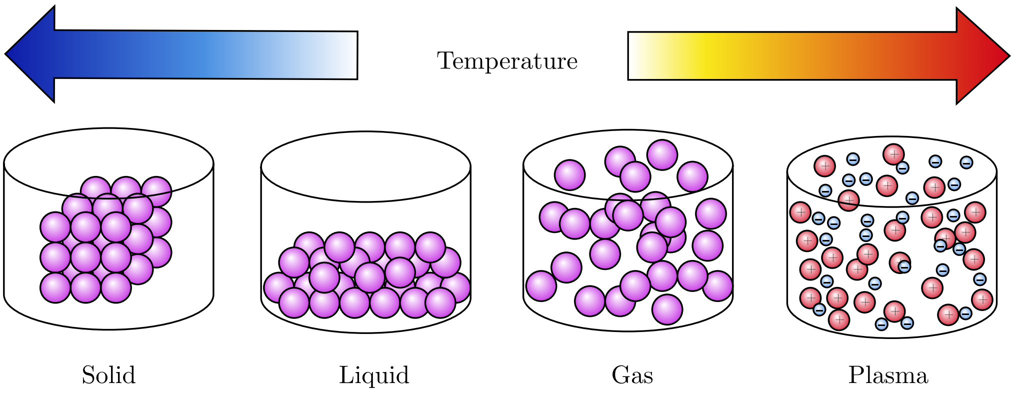


Figure 1.1.: States of matter. Adapted from [Sha+13].

After this description of the change of states in matter, we can take from Chen's book [Che18] a definition of plasma: "*A plasma is a quasineutral gas of charged and neutral particles which exhibits collective behavior*".

It is important to note that it is not necessary for the gas to be fully ionized to behave like a plasma. The degree of ionization is modeled by Saha's equation [Che18], $n_i / n_n \propto (T^{3/2} / n_i) e^{-U_i / k_B T}$. This equation shows how the ratio of ionized particles n_i to neutral atom densities n_n depends on temperature T, their ionization energy (the energy required to ionize an atom), and the Boltzmann constant k_B. As temperature increases, the ratio of ionized particles n_i to neutral particles n_n also increases, meaning more particles become ionized at higher temperatures.

This state of matter has unique properties. Because most substances we perceive in our daily lives are in solid, liquid, or gaseous states, some plasma physical properties appear counterintuitive. However, most matter in the universe is actually in the plasma state. For instance, ions and electrons act as different particle populations, so they have different distribution functions, meaning they have different velocities $V_i \approx \sqrt{(T_i / m_i)}$ and $V_e \approx \sqrt{(T_e / m_e)}$. The velocity of electrons is on the order of 40 times higher than that of ions, $V_e > 40 \cdot V_i$. An example of plasma is the gas contained in neon lamps, which can be held with a bare hand. This is possible thanks to the low ionic temperature, while the electronic temperature can reach 20,000 K [Che18]. In contrast, the plasma in ITER is expected to have $V_i \approx 1000$ km/s and $T \approx 100$ MK. It is also important to note that in a plasma mixture of different species, ions of each species can have different temperatures and density distributions.

The definition of plasma needs to be done in quantitative terms, as not every ionized gas behaves like plasma. The following three conditions must hold [Che18; Rax11]:

$$\begin{cases} \lambda_D = 69\sqrt{(T_e/n)} & \ll L \quad \text{(quasineutrality)} \\ N_D = n\frac{4}{3}\pi\lambda_D^3 & \ggg 1 \quad \text{(collective behavior)} \\ \omega \cdot \tau & > 1 \quad \text{(enough collisions)} \end{cases}$$

where λ_D is the Debye length, ω is the proper frequency of plasma waves, τ is the mean time between collisions, N_D is called the plasma parameter, representing the number of particles inside a Debye sphere, and L is the length of the system.

Quasineutrality refers to the property of plasma in which the densities of positive and negative charges are nearly equal over large volumes. In other words, even though the plasma contains charged particles—electrons and ions—on average, the plasma is electrically neutral when viewed on macroscopic scales. This is because the positive charges (due to ions) and negative charges (due to electrons) tend to cancel each other out. However, small local deviations from neutrality can exist on very short length scales.

The condition $\lambda_D \ll L$ is a mathematical expression of quasineutrality, where λ_D, the Debye length, is much smaller than the overall size of the plasma L. The Debye length is the scale over which electric fields within the plasma can affect charged particles. Beyond this length, the shielding effects of surrounding charges neutralize any electric field generated by an individual particle. Essentially, quasineutrality implies that the plasma behaves as if it is neutral beyond a distance of λ_D, even though it's composed of charged particles.

Collective behavior in a plasma refers to the fact that individual particle interactions are influenced by the presence of many other charged particles, leading to behavior that cannot be understood by considering just a few particles in isolation. This is different from neutral gases, where individual particle collisions can be treated independently.

The final condition $\omega \cdot \tau > 1$ ensures that the gas exhibits plasma-like behavior instead of behaving as a neutral gas [Che18]. Here, ω is the natural frequency of plasma oscillations, and τ is the mean time between collisions. For the plasma to behave collectively and maintain its quasineutrality, the particles must interact frequently enough. When $\omega \cdot \tau > 1$, the particles are colliding frequently compared to the characteristic timescale of plasma oscillations, ensuring that the plasma exhibits collective effects.

The proper plasma oscillations described by ω tell us how fast the electrons or ions in the plasma will oscillate if they are displaced from their equilibrium positions. This is a characteristic frequency of the plasma and is independent of temperature, depending only on the electron or ion density. For instance, the electron plasma frequency is of the order of gigahertz (GHz) for typical laboratory plasmas, much higher than the frequency of MHD waves, which are discussed later in this thesis.

It is interesting to know that plasma is the most abundant state of visible matter in the universe: stars, nebulas, and galaxies. Although the plasma state can endure only in a vacuum, and we live in an atmosphere that does not favor its existence, plasma is present in many earthly phenomena: fire, lightning, auroras, some types of televisions, and fluorescent tubes. Plasma can exist at many temperature scales, from a few hundred degrees Kelvin to several hundred million in Tokamak reactors. The electron N_D varies from 10^4 m^{-3} in interstellar space to 10^{31} m^{-3} in the solar core, λ_D can vary from 100 m in interstellar space to 1 μm in the stellar core, and the proper frequencies ω vary from a few 10^4 rad/s to 10^{14} rad/s. A diagram indicating the different types of plasmas and their parameters can be found on page 12 of Ref. [BB04].

1.4. Plasma confinement

Electric particles moving in a homogeneous magnetic field experience a force perpendicular to the field and their velocity, as depicted in (Fig. 1.2 - right). The Lorentz force equation describes their movement accurately:

$$F = eZ(\mathbf{E} + \mathbf{v} \times \mathbf{B}). \tag{1.7}$$

The Lorentz force does not have a component along the magnetic field B, it is always directed perpendicular to the magnetic field. It causes the particle to have a helical movement around the field lines (Fig. 1.2 - left). The helical trajectory of these particles is called Larmor orbits, or gyro-orbits.

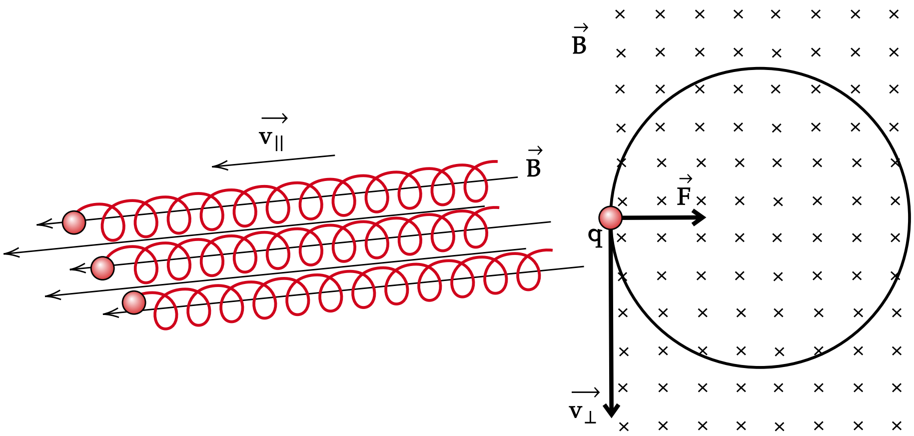


Figure 1.2.: Lorentz force: parallel and perpendicular velocities motion of particles in a magnetic field. The resultant combined motion is helical (Larmor orbits on the left).

The Larmor radius $\rho = \frac{mv}{\|q\|B}$ and the cyclotron frequency $\omega = \frac{\|q\|B}{m}$ describe the circular movement component (also known as gyro-radius and gyro-frequency, respectively). The Larmor radius depends on the respective kinetic energies of the electrons and ions. The linear movement depends on the particle velocity component along the field lines. The cyclotron frequency is ≈150 GHz for the electrons and ≈40 MHz for the ions.

Several magnetic configurations allow partial confinement of charged particles. For instance, the cyclotron was one of the first particle accelerators. It used a constant magnetic field to confine the particles and accelerate them in a spiral trajectory before releasing them into the target. Since charged particles follow magnetic field lines with a parallel velocity, it is natural to think about bending the magnetic field to confine the particles. Another example is the magnetic mirror: two parallel rings of currents can create a potential magnetic well that can partially confine particles. However, for the case of fusion, we need a special magnetic field topology configuration so we can confine particles long enough (many orbits) while they are heated and collide. We need to bend the field lines so all particles are confined in a volume. The tools we have to do so are external magnets and the magnetic field produced by the plasma current itself.

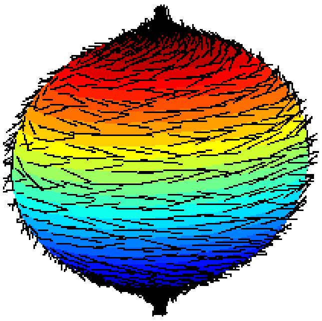


Figure 1.3.: Hairy ball: the vector field is not confined on the surface (from [24d]).

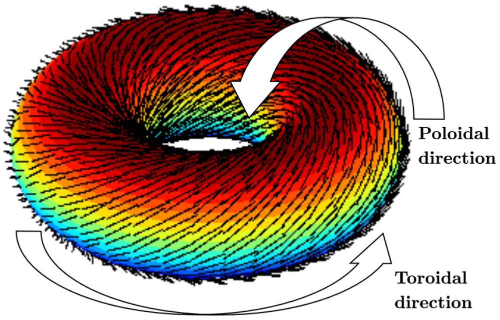


Figure 1.4.: Hairy torus: the vector field is confined to the surface (from [24d]). Also the toroidal and poloidal directions are indicated.

Two theorems from topology explain how to configure a vector field on a surface so that it does not have singularities. The hairy ball theorem tells us that a vector field cannot be confined to a spherical surface without creating a singularity (any continuous tangent vector field on the sphere has at least one point where the field is zero [BK17]), see Fig. 1.3. In contrast, the geometrical shape that allows us to “comb” magnetic field lines and close them is a torus, Fig. 1.4. Note how the field lines twist around the torus. These helicoidal magnetic lines are closed, and particles can theoretically circulate the torus without leaving it.

The hairy ball theorem is a special case of the Poincaré-Hopf theorem, which tells us that [Ric08] “for any vector field on a closed surface S with only finitely many zeros, the sum of the indices of all zeros equals the Euler number of the surface”. Without entering into the details of how to calculate the indices and the Euler characteristic, we can say that the Euler characteristic of the 2-sphere is two [Ric08]. Therefore, there must be at least one zero. However, in the case of the torus, the Euler characteristic is

0; therefore, it is possible to create a field without zeros and "comb a hairy doughnut flat". It is important to note that this is only a necessary condition; not all vector fields that can be held in a torus are free of zeros.

The Lorentz force and the hairy ball theorem help build an intuition of how one particle may follow this ideal field configuration. However, we have ignored many physical factors: the situation becomes more complex if we consider a system of particles, as the electric field causes them to interact with each other. In addition, the Lorentz force assumes a constant field. In real scenarios, the trajectory passes through inhomogeneous fields, experiencing additional force components called drifts. Moreover, accelerated charges emit radiation, losing kinetic energy. It is worth mentioning that, since the Larmor radius is different for electrons and ions, they will have different trajectories.

In Fig. 1.4, the toroidal and poloidal directions are indicated. The toroidal direction is the direction around the torus, and the poloidal direction is the direction in the plane perpendicular to the toroidal direction. The toroidal direction is toroidal direction is the long direction around the "hole" in teh torus. The poloidal direction is the short way around the "bend" cylinder.

In this text, we will not discuss particle drifts in detail; they are discussed in many plasma physics references. However, it is important to mention the basic cases. A Larmor orbit can be drifted: in the presence of external forces, when the magnetic field is inhomogeneous, in the presence of an electric field, when a static magnetic field has curvature, or when the electric field is time-dependent. The drift directions depend on the charge sign in the case of drifts due to magnetic field gradients and field curvature. Therefore, when ions and electrons are present in an inhomogeneous curved magnetic field, they drift in opposite directions, causing a charge separation.

There is another reason, besides topology, to choose a torus. If the field is only toroidal, one might think that particles can be confined in the toroidal direction, orbiting around a center. However, this configuration has a problem: the particles will drift away from the torus because of the presence of an inhomogeneous curved magnetic field. In Fig. 1.5, an "only toroidal field" device is shown (left), along with the tokamak (center) and stellarator (right) field configurations. The toroidal magnetic field has a gradient in the radial direction, which produces a charge separation, and the electric field generated by this charge separation will cause a $\mathbf{E} \times \mathbf{B}$ drift, pushing the particles away from the torus.

To cancel the $\mathbf{E} \times \mathbf{B}$ drift, the idea is to average the drift $\mathbf{E} \times \mathbf{B}$ for each Larmor orbit around the torus so that the result is zero. It is necessary to add a "twist" to the toroidal field line so that the drift $\mathbf{E} \times \mathbf{B}$ rotates around the torus. This way, the drift for each Larmor orbit is compensated around the torus. This has been often compared to holding honey with a spoon; if the spoon is straight, the honey will fall, but if the spoon is twisted, the honey will remain.

In other words, it is necessary to add a poloidal field component to the magnetic field so that $\mathbf{B} = \mathbf{B}_{\text{toroidal}} + \mathbf{B}_{\text{poloidal}}$. This poloidal field can be generated by external coils, as in the case of the stellarator, or by the plasma current itself, as in the case of the tokamak. This is depicted in Fig. 1.5.

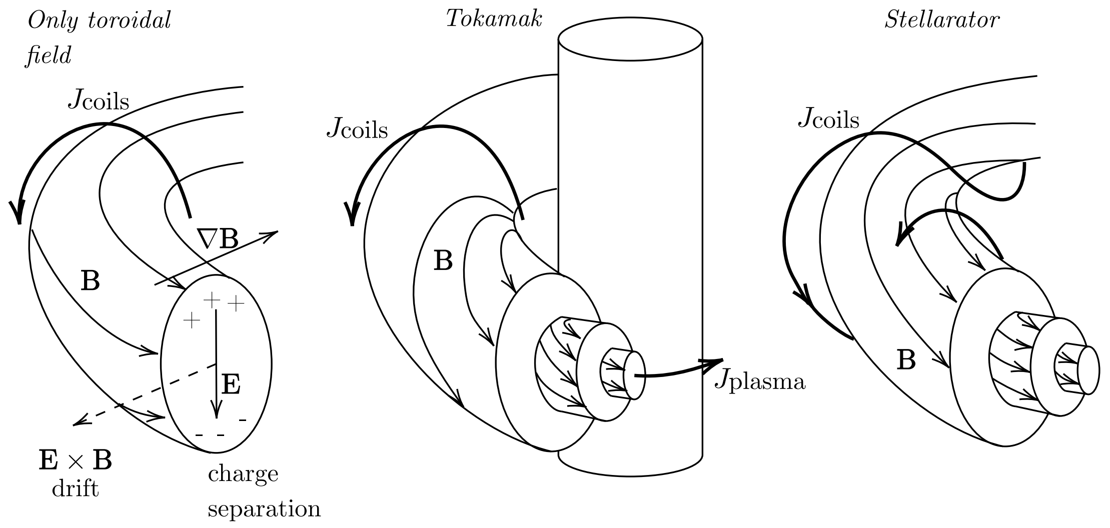


Figure 1.5.: Toroidal field vs Tokamak vs Stellarator magnetic configurations. Adapted from [Igo14].

In the case of the tokamak, the plasma current is generated by a central solenoid, which induces a current in the plasma. The plasma current $\mathbf{J}_{plasma}$ generates the poloidal magnetic field. The orientation of the poloidal field is determined by the right-hand rule: imagining the thumb of the right hand pointing in the direction of the current $\mathbf{J}_{plasma}$, closing the hand curls the fingers in the direction of the magnetic field.

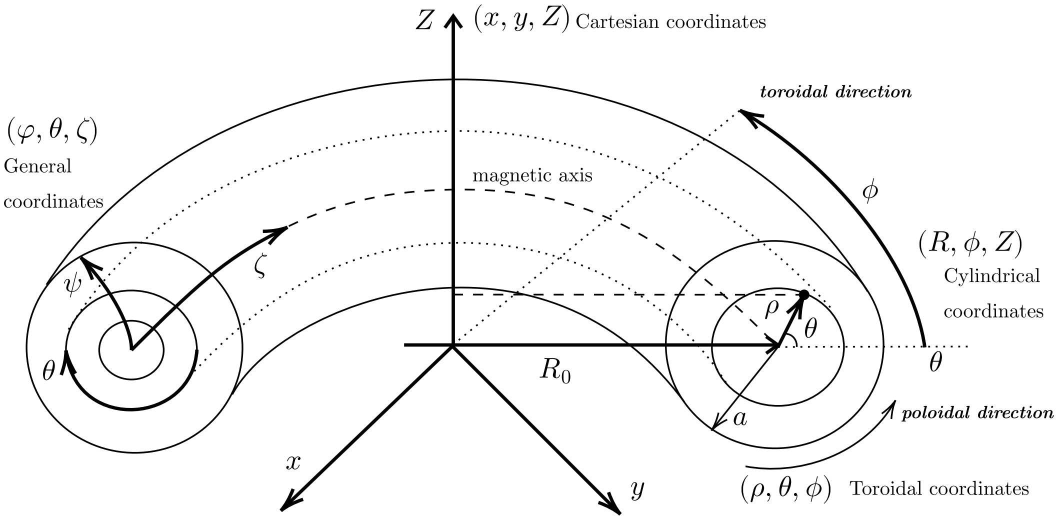


Figure 1.6.: Toroidal geometry and alternative coordinate systems: general coordinates (or curvilinear) (φ, θ, ζ), cartesian (x, y, Z), polar (R, ϕ, Z), and toroidal (ρ, θ, ϕ). Adapted from [Sha+13]

To describe the magnetic field in a torus, it is necessary to choose an appropriate coordinate system. Alternative systems are shown in Fig. 1.6. In a torus, the two main directions are the toroidal and poloidal directions. The toroidal angle is the angle around the torus, and the poloidal angle is the angle in the plane perpendicular to the toroidal direction. The major radius is the distance from the center of the torus to the magnetic axis; the minor radius is the distance from the magnetic axis to the plasma. The aspect ratio of the device is given by the ratio of the major to the minor radius.

The Cartesian coordinates (x, y, Z) are the most familiar system, but they are not ideal for describing the toroidal geometry of a fusion device. The magnetic field lines in these devices follow curved paths, making Cartesian coordinates cumbersome for representing the circular and helical shapes inherent to the plasma. The polar coordinates (R, ϕ, Z) are an extension of Cartesian coordinates, which can be useful for some applications in axisymmetric systems like tokamaks.

The toroidal coordinates (ρ, θ, ϕ) are particularly useful for the torus-shaped geometry of tokamaks. In this system, ρ is the radial distance from the magnetic axis, θ is the poloidal angle (around the small circular cross-section), and ϕ is the toroidal angle (around the large axis of the torus). This system aligns better with the magnetic field structure, where field lines follow helical paths in concentric surfaces.

The general coordinates (curvilinear coordinates) (φ, θ, ζ) are often used in stellarators, which lack the axisymmetry of tokamaks, or in tokamaks when 3D effects are considered. Unlike toroidal coordinates, curvilinear systems can account for the helical, non-symmetric magnetic field lines. Their rigorous definition requires tensor calculus, which is out of the scope of this introduction. A specialized monograph on the subject is Ref. [Dha12] and for the sign conventions Ref. [SM13].

1.4.1. Magnetohydrodynamics (MHD) equilibrium

Three lines of research in plasma physics derive from the triple product [Fre14]: heating, transport, and MHD. However, density and temperature are roughly proportional to plasma pressure, which is studied by magnetohydrodynamics theory, while confinement time (transport) is studied by kinetic theory.

In kinetic theory, the Boltzmann equation describes the evolution of the distribution function f, which represents the probability of finding a particle with a given velocity at a given position. The Boltzmann equation includes the effects of external forces and collisions:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \frac{\mathbf{F}}{m} \cdot \frac{\partial f}{\partial \mathbf{v}} = \left(\frac{\partial f}{\partial t} \right)_{\text{coll}} \tag{1.8}$$

where the force $\mathbf{F}$ acting on a particle with charge q is given by the Lorentz force:

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}), \tag{1.9}$$

and the collision term $\left(\frac{\partial f}{\partial t}\right)_{\text{coll}}$ accounts for particle-particle interactions.

It is worth noting that MHD equations can be derived from kinetic theory ([GP04] chapter 3) by appropriate averaging. The MHD equations describe the macroscopic behavior of the plasma, treating it as a continuous fluid rather than a distributions of individual particles. That is why MHD is appropriate for describing the equilibrium and stability of magnetically confined plasmas.

Many references offer a more detailed and rigorous treatment of MHD and kinetic theory with applications to magnetic confinement devices, such as Refs. [Fre14; Rax05; Rax11; BB04; GP04; GKP10; Zoh15; HM03; Jar10; WC11]. It is important to note that there are many variations or “flavors” of MHD models. The simplest model is ideal MHD, which considers the plasma as a single superconducting fluid.

The MHD equations are a set of coupled, highly non-linear differential equations that describe the behavior of a magnetized plasma as a fluid. These equations are derived from first principles like the conservation laws of mass, momentum, and energy, combined with Maxwell’s equations for electromagnetism, or by averaging the kinetic equations.

The MHD equations are valid when [Fre14]: the plasma is highly collisional, characteristic dimensions much larger than an ion gyro radius and the resistive effects can be neglected despite increased resistivity from collisions. They also describe low-frequency phenomena, where MHD frequencies are much smaller than the gyro frequency of the particles and the proper frequencies of the plasma, as mentioned in the order of GHz.

In MHD, the plasma behavior is described by a set of equations that combine the principles of electromagnetism and fluid dynamics. These equations govern the behavior of the magnetic fields, electric fields, and the motion of plasma particles. The change in particle orbits (gradients of pressure) can create what are called diamagnetic fields and diamagnetic currents. The fields themselves are responsible for the plasma confinement and the particle orbits. Perturbations to the field can create waves, and the particles can interact with the waves and vice versa. This phenomenon is all coupled together. However, the coupling between individual particle orbits and MHD waves is not directly modeled by ideal MHD.

The MHD equations include Maxwell’s equations for modeling the electromagnetic fields. The Gauss law for electric fields is given by:

$$\nabla \cdot \mathbf{E} = \frac{\rho_e}{\epsilon_0}, \tag{1.10}$$

where ρ_e is the charge density. In MHD, where the net charge density ρ_e is often negligible (since the plasma is quasineutral), this simplifies to:

$$\nabla \cdot \mathbf{E} = 0. \tag{1.11}$$

The magnetic field **B** is divergence-free, leading to the condition:

$$\nabla \cdot \mathbf{B} = 0. \tag{1.12}$$

In other words, there are no magnetic monopoles, and the magnetic field lines form closed loops.

Faraday's law of induction describes how changing magnetic fields induce electric fields:

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}. \tag{1.13}$$

Ampère's law, which relates the magnetic field to the electric current density **J**, is:

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J} + \mu_0 \epsilon_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{1.14}$$

In MHD, the displacement current term $\mu_0 \epsilon_0 \frac{\partial \mathbf{E}}{\partial t}$ is often negligible due to the timescales involved, so Ampère's law simplifies to:

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J}. \tag{1.15}$$

The generalized Ohm's law in MHD is given by:

$$\mathbf{E} + \mathbf{v} \times \mathbf{B} = \eta \mathbf{J}, \tag{1.16}$$

where η is the resistivity, and **v** is the plasma velocity. In ideal MHD, the resistivity is zero ($\eta = 0$), and the fluid is superconducting. Therefore, ideal MHD does not model resistivity effects like magnetic reconnection. This equation represents how the electric field **E** is influenced by the magnetic field and the motion of the plasma.

The continuity equation, which expresses mass conservation, is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \tag{1.17}$$

The momentum equation (also called the MHD Euler equation in [Rax11]), which describes the force balance on a plasma element, is:

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\frac{1}{\rho} \nabla p + \frac{1}{\rho} (\mathbf{J} \times \mathbf{B}), \tag{1.18}$$

where $\mathbf{J} = \frac{1}{\mu_0} \nabla \times \mathbf{B}$. This equation shows that plasma motion is influenced by both pressure gradients and the Lorentz force $\mathbf{J} \times \mathbf{B}$. Note that $q\mathbf{E}$ in the Lorentz force is a current.

The magnetic induction equation describes how the magnetic field evolves over time:

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B}) - \nabla \times (\eta \mathbf{J}). \tag{1.19}$$

This equation also describes the convection and diffusion of the magnetic field (this equation is called the convection-diffusion equation in [Rax11]), where the first term represents convection created by the plasma flow, and the second term represents the diffusion of the magnetic field due to resistivity, which again is zero for ideal MHD.

The equation of state relates the pressure p to the plasma density ρ and temperature

T:

$$p = p(\rho, T). \tag{1.20}$$

Some models assume an ideal gas equation of state, where $p = (\rho/m)k_B T$, with k_B being the Boltzmann constant. This leads to the energy equation $(\mathrm{d}/\mathrm{d}t)\left(p/\rho^{\gamma}\right) = 0$, where γ is the ratio of specific heats, and $\mathrm{d}/\mathrm{d}t$ is the fluid derivative.

The necessary condition for equilibrium is that time derivatives of the vector field quantities are zero. Setting the time derivatives to zero in Ampère's law and the momentum equation gives the MHD conditions for equilibrium:

$$\begin{cases} \mathbf{J} \times \mathbf{B} & = \nabla P \\ \nabla \times \mathbf{B} & = \mu_0 \mathbf{J}, \end{cases} \tag{1.21}$$

These conditions are valid for both tokamaks and stellarators. The first equation describes the force balance between the pressure gradient and the Lorentz force, while the second equation governs the relationship between the magnetic field and the plasma current. Fig. 1.7 shows the force equilibrium in a tokamak, where the magnetic field confines the plasma.

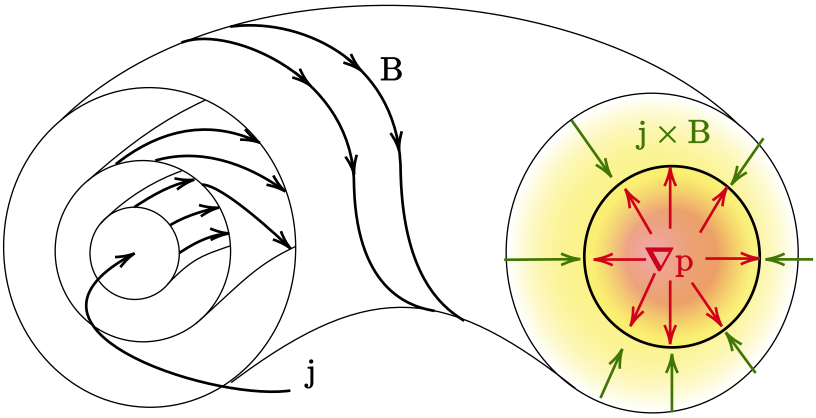


Figure 1.7.: Equilibrium of forces in a tokamak

In equilibrium, the plasma satisfies conditions like $\mathbf{J}\cdot\nabla P = 0$ and $\mathbf{B}\cdot\nabla P = 0$, meaning that the current and magnetic fields are aligned with the pressure gradient isobaric surfaces. This fact can be used to construct a coordinate system that follows flux lines [HM03; Rax11]. The magnetic field can be expressed in flux coordinates as:

$$\mathbf{B} = \frac{1}{2\pi}(\nabla\Psi_T \times \nabla\theta - \nabla\Psi_P \times \zeta) \tag{1.22}$$

where Ψ_T and Ψ_P are the toroidal and poloidal fluxes, respectively. The fluxes are calculated by either surface integrals or volume integrals of the magnetic field as in Ref. [HM03] pages 53ff. When the equilibrium is axisymmetric (as in tokamaks), the magnetic surfaces are nested toroids like in Fig. 1.7. In stellarators, the existence of nested surfaces depends on the external coil design. It is worth noting that there are

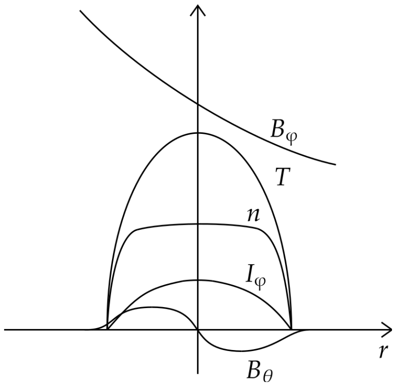


Figure 1.8.: Typical plasma profiles of temperature T, density n, toroidal field B_φ, poloidal field B_θ, and toroidal current B_φ varying with minor radius r. Adapted from Ref. [Rax11].

alternative flux coordinate definitions, for example, Boozer coordinates or Hamada coordinates, which are appropriate for stellarators and 3D equilibria in tokamaks. As explained in Hazeltine's book [HM03] there are also straight-line coordinates: for an arbitrary system of coordinates, if the field lines are drawn in a ζ-θ plane, the field lines look curved. However, if "straight-line" coordinates are used, each field line looks straight in the Ψ_T-Ψ_P plane. To summarize, we have Cartesian, polar, toroidal, curvilinear coordinates, and some flux coordinate alternatives, each appropriate for different purposes.

When there is axisymmetry, the MHD equilibrium of Eq. 1.21 can be transformed into the Grad-Shafranov equation [Fre14; Rax11]. The equilibrium can then be solved by taking into account the poloidal flux, the pressure profile, and the poloidal current profiles. This task is often done with numerical codes such as HELENA [HGK+91; Goe81; Poe+96]. Although this equation is nonlinear and usually solved numerically, there are some cases where analytic solutions can be found (Solovev solutions).

In the case of stellarators, the MHD equilibrium described by Eq. 1.21 has to be solved directly, and the 3D plasma confinement theory mathematics are more complex [Hel14]. 3D equilibria is solved numerically with codes such as VMEC [HW83; HVM86] or its variants [Gar90; Str97; Spo+01].

Many quantities help to characterize the plasma equilibrium that cannot be measured directly. The use of a equilibrium reconstruction like EFIT or a surrogate model [Lao+22] is necessary. Plasma profiles, such as temperature T, density n, the toroidal current I_φ, and magnetic fields B_φ and B_θ, vary with the minor radius r of the plasma, as shown in Fig. 1.8.

The plasma β represents the ratio of plasma pressure to magnetic pressure (from [WC11]):

$$\beta \equiv \frac{\text{thermal pressure}}{\text{magnetic pressure}} = \frac{p}{B^2/2\mu_0}, \tag{1.23}$$

The plasma β is a measure of the plasma pressure relative to the magnetic pressure, and it quantifies the importance of plasma pressure in the equilibrium of the plasma.

A high β indicates that the plasma pressure is significant compared to the magnetic pressure, which can affect the stability and confinement of the plasma. It is important to note that this quantity varies with the minor radius r of the plasma, so alternative averaged definitions of β exist.

The safety factor q, which describes the twisting of magnetic field lines in the tokamak, is defined as:

$$q \equiv \frac{d\Psi_T}{d\Psi_P} = \frac{d\zeta}{d\theta}. \tag{1.24}$$

The q factor is usually set slightly greater than 1 in the core of the plasma to ensure stability against internal kinks, and it increases monotonically toward the edge of the plasma.

The rotational transform ι (the q-profile of stellarators) describes the number of toroidal rotations per poloidal rotation:

$$\iota \equiv 2\pi \frac{d\Psi_P}{d\Psi_T} = 2\pi \frac{d\theta}{d\zeta}. \tag{1.25}$$

The reason to define ι for stellarators is likely due to the fact that the q factor evolves toward negative values at the edge (the shear is usually reversed in stellarators compared with tokamaks), and ι gives a monotonically increasing quantity.

The magnetic shear s, which measures the variation of the safety factor with radius, is given by:

$$s \equiv \frac{dq}{d\psi_P} = \frac{d^2\psi_T}{d\psi_P^2}. \tag{1.26}$$

Magnetic shear is beneficial for plasma stability, as it prevents the propagation of perturbations along the perpendicular field direction. A reversed shear is key for the production of internal transport barriers (ITB) in advanced tokamak scenarios [Sha21].

Finally, at certain surfaces the fields join up with them selves again when there is an integer number of toroidal (n) and poloidal turns. This happens when the safety factor q is rational, given by:

$$q \equiv \frac{d\zeta}{d\theta} = \frac{m}{n}, \tag{1.27}$$

where m and n are integers, representing the poloidal and toroidal mode numbers, respectively. It can be shown, using Hamiltonian theory, that resonant magnetic perturbations (see [Rax11], pages 165-167) are located in this surfaces.

Therefore, the q-profile, rotational transform ι, and magnetic shear s are essential for understanding the behavior of the plasma and designing magnetic confinement devices. It is important to note that these quantities are dimensionless. By controlling these parameters, scientists and engineers can optimize plasma stability and confinement, leading to more efficient and reliable fusion reactors. It is important to note that these quantities are not measured directly, and their values are inferred from equilibrium reconstruction codes.

1.4.2. Tokamaks vs Stellarators

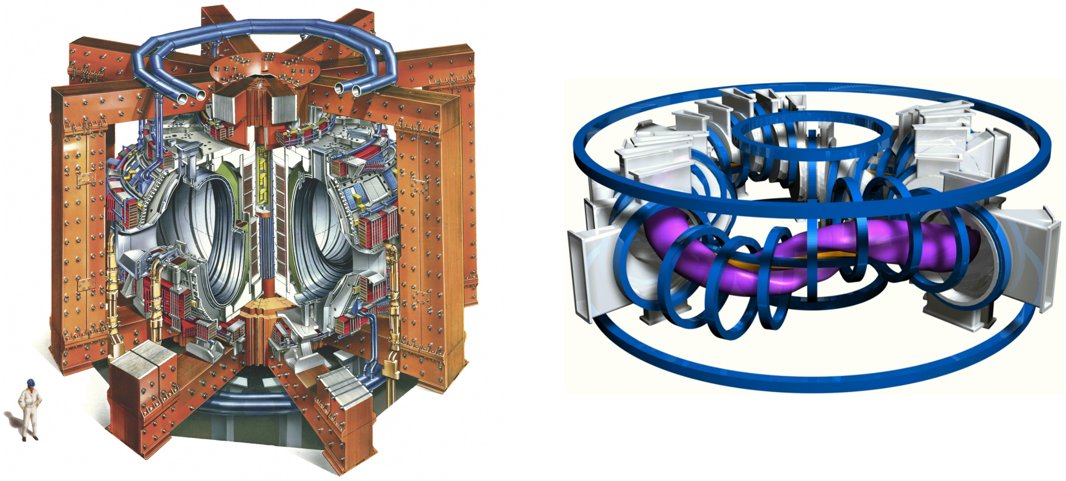


Figure 1.9.: JET tokamak (from Eurofusion) and TJ-II stellarator (from CIEMAT).

In magnetic fusion, achieving ignition—the point where plasma self-heats and sustains the fusion reaction—can be approached in three ways [Sha21]: increasing energy confinement time, increasing magnetic field strength, or increasing beta (the ratio of plasma to magnetic pressure). Each of these methods has distinct technological challenges and physical limitations, but together they offer different research lines for the design of fusion devices.

One approach to achieving ignition is by increasing the energy confinement time τ_E. This involves ensuring that the plasma retains energy for longer periods before it cools. A longer confinement time can be achieved by increasing the volume of the fusion reactor, as larger volumes help sustain energy. For example, in a reactor generating 1 GW of power, a plasma volume of approximately 1000 m^3 is required. The ITER project is designed to come close to this critical size. Current reactors like the Joint European Torus (JET), with volumes around 100 m^3, are still subcritical in volume, meaning they cannot yet achieve ignition based on confinement time alone [Sha21].

Another method to achieve ignition is by increasing the magnetic field B, which enhances the plasma confinement. However, raising the magnetic field strength beyond 5 T is technologically difficult due to the increased pressure on magnetic coils. The magnetic pressure on the structure rises significantly with higher fields, making it a major engineering challenge. For instance, magnetic pressures around 1 kg/cm^2 for a 0.5 T field escalate to approximately 400 kg/cm^2 for a 10 T field. Projects such as Alcator C-MOD and IGNITOR have been designed to push the boundaries of magnetic field strength, while high-temperature superconducting magnets offer a promising future solution for increasing B [Sha21].

Finally, increasing beta β, the ratio of plasma pressure to magnetic pressure, presents another path to ignition. Higher beta values allow more plasma pressure to be confined for a given magnetic field. This method is limited by physical constraints, specifically magnetohydrodynamic (MHD) instabilities, which cap the maximum beta. However, optimizing the magnetic field topology, such as using spherical tokamak designs, can achieve higher beta values. Spherical tokamaks, like MAST and NSTX, have demonstrated much higher beta levels than traditional machines, making them a promising concept for future fusion reactors [Sha21].

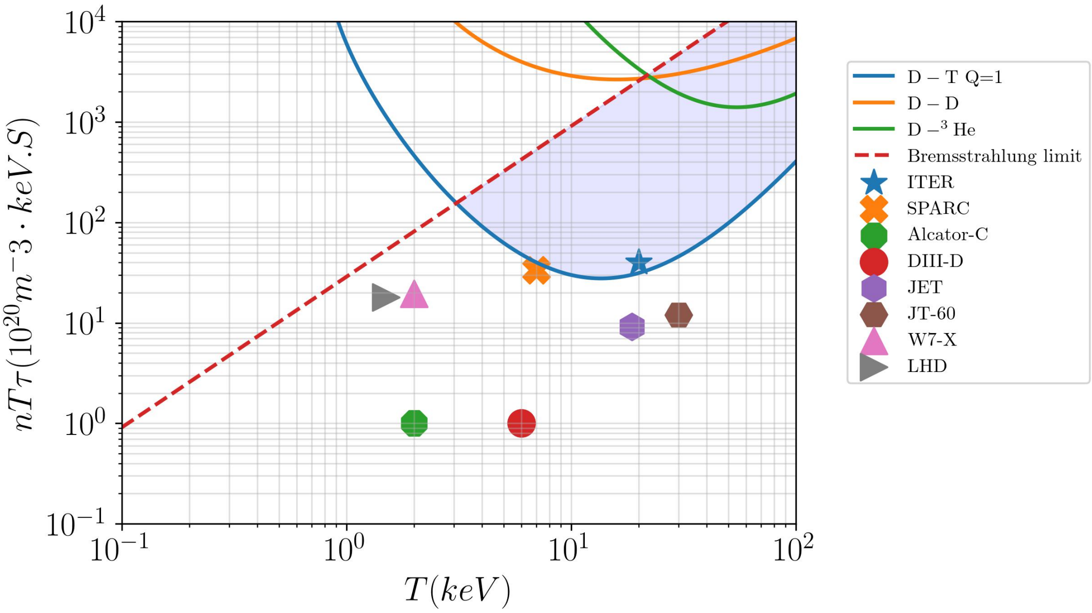


Figure 1.10.: Triple product of some Tokamaks and Stellarators

In Fig. 1.10, we compare a sample of different triple products achieved by tokamaks and stellarators. We can see that the triple product of tokamaks under construction is close to the ignition threshold. Additionally, the performance of optimized stellarators is approaching that of tokamaks. However, tokamaks are able to achieve higher temperatures than stellarators. The triple product depends on the experimental conditions, so Fig. 1.10 offers only a sample of many experiments. For a more exhaustive comparison of the triple product between stellarators and tokamaks, consult Ref. [WH22].

It’s worth including for comparison other fusion reactions that are being studied, D-D and D-He^3.

$$D\,(^2_1\mathrm{H}) + D\,(^2_1\mathrm{H}) \rightarrow {}^3_2\mathrm{He}(0.82\,\mathrm{MeV}) + {}^1_0\mathrm{n}(2.45\,\mathrm{MeV}) \tag{1.28}$$

$$^2_1\mathrm{H} + {}^3_2\mathrm{He} \rightarrow {}^4_2\mathrm{He}\,(3.6\,\mathrm{MeV}) + {}^1_1\mathrm{p}\,(14.7\,\mathrm{MeV}) \tag{1.29}$$

Their triple product dependency with temperature is included in Fig. 1.10. Their reactivity is much lower than that of D-T, therefore the triple product needed to achieve ignition is an order of magnitude higher than for D-T.

Stellarators and tokamaks have both advantages and disadvantages compared to each other. Tokamaks achieve higher triple products than stellarators, heat the plasma to higher temperatures, and have better particle confinement (neoclassical). Stellarators, on the other hand, have fewer current-drive instabilities because the poloidal field is produced by external coils, and they do not experience disruptions, or runaway electrons [Boo21]. Stellarators offer many possible magnetic configurations and optimization possibilities. However, once the coils are optimized, the operational margin is usually less flexible compared with tokamaks.

The major physical difference between tokamaks and stellarators is the axisymmetry of tokamaks and the non-axisymmetry of stellarators, as it can be appreciated in Fig.1.9. The axisymmetry allows the conservation of particle toroidal angular momentum [Hel14] (also called canonical angular momentum), which enables better particle confinement than stellarators. In stellarators, the lack of axisymmetry produces many local minima in the field lines around the torus. The particles are easily trapped in these minima and drift away quickly (this trapped particle orbits of local minima—also called banana orbits—do not precess around the torus like in a tokamak). An analogous effect is produced in tokamaks when there are gaps or misalignments between toroidal coils, producing small perturbations to the axisymmetry; this 3D effect is called error field in tokamaks. Therefore, the optimization of magnetic configurations in stellarators must take into account trapped particle orbits to improve particle confinement, aiming for what is called quasisymmetry. In addition, other parameters like transport or plasma wall wall interactions can be optimized. For a more detailed comparison between tokamaks and stellarators, see Ref. [Hel+12; Boo21].

1.5. Waves in plasmas

Wave phenomena is ubiquitous in all physical disciplines, and plasmas are no exception. Waves propagate energy perturbations through matter or fields. The waves in plasmas can be classified into three main categories [Rax11]-chapter 8: electromagnetic waves (light), heating waves, and destabilizing waves.

Electromagnetic waves (light, x-rays) are the same as in a vacuum, and they are used for diagnostic purposes. They have frequencies much higher than the rest, in the order of THz, and the characteristic wavelength is the shortest of all (smaller than the electron gyrofrequency). Heating waves are used to heat the plasma and have frequencies in the order of GHz (frequencies ranging from ion gyrofrequency, lower-hybrid frequency, pulsating plasma frequency to the electron gyrofrequency) and characteristic wavelengths in the order of cm-mm.

Destabilizing waves are characteristic of MHD instabilities and turbulence. They have frequencies in the order of kHz for MHD waves and kHz-MHz for turbulence. Their characteristic wavelength is in the order of the machine's major R radius or the minor radius a (meters) for MHD waves, and between the ion gyroradius and the electron gyroradius for turbulence (cm-mm).

The characteristic frequencies of MHD waves are comparable to the precession frequency of confined ions or any other bouncing frequency of confined particles. This is important because a wave can interact with distributions of particles through an effect known as Landau damping. If the wave frequency is close to the confined bouncing frequency of confined particles, the wave can gain energy from the slowing-down distribution to grow and become an instability; otherwise, the waves get damped and disappear. An immediate consequence of the formation of resonant waves is a perturbation of the equilibrium configuration of the plasma, leading to the deterioration of confinement. Particle and energy loss occurs, reducing the triple product and, consequently, plasma performance. In this thesis, we focus our attention on MHD waves and MHD instabilities. Their ability to generate growing stationary waves on the scale of fusion devices, when resonating with confined particle orbits, makes them an important phenomenon to understand and control in fusion devices.

Other types of waves worth mentioning are sound waves, which are compressional waves that propagate through the plasma at the sound speed, and drift waves, which are driven by density gradients and propagate perpendicular to the magnetic field. The velocity of propagation for the different waves has to be calculated using their dispersion relation $\omega(\mathbf{k})$ and the wave vector $\mathbf{k}$. Additionally, waves can have different velocities in different directions; this is called anisotropy. The waves can be classified by their polarization, the direction of the electric field with respect to the magnetic field. The waves can be longitudinal, transverse, or elliptical. They can also be classified by their phase velocity, the velocity of the wavefronts, and the group velocity, the velocity of the energy transport. Discussing all these details is beyond the scope of this introduction, but detailed discussions are available in the plasma physics literature.

In conclusion, waves in plasmas are a multiscale, complex phenomenon. The waves can interact with particles, fields, and couple with other waves. However, in this thesis, we focus on MHD waves and instabilities, particularly Alfvén waves, which are the ones observed in our diagnostic data and are relevant for the macroscopic plasma confinement in burning plasmas.

1.5.1. MHD waves in homogeneous plasmas

The MHD wave equations can be derived from a perturbed MHD equilibrium $\mathbf{J_0} \times \mathbf{B_0} = \nabla P_0$, where the subindex 0 denotes the equilibrium quantities. The perturbed quantities are [Sha+13]: $\mathbf{B} = \mathbf{B_0} + \delta\mathbf{B}$, $\mathbf{J} = \mathbf{J_0} + \delta\mathbf{J}$, $\mathbf{v} = \delta\mathbf{v}$, $P = P_0 + \delta P$, $\rho = \rho_0 + \delta\rho$, and $\mathbf{E} = \delta\mathbf{E}$, where ρ is the plasma density.

It can be shown [Fre14] (page 51) by conservation of the magnetic flux in the ideal non-resistive plasma MHD approximation that magnetic lines are *frozen* into plasma, so magnetic lines move with the plasma. This means that plasma displacement ξ from the equilibrium $\delta\mathbf{v} = \frac{\partial\xi}{\partial t}$ translates into a magnetic signal that we can measure.

The MHD wave equations are obtained by linearizing the MHD equations around the equilibrium configuration. After some algebra, the following equation can be obtained [Sha+13]:

$$\frac{\partial^2\xi}{\partial t^2} = c_S^2 \nabla \operatorname{div}\xi + V_A^2 \nabla_\perp \operatorname{div}\xi_\perp + V_A^2 \frac{\partial^2 \xi_\perp}{\partial z^2} \tag{1.30}$$

This equation describes linear MHD perturbations in ideal homogeneous plasmas.

Where the speed of sound is

$$c_S = \sqrt{\frac{\gamma p_0}{\rho_0}}, \tag{1.31}$$

and the Alfvén speed is

$$V_A = \frac{B_0}{\sqrt{\mu_0 \rho_0}}. \tag{1.32}$$

The vector equation 1.30 can be separated into three wave equations for the shear Alfvén waves $\frac{\partial^2 \xi_\perp}{\partial t^2} = V_A^2 \frac{\partial^2 \xi_\perp}{\partial z^2}$, the compressional Alfvén waves $\frac{\partial^2 \operatorname{div} \xi_\perp}{\partial t^2} = V_A^2 \Delta_\perp \operatorname{div} \xi_\perp$, and the slow magnetosonic waves $\frac{\partial^2 \xi_z}{\partial t^2} = c_S^2 \frac{\partial^2 \xi_z}{\partial z^2}$. Shear Alfvén waves are transverse waves that propagate along the magnetic field lines, compressional Alfvén waves are longitudinal waves that propagate along the magnetic field lines, and slow magnetosonic waves are longitudinal waves that propagate perpendicular to the magnetic field lines. This is depicted in Fig. 1.11. Of the three types of waves, the shear Alfvén waves are the easiest to excite [Sha+13].

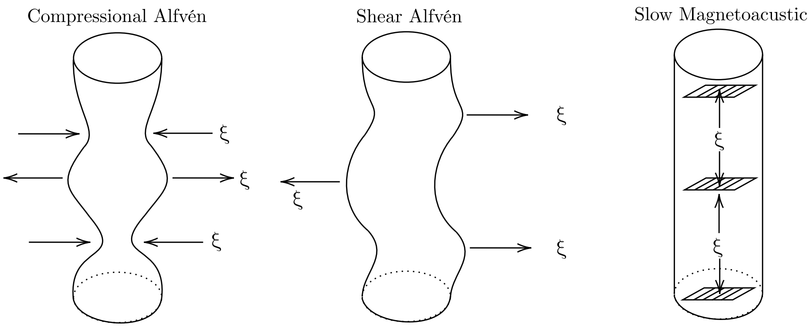


Figure 1.11.: Types of MHD wave perturbations of ideal homogeneous plasmas in a cylinder: Compressional Alfvénic waves (CA), shear Alfvén Waves (SA), and slow magnetosonic waves (SM). Adapted from [Sha+13].

In the case of shear Alfvén waves, the perturbed magnetic field lines can vibrate transversally like a guitar string. The waves propagate in the parallel direction with respect to the field. The dispersion relation can be written as

$$\omega = k_{||} V_A, \tag{1.33}$$

where the wave vector $k_{||}$ is parallel to magnetic field lines. We can rewrite V_A by expressing the density as the sum of the mass of the species in the plasma $\rho_0 = \sum n_i m_i$:

$$V_A = \frac{B}{\sqrt{\mu_0 \sum n_i m_i}}. \tag{1.34}$$

The Alfvén speed is proportional to the magnetic field B, and inversely proportional to the square root of the mass density of the plasma $\sum n_i m_i$. When the plasma is composed of species with the same charge/mass ratio, the mass density [Hei08] is proportional to electron density, $V_A \propto B/\sqrt{n_e}$. In these cases, substituting into the dispersion relation, we have

$$f \propto \frac{B}{\sqrt{n_e}} \propto \frac{1}{\sqrt{n_e}}. \tag{1.35}$$

This equation allows us to distinguish Alfvén modes from other plasma waves, because they are strongly correlated with the plasma density diagnostic signal.

Waves are often expressed with a Fourier expansion in which *one* harmonic (or *one mode*) represents a displacement ξ. A harmonic can be defined for toroidal geometry as follows [Igo14]:

$$\xi(\rho,\theta,\phi,t) = \xi_r(\rho)\cos\big(n\phi - m\theta - 2\pi f t\big)\cdot e^{\gamma t}, \tag{1.36}$$

where ξ_r is the radial amplitude function (depending on the radial coordinate ρ), ϕ and θ are the toroidal and poloidal harmonic angles, respectively, and m, n are the toroidal and poloidal harmonic numbers associated. The above parameters determine the spatial distribution and the temporal evolution of the mode. A sea of such modes can exist in the plasma

1.5.2. Alfvén waves in inhomogeneous plasmas

In the beginning, it was believed that shear Alfvén waves would be strongly damped and not be discretely observable. The reason is that in an inhomogeneous plasma with $V_A = V_A(r)$ and $k_{||} = k_{||}(r)$, the frequency varies with radius. Additionally, the varying q-profile (varying shear) along the minor radius changes the direction of wave vectors. Therefore, the waves propagate with different velocities and directions, dispersing quickly. This type of mode is known as a continuum mode, where mode perturbations are highly localized in the radial direction, but these perturbations are highly damped. In Fig. 1.12, we can see the dispersion relationships of the different types of modes and their radial distribution. The first row shows the continuum type. An antenna sending a signal with a frequency in the continuum will create a highly localized radial structure but with high damping.

The first discovery contradicting this expectation was the experimental finding of the Global Alfvén Eigenmodes (GAE) in plasma cylinders [RCM82]. In cylindrical geometry, the length of the cylinder L determines the smallest parallel wave vector as $k_{\min} = \frac{2\pi}{L} > 0$. Therefore, the lowest shear Alfvén (SA) frequency remains above zero. It was found experimentally that a resonance of frequency $\omega_{\mathrm{GAE}} < \omega_A$ can exist if the current profile in the cylinder creates a minimum in the Alfvén continuum. This is allowed by the condition:

$$\frac{1}{k_{||}}\frac{\mathrm{d}k_{||}}{\mathrm{d}r} = -\frac{1}{V_A}\frac{\mathrm{d}V_A}{\mathrm{d}r} \tag{1.37}$$

which occurs in the vicinity of a minimum in the Alfvén continuum,

$$\left(\frac{\mathrm{d}\omega_A(r)}{\mathrm{d}r}\right)_{r_{\min}} = 0. \tag{1.38}$$

These are depicted in the second row of Fig. 1.12. The perturbation can extend along the radial direction and the mode can be excited.

The GAE wave propagation is analogous to total internal reflection in optical fibers [Sha21]. The GAEs propagate through a waveguide in the plasma defined by a maximum in the radial refractive index $N_r = c/k_r$ (a minimum in frequency $\omega_A(r)$).

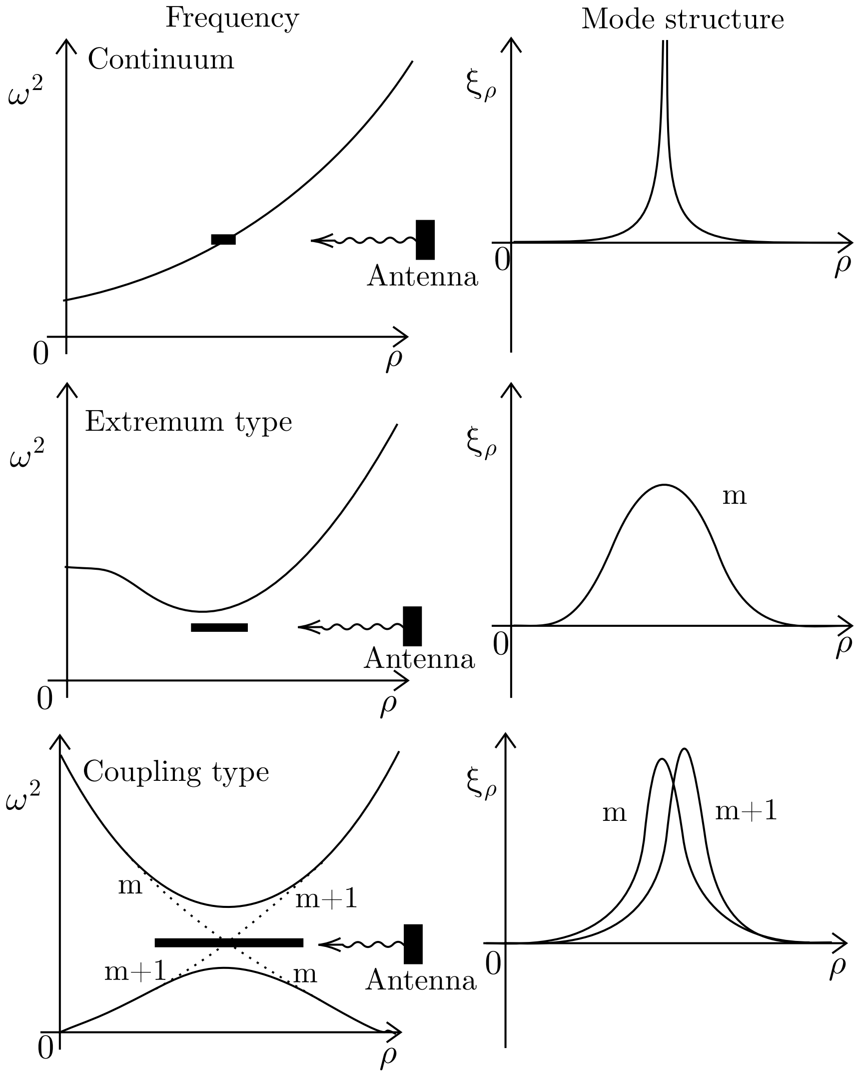


Figure 1.12.: Dispersion relationships and Mode structures. Adapted from [Pin96]

There is a third type of shear Alfvén wave that appears in inhomogeneous plasmas: the coupling type. It was found [CCC85] that due to the non-uniform toroidal magnetic field, the continuum is broken up into small continuum bands, and new discrete toroidal eigenmodes can exist inside the continuum gaps.

The parallel wave vector for the mode m in toroidal geometry is given by:

$$k_{\parallel m}(r) = \frac{1}{R}\left(n - \frac{m}{q(r)}\right). \tag{1.39}$$

The coupling between adjacent modes, m and $m+1$, is given by the condition:

$$k^2_{\parallel m-1} V_A^2 = -k^2_{\parallel m} V_A^2. \tag{1.40}$$

The condition for the safety factor q at which TAE modes can occur can be calculated by substituting Eq. 1.39 into Eq. 1.40:

$$\frac{1}{R}\left(n - \frac{m}{q(r)}\right) = -\frac{1}{R}\left(n - \frac{m+1}{q(r)}\right), \tag{1.41}$$

and then solving for q, which gives:

$$q = \frac{m - 1/2}{n}. \tag{1.42}$$

Substituting this value of q into Eq. 1.39, and the result into $\omega = k_{\parallel m}(r) V_A$, we obtain the frequency of the gap formed by the coupling of the m and $m+1$ modes. This mode is known as the Toroidal Alfvén Eigenmode (TAE) and has a frequency:

$$\omega_{\mathrm{TAE}} \cong \frac{V_A(r_{\mathrm{TAE}})}{2R_0 q(r_{\mathrm{TAE}})}. \tag{1.43}$$

In the last row of Fig. 1.12, we can see the dispersion relationship of the coupling type. The two harmonics of the mode are localized in the radial direction and the mode can be excited easily, overcoming damping.

The formation of frequency gaps is a common wave phenomenon observed in various physical systems [Hei08]. Examples include the electron band gap in conductors and Bragg reflection in optical filters. As Lord Rayleigh noted, any periodic modulation of the refractive index N creates a band gap. In conductors, the periodic ion lattice causes changes in N in the electron wave equation, while in photonic crystals, alternating layers of materials with different N produce a frequency gap. Similarly, in optical fibers, periodic variations in the core's refractive index create a "fiber Bragg grating".

These gaps arise due to the interference of counterpropagating waves reflected backwards by the periodic modulations [Hei08]. In toroidal plasmas, variations in the magnetic field B, which occur due to rotational transform and poloidal currents, result in a variation of the Alfvén speed along a field line, effectively acting like a periodic refractive index. Since the Alfvén speed is proportional to B, these variations lead to frequency gaps. Additionally, factors like elliptical or helical geometry, changes in compressibility, and electron dynamics can introduce further couplings and influence gap formation. The nomenclature of the different types of shear Alfvén eigenmodes can be found in Table 1.1.

The equilibrium stability can be analyzed with codes like the MHD linear code

Acronym	Name	Necessary condition
RSAE	Reversed-shear	$q_{\min}$ (Also known as Alfvén cascades)
BAE	Beta	Compressibility
GAE	Global	Minimum in Alfvén continuum
TAE	Toroidal	m and $m+1$ coupling (toroidicity)
KTAE	Kinetic	Electron dynamics
EAE	Ellipticity	m and $m+2$ coupling (elipticity)
NAE	Noncircularity	m and $m+3$ coupling (triangularity)
MAE	Mirror	n and $n+1$ coupling
HAE	Helicity	n and m coupling combinations

Table 1.1.: Nomenclature of shear Alfvén eigenmodes. Theoretical and experimental observations are available in Ref. [Hei08].

MISHKA [Mik+97] or its CASTOR variants [Ker+98; BK99; SG16]. Non linear stability can be studied with codes like LIGKA [Lau+07] with HAGIS [Pin+98] or FAR3D [Spo+21].

It is worth noting that by identifying the Alfvén mode type, their frequency and mode number, we can infer equilibrium properties using their propierties. This method is called MHD spectroscopy [Fas+02; Sha+01].

1.5.3. Energetic particle drive

The waves explained in the last section need a source of free energy to be excited, also known as a drive. Inhomogeneous plasmas in fusion devices have many types of gradients: pressure, current, magnetic field, shear, etc., which can act as sources of free energy. To compile this introduction, information from monographs about energetic particle instabilities [Sha+13; Hei08; Pin+15; GPT14] and detection-control [Igo14] has been used. The historical development of the field (experiments and theory) can be consulted in Refs. [Hei08; GPT14].

The Alfvén wave instabilities are extensively studied because energetic particles (EPs) can transfer energy to plasma waves. Gradients in the EP distribution come either from external heating, like Ion Cyclotron Resonance Heating (ICRH) or Neutral Beam Injection (NBI), or from the nuclear fusion reactions, like alpha particles. In burning plasma, He^{2+} ions coming out of the deuterium-tritium reaction have a high energy of 3.5 MeV and fulfill the conditions mentioned to drive the different kinds of Alfvén waves. With the energy coming from EPs, the instabilities can increase their amplitude (positive γ), causing particle and energy losses, reducing the reactor efficiency due to loss of energetic particles (reduction of alpha confinement). On fusion fuel-less experimental devices, EPs are provided by heating systems, allowing researchers to study this phenomenon before achieving ignition.

The general expression for power transfer between a particle and the fields of a long-wavelength, low-frequency wave is given by the rate of change of energy $\frac{dW}{dt}$.

According to Ref. [Hei08], the formula is:

$$\frac{dW}{dt} = eZ\mathbf{v}_d \cdot \mathbf{E}_\perp + eZv_\parallel E_\parallel + \mu\frac{\partial B_\parallel}{\partial t}. \tag{1.44}$$

The terms in the formula are defined as follows: e represents the elementary charge of an electron, while Z is the charge state of the particle, typically greater than 1 for ions. The drift velocity of the particle orbits is denoted by $\mathbf{v}_d$, and $\mathbf{E}_\perp$ refers to the transverse electric field, which is perpendicular to the magnetic field. The velocity component of the particle orbits parallel to the magnetic field is given by $v_\parallel$, and the parallel electric field is represented as $E_\parallel$. Additionally, μ is the magnetic moment of the particle, and $\frac{\partial B_\parallel}{\partial t}$ describes the time variation of the magnetic field parallel to the particle trajectory.

The first term $eZ\mathbf{v}_d \cdot \mathbf{E}_\perp$ describes the power transfer due to the drift motion of the particle in the transverse electric field. For shear Alfvén waves, this term is often dominant because the transverse electric field can be significant, and the drift velocity of the particle orbits is the main contributor to energy transfer between the particle and the wave.

The second term $eZv_\parallel E_\parallel$ represents power transfer due to the particle's parallel velocity $v_\parallel$ in the presence of a parallel electric field $E_\parallel$. In shear Alfvén waves, this term is generally small because the electric field component parallel to the magnetic field is small unless mode conversion occurs (to a wave with a large electrostatic component).

The third term $\mu\frac{\partial B_\parallel}{\partial t}$ corresponds to the power transfer due to the time variation of the parallel magnetic field. Since the magnetic perturbation of shear Alfvén waves is mostly transverse, this term is typically small, especially in low-β tokamaks (where plasma pressure is much smaller than the magnetic pressure).

Alfvén Eigenmodes (AEs) can be excited by EPs if three primary conditions are satisfied, as explained in Chapter 9 of Igochine's book [Igo14].

First, the drift frequency resulting from the radial gradient of EPs pressure P_α must exceed the Alfvén frequency $\omega_{\mathrm{AE}} = 2\pi f_{\mathrm{AE}}$. This condition is mathematically expressed as [Igo14]:

$$\omega_{d_\alpha} \equiv -\frac{m}{r}\frac{T_\alpha}{e_\alpha B_0}\frac{d\ln p_\alpha}{dr} > \omega_{\mathrm{AE}}, \tag{1.45}$$

where m represents the poloidal harmonic, r is the minor radius, T_α is the EP temperature, e_α is the EP charge, B_0 is the equilibrium magnetic field strength, and p_α denotes EP pressure. This condition is necessary because in curved fields, power transfer can occur, but only the drift velocity $v_d \propto \omega_d$ contributes, as shown in the power transfer equation.

Second, the free energy from the radial gradient of EP pressure needs to be coupled to the energy of the AEs through wave-particle resonances, such as the Landau resonance, where the parallel velocity of alpha particles $V_{\parallel_\alpha}$ equals the Alfvén velocity V_A.

To the later we can also add a resonance condition for the particle orbits

$$\omega + p\omega_\theta - n\omega_\zeta \simeq 0 \tag{1.46}$$

where ω denotes the wave frequency of the Alfvén eigenmode (AE), the orbit poloidal frequency is given by ω_θ, ω_ζ is the toroidal orbit frequency. The integer p indicates that resonance can occur at different harmonics of the poloidal orbit, and the integer n is the toroidal mode number.

Finally, the power transfer from EPs to the AEs, denoted as P_α, must exceed the wave damping caused by the thermal plasma. This is represented by the inequality:

$$\gamma_\alpha = \frac{P_\alpha}{2W_{\mathrm{AE}}} > \gamma_{\mathrm{plasma}}. \tag{1.47}$$

where W_{AE} is the wave energy, and γ_{plasma} is the damping rate of the thermal plasma. The main mechanisms of damping are four [Pin+15]: Landau damping due to collisions with ions, or collisions with trapped electrons, radiative damping and continuum damping.

Non linear effects that can also be observed should be mentioned. They reveal several behaviours in the mode amplitude and frequency evolution that can be observed in diagnostic data, including [Sha21]:

(a) the amplitude grows until saturation at nearly constant frequency, following the Alfvén scaling law $f \propto \frac{B_0}{\sqrt{n_i}}$;

(b) the amplitude becomes modulated at constant frequency, leading to phenomena such as “pitchfork splitting” and chaotic structures in Fourier spectrograms;

(c) the amplitude shows rapid bursts over time, with the mode frequency sweeping, commonly referred to as “chirping” modes.

In conclusion, it is important to note that many other types of instabilities can be observed in fusion devices, such as fishbones (internal kink driven by EPs), neoclassical tearing modes (NTMs), edge localized modes (ELMs) in tokamaks, and magnetic islands rotating with the plasma. More information can be found in Refs. [WC11; Igo14]. The abundance of instability phenomena presents a challenge to detect and classify the different types of instabilities.

1.6. Plasma diagnostics

There is a large variety of diagnostics used in fusion plasmas. They can be active or passive, meaning that they either interact with the plasma or not. They can also play a role in real-time control or be used for data collection and later analysis. The plasma diagnostics used in Tokamaks, which can also be used in Stellarators, are extensively discussed, for example, in Igochine’s book [Igo14] and in chapter 10 of Wesson’s book [WC11]. The following text provides a brief introduction.

Light is studied in the infrared, visible, and X-ray regions by cameras, bolometers, and spectroscopy devices. Radiation is measured by particle counters, Faraday cups (which measure the energy of fast ions), scintillators, and bolometers. Bolometry and soft X-ray diagnostics are capable of reconstructing 2D profiles of the plasma.

Various diagnostics are used to measure MHD perturbations, plasma current, electron density, and plasma energy in fusion plasmas. Mirnov coils, Electron Cyclotron Emission (ECE), Far Infrared (FIR) interferometry, and reflectometry are particularly effective for capturing MHD activity. Plasma current is commonly measured with Rogowski coils, while electron density is measured using interferometry and reflectometry. Diamagnetic loops provide measurements of the total plasma energy content. These complementary diagnostics offer a comprehensive understanding of plasma behavior and are essential for controlling and studying fusion plasmas in fusion devices. Cameras in the visible and infrared regions are used to monitor the heating of plasma-facing components.

Mirnov coils (also called pickup coils) are the most commonly used diagnostic for detecting MHD perturbations. Their construction is very simple, they consist of a coiled wire. The magnetic flux is proportional to the number of turns N in the coil, the area A defined by the turns, and the alignment $\mathbf{B} \cdot \mathbf{n}$ with the measured magnetic field: $\Phi = N \cdot A \cdot \mathbf{B} \cdot \mathbf{n}$ [WC11]. The magnetic flux can be measured by integrating the voltage induced in the coil $V \propto -\frac{d\Phi_B}{dt}$. These coils measure the time-varying magnetic flux and are effective in capturing oscillatory magnetic perturbations, such as Alfvén Eigenmodes (AEs), which have frequencies ranging from 10 kHz to 500 kHz.

By placing arrays of Mirnov coils toroidally and poloidally around the plasma, researchers can determine the mode numbers and propagation direction of these instabilities. However, Mirnov coils are most sensitive near the plasma edge, making them less effective for detecting perturbations deep in the core. Other tools like ECE and FIR interferometry can fill this gap by providing better resolution of the internal structure of the plasma.

When measuring the frequency and toroidal mode number of MHD modes using Mirnov coils in rotating plasmas, it is essential to account for the Doppler effect. In devices like tokamaks, plasma often undergoes significant toroidal rotation, particularly when driven by external heating methods such as neutral beam injection (NBI). This rotation causes a Doppler shift in the observed frequency of MHD modes, including Alfvén eigenmodes (AEs), between the laboratory reference frame and the plasma's reference frame.

The observed frequency $f_{\mathrm{LAB},n}$ in the laboratory frame and the mode frequency $f_{0,n}$ in the plasma reference frame are related through the Doppler shift as $f_{\mathrm{LAB},n} = f_{0,n} + n f_{\mathrm{rot}}(r)$, where n is the toroidal mode number, and $f_{\mathrm{rot}}(r)$ is the plasma's local rotation frequency [Sha21]. The Doppler shift increases or decreases the observed frequency based on the direction and speed of the plasma's toroidal rotation. Ignoring this effect could lead to misinterpretation of the actual mode frequency, which is critical for diagnosing plasma stability and dynamics.

Furthermore, when several MHD modes with neighboring toroidal mode numbers are excited in a rotating plasma, the frequency separation between these modes is approximately the plasma's rotation frequency: $f_{\mathrm{LAB},n+1} - f_{\mathrm{LAB},n} \approx f_{\mathrm{rot}}(r)$. This frequency separation can be used to estimate the plasma's rotation frequency.

ECE measures the electron temperature in localized regions, making it useful for detecting temperature perturbations associated with MHD instabilities like Alfvén modes. This diagnostic provides high spatial resolution and helps in analyzing the mode structure inside the plasma. FIR interferometry, on the other hand, measures the line-integrated electron density across different lines of sight. With a high sampling rate, FIR interferometry can detect density perturbations related to MHD modes deep in the plasma core, which are often missed by Mirnov coils. Reflectometry, which analyzes the reflection of microwaves from density layers in the plasma, also plays a significant role in diagnosing density fluctuations caused by AEs.

To measure the plasma current I_p, Rogowski coils are typically employed. These coils surround the plasma column and detect the magnetic fields generated by the current. As explained in [WC11], Ampère's law relates the integral of the magnetic field strength around a closed loop with the total current enclosed by the loop $I = \frac{1}{\mu_0} \oint \mathbf{B} \cdot \mathrm{d}\mathbf{l}$, so the plasma current can be obtained by integrating the loop voltages over time. Rogowski coils provide real-time measurements of the net plasma current and are widely used in fusion experiments, including TJ-II and JET. They are particularly effective for measuring plasma current in the presence of MHD instabilities, as they can detect the fast changes in current associated with these perturbations.

The energy content of the plasma, specifically the magnetic energy W, is measured using diamagnetic loops. In TJ-II, the loops consist of primary and compensating loops, which subtract the background magnetic field to provide accurate energy measurements [24a]. This diagnostic is essential for understanding the overall energy dynamics of the plasma.

Finally, for measuring electron density, interferometry and reflectometry are the primary tools. Laser interferometry, particularly FIR interferometry, provides continuous and accurate line-integrated measurements of electron density. Though it lacks spatial resolution, it is fast and reliable, making it one of the standard methods in fusion research. Reflectometry complements this by offering localized measurements, particularly in areas with steep density gradients.

1.7. Thesis problem

Understanding, predicting, and controlling plasma instabilities is of major importance on the path towards achieving steady-state burning plasma scenarios, where energetic particles must remain well confined in order to transfer their energy to the plasma. In this respect, the accurate identification of various plasma instabilities that might tend to de-confine particles and their energy is required to develop possible control techniques. Machine learning offers the possibility to detect patterns in diagnostic data that can identify the presence of instabilities. This thesis focuses on the identification of plasma instabilities using machine learning (ML) techniques. The ultimate goal for control in fusion devices is to identify instabilities in real time or predict their appearance. Many approaches are possible to address this problem.

One supervised approach could consist of the following process: first, building a database using synthetic experimental data for various calculated instabilities such as MHD and energetic particle-driven modes; second, training a multi-class classification model based on supervised algorithms; and finally, applying the model to test cases and experimental data to investigate the portability of the AI model between machines. However, this approach presents several challenges. For instance, synthetic data is not always representative of real diagnostic data, and therefore models trained this way may be biased towards the synthetic data and may not generalize well to real test data. Additionally, simulation codes are not always accurate or fast enough to construct large databases. Indeed, MHD linear codes give the frequencies of unstable waves and are fast enough to construct large databases. However, they do not provide the growth rate of the waves, and non-linear codes that determine which waves are driven are computationally expensive. During this PhD research, a scarcity of both labeled data and systematic methods to label them has been found.

Another supervised approach consists of manually labeling the data and then training the model to predict the labels. This approach is more accurate, but it is highly time-consuming and requires people with expertise in fusion. As shown in previous sections, the number of possible types of instabilities is staggering. In this thesis, we will explore the possibility of using unsupervised learning to detect instabilities in the data. This approach is more flexible and does not require labeled data. A model can learn the patterns in the data and detect the instabilities without the need for human intervention. Models can also be trained with a small amount of labeled data and then used to detect instabilities in the rest of the data. Then, simulations can be used for data augmentation purposes. This approach is more efficient, and it can be used to detect instabilities in real time if enough data is labeled.

2. Data driven science

> *"machine learning is just glorified curve fitting"*
>
> — popularized by Judea Pearl.

Summary

2.1. Introduction

Machine learning (ML) needs to be understood as the final step in a journey that begins with data. Therefore, it is convenient to frame ML from the data science perspective. Data science is a relatively new area that aggregates several classic disciplines. The role of a data scientist merges those of a data analyst, a statistician, and a machine learning engineer. The first task in a data science project involves data extraction, process and analysis, then statistics or signal analysis helps to describe the data available and the possible machine learning applications. Finally, it is necessary to implement the ML solutions and evaluate their performance. Moreover, different technologies and solutions are involved from data collection to data storage. In case of need, some tasks overlap with those proper of data architects: designing database applications and extracting transform-load pipelines (ETLs), developing simple end-user applications, or deploying and maintaining models through their life cycle. That means evaluating and re-training periodically the machine learning pipelines and sometimes automating the process.

Before continuing, we should advance informal definitions of machine learning (ML). One could be: "computers program themselves with data". Or a more detailed one: *Given some data, computers can learn to detect patterns (optimal representation of data), and then perform complex actions according to some constraints (optimal decisions).* In other words, we can model complex functional relationships between the data available and the solution to the problem of interest. ML models act as a function between input data and the solution output of the corresponding use case. The solution is found by solving a optimization problem in most cases.

We can use ML algorithms to model physics or approach many scientific and engineering problems. The (data-driven science) terms emphasize the application of ML and other algorithms to modeling physical and engineering applications.

Regarding the problem of this thesis work, Alfvén waves are magnetohydrodynamic waves propagating through the plasma, the vacuum field and the plasma-facing components of the fusion reactor. The experimental data is collected by the different diagnostics and stored in different databases. The types of data available to detect Alfvén instabilities are mainly two, raw time series and their processed spectrograms.

The signal processing field is a vast research area that looks for alternative signal representations of signals. Through alternative representations patterns can be recognized, or data can be compressed. More importantly for our purpose, it can help to study diagnostics signals and to separate noise from features in signals and images.

How can ML help to represent signals? ML models can represent time-dependent signals. In general, most successful models in ML are usually parametric models, and optimization (in particular gradient descent) is the most extended method to find model parameters. In this thesis, we use the connection between optimization and ML to explore new interpretable representations of signals.

Considering the matter from another standpoint: MHD experts just “look” at the spectrograms of Mirnov coils to identify the magnetic modes and different types of instabilities. Therefore, we can imitate this natural perception of fusion researchers using artificial vision (computer vision). Which is the field that studies problems regarding the use of images, from processing to detection of objects. It incorporates all classes of algorithms from classic, to ML and DL approaches.

In this chapter, we develop the necessary concepts and present the positioning of the thesis with respect to the state-of-the-art in the context of machine learning, signal analysis, and computer vision. Finally, we discuss current applications of these disciplines to the identification of plasma instabilities.

2.2. Machine learning

Nowadays, probably the more popular words (and abused) are Artificial intelligence (AI). A formal definition of AI is a more elusive or confusing concept than defining ML, because it has not been reached yet, or perhaps as a consequence of our difficulties in defining rigorously human intelligence or human conscience. However, we can agree that AI involves the imitation of human cognitive capabilities (or even overcoming them). In addition, AI must be a superset of ML. Because AI has to contain all modern ML, the term has been popularized as a synonym of ML. To recover the original intended meaning of AI, we talk about Artificial General Intelligence, AGI.

Thanks to the progress of computational power, ML has achieved many successes in specialized tasks in recent years, especially regarding artificial vision o modeling human language. AI is being popularized as a way to name all ML applications. The

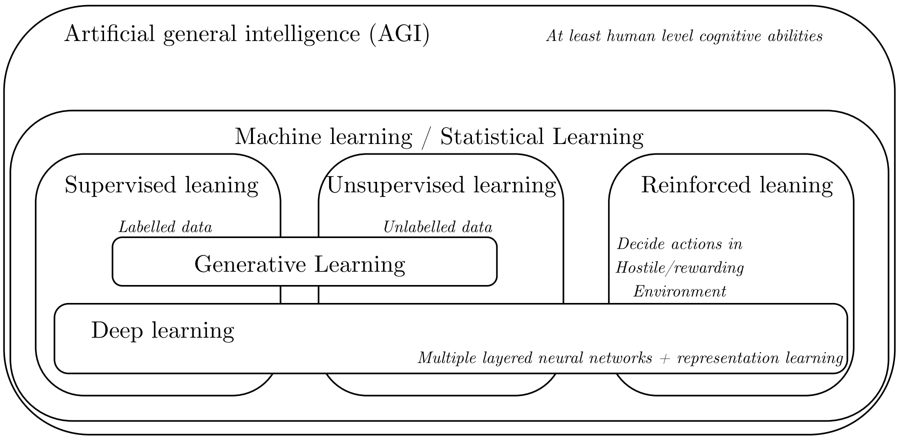


Figure 2.1.: Disciplines of Artificial intelligence (Adapted from [Pri23], page 2)

specialized tasks that ML can perform are very varied: they range from detection of objects in an image [VV24], to solving equations [RPK17], or discovering physical equations in data [BPK16], or helping robots to learn how to walk [Rud+22].

The availability of labeled data, unlabeled data, or an environment (real or simulated) determines the ML type, as approaches are substantially different. The areas of AI/ML are represented in Fig.2.1. When labeled data is available the family of problems to solve is known as supervised learning, because the “answers” or “ground truth” are known through the labels and they can be used as part of the learning algorithms. If there is no ground truth, multiple approaches are available to detect patterns in data and create artificial labels, or to solve the problem of interest in each use case. In addition, generative learning closes the gap between supervised and unsupervised learning, it uses data to self-supervise pattern detection, by making models to simulate data until it can not be distinguished from the real data. Finally, if a real or simulated environment can be used in the learning process, we use reinforcement learning, in which actions of the algorithm are evaluated using their response to the environment.

Deep learning (DL) is a subset of ML characterized by the accumulation of hidden layers in its neural networks (NNs) and the learning of data representations. For that, DL models have extremely high complexity. For instance, nowadays, a “small” model has millions of parameters. This allows ML models to continue learning patterns when big databases are available. When the data is very abundant, DL can outperform other ML methods, which do not have enough complexity to continue the learning. Another key characteristic is that the deep networks are capable of extracting patterns and defining a data representation of their own, via trained convolution filters in convolutional neural networks (CNN) [KSH17], latent representations in autoencoders (based originally in helmholtz machines [Day+95; KW19]), or attention mechanisms in transformers [Vas+23]. As it is illustrated in Fig.2.1, DL is nowadays used in all types of ML applications.

The literature on ML is vast, and it is not the purpose of this thesis to review it.

However, we give the following references, consulted to write this section: an attempt to rigorously review the field is given by Kevin P. Murphy's book [Mur22], some introductions to deep learning are the books by Goodfellow et al. [GBC16] and Prince [Pri23], and classical approachs to ML is in the book by James et al. [Jam+14] and Hastie et al. [HTF09], and the Bayesian approach to ML is cover by Bishop [Bis16]. The reference in data driven approaches to science and engineering is the book by Brunton et al. [BK22].

2.2.1. Machine learning definition

We can attempt to define ML using the popular Tom Mitchell's definition ([Mit97], page 2): "A computer program is said to learn from experience E with respect to some class of tasks T, and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." According to this definition there are many types of ML [Mur22], depending on the class of tasks T, the type of experience E, and the performance measure P. However, the most common task is to learn a function that maps input data X to output data Y (also called target). The data X is compound of predictor variables $X_1, ..., X_p$ or features. We use lowercase to indicate an observation $x \in X$ and its output $y \in Y$.

Assuming that an unknown function f exists such that $Y = f(X) + \epsilon$, where ϵ is an irreducible error. The objective of statistical learning is to approximate the function f by an estimation $\hat{f}$ using data. To this goal, data X in a training dataset is used in order to make the predictions $Y \approx \hat{Y} = \hat{f}(X)$ [Jam+14].

A probabilistic definition of ML is based on the fact that X and Y are sets of random variables governed by probability distributions [Mur22]. The power of a probabilistic approach is having a unifying theory for ML that handles uncertainty. If the function $\hat{f}$ is an estimate, again, it is more appropriate to interpret $\hat{Y}$ probabilistically to model the errors of estimation.

Because Y is a function of X and an error ϵ, it is usual to split X into several sets of data to perform cross-validation of the model performance: datasets are often divided into subsets to train and evaluate machine learning models effectively.

A common split might be 60% for the training set, 20% for the validation set, and 20% for the test set. However, when data is scarce, traditional splitting can lead to insufficient data in each subset. In such cases, K-fold cross-validation methods are used. One of the most computationally demanding of these is the leave-one-out cross-validation, where the model is trained multiple times, each time leaving out a different single data point for validation. This approach maximizes the use of limited data for both training and validation.

The model parameters are "fitted" in the training dataset and validated in the test dataset. The model learns with the training data, that is why we say that models are trained (or fitted) when we are just optimizing its parameters. Sometimes we can use pre-trained models with external data and train the model again, in that case, we are doing transfer learning, or in case of optimizing the parameters of part of the model, this method is called fine-tuning.

Finally, there as many ML types as types of data X and Y are used. The input X can be scalars, tabulated data, vectors of categorical variables, vectors of floats or integers, time series, or images. If the input X and the output Y are a sets of different types of data, the resulting models are known as multimodal. It is worth mentioning that each modality has different statistics and combining them can be quite difficult.

2.2.2. Parametric vs Non-parametric models, Linear vs Nonlinear models

Depending on how our ML models model the relationship between X and Y there are two types of models parametric and no parametric models.

The parametric models use a probability $p(Y|\boldsymbol{\theta})$. Therefore, instead of aiming for the distribution functions themself, we search for the parameters $\boldsymbol{\theta}$ that model Y distributions. For instance, when fitting a line to some data distribution often we are only interested in the slope θ_1 and the intercept θ_0 so $Y = \theta_1 X + \theta_0$, but not the distribution of slopes and intercepts that fits the data. Even if we model the noise of the data around the straight line with a Gaussian distribution, the mean and standard deviation of such distribution are the only parameters of interest.

The parametric models used in this work are linear regression and neural networks. In general, they have the advantage of being simple, fast and requiring less data than non-parametric models. On the downside, they have the weakness that it is necessary to assume a function form, they have limited complexity and may not fit well to certain data structures.

Non-parametric models model the relationship of X and Y via more general assumptions of a geometrical, algebraic or probabilistic kinds. For instance, to predict the class label of an observation, we can work out the geometric distance (in the features space) to previous observations, and give it the class label of the closest group of observations, this is the approach behind k-nearest neighbors (kNN) [CH67]. Other examples of non-parametric models are support vector machines [CR08], tree models [Kas80; Bre+17; Bre01], Gaussian processes [RW05], or principal component analysis (PCA) [FRS01; Hot33].

Non-parametric methods do not assume hypotheses about the form of the function f [Jam+14]. They only attempt to make a fit so that the results $\hat{Y}$ best match Y. They usually have the advantage of being able to estimate a wider variety of data structures, as they do not assume a form for f, but they usually require many more observations and computation time and are prone to overfitting. In this work, non-parametric methods have been used for clustering analysis: k-means [AV07], DBSCAN [Sch+17] and HDBSCAN [CMS13].

Another dichotomy in model classification is that of linear vs non-linear models. For example, in the classification problem, the categories can be linearly separable or not. An easy example to explain is support vector machines; it is easy to draw or imagine datasets in the plane or space that are not separable by a plane but are separable by a paraboloid, for example (the plane equation used to define the margins is linear in its weights).

It is important to mention that a linear model can become non-linear by replacing the product of parameters with a transformation function $\phi(X)$, known as basis expansion [Kra16] so that $\hat{f}(X) = \boldsymbol{\theta} \cdot \phi(X)$. A method of this type is known as the "kernel trick", which modifies support vector machines, logistic regression (classification), or linear regression. The "kernel trick" has been used in thesis PhD to add non-linearity to the linear regression so it can be used in complex signal representations. Other models are non-linear in their nature, such as decision trees and deep neural networks.

2.2.3. Frequentist vs Bayesian approaches

The frequentist statistics emphasizes parameter estimation (like the mean and standard deviation) and assume that frequencies can be interpreted as probabilities if there is enough data. In contrast, Bayesian statistics emphasize working with probability distributions (there are no parameters, there are probability distributions), and the unsettled fact is that priors define probability as belief.

A fully probabilistic approach implies that random variables come from probability distribution functions that should follow the Bayes theorem. Therefore, we could say that any "intelligent" computer program is somehow trying to solve the Bayes theorem

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}, \tag{2.1}$$

where $p(X|Y)$ is called likelihood function, $p(Y)$ is the prior probability, and $p(X)$ is known as evidence.

The frequentist approach has conceptual difficulties handling sources of uncertainties in data. For example, it is common practice to remove outliers before fitting a model instead of accounting for their presence in the model itself. In this case, the frequentist approach assumes that the uncertainties come from having "incorrect" data. Otherwise, why does it need to be removed?

Robust regression models offer a frequentist alternative to outlier removal, but they lack a unified modeling strategy. In contrast, a fully probabilistic approach changes the model so the probability distribution functions account for observations that could be considered outliers, for example using priors, as in chapter 8 of Ref. [SS12]. Of course, accounting for outliers requires extra modeling effort in both approaches.

The Bayesian approach offers a more general interpretation of probability. Therefore, it is an adequate formalism to define problems with uncertainty. However, it is often not possible to implement a practical use case because the general integration method used to solve the realistic posterior distributions $p(\theta|\mathbf{x})$ are Monte Carlo simulations and they are slow, expensive, and do not scale well with data size. In contrast, frequentist approaches are faster and scale well with data size. A good compromise is to use the Bayesian "mindset" for hypothesis testing, when having little data, or in special cases (for instance when variable independence is a reasonable assumption), and to use a frequentist approach when the data is abundant. After all, the Bayesian approach acknowledges fundamental limitations that the frequentist approach can not do: model parameters are indeed random variables governed by unknown probabilistic distributions, which might have their respective priors.

2.2.4. The four basic problems: regression, classification, clustering and dimensionality reduction

The four workhorses of ML are regression, classification, clustering and dimensionality reduction. Most ML applications involve at least one of them. The formulation of these problems is tightly linked to the models we use to solve them.

Regression

The simplest ML regression example is *linear regression*. If we want to fit a line $Y = \theta_1 X + \theta_0$ to some data, we need to estimate the parameters $\boldsymbol{\theta}$. Let us consider for the sake of simplicity and clarity of the notation a straight line in 2D a space. In that case, we need to find only 2 parameters. Then, the generalization to higher dimensions is straightforward.

As it is mentioned in our informal definition of ML. A learning process involves a performance metric P. For simplicity, we discuss first the Mean Squared Error (MSE), which is proper of regression

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i(\boldsymbol{\theta}))^2, \tag{2.2}$$

where y_i are data points in the output Y, related with x_i in X.

The usual approach to linear regression is to minimize the mean-squared error (MSE)

$$\boldsymbol{\theta}_{mse} = \arg\min_{\boldsymbol{\theta}} MSE(\boldsymbol{\theta}) = \arg\min_{\{\theta_1,\theta_0\}} \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i(\theta_1,\theta_0))^2 \tag{2.3}$$

therefore, minimizing the error ϵ between the data points y_i and their model predictions $\hat{y}_i$. It can be demonstrated ([Mur22], page 9) that this is equivalent to maximizing the likelihood (minimizing negative log-likelihood, NLL)

$$\boldsymbol{\theta}_{mle} = \arg\min_{\boldsymbol{\theta}} NLL(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \left\{ -\frac{1}{N}\sum_{i=1}^{N} \log p(y_i|\hat{y}_i(X;\boldsymbol{\theta})) \right\}. \tag{2.4}$$

When the likelihood is a Gaussian distribution $\mathcal{N}(y|\hat{y}(X;\boldsymbol{\theta}),\sigma^2)$ then

$$NLL(\boldsymbol{\theta}) \propto \frac{1}{2\sigma^2} MSE(\boldsymbol{\theta}) \tag{2.5}$$

This means, when we consider that the regression errors follow a Normal distribution, then NLL and MSE are equivalent. Therefore, we find the parameters that give the less probable error. This is an important result because maximizing the likelihood (or equivalently, minimizing the negative log-likelihood) can be seen as solving Bayes' theorem with uniform priors and ignoring the evidence.

However, the MSE can be calculated much more easily. Therefore, given the assumption of normally distributed errors with constant variance, the MSE is said to be indeed a good estimator of the NLL because they are proportional up to an additive integration constant. When good estimators are found, Bayesian and frequentist approaches are linked. More regression metrics can be found in the appendix A.

Classification

The next most extended application is to solve a classification problem, we can formulate it again as learning a mapping from an input vector X to an output vector Y [Mur22], where $Y \in \{0,...,C\}$, with C being the number of classes. If $C = 1$ we have a *binary classification* problem. The values of Y would be $\{Y = 0, Y = 1\}$, or $\{Y = \text{No}, Y = \text{Yes}\}$, or more interestly for our work $\{Y = \text{No plasma Instabily}, Y = \text{Plasma Instability}\}$. Logically, if C is greater than 1 we have a multiclass classification problem. For example, $\{y = NTM, y = TAE, y = AEA, y = EGAM...\}$ could be the classes for a classification model that identify different Alfvén eigenmodes.

Given an array of predictor variables $X = (X_1, X_2, ..., X_p)$, then $p(Y|X)$ is the probability distribution for the different classes. Thus our best guess of the true class will be given by

$$\hat{Y} = \hat{f}(X) = \underset{c=0...C}{\arg\max}\left\{p(Y = c|X)\right\}, \tag{2.6}$$

or in other words, the class will be decided by choosing the most probable one (argmax is an operation that finds the argument that returns the maximum value of an objective function. Thus, a decision problem, classification, can be understood as an optimization problem).

The evaluation metrics of classification problems are mostly based on the confusion matrix.

Data Class	Classified as 0	Classified as 1
0	true negative (tn)	false positive (fp or Type I Error)
1	false negative (fn, or Type II Error)	true positive tp

Table 2.1.: Confusion Matrix

This is elaborated by counting the number of correct classifications and the number of errors for each class label. More metrics are available in the appendix A.

Clustering

Sometimes, there is no Y available. For instance, there is data X but is not labeled. Finding an artificial set of labels $L \equiv Y \in \{0,...,C\}$ that best suits a data distribution is known as clustering, belonging to unsupervised learning.

Clustering is the task of dividing a set of observation $x_i \in X$ into groups, or clusters, in such a way that objects in the same cluster are more similar to each other than to those in other clusters. The goal is to find a mapping from the input space X to a set of labels $L \in 0, \ldots, C$ where C is the number of clusters. Unlike classification, the labels L are not provided beforehand. Instead, the algorithm must discover these labels based on the data distribution. As we mentioned, most clustering algorithms are non-parametric. Although we only mentioned the methods used in this work many others exist. The choice of clustering algorithm and the number of clusters C can significantly impact the results, and different metrics such as silhouette score can be used to evaluate the quality of the clustering. However, the best evaluation involves expert knowledge of the dataset and profiling each group by the input features.

Dimensionality reduction

The fourth type of ML algorithm more extended is dimensionality reduction: adding features (predictor variables) to solve an ML problem might sound appealing as we are adding more information. However, because of the so called *curse of dimensionality* (Term introduced by Richard E. Bellman), this can be very counterproductive.

Suppose we have a Cartesian input space $\mathscr{X}$ having two dimensions, let us sample a fixed squared area with 100 points (observations). If we want to sample a cartesian input space of three dimensions and we require the same data density, i.e. at least 1000 observations. In this example, the amount of data necessary grows exponentially as 10^N, with N the number of dimensions. Therefore, adding more features requires much more data in order to have enough samples.

If we have a dataset in matrix form X, this matrix needs to have more rows than columns (more observations than features). When the number of dimensions is high there is a variety of techniques to transform variables in a new set of lower dimensionality (dimension reduction) or to select the most relevant variables (feature selection). In Table 2.2 and Table 5 of appendix B, we summarize the different types of dimensional reduction and variable selection suggested by the literature [BSA15]. These techniques are essential to tackle the problem of high dimensionality, and a requisite for successful applications.

Dimensionality reduction techniques depend on the type of variables being used, as shown in Table 2.2 below.

Dimensionality Reduction Technique	Acronym	Data Type
Principal Component Analysis	PCA	Numerical
t-Distributed Stochastic Neighbor Embedding	t-SNE	Numerical
Self Organizing (Kohonen [Koh82]) Maps	SOM	Numerical
Contingency Tables	CA	2 categories
Multiple Correspondence Analysis	MCA	More than 2 categories
Multiple Factor Analysis	MFA	Numerical and categorical
Factor Analysis for Mixed Data	FAMD	Numerical and categorical

Table 2.2.: Summary of Dimensionality Reduction Techniques as a Function of Data Type

Some techniques like t-SNE [MH08] are more suitable for data representation in lower dimensional spaces than dimension reduction. For instance, PCA is used for both use cases. In this thesis work, we have used t-SNE to visualize clustering analysis.

The feature selection methods are basically 3 classes: filtering (keeping most correlated variables), wrapping (recursive permutation of features, train and selection by performance), embedded (feature selection someway is incorporated in the model) [BSA15].

It is also worth mentioning latent-representation methods, most of which are part of generative learning. These are methods that learn optimal representations of data, usually of lower dimensions. These representations can be understood as a nonlinear dimension reduction, and they can even be used to create new data instances. The most extended right now are the DL autoencoders (AEs) or variational autoencoders (VAEs).

2.2.5. Fundamental limitations for ML

The accuracy with which we can estimate f and make predictions of Y depends on the model we use, but there is a fundamental limit intrinsic to the data (X, Y), so that even if we obtain a perfect reproduction $Y = f(X)$ we will always get an error in predictions, because in reality Y *is a function of both* X *and a random error term* ϵ [Jam+14], so

$$Y = f(X) + \epsilon. \tag{2.7}$$

Since y is a function of both X and ϵ, *we cannot obtain* $\hat{Y}$ *perfectly, exclusively from the data* (X, Y).

Assuming that $\hat{f}$ and X are fixed, it can be shown [Jam+14] that

$$E(Y - \hat{Y})^2 = E[f(X) + \epsilon - \hat{f}(X)]^2 = E[f(X) - \hat{f}(X)]^2 + Var(\epsilon) \tag{2.8}$$

where E in this case stands for the expected value (average for all points in the data).

The last term ϵ in Eq.2.8 determines the irreducible error, which will always be present *regardless* of the model we choose. We can find the one that eliminates the most reducible error in our problem, but “there is no universally best model” [Mur22]. In other words, there is no best model a priori. This result is also known as the *No Free Lunch* theorem. (Wolpert 1996 [Wol96]) .

Another important result regarding the irreducible error is the *bias-variance trade-off*. It can be demonstrated that, for unseen data (a test set) the model’s MSE error can be decomposed into a bias term and a variance term. That is it for each test point x_{test}, y_{test} we have, that the expected test MSE is (page 34 of [Jam+14]).

$$MSE_{test} = E(y_{test} - \hat{f}(x_{test}))^2 = \mathrm{Var}(\hat{f}(x_{test})) + [\mathrm{Bias}(\hat{f}(x_{test}))]^2 + \mathrm{Var}(\epsilon). \tag{2.9}$$

The bias refers to the error introduced by approximating a real-life problem, which may be complex, using a simpler model (fewer parameters or degrees of freedom than necessary). It represents the difference between the true underlying relationship and the model’s approximation of that relationship. High bias means the model is too simple and may not capture the true complexity of the data.

The variance is related to the amount by which the estimated model would change if we used a different training dataset. It measures the sensitivity of the model to the specific data points used for training. High variance means the model is too complex and may overfit the training data, leading to large changes in $\hat{f}$ with different training sets.

We can conclude that if we want a good model, we need to reduce both the bias and variance. However, it is not possible to reduce the bias (increase model complexity) without increasing the variance and vice versa.

In general, not only do we want to make predictions, but also to understand the behavior of the function $f(X)$, i.e., the relationship between the different predictor variables $X_1, ..., X_p$ [Jam+14]. This helps us understand and explain the phenomenon we are statistically analyzing. A linear parametric estimation, for example, assumes a dependence of the form $f(X) = f(\theta_0 + \theta_1 X_1 + \theta_2 X_2 + ... + \theta_p X_p)$, estimating $f(\boldsymbol{\theta}^T \cdot \mathbf{x})$ reduces to estimating these parameters θ_i associated with each variable. These parameters can help to interpret the predictions of the model, but the interpretability decreases fast with the number of parameters, and vice versa. This is known as *Trade-Off Between Prediction Accuracy and Model Interpretability* (page 24 of Ref. [Jam+14]).

Finally, it is worth mentioning that even in the case of DL models, scaling laws has been found recently that limits the improvement of performance, even adding more data or computation [Kap+20; Hen+20; Bah+24; Wel24].

2.2.6. Deep Learning and optimization of neural networks

Most problems in ML involve minimizing a loss function or maximizing a reward. We have already shown that regression involves minimizing the NLL , and classification involves maximizing the probabilities of separating classes. Now, we introduce a simple neural network to further discuss this concept.

Neural networks have been researched for many years. Like many bio-inspired algorithms, the motivation is to mimic neuron function. It is important to emphasize that, for many years, it was not well understood how to train them. The challenge was making the neurons learn correctly without overfitting. Additionally, the computational power required was not broadly available until recent years. For example, a photograph of the "Knobby" ADALINE network can be seen in Fig.2.3. An ADALINE network was the first neural network to learn how to recognize digits in a primitive segmented display [Are08] and other filtering applications [24e]. Once the training problems problems were solved, neural networks advanced rapidly. Let us look at a simple example. Fig. 2.2 shows the structure of an Adaline-type neuron.

From the left, we have an input of 1 (the bias term) and the other variables to be learned. Each variable is assigned a weight (in NNs the parameters θ are called weights $\mathbf{w}$), forming the linear combination $z = \mathbf{w}^T\mathbf{x} + \mathbf{w}_0$ of variable values and weights (Input Function). This generates a prediction through the activation function $g(z) = z$ (linear activation in this case). Then, the result is quantized (thresholded), for instance, 1 if $z \geq 0$, 0 otherwise. Learning involves changing the weights (model parameters) so that the prediction of the target variable is accurate. To achieve this, an error or cost function is defined and minimized to adjust the weights and improve the prediction. In this example, we use SSE (sum of squared errors), but there is flexibility in choosing the appropriate one. We want to minimize [20]:

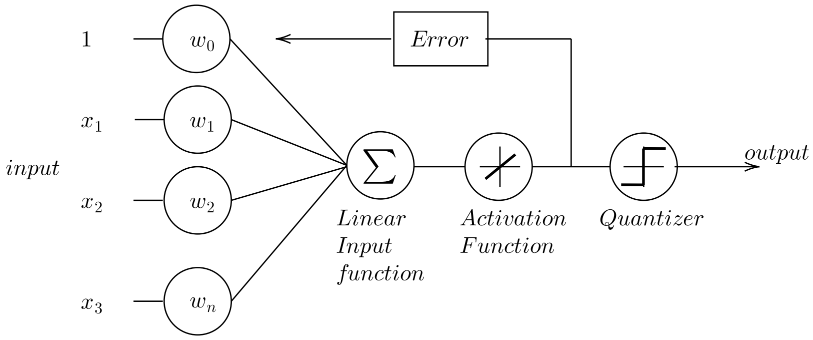


Figure 2.2.: Adaline: a simple neural network. Based on [20].

$$J(\mathbf{w}) = \sum(\text{target}(i) - \text{output}(i))^2 \tag{2.10}$$

Here, a crucial nonlinear optimization method comes into play: gradient descent. The gradient of the error function

$$\Delta w_j = -\eta \frac{\partial J}{\partial w_j} \tag{2.11}$$

is used to find the direction in which changes to the weights improve the prediction. $\mathbf{w} = \mathbf{w} + \Delta\mathbf{w}$. This concept is illustrated in the following figure: in each iteration, the prediction of a data sample is computed, the prediction error is calculated, and then each model parameter j is slightly changed by a value Δw_j. The prediction error is progressively reduced as the values of the loss $J(\mathbf{w})$ approach a minimum.

The rate at which we descend along the curve is given by the parameter η, which

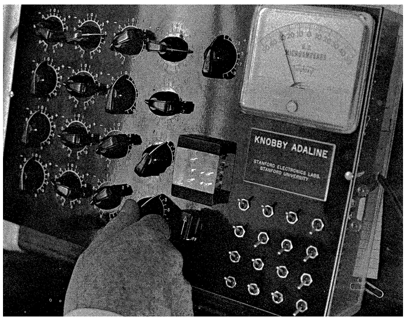


Figure 2.3.: “Knobby Adaline” (Stanford Today magazine © 1963)

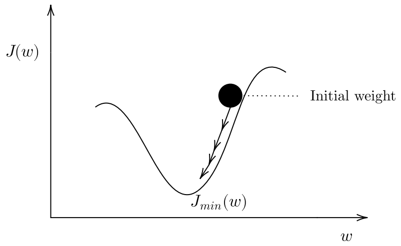


Figure 2.4.: Gradient-descent cartoon.

is known as the learning rate. In a real case, we have a large number of dimensions. It is impossible to determine the shape of the curve, so adjusting this parameter is essential for the method to converge. That means avoiding overshooting the minimum or learning too slowly. It is also important to avoid local minima.

Seen this way, optimization is a special search problem. Many ML optimizers are based on variations of pure gradient descent. Some of the most used are ADAM (Adaptive Moment Descent [KB17]) and SGD (Stochastic Gradient Descent [Rud17]). It is important to emphasize that optimization problems can be extremely complex. Many of them are $\mathscr{NP}$-hard problems, and in such cases it is hard to tell if the solution found is the best one. When this happens, a good enough solution is already celebrated.

To generalize gradient descent to more complex architectures, every network component (added layers, activation functions, etc.) must be differentiable, so automatic differentiation algorithms can properly calculate how much the weights should change in each iteration.

Neural networks architectures

To assemble a network model, it is sufficient to add the desired layers and neurons and apply this principle systematically to each neuron in each layer. The weights of one layer depend on the weights of the previous layer, and the process of differentiation extends to the previous layers, layer by layer. This is called backpropagation.

The most commonly used model, even as part of more complex architectures, is the multilayer perceptron (MLP). An MLP is a neural network consisting of multiple layers of neurons, each fully connected to the neurons in adjacent layers. An MLP typically includes an input layer, one or more hidden layers, and an output layer. Each neuron in a layer computes a weighted sum of its inputs, applies a nonlinear activation function to this sum, and passes the result to the next layer. Mathematically, if $\mathbf{x}$ is the input vector, the output $\mathbf{h}^{(l)}$ of the l-th hidden layer is given by $\mathbf{h}^{(l)} = g(\mathbf{w}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)})$, where $\mathbf{w}^{(l)}$ and $\mathbf{b}^{(l)}$ are the weights and biases of the l-th layer, respectively, and $g()$ is a nonlinear activation function such as ReLU or sigmoid. The depth (number of layers) and width (number of neurons per layer) of an MLP can be adjusted to suit the complexity of the problem, with deeper networks often capable of capturing more intricate relationships in the data.

In addition to MLPs, there are many types of layers, but they follow two missions: (1) learn features from data automatically or (2) facilitate the optimization process (avoid overfitting and other problems during gradient descent).

Convolutional Neural Networks (CNNs) are particularly effective for processing data with a grid-like topology, such as images. CNNs consist of convolutional layers that apply filters to the input, pooling layers that reduce the spatial dimensions of the feature maps, and fully connected layers that make final predictions. Recurrent layers, such as LSTMs [HS97], are particularly suitable for sequential data like time series and text. However, a more modern approach to capturing features in sequential data is the use of attention layers, as in transformers. These are highly effective for tasks requiring focus on specific parts of the input in very long sequential data, such as natural language processing. They are directly responsible for the revolution of ChatGPT [Liu+23] and other large language models of 2022.

The other types of layers that facilitate training are: Batch normalization layers, which normalize the inputs of each layer to stabilize and accelerate the process; Dropout layers, which help avoid overfitting by randomly setting a fraction of input units to zero during training; and Residual layers, often used in deep networks, which facilitate training by allowing gradients to flow more easily.

It is commonly agreed that the first model that started the DL revolution was AlexNet [KSH17]. AlexNet, used for image classification in the ImageNet dataset, outperformed the state-of-the-art (SOTA) models at the time. It has all the main ingredients that made DL possible, including automatic feature extraction with deep stacking of CNN layers; regularization to avoid overfitting (data augmentation and dropout); and training on multiple GPUs with a large dataset.

MLP networks with approximately 100 neurons and one hidden layer, along with their variations, were the first successful applications in ML. In recent years, model sizes have increased exponentially. Nowadays, a model like AlexNet with 60 million parameters and 650,000 neurons is considered small in comparison with large language models like Llama 3.1, which has 405 billion parameters ([fac24]).

But what exactly are the advantages of adding so much depth? For example, in computer vision, deep networks have allowed automatic feature extraction. Previously, to recognize a face, it was necessary to specify which geometric parameters (features) allowed a face to be identified. Another advantage is improved performance in areas where classical ML was stagnating, taking advantage of increased amounts of data. However, if the amount of data is not enough, these advantages might disappear.

Remarkably, it was not believed that models with this many parameters could be trained due to the bias-variance trade-off. Nevertheless, they work. Exactly why deep neural networks work is still an open question. Although big datasets and regularization appear to play a big role in it, understanding how DL makes predictions is fundamental to deploying high-risk applications, like driving a car (or a tokamak).

2.2.7. Regularization: avoiding overfitting and facilitating training

Given its importance in DL and this thesis work, we further discuss what is regularization and how is implemented.

If the gradient descent is like a boat sailing the sea, sometimes the sea is very rough and stormy. Not always is easy to find the direction to port (the optimal solution - the best model parameters). Convex optimization offers some guarantees (that the solution found is optimal). However, real-life loss function landscapes are uncharted territory, and they can be highly nonconvex, so there are multiple local minima to sort out.

To ameliorate the situation, some techniques that modify the loss function landscapes have been developed. There are 4 types of regularization strategies: (1) Smooth the loss function so the gradient descent does not keep trapped in local minima (2) inflate data artificially (increase variance of data) (3) combine different models (with different parameters, bias-variance) (4) find wider minima during optimization. Most popular techniques are summarized in Fig.2.5 (More details in chapter 9 [Pri23]). In this thesis, we used Explicit regularization that modifies the loss function, and use random features that create noisy weights in input layers.

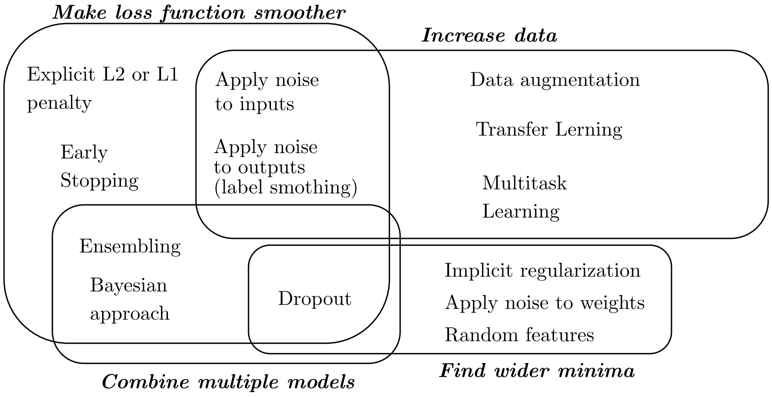


Figure 2.5.: Regularization types (Adapted from [Pri23].

In the case at hand, we can use explicit regularization: Lasso (L1 regularization) [Tib96] and Ridge (L2 regularization) [HK00] , which add a penalty term to the loss function to prevent overfitting and limit the influence of outliers.

The penalization of weights is done using different norms, that give different relevance to the parameters. We could use for instance the ℓ_0 norm, denoted as $\|\mathbf{x}\|_0$, which is defined as the number of non-zero elements in a vector $\mathbf{x}$, noted as $\|\mathbf{x}\|_0 = \sum_{i=1}^{n} \mathbf{1}(x_i \neq 0)$ where $\mathbf{1}$ is the indicator function. When minimized, it promotes the sparsity of weights during training. The problem is that the indicator function is not differentiable, so we can not use gradient descent.

The ℓ_1 norm, also known as Manhattan or city-block norm, is the sum of the absolute values of the elements of $\mathbf{x}$, given by $\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$. It promotes sparsity (more relaxed than ℓ_0) and it is differentiable.

The ℓ_2 norm, or Euclidean norm $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$, is the square root of the sum of the squares of the elements of $\mathbf{x}$. These norms measure the size or length of vectors in different ways. It produces a shrinkage of all the parameters, in other words, any parameter is much bigger than the others. This situation is favorable when there are difficulties in the optimization (backpropagations are more stable).

The lasso loss function can be written as

$$\text{Lasso:} \quad \underset{\boldsymbol{\theta}}{\arg\min} \left\{ \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \lambda \|\boldsymbol{\theta}\|_1 \right\}, \tag{2.12}$$

and the Ridge regression loss is

$$\text{Ridge:} \quad \underset{\boldsymbol{\theta}}{\arg\min} \left\{ \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \lambda \|\boldsymbol{\theta}\|_2^2 \right\}. \tag{2.13}$$

The frequentist interpretation or regularization is usually explained with Fig.2.6. The equipotential curves of the loss function are ellipses. On the left, the surfaces of the square represent norm values equidistance to the origin for different penalty λ_1 strengths of L_1 norm. Comparing, to the Euclidean norm on the right we can see that distances are measured differently. The parameters β_1 and β_2 for an arbitrary given module $\|\mathbf{x}\|_1$ can lay along the squared lines. However, if this norm $\|\mathbf{x}\|_1$ is minimized, the parameter values will tend to be located on one of the axes. In contrast, the L_1 norm promotes to have some mixture in β_1 and β_2.

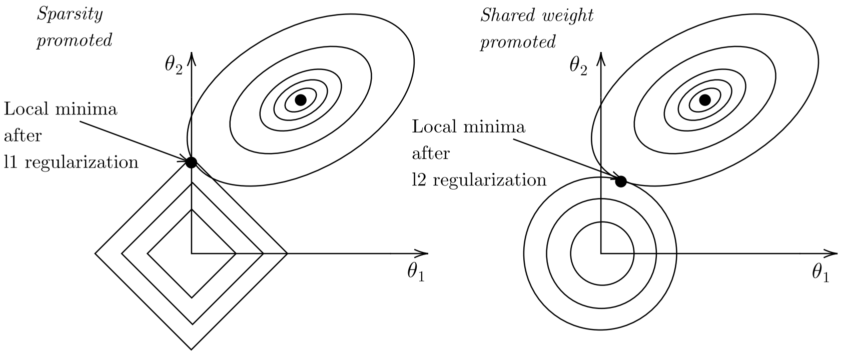


Figure 2.6.: ℓ_1 and ℓ_2 effect on a 2-parameter model.

It is worth noting that the Bayesian approach to add regularization is adding prior parameter distributions like the Laplace prior (most probable weight is 0 or very close to 0). The effect of regularization can be also understood as a feature selection, only the most important are kept.

To conclude, it's worth summarizing the ingredients that made the DL revolution possible, regarding the training process: (1) automatic differentiation and better gradient descent algorithms (2) regularization (3) massive parallelization using GPUs. Concerning the feature extraction (1) Deep CNNs for images, (2) Attention mechanism for transformers (text).

2.2.8. Approaches to implement a ML solution

Let us comment on the 3 basic approaches to developing an ML application. The differences between classical algorithms, ML or DL approaches to design a ML application are summarized in Fig.2.7.

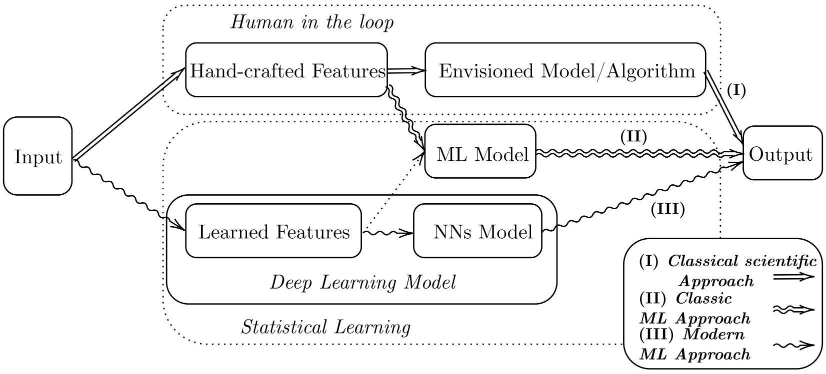


Figure 2.7.: Approaches to data modeling (Adapted from [Sze22] page 189 and [GBC16] page 10).

It is important to remark that ML is not a new subject by any means. Scientists have developed over the years a huge variety of algorithms to interpret the data collected. We call this approach, classical scientific approach (I) in Fig.2.7. In this approach, all data is analyzed and studied, then the experience gained by scientists mining and knowing the data can be added to a mathematical model.

Classic ML approaches (II) still relied on hand-crafted features and the Mapping to the output is done mainly with a statistical model. Good performance and automatic learning can be achieved while understanding the model input. In contrast, DL approaches (III) learn the features, for example by CNN networks or latent representations, and then more NN layers map the learned representation to the output.

Although it depends on the particular case. The advantages of more classical approaches are interpretability and potential generalization outside the available dataset (classical approaches potentially work for any tokamak), while the advantages of more modern approaches are better performance if enough data is available. The disadvantages of classical approaches are that feature extraction and mathematical modeling can be time-consuming, while the drawbacks of DL are mainly the concerns about its lack of interpretability and generalization.

The 3 approaches explained have been explored in this thesis work. For this reason, 3 major research areas are investigated: signal processing, ML and computer vision (for feature extraction).

2.3. Signal analysis: alternating data representations

The data contained in time signals and images is incredibly dense, easily reaching millions of samples per instance. Consequently, disentangling relevant information from noise can be extremely difficult. In the area of signal analysis, data is transformed into alternative representations that help recognize patterns and interpret the signals.

Many applications have emerged from researching alternative data representations. For instance, data can be compressed, reducing memory and computational resources. In addition, noise can be removed, or certain types of features can be enhanced.

In natural language, words stored in a dictionary form sentences and transmit information. Moreover, the choice of words depends on the dictionary used, as many different languages are spoken worldwide. Stéphane Mallat used this analogy to construct a theoretical framework for modern signal analysis, where a signal can be represented using different dictionaries. His monograph [Mal09], "A Wavelet Tour of Signal Processing", covers most established techniques in the field. The following introduction to signal analysis is based on notes from its introductory chapter. For convenience, we use the same notation of this book. In addition, a monograph on Sparse signal coding that can be consulted is Elad's book [Ela10] or Hastie's book [HTW20] on statistical learning with sparsity.

2.3.1. Fourier transform

Modern signal analysis began in the early nineteenth century with Fourier's harmonic analysis. Fourier, while studying the heat equation, discovered that any periodic signal can be represented as a sum of sine and cosine functions. He presented this result to the Paris Institute in 1807. The coefficients (expressed as amplitude and phases) of this series of trigonometric functions (harmonics) reveal patterns in the signal.

For continuous functions $f(t)$ (or signals), the Fourier transform (FT) is defined by projecting the signal onto a basis of complex exponentials, using the following integral:

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t) \exp^{-i\omega t} dt, \tag{2.14}$$

expressing the signal in terms of its harmonic frequencies ω. This equation solves the analysis problem, or decomposition of the signal. The signal can be recovered from the frequency space using

$$f(t) = \int_{-\infty}^{+\infty} \hat{f}(\omega) \exp^{i\omega t} d\omega, \tag{2.15}$$

which solves the synthesis problem, or reconstruction of the signal.

Without detailing the rigorous definition of the discrete version, we can say that the integrals are replaced by discrete sums over the signal samples. The most widely used algorithm for calculating the Fourier transform in digital signals is quite fast, with $\mathcal{O}(N \log N)$ time complexity (N is the number of signal samples). Interestingly, this algorithm was introduced by IBM researchers while the U.S. military was searching for a fast way to detect Soviet nuclear tests. Later, it was found that Gauss had already developed it centuries earlier.

For discrete signals (all digital signals), the signal is projected onto a discrete orthogonal basis:

$$\mathscr{D} = \{\exp^{i2\pi kmt}\}_{m\in\mathbb{Z}}, \tag{2.16}$$

where the coefficients have amplitude values between 0 and 1 (and a phase between -∞ and ∞, or wrapped between 0 and 2π). This basis constitutes our first type of dictionary $\mathscr{D}$. If the signal is regular enough, very few frequencies will represent it, and in such cases, FT offers a sparse representation of the signal.

FT is ubiquitous in signal analysis. Its most remarkable property is that it “diagonalizes time-invariant convolution operators”, [Mal09]. In other words, in the frequency space, convolution operations between two functions become simple multiplications. Consequently, FFT is the foundation for other signal transformations.

2.3.2. Wavelet transforms

An alternative to Fourier bases is wavelet bases. The first wavelet function, introduced by Haar in 1910 [Haa10], is defined as:

$$\psi(t) = \begin{cases} 1 & \text{if } 0 \le t < \frac{1}{2}, \\ -1 & \text{if } \frac{1}{2} \le t < 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2.17}$$

Using the Haar wavelet, a signal is represented by a basis of piecewise stepped waveforms, in contrast to the trigonometric functions of FT. The Haar waveform can be located at a given time t using dilations $1/\sqrt(2^j)$ and translations $\frac{t-2^j n}{2^j}$. These are used to construct a dictionary:

$$\mathscr{D} = \left\{\psi_{j,n}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-2^j n}{2^j}\right)\right\}_{(j,n)\in\mathbb{Z}^2}. \tag{2.18}$$

The scale is roughly equivalent to FT frequencies, being larger for lower-frequency components and shorter for high-frequency components.

This wavelet orthonormal basis is used to reconstruct the signal $f(t)$ as:

$$f(t) = \sum_{-\infty}^{+\infty}\sum_{-\infty}^{+\infty}\langle f, \psi_{j,n}\rangle\psi_{j,n}, \tag{2.19}$$

where $\langle f, \psi_{j,n} \rangle$ is the inner product defined by:

$$\langle f, \psi_{j,n} \rangle = \int_{-\infty}^{+\infty} f(t) \psi_{j,n}(t) dt. \tag{2.20}$$

These coefficients $\langle f, \psi_{j,n} \rangle$ will be large if the signal has localized stepped variations (or transitions). The Haar wavelet is particularly useful for processing "square-like" signals. Under the right conditions, this wavelet can provide a sparse representation of the signal, while FT may struggle with stepped discontinuities.

The next major breakthrough in wavelet analysis came in the 1980s. While Meyer was attempting to prove that a piecewise function could not provide a better approximation of continuous signals [Mey86] (unaware of Strömberg's proof [Str82]), he ended up creating a new family of continuously differentiable wavelets [MM04]. This led to a race to extend the available functions, with significant contributions from Daubechies (compactly supported wavelets [Dau88]) and Mallat-Meyer (multiresolution expansion [Mal89]). The culmination of this research was the discovery of a link between wavelet bases and filter banks, which allowed for fast calculations ($\mathcal{O}(N)$) using cascades of mirrored filters [Mal09]. Mallat demonstrated that any set of conjugate mirrored filters can define a wavelet family with orthonormal bases. This relationship between filter banks and wavelet families was inspired by an algorithm from the computer vision field called pyramid filters [BA87].

An important property of these filter bank decompositions is that each decomposition level corresponds to a different scale, in contrast to the Fourier transform, which has the same basis vector scale for all frequencies. These abstract concepts are better illustrated with examples. The three mother wavelets represent the horizontal, vertical, and diagonal features of the signal across different scales (or levels).

A 2D gray image of N pixels can be represented as $f(x) = f(x_1, x_2)$, using three mother wavelets $\psi^1(x)$, $\psi^2(x)$, $\psi^3(x)$ (horizontal, vertical, and diagonal details). The signal can be expanded using the dictionary:

$$\mathcal{D} = \left\{ \psi_{j,n}(x) = \frac{1}{2^j} \psi^k \left(\frac{t - 2^j n}{2^j} \right) \right\}_{j \in \mathbb{Z}, n \in \mathbb{Z}^2, 1 \le k \le 3}. \tag{2.21}$$

An example of an image decomposition using this dictionary is shown in Fig.2.8, where for each level, we have the approximation image CA, horizontal CH, vertical CV, and diagonal components CD.

2.3.3. 2D Fourier transform

Alternatively, a 2D version of the FFT for a rectangular image $f(x_1, x_2)$ of $N = N_1 \times N_2$ pixels can be implemented using two separable bases: $\mathcal{B}_1 = \{\exp^{i2\pi k_1 n_1 / N_1}\}_{0 \le k_1 \le N_1}$ and $\mathcal{B}_2 = \{\exp^{i2\pi k_2 n_2 / N_2}\}_{0 \le k_2 \le N_2}$, or:

$$\mathcal{D} = \{\mathcal{B}_1, \mathcal{B}_2\} = \left\{ \{\exp^{i2\pi k_1 n_1 / N_1}\}_{0 \le k_1 \le N_1}, \{\exp^{i2\pi k_2 n_2 / N_2}\}_{0 \le k_2 \le N_2} \right\}. \tag{2.22}$$

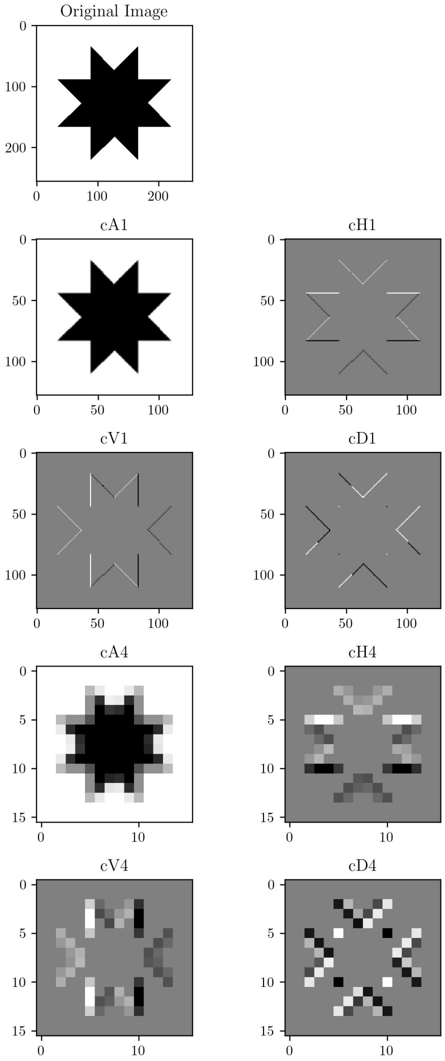


Figure 2.8.: Example of DWT decomposition levels 0, 1, and 4.

The analysis is defined in 2D as:

$$\hat{f}(k_1,k_2) = \frac{1}{N_1 N_2} \sum_{(x_1=0)}^{N_1-1} \sum_{(x_2=0)}^{N_2-1} f(x_1,x_2) \exp^{-i2\pi\left(\frac{k_1 x_1}{N_1}+\frac{k_2 x_2}{N_2}\right)}, \tag{2.23}$$

or for continuous functions:

$$\hat{f}(\omega_1,\omega_2) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x_1,x_2) \exp^{-i(\omega_1 x_1+\omega_2 x_2)} dx_1 dx_2. \tag{2.24}$$

A toy example of FFT2D use is shown in Fig.2.9, which reveals high-frequency details of the image.

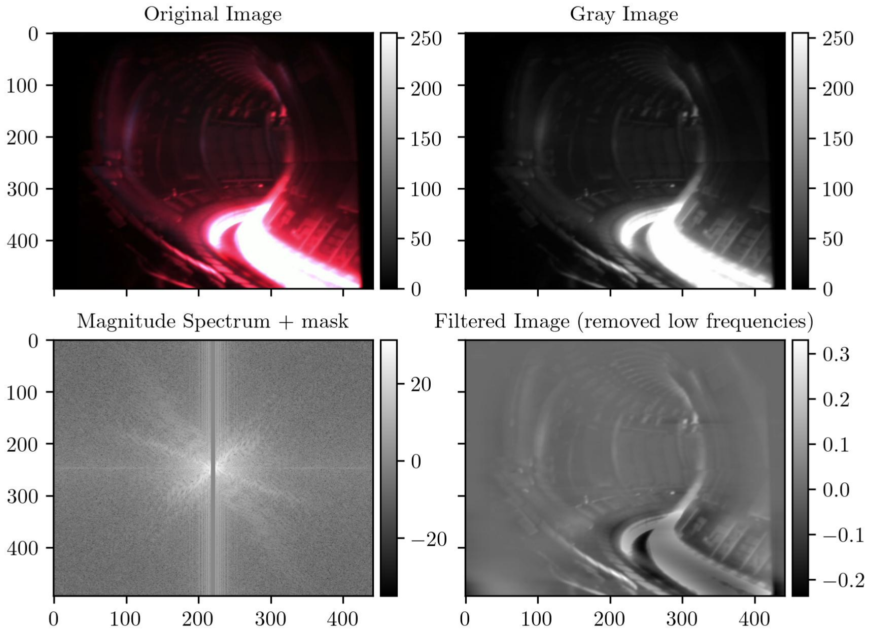


Figure 2.9.: Example of 2D Fourier decomposition and high-pass 2D filter.

2.3.4. Time-frequency analysis: Spectrograms and scalograms

When a signal is measured and digitized, a continuous function is approximated by a discrete one. It is worth mentioning that the discretization using Fourier and Wavelet bases is a linear expansion. However, complex signals can benefit from a nonlinear basis expansion, where basis vectors concentrate along edges, contours, or other localized features to improve the signal approximation.

The reader might have noticed that FT does not provide time-localized information. When transforming to the frequency domain, we lose time-specific features. However, in natural sounds, music, or plasma signals, different features occur at different times. To address this, let us introduce time-frequency dictionaries: Gabor or Windowed-Fourier, and the Continuous Wavelet Transform (CWT).

The first author to propose such a basis was Gabor in 1946. Inspired by quantum mechanics' wave-particle duality, he referred to the elements of this basis as "atoms." He applied a window function g to the Fourier basis to localize the waveform in time. The time-frequency "atoms" $g_{(u,\xi)}(t)$ form a dictionary:

$$\mathscr{D} = \left\{ g(t-u)\exp\{i\xi t\} \right\}_{(u,\xi)\in\mathbb{R}^2}. \tag{2.25}$$

Typically, Gabor's atoms use a Gaussian window function:

$$g(t-u) = \exp\left\{ \frac{-(t-u)^2}{2\sigma^2} \right\}. \tag{2.26}$$

In addition, the discrete Cosine transform is commonly used instead of the complex Fourier basis. The coefficients of this dictionary are calculated using the Short-Time Fourier Transform (STFT), where the windowed signal is projected onto a Fourier basis:

$$STFT\{f(u,\xi)\} = \langle f, g_{u,\xi}\rangle = \int_{-\infty}^{+\infty} f(t)g(t-u)\exp^{-i\xi t}dt. \tag{2.27}$$

This is equivalent to computing the Fourier transform on a windowed segment of the signal. In practice, the signal is digitized, and FFT is used. The result can be interpreted as a 2D image representation (time-frequency spectrogram).

It is important to note the trade-off between time and frequency resolution. Increasing the window size δ_t reduces frequency uncertainty δ_ω but delocalizes the waveform in time, and vice versa. This phenomenon is known as the Heisenberg uncertainty principle: $\delta_t\delta_\omega \geq \frac{1}{2}$, or the time-frequency trade-off.

So far, we have constructed dictionaries with linear bases of various functions: orthonormal bases, continuous or piecewise. Other orthonormal bases include the Local Cosine Transform and wavelet packets. The issue with orthogonal bases is that they can be too rigid for representing complex signals, as the projection between basis components may not separate features well. Sometimes redundancy is needed to represent complex regions of an image.

While omitting the full mathematical framework, it is essential to define dual frames to establish the mathematical conditions for a redundant basis to provide stable analysis and synthesis. Analogous to our original metaphor, dictionaries composed of frames have more "words" to describe the same signal. The duality in frames ensures stable analysis and synthesis, while redundancy (determined by frame bounds) improves the representation. Thus, frames can be seen as a generalization of bases. Both STFT and CWT are rigorously defined with dual frames. Additionally, to address the lack of directionality in wavelets for images, new frame families have been introduced: Gabor wavelets (which model image features similarly to the V1 brain cortex [OF97a]), Steerable wavelets, and the Curvelet transform.

2.3.5. Curvelet transform

The Curvelet transform, introduced by Candés and Donoho [CD05], enhances the representation of multiscale and multidirectional features, such as edges and curves. The synthesis problem is defined as in page 198 of [Mal09]:

$$f(x) = \sum_{j\in\mathbb{Z}} \sum_{a\in\Theta} \sum_{m\in\mathbb{Z}} \langle f, c^{\alpha}_{j,m}\rangle c^{\alpha}_{j,m}(x), \tag{2.28}$$

where α represents a direction and m a translation along that direction, with a scale width of 2^j. Translations along a direction are almost symmetrical, so we can describe the process in terms of both scale and angles. This concept is better understood with an example: in Fig.2.10, wedges of different scales are represented. Notice how they capture more directional details compared to the DWT shown in Fig.2.8.

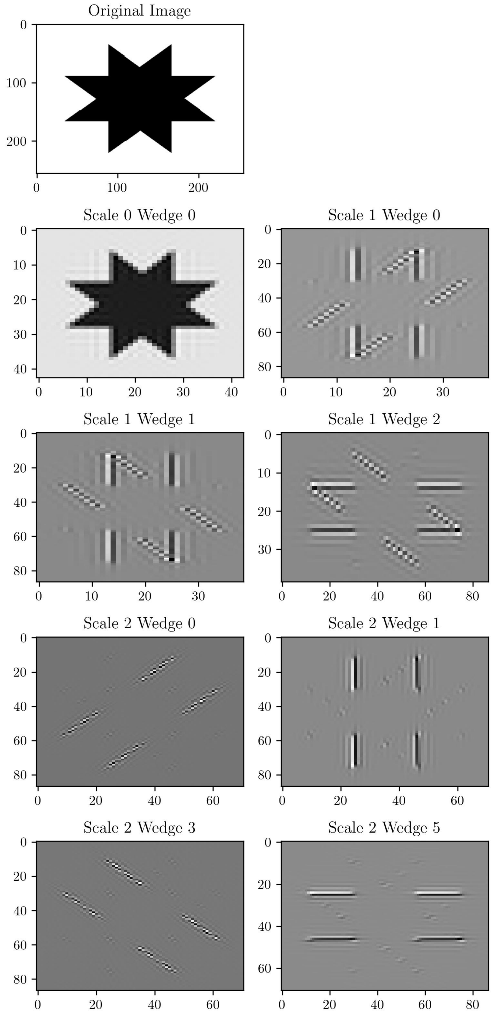


Figure 2.10.: Curvelet decomposition example showing only some wedges.

Each curvelet $c^{\alpha}_{j,m}(x)$ represents localized ridges in the image. In the fast Curvelet algorithm, each edge curvelet $c^{\alpha}_{j,m}(x)$ corresponds to a “wedge” $\hat{c}^{\alpha}_{j,m}(\omega)$ in the frequency domain. Therefore, Curvelets are calculated by tessellating the Fourier space.

2.3.6. Basis pursuit

Can we go further than frames? We can reformulate the analysis and synthesis problem as an optimization problem to obtain a sparse representation using redundant dictionaries. Given a dictionary $\mathscr{D} = \{\phi_p\}_p \in \Gamma$ and a signal f, the ideal representation is obtained by minimizing the Lagrangian $\mathscr{L}_0$: we seek a combination of vectors Λ that reconstruct the signal f_Λ with minimal error $||f - f_\Lambda||^2$:

$$\mathscr{L}_0(\lambda_0, f, c) = ||f - f_\Lambda||^2 + \lambda_0^2||\Lambda||_0. \tag{2.29}$$

The main obstacle is that this problem is NP-hard (non-deterministic polynomial-time hard), making it impractical to solve exactly. However, finding a "good enough" solution is a tractable alternative. Several approaches exist: (1) thresholding with dictionaries of orthonormal bases, (2) greedy algorithms, and (3) basis pursuit with l_1 relaxation.

The "best basis theorem" [DJ+94] states that the best basis for sparse approximation can be obtained when working with dictionaries of orthonormal bases, provided the basis is conveniently thresholded. Bases like wavelet packets, local cosine bases, or bandlet orthonormal bases can be used. However, orthogonal bases may lack sufficient degrees of freedom (too rigid) to represent complex signals.

Greedy algorithms, such as matching pursuit or orthogonal matching pursuit [MZ93], can be very useful due to their defined convergence properties (complexity $\mathcal{O}(MN\log(N))$ with M iterations). They can also be applied to find the best basis in Gabor dictionaries with carefully designed time-frequency locations, such as those introduced by Qian and Chen, with a complexity of $\mathcal{O}(N^2\log(N))$. These algorithms are fast but can fall into local minima, as greedy strategies are not always optimal.

In this thesis work, a novel approach to obtain a sparse basis decomposition using l_1 and l_2 norms is introduced.

2.4. Computer vision

Computer vision (CV) studies algorithms and models for processing, analyzing, and interpreting images and videos. The ultimate goal is to imitate human visual perception. To tackle the complexity of visual understanding, the field is divided into specialized tasks or topics.

Among the tasks studied in CV are [Sze22]: image formation, image processing, image recognition (image classification, object detection, segmentation, etc.), feature detection and matching (points, patches, edges, ridges, contours, straight lines), image alignment and stitching, motion estimation, computational photography, structure from motion, simultaneous localization and mapping, depth estimation, 3D reconstruction, rendering, and mathematical modeling (ML, Bayesian approaches, and DL).

Deep learning has significantly disrupted this field. However, the knowledge base of CV still contains approaches of all types (Fig.2.7). The classic computer vision approach differs from the deep learning approach in several aspects. In classic CV, image features are extracted by designing a workflow of algorithms and mathematical models. For example, a composition of wavelet filters can remove grain noise in a photograph, and then objects can be detected by comparing them to a database.

Signal analysis introduced earlier in this chapter, such as Fourier transforms and wavelet transforms, are part of CV algorithms and represent a whole field on their own. For instance, wavelets are extensively used for image compression and manipulation (filtering, merging, imprinting).

To illustrate how diverse the problems and approaches in CV are, we can follow a rough timeline [Sze22] of the most active research topics in CV over the last decades: (1970s) Digital image processing, Blocks world, Line labeling, Generalized cylinders, Pattern recognition, Stereo correspondence, Intrinsic images, Optical flow; (1980s) Structure from motion, Image pyramids, Shape from shading/texture/focus, Physically-based modeling, Regularization, Markov random fields, Kalman filters; (1990s) 3D range data processing, Projective invariants, Factorization, Physics-based vision, Graph cuts; (2000s) Particle filtering, Energy-based segmentation, Face recognition and detection, Image-based modeling and rendering, Texture synthesis and inpainting, Computational photography; (2010s) Feature-based recognition, Category recognition, ML, Modeling and tracking humans, Semantic segmentation, SLAM and VIO, Deep learning; (2020s) Vision and language.

These algorithms were developed over years of research and now require very few computational resources. They do not require a dataset of images. Their effectiveness relies on assumptions about the image properties, making the mathematical manipulation of the image deterministic and highly interpretable. For instance, the strength of wavelets lies in recognizing multiscale features in signals and images. However, it is necessary to experiment with different algorithms and parameters for each image.

In contrast, feature extraction in DL models is done automatically. For example, using convolutional neural networks (CNNs), features are learned using inductive bias, so deep neural networks need to be trained with labeled datasets. Roughly speaking, a CNN can be understood as a concatenation of image filters, where the parameters are learned using statistics from a given dataset to solve a specific problem. The fundamental difference between classic CV and DL approaches is that in classic CV, image features are calculated, whereas in DL, image features are inferred.

If the amount of data is sufficient, deep learning can outperform classic CV. However, large image datasets are needed to achieve this advantage, and DL performance collapses when the patterns to detect are absent from the training dataset [ON21]. Additionally, DL models often require multiple GPUs, and datasets need careful preparation before use. For instance, images must be downsampled, centered, and cropped to ensure objects occupy most of the image area. Moreover, labeling images is still a time-consuming task necessary for training DL models. Therefore, CV and signal processing are useful when data is scarce or computational resources are constrained. CV is also used to prepare data for deep learning algorithms, often enhancing DL performance [OMa+20].

2.4.1. Feature extraction with edge and ridge filters

We can conclude that classic CV can be valuable for fusion applications since its models are much more interpretable than DL models, and CV algorithms can be applied easily to data from different machines if their assumptions hold.

In this thesis, we have explored the use of feature extraction algorithms, particularly ridge detectors combined with automatic threshold selection, to mark modes in spectrograms. Before continuing, we should clarify the difference between edges and ridges and provide some details about our image data.

Given a color image, we most likely have three matrices of pixel data corresponding to the RGB channels. It's worth noting that other color schemes, such as HSV, define color with three matrices as well: Hue (color frequency), Value (brightness), and Saturation (amount of color mixed with white). In contrast, grayscale images have only one matrix of pixels. In the case of spectrograms, the time-frequency amplitude of the STFT is used to form an image. Consequently, spectrograms are grayscale images. However, we use false color palettes to distinguish details better, as color improves our perception of contrast and dissimilarity.

The fluctuation and structure of pixel intensity determine the features of an image. The pixel magnitude value (intensity) can be seen as a map of a landscape with valleys, plateaus, mountains with ridges, and abrupt changes like edges or spikes caused by noise. In Fig.2.11, we depict an oversimplified 1D cut of an image's pixel intensity, highlighting edges and ridges.

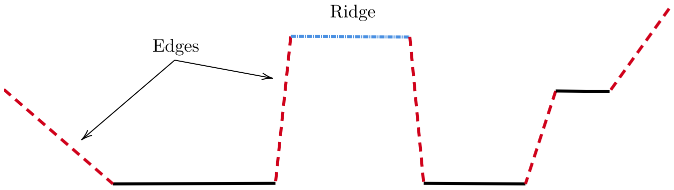


Figure 2.11.: Edges and ridges.

To mathematically define ridges and edges, we use calculus: the gradient (first derivatives) and the Laplacian or Hessian matrix (second derivatives). The gradient value of an image indicates changes in pixel intensity, and the zero-crossings of the Laplacian indicate maxima or minima.

Without diving into the mathematical details, the magnitude, gradient, and Laplacian are the basis for different detectors. For example, pixel magnitude is used in the "watershed" [NS94; BLM14] segmentation algorithm, the gradient is used in the Sobel edge detector [Sob+22], and the Laplacian is used in the "Canny" edge detector [Can86]. Additionally, the eigenvalues of the Hessian matrix are used in most ridge detection algorithms.

To illustrate the difference between edge and ridge detector results, we could have used classic benchmark images like Lena, the cameraman, or the peppers. Instead, we used a fusion-related image: a photograph from JET's shot 99971 (a fusion power record shot). The results of applying the Sobel edge detector and Meijering ridge detector are shown in the first row of Fig.2.12. In the second row, the image is binarized with the automatic threshold algorithm called Sato. Four observations can be made: (1) Thresholding the original image without feature enhancement reveals few details. (2) Using edge and ridge detectors reveals many details of the in-vessel components, such as the tiles or the divertor edges, though the results differ. (3) Edge detectors mark borders around structures, while ridge detectors label the structures themselves (e.g., look at the ceiling tiles). (4) The edge detector detects both increasing and decreasing pixel fluctuations, making the result noisier.

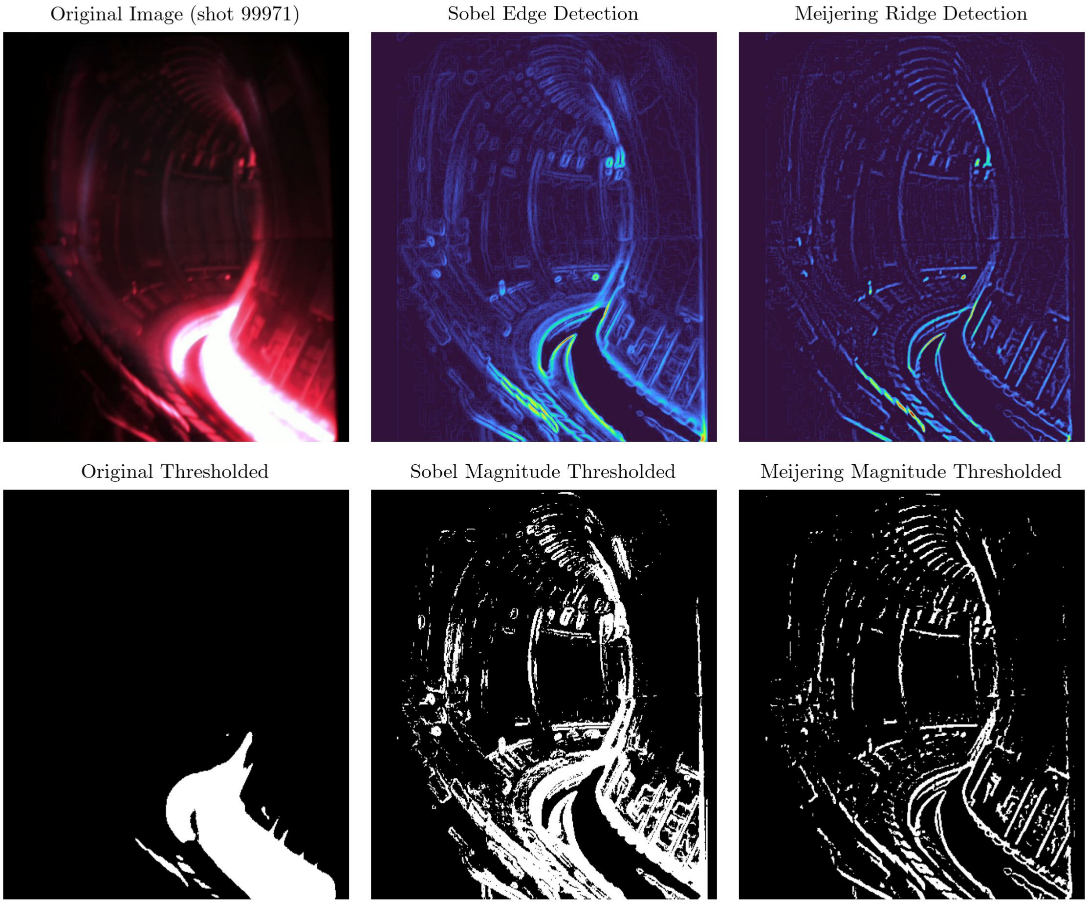


Figure 2.12.: Comparison between edge detection and ridge detection, and their thresholded results.

We can conclude that modes present in spectrograms can be detected using segmentation, edge, and ridge detectors. However, ridges more accurately model the presence of time-frequency structures. Interestingly, the human eye detects modes using edge detectors: experiments identify neurons called simple cells in the visual cortex V1 as oriented edge detectors (directional Gabor filters [OF97a]), and later, other parts of the brain identify the objects.

There are several types of ridge detectors available in open-source libraries like Scikit-Image [Van+14]. In this work, we tested Meijering, Sato, Frangi, and Hessian ridge detectors. These algorithms are extensively used in microscopy imaging to detect vessels and other structures. Sometimes, ridge structures appear as black in the images, so the algorithms have an option to detect them. Another feature of these algorithms is that they can be applied at multiple scales to detect both broad and narrow ridges simultaneously, which can also be useful for detecting modes.

The mentioned ridge detectors were applied to the JET photograph to compare the different results (Fig.2.13). Once the ridges are enhanced, the same threshold technique is applied to all of them. The parameter σ (or a list) determines the scale in pixel units to which the detectors are applied. It can be seen that Hessian tends to be noisier, the Frangi method detects broader ridges, and Meijering and Sato give intermediate results.

2.4.2. Automatic thresholding

Thresholding is used to separate the image's foreground and background. Thresholding an image is not a trivial task for two reasons. First, a single value for the entire image might not work due to abundant local features. Second, the result is very sensitive to the selected value, as pixel intensity changes rapidly. It is also important to note that in color images, the results change if channels are thresholded separately or after converting the image to grayscale. To address these issues, several automatic thresholding algorithms have been developed over the years, including Isodata, Li, the Mean method, the Minimum method, Otsu, the Triangle method, and Yen. These algorithms analyze the modality of the pixel intensity distribution to determine a global value. All are available in the Scikit-Image library [Van+14].

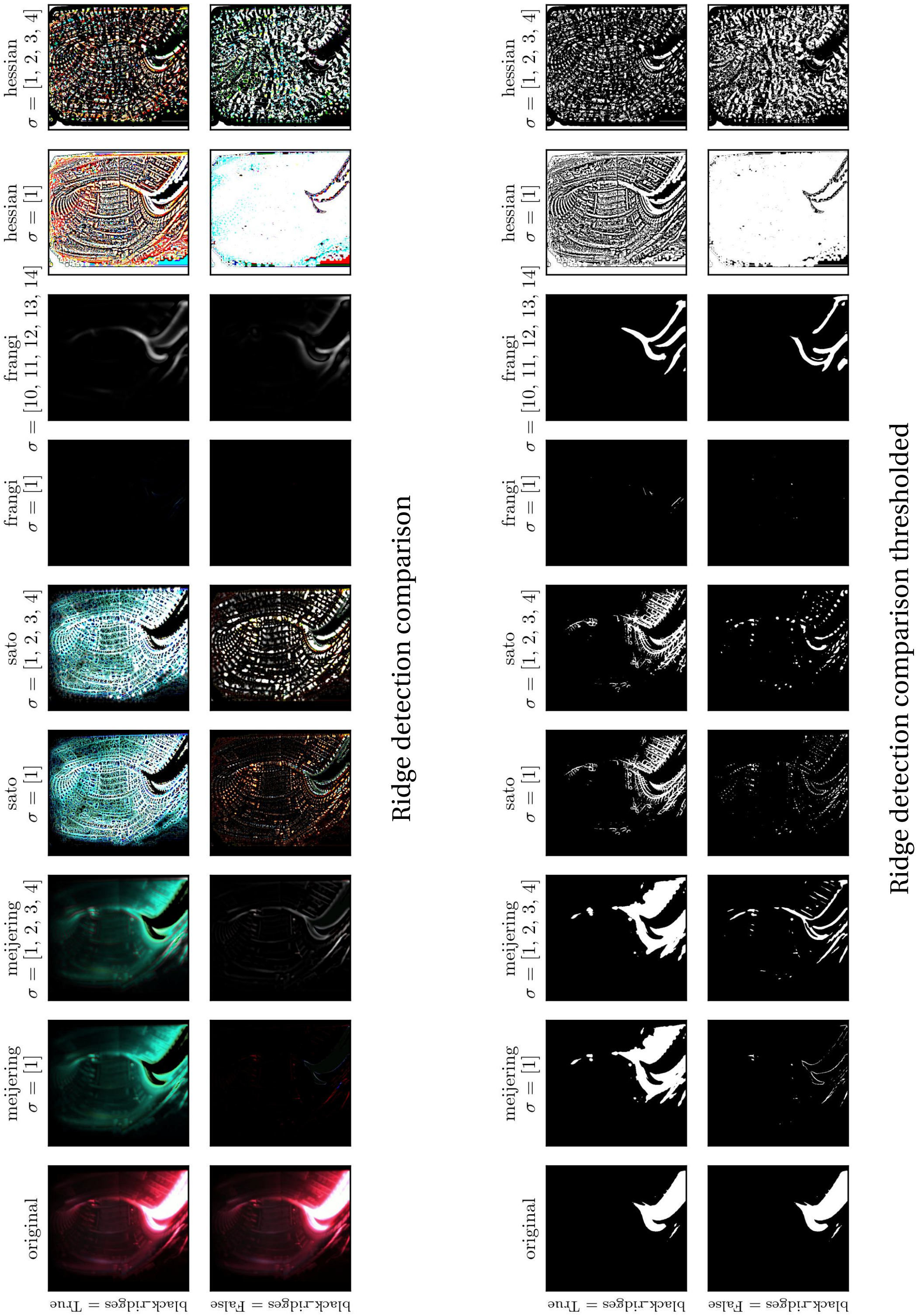


Figure 2.13.: Examples of different ridge detection algorithms on JET in-vessel view (shot 99971).

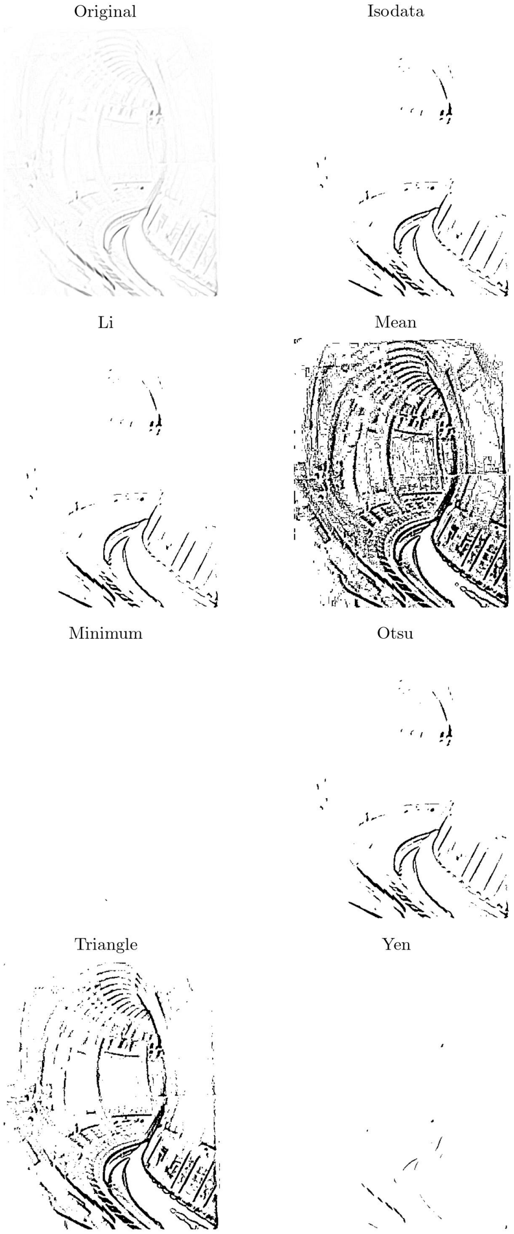


Figure 2.14.: Comparison of different automatic thresholding algorithms on Features from Meijering ridge detector of Fig. 2.12.

In Fig.2.14, the following automatic threshold methods (Isodata, Li, the Mean method, the Minimum method, Otsu, the Triangle method, and Yen) are used to binarize the same image. They all use different approaches for global auto-thresholding. To give some details (from page 69 of Ref. [Kri16]): the Isodata method iteratively refines the threshold by computing the means of foreground and background regions until convergence; Li's method uses an iterative cross-entropy thresholding approach; the Mean method sets the threshold at the image's mean intensity; the Minimum method assumes a bimodal histogram and iteratively identifies the threshold as the minimum point between two peaks; Otsu's method maximizes the variance between classes; and the Triangle method uses the histogram's peak, assuming the peak is not centered, and sets the threshold in the largest region. These methods can be used like black boxes as they do not have hyperparameters to adjust. Depending on the image statistics, some methods perform better than others, so comparing the results is necessary before incorporating one into a CV pipeline.

2.5. Machine learning and Computer Vision on detection of Alfvén instabilities

To this date and to the best of our knowledge, several works have approached the detection of EP modes using supervised ML with data from different tokamaks: in COMPASS [Škv+20] generative neural networks (Variational autoencoders and Gaussian Mixture models) have been trained to detect chirping modes; in NSTX [Woo+20] the use Random Forest tree models to predict several labels (quiescence, fixed-frequency, frequency sweeping, chirping, and fishbone-like and abrupt large events); in DIIID, the availability of a big labeled dataset of the ECE and CO2 interferometers spectrogram (with 4 labels: LFM, BAE, RSAE, TAE, EAE) has produced many results [Jal+21; Jal+22; Kap+22a; Gar+23] with different model architectures ranging from RCN, MLP or LSTM architectures. All of the works mentioned mainly use models that are trained on labeled spectrogram images, with various neural network architectures.

In TJ-II stellarator, previous work from Bustos et al. [Bus+21], used supervised deep learning to extract modes automatically from Mirnov coils signals spectrograms. They manually annotated ~500 spectrograms to create a training dataset for image segmentation. The trained model was able to segment the spectrogram images, recovering 989 modes on the full dataset.

Because labeling spectrograms is a costly and tedious task. These works motivated us to research an unsupervised labeling technique of mode signals. Because deep learning networks can potentially boost their performance by employing more training data [Hes+17], the capability to create a bigger labeled database of modes with intershot analysis is a necessary step to generate enough data to train real-time accurate models.

Another application close to this thesis work is the toroidal mode number calculation [Kle+08] of Mirnov coil signals using sparse regression of the complex Fourier spectra. The SparSpec algorithm [BCB07] original purpose was calculating periods (frequencies) in unevenly sampled astrophysical signals. This work uses a ℓ_1 regularization and an optimization algorithm based on step coordinate descent [BCB07]. In this model, the detection of MHD activity requires the existence of a coherent signal detected by the array of magnetic diagnostics. Other approaches using coherence or cross-power spectra can identify mode activity [Mel+12]. In fact, all methods for mode number calculation can be used to extract patterns from MHD spectra using the 3D phase information as well. They are based on: phase fitting of two or multiple coils, using the FFT or wavelets [Pok+10]; using Kalman filters to fit the phase [ACt13]; using

SVD of the matrix of raw signals [Nar92] or the Fourier complex matrix [HA07]; or applying unevenly sampled spectral methods like the Lomb's periodogram [Zeg+06]. In addition, phase coherence between different signals can be used to enhance the mode trace [Pok+10; Tin+22].

It is important to mention that there is a growing research effort on other applications of ML to fusion use cases. To give some examples, experiments with real control of plasma equilibrium by reinforcement learning [Deg+22] have been carried out, or very recent work avoiding tearing instability [Seo+24] has reached real experiments. One of the earliest applications of ML to Fusion was disruption prediction [Her+96; Pau+02; Mur+08], and it has undergone significant development [Mur+09; OMM13; Veg+13; Veg+15; Veg+16; Mor+16; RG18; Rea+18; KST19; Veg+22; Zhe+23; She+23; Mur+24]. Therefore, disruption prediction literature is already quite extensive, an historical review on the topic [Rea23] can be consulted by the interested reader. Although disruption prediction sometimes observe MHD features of the plasma [FMR22], these works will not be further discussed, as this thesis work focuses on the identification of Alfvénic type of instabilities.

Research using full unsupervised learning approaches is less studied. This might be caused by its intrinsic difficulties: the labels and patterns detected are often difficult to interpret, and when they do, the result might appear evident. To provide some references: in [Veg+09] similarity search is employed to query databases for close waveforms and images; SVD and cluster trees are used in [PB09; Yam+10] to find different MHD modes and Alfvénic activity [Bla+] in multichannel Mirnov coil data with some configuration parameters of the H-1 heliac stellarator, work from [HBP14] uses SVD and STFT feature extraction, complementing each other to label MHD modes in H-1 NF spectrograms. And finally, [Veg+20] uses unsupervised clustering of multi-signal patterns and develope supervised classifiers of multisignal plasma relevant events.

In contrast, the research on the application of classic CV and wavelet for image processing in Fusion applications is limited (outside Camera diagnostics). To our knowledge, only previous work [Kap+22a] from DIIID uses computer vision algorithms, morphological filters specifically, to prepare data for using ML. For automatic identification of MHD modes: Ridge detection and multi-wavelet convolutional neural networks have been explored in two master theses [Col23; Oro23] respectively. This gap in knowledge regarding the use of CV to extract information from plasma spectrograms motivated the research presented in this thesis work.

3. Mode identification using sparse regression and Gabor's random dictionaries

> *"the search for the Holy Grail of an ideal sparse transform adapted to all signals is a hopeless quest."*
>
> — Stéphane Mallat [Mal09]. page 1.

Summary

3.1. Introduction

Given plasma diagnostic data in a time-series format, such as a Mirnov coil signal, our goal is to identify the signal's modal components. These components correspond to distinct types of finite frequency oscillations that can be observed in the signal's spectrum. To illustrate this problem, we begin with a musical analogy: given a sound sample of a musical instrument, our goal is to identify different musical notes from different instruments. In Western music, there are 7 musical notes, and each one is composed of several harmonics, which are groups of specific time-frequency oscillations associated with the stationary waves produced in the instruments. One possible data-driven approach would be to use a spectrogram to identify which frequencies are present in the sound. Since the harmonic frequencies of different instruments are well known, we could compare the information obtained in the spectrogram with our prior knowledge of musical sounds, finally obtaining a transcription of the sound. Once the transcription is done, if we repeat the process with many songs, we might be able to have enough data to train a computer to do automatic score transcriptions.

A musical note has many harmonics associated with a fundamental frequency: their frequencies, higher or lower than the fundamental, determine the tone of the instrument. Similarly, plasma modes can have many harmonics. Like a musical instrument, the waves in the plasma have a three-dimensional distribution, which we also want to study.

Musical instruments are usually tuned before recording, and their production is highly standardized making differences between different instruments very subtle. This makes the frequency patterns almost equivalent between many situations, and subtle variability is in the hands of the musicians and their will. This advantage allows us to recognize recorded songs with our phones very easily. However, the situations inside a fusion device are different: physical conditions change rapidly and frequencies do not remain stationary. The musical analogous would be a guitar changing in length or string tensions while playing. Although a shot can be repeated and instabilities appear again with similar signal patterns (time-frequencies in the case of the spectrograms), the goal of experiments is often changing physical conditions to understand the plasma behavior. The changing experimental conditions difficult to establish a fixed library of prior knowledge about plasma patterns.

In the spirit of the analogy, we can find an inspiration to approach the problem of plasma modes identification: the sparse properties of musical language used for transcribing music are extraordinary. Complex sounds between 20Hz and 20 kHz are encoded, employing only seven musical notes and different indications for rhythm. A continuous range of frequency is transformed in a very sparse code of musical notes. This is possible because fundamental frequencies are not dense in the audible range of frequencies 20Hz and 20 kHz. Therefore, a sparse codification like the musical language, contains much meaningful information. Given the fact that instability frequencies are sparse in the frequency range between 0 and 500 kHz, we use frequency sparsity as the prior knowledge of our plasma modes identification. In this chapter, we aim to find groups of sparse frequencies that can represent waveforms extracted from plasma diagnostics.

3.2. Sparse coding with a musical example

In order to demonstrate how the algorithm Sparse Random Mode Decomposition (SRMD) [RST22] works, we use a well-known piano score to extract the harmonic content of the signal using SRMD. The piano piece is a recording of "The Entertainer" by Scott Joplin (1902) which was registered on a piano roll and recorded in audio years later. In Fig. 3.1, a copy of the original score is shown. The raw audio waveform can be seen in Fig. 3.4 along a Fourier Spectrogram of the signal. In addition, the musical note frequencies can be calculated using $f(n) = 440(2^{(1/12)})^{(n-49)}$ Hz for each n key from 1 to 88 [BK22] (page 80). The A (La) note of the central octave is the key 49 and the fundamental frequency oscillates at 440 Hz. The notes around the central octave are represented in Fig. 3.2, over the audio spectrogram.

The problem of musical transcription can be understood as a linear regression

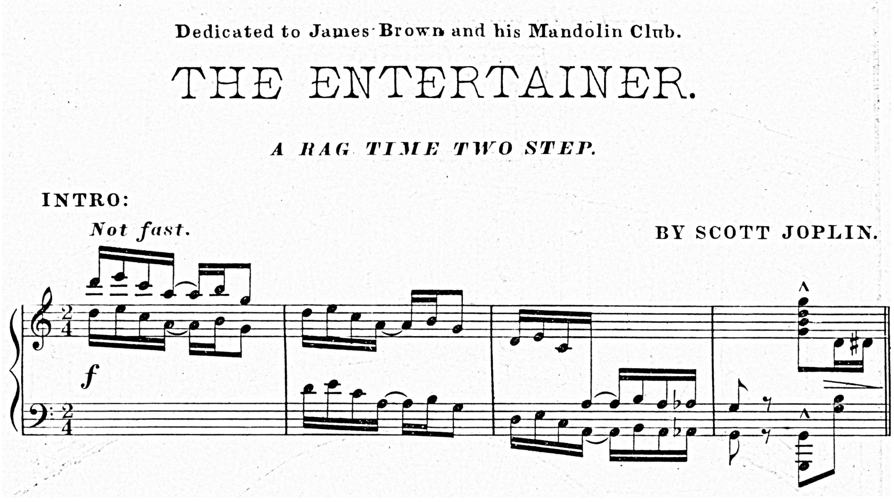


Figure 3.1.: Piano score: First 5 bars of The Entertainer by Scott Joplin (1902) from NYPL's open archive.

problem. Given the signal $\boldsymbol{s}$, a matrix of musical waveforms $\mathbf{D}$, find the weights $\boldsymbol{c}$, such that $\boldsymbol{s} = \mathbf{D} \cdot \boldsymbol{c}$. The matrix $\mathbf{D}$ is known as the dictionary matrix. In the case of SRMD, the columns of $\mathbf{D}$ are composed of real Gabor atoms, with random frequencies, phases, and time locations (hence the word *Random* in SRMD).

The goal is to find a representation that keeps the modes corresponding to the piano harmonics but discards undesired features like noise. In order to construct a dictionary matrix, we assume that the audio signal registered by the microphone can be reproduced as a linear combination of Gabor atoms: $s_i = W(t-\tau_i)\cos(2\pi f_i t + \phi_i)$, where $W(t-\tau_i) = \exp\left(-\frac{(t-\tau_i)^2}{2\sigma^2}\right)$ is a Gaussian window function of bandwidth σ. The term $\cos\left(2\pi f_i t + \phi_i\right)$ reconstructs the oscillatory structure and the window function modulates de amplitude over a specific time τ. A representation of different Gabor atoms can be seen in Fig. 3.3

The audio signal can be reconstructed using the linear combination of all atoms, given that we know the coefficients c_i, that represent each atom amplitude.

$$\boldsymbol{s} \approx \sum_i c_i \cdot s_i = \sum_i c_i \cdot \exp\left(-\frac{(t-\tau_i)^2}{2\sigma^2}\right)\cos(2\pi f_i t + \phi_i). \tag{3.1}$$

To calculate the coefficients c_i it is possible to use a linear regression fit, where the common practice is to minimize the squared error $||\boldsymbol{s} - \mathbf{D} \cdot \boldsymbol{c}||^2$. However, the values of coefficient c_i will also represent the noise in the audio signal.

Signal representations can be also obtained by a projection over a basis. The Gabor transform of a signal $\boldsymbol{s}(t)$ is the projection of this signal onto the Gabor dictionary $\mathcal{D}_{\tau,f} = \left\{W(t-\tau)\exp\left\{i2\pi f t\right\}\right\}_{(\tau,f)\in\mathbb{R}^2}$. Such transform inspired the actual windowed Fourier transform used to plot the spectrograms, the STFT explained in chapter 2. It allows us to calculate time-frequency representations (spectrograms) efficiently. We use it in Fig. 3.2 and Fig. 3.4 as a reference representation to evaluate the result of

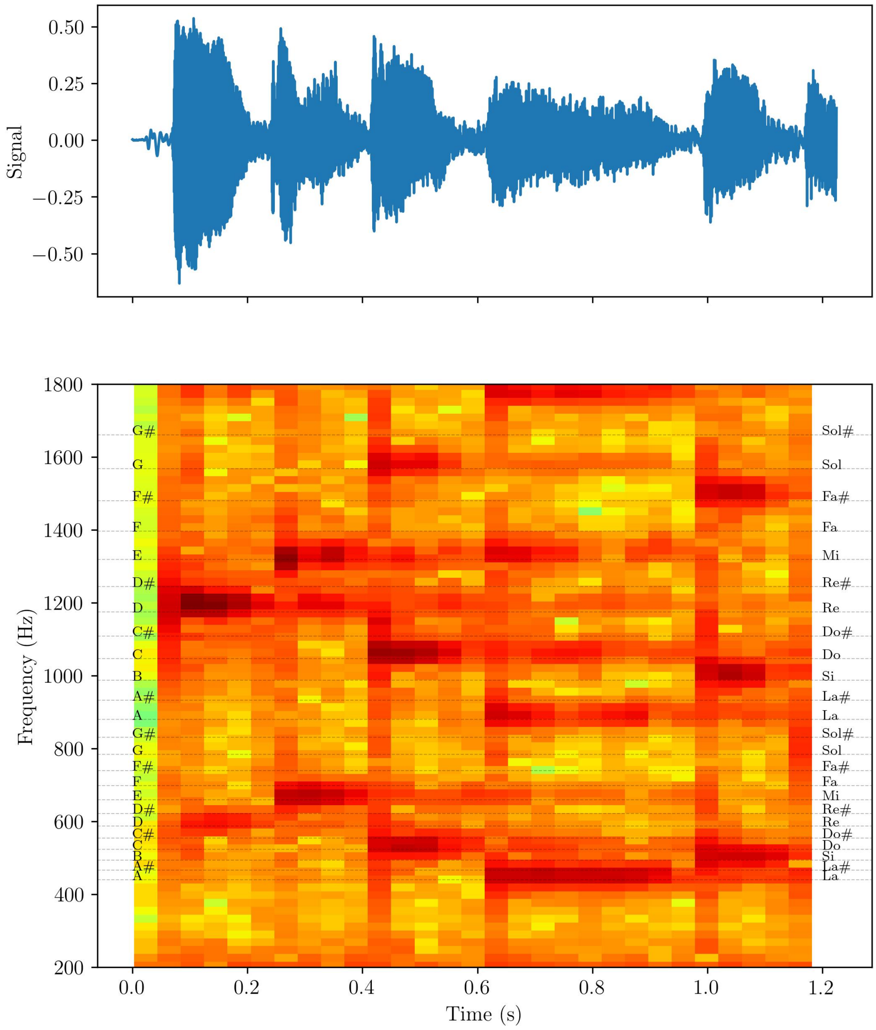


Figure 3.2.: Raw audio signal of The Entertainer's first bar and its spectrogram.

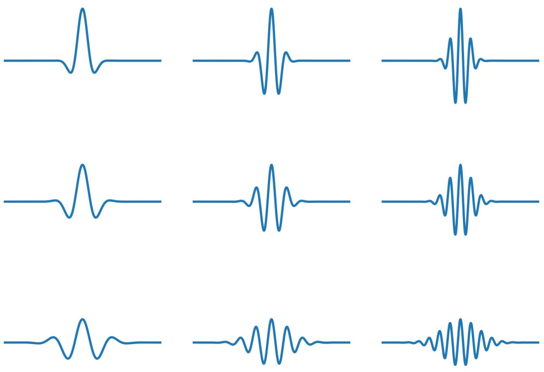


Figure 3.3.: Gabor atoms of different frequencies and bandwidth.

SRMD. However, in the case of the Fourier spectrogram, the signal and noise are not directly separated.

To extract the modal amplitudes without noise interference, sparsity of $\boldsymbol{c}$ has to be promoted in the optimization algorithm. A vector is sparse if the majority of components are 0. If we sum the absolute values of all vector components, we obtain the ℓ_1 norm of a vector. Therefore, we can express sparsity mathematically using : $||\boldsymbol{c}||_1 = \sum_i |c_i| \approx 0$.

We promote the sparsity of the code vector by adding a regularization term to the square error. Therefore, we obtain the code $\boldsymbol{c}$ by solving the following optimization problem

$$\begin{pmatrix}\text{Encoder}\\ \text{equation}\end{pmatrix} \quad \boldsymbol{c} = \arg\min_{\boldsymbol{c}} \{||\boldsymbol{c}||_1\} \text{ such that } ||\mathbf{D}\cdot\boldsymbol{c} - \mathbf{s}||_2^2 < r||\boldsymbol{s}||_2, \tag{3.2}$$

which we have called (encoder equation). Once the code is known, the decoder equation is a straightforward linear combination:

$$\begin{pmatrix}\text{Decoder}\\ \text{equation}\end{pmatrix} \quad \boldsymbol{s} = \mathbf{D}\cdot\boldsymbol{c} \tag{3.3}$$

In practice, the optimization algorithm does not always set to zero no-sparse components of $\boldsymbol{c}$. The values of $\boldsymbol{c}$ are divided into 100 quantiles and the bottom quantiles below a given threshold value are discarded. This collection of atoms composes a signal, as words in a dictionary represent sentences. The terms *dictionary* and *atoms* were proposed by Mallat and Zhang [MZ93] in 1993. If the matrix product $\mathbf{D}\cdot\boldsymbol{c}$ reconstructs the signal $\mathbf{D}\cdot\boldsymbol{c} = \boldsymbol{s}$, the role of the vector $\boldsymbol{c}$ is both the selection of columns in $\mathbf{D}$, and setting the amplitude of each Gabor atom. For this reason, the vector of amplitudes $\boldsymbol{c}$ is also known as *code*. Following the Mallat's notation introduced in chapter 2, our dictionary can be defined as

$$\mathscr{D}_{\tau,f,\phi} = \left\{W(t-\tau)\cos(2\pi f t + \phi)\right\}_{(\tau,f,\phi)\in\mathbb{R}^3}, \tag{3.4}$$

where τ, f and ϕ are randomly chosen from uniform distributions of time shifts, frequency and phase shifts.

Once the signal is encoded we can create groups of atoms close in time and frequency. For that, SRMD uses the DBSCAN algorithm (density-based spatial clustering of applications with noise). The result of applying SRMD can be seen in Fig. 3.4: the scatter plot over the spectrogram represents the set of points $\{\tau_i, f_i\}$, each dot represents the time-frequency position of a Gabor's atom encoded by SRMD. The result has been cropped so we can identify easily which cluster corresponds to each musical note. The raw signal above the spectrogram reflects the oscillations of the different clusters with the same color palette. The reader able to read the musical score in Fig. 3.1 would identify the melody notes of the first bar (B,E,C,A,D,G) with the cluster centers. In addition, it can be seen that some modal groups are individual notes merged together because of the result of the cluster analysis.

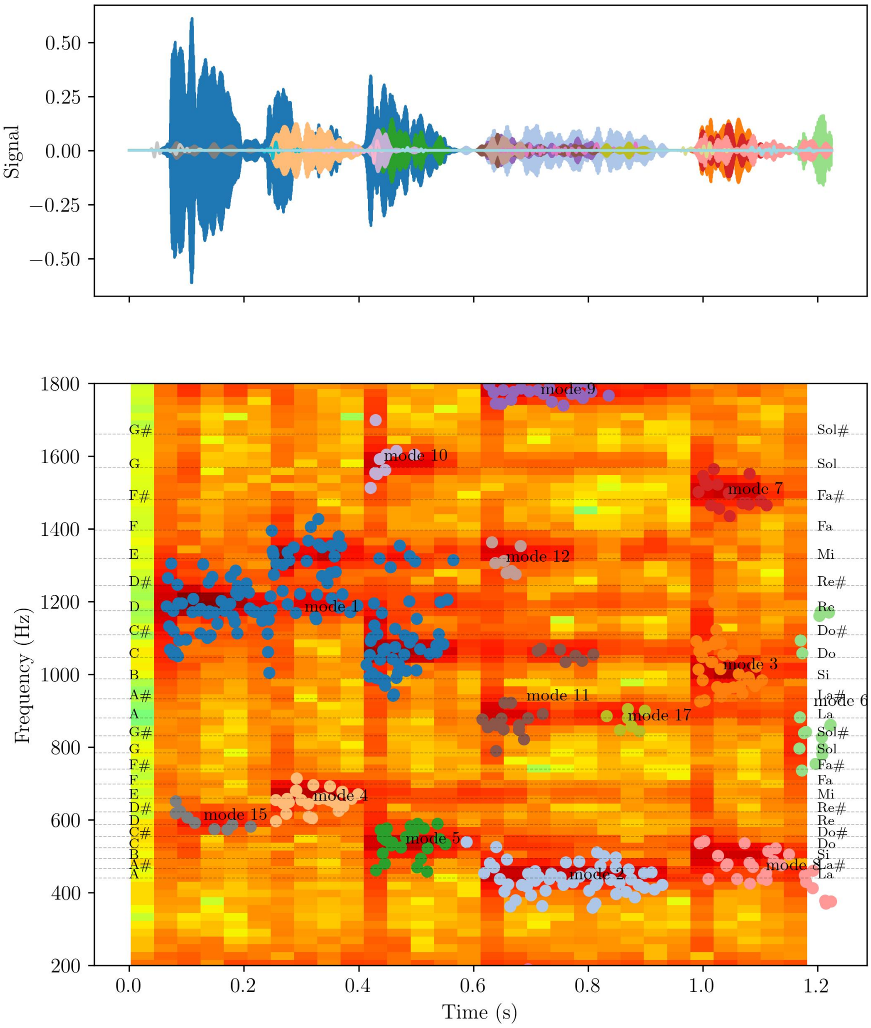


Figure 3.4.: Mode decomposition obtained by SRMD: raw signal and time-frequency location over spectrogram

To summarize SRMD (algorithm 1), the basic steps are 3: create a dictionary of random atoms (1), run an optimization algorithm to determine the sparse collections of atoms that can generate the signal, in other words, obtain the sparse code (2), and run a cluster analysis algorithm to identify a mode out of a group of close atoms in the time-frequency space. Once the analysis is completed the result can be seen in Fig. 3.4.

Before applying SRMD to magnetic signals we need to discuss its limitations: Mirnov coil signals are sampled at 1 or 2 MHz, that is 1 or 2 million samples per second of signal. Without taking into account the coil startup and plasma formation time, TJ-II shots duration is approximately 1 second. In addition, the Alfvénic activity that we want to analyze has a very high frequency between 100 and 400 kHz. This discards the possibility of downsampling the signal: because of the Nyquist limit, which is approximately half the sampling frequency, we need all samples registered. Therefore

Algorithm 1: SRMD algorithm from [RST22]

```
Input: signal $\boldsymbol{s}$
Parameters: number of atoms N, maximum frequency $f_{max}$, window size $\Delta$,
            noise noise level $r$, quantile threshold $thr$
Output: $\boldsymbol{c}$, labels, $\boldsymbol{\tau}$, $\boldsymbol{f}$, $\boldsymbol{\phi}$
Steps:
1 Construct dictionary matrix drawing random $\tau_i$, $f_i$, $\phi_i$ from uniform
    distributions
      $\mathbf{D} = [\boldsymbol{s}_i(t)] = \left[\exp\left(-\frac{(t-\tau_i)^2}{2\sigma^2}\right)\cdot\cos\left(2\pi f_i t + \phi_i\right)\right]$
2 Solve the encoder equation
      $\begin{pmatrix}\text{Encoder}\\ \text{equation}\end{pmatrix}$ $\boldsymbol{c} = \arg\min_{\boldsymbol{c}}\{||\boldsymbol{c}||_1\}$ s.t. $||\mathbf{D}\cdot\boldsymbol{c}-\mathbf{s}||_2^2 < r||\boldsymbol{s}||_2$
  Discard lower quantile of $\boldsymbol{c}$ values: $c_i = 0$ if $Q(c_i) < thr$.
  getting $\boldsymbol{c}$ $\boldsymbol{\tau}$, $\boldsymbol{f}$, $\boldsymbol{\phi}$
3 Cluster $\boldsymbol{\tau}$-$\boldsymbol{f}$ space using DBSCAN, labeling the modes
  Return $\boldsymbol{c}$, labels, $\boldsymbol{\tau}$, $\boldsymbol{f}$, $\boldsymbol{\phi}$
```

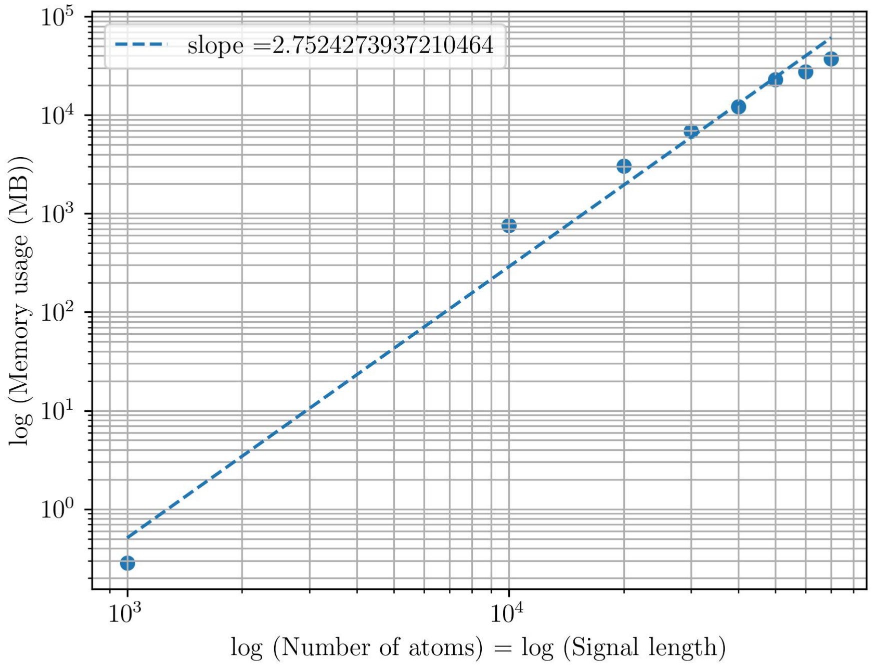


Figure 3.5.: SRMD memory usage against signal length and number of atoms (in log scale)

SRMD should be able to cope with 1 million samples. However, the allocation of matrix **D** demands high amounts of memory as the number of atoms has to be very high to correctly describe the signal. Moreover, the optimization algorithm also is computationally expensive, requiring time to perform all computations needed.

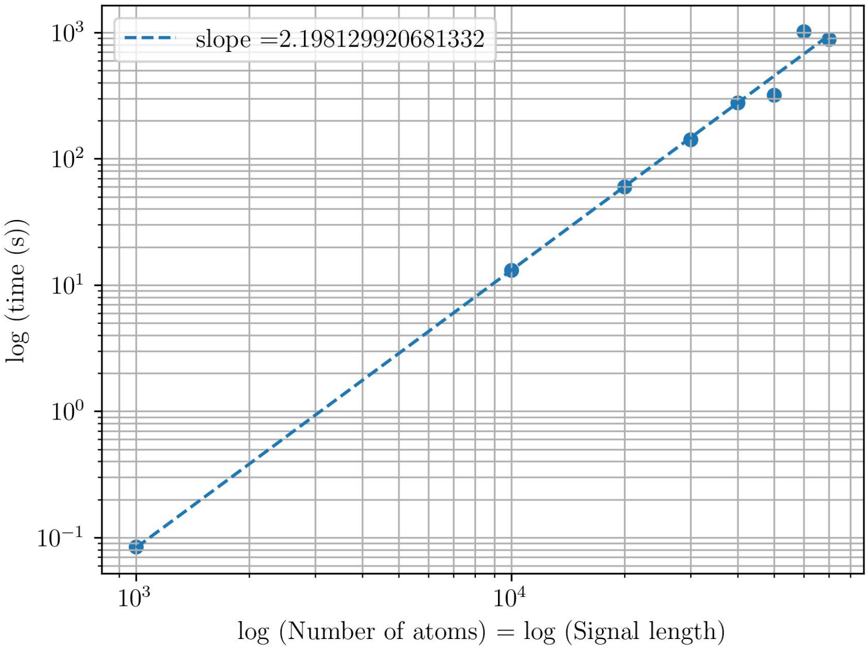


Figure 3.6.: SRMD execution time against signal length and number of atoms (in log scale)

To demonstrate the computing resources needed by SRMD and its scaling in memory and time: we run SRMD on a signal increasing the number of samples. The dictionary size increase with the number of samples. We set the number of atoms equal to the number of samples so dictionary matrix is a square matrix of size: 1000, 10000, 20000, 30000, 40000, 50000, 60000, and 70000 samples/atoms respectively (the size of the matrix gives the number of atoms and samples, hence the samples/atoms notation). In Fig. 3.5 it can be seen that the memory required increases almost with the cube of the signal size. On the other side, in Fig. 3.6, the execution time grows approximately as the square of the signal length, this make sense as the dictionary matrix **D** is chosen to be a square matrix for this experiment (same number of atoms as the signal length, in practice there will be more atoms than samples). The experiment ended with 80000 samples/atoms as the program was not able to allocate 47.7 GiB on memory. Therefore, 80k samples is a figure far from the 1 million samples per second needed. The situation would be even more adverse in the case of other fusion machines. For instance, JET pulses are approximately 10 seconds long, and future ITER's hybrid scenarios aim for more than 1000 seconds. Because of these reasons, it is necessary to adapt SRMD to work with high-resolution time series which are very common in plasma diagnostics.

"There is not a universally ideal representation for all signals" as Mallat said in its reference book [Mal09], which is the reason why a good data analyst compares several representations. Also, the convenience and computational cost vary widely between different methods of representation. For instance, the ℓ_0 norm is equivalent to counting the number of components in a vector different than 0. Using ℓ_0 in regularization to obtain a sparse code, that makes the optimization problem an $\mathscr{NP}$ hard problem [Til15]. That is the reason why the ℓ_1 convex relaxation is used instead. However, the effort to obtain a sparse representation of data is not justified only by the need for separating signal from noise: sparsity is an important prior in many domains, and the reasons are deep. If we accept Ocham's razor principle, in the situation of having two different competing representations of a signal, the one with fewer components, i.e. the simpler one, should be the correct one. This principle translates mathematically in choosing the vector of weights $\boldsymbol{c}$ with more null components, in other words, the sparsest one.

The importance of sparsity in physics, statistics, and nature can not be understated. In physics, generally, the coordinate system that facilitates calculations (*generalized coordinates*) and formulation of physical laws has the minimum degrees of freedom. For instance in biology, blocks of neurons called simple cells, located in the visual cortex V1 and organized in columns, specialize in detecting edges with a determined orientation. This discovery allowed Hubel and Wiesel to win the Nobel Prize of Medicine in 1981. Later, the way simple cells in visual cortex V1 operate was modeled by a sparse coding of images via an overcomplete Gabor's atoms dictionary [OF97b]. Using a filter bank of Gabor atoms is a bio-inspired method of image feature extraction in the machine learning literature.

Dimensional reduction also involves sparse assumptions, and its need is justified because the data required to correctly sample learning spaces increase exponentially when adding dimensions. This fact is known as *dimensionality curse* and it makes difficult the statistical learning of algorithms. Therefore sparse representations can help to learn patterns in high dimensional spaces. Finally, the parsimonious principle is used in statistics to improve model generalization (it avoids overfitting), and model interpretability. Because of this reasons, using sparsity is a key concept for data-driven physics and machine learning, and it is gaining attention in plasma physics [Kap+23], with applications as different as calculating mode numbers [Kle+08] or optimizing stellarators [Kap+22b].

3.3. TJ-II stellarator signals

Raw signal plots from Mirnov coils are indistinguishable (at least at first sight) from an audio waveform: like the first bar of the Entertainer 3.2 and the TJ-II shot 38339 in Fig.3.8. The properties of the signal are revealed by close inspection of the spectrogram Fig.3.8, or by the use of algorithms like SRMD.

The variation in the magnetic field is registered in TJ-II's array of Mirnov coils. These coils record the voltage induced by the plasma magnetic fluctuations. Particularly, the signals are extracted from the MIR5C coil, belonging to the straight line array. The spectrograms generated using this coil are exceptionally clear because it is positioned inside the vessel very close to the plasma's last-closed surface. In Fig.3.8 we illustrate the spectrogram of a typical shot observed in TJ-II measured by the MIR5C.

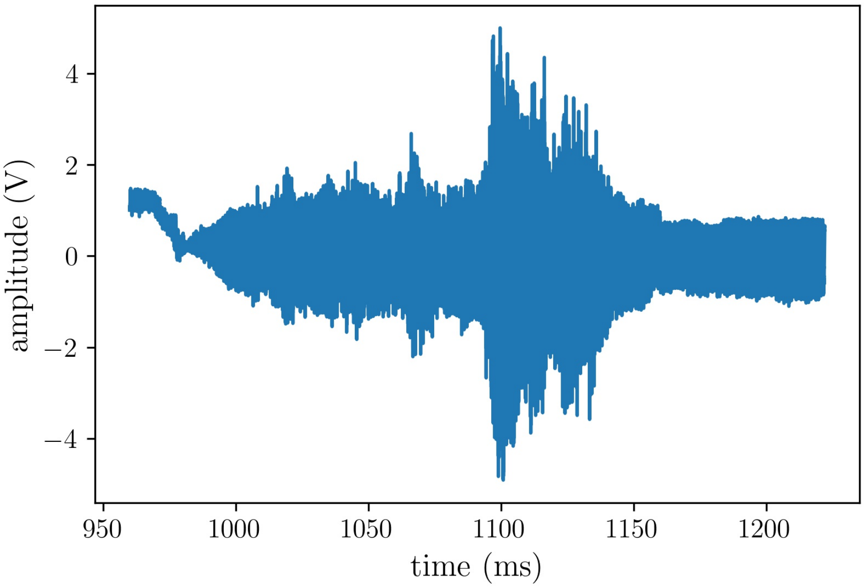


Figure 3.7.: Shot 38399: MIR5C coil raw signal.

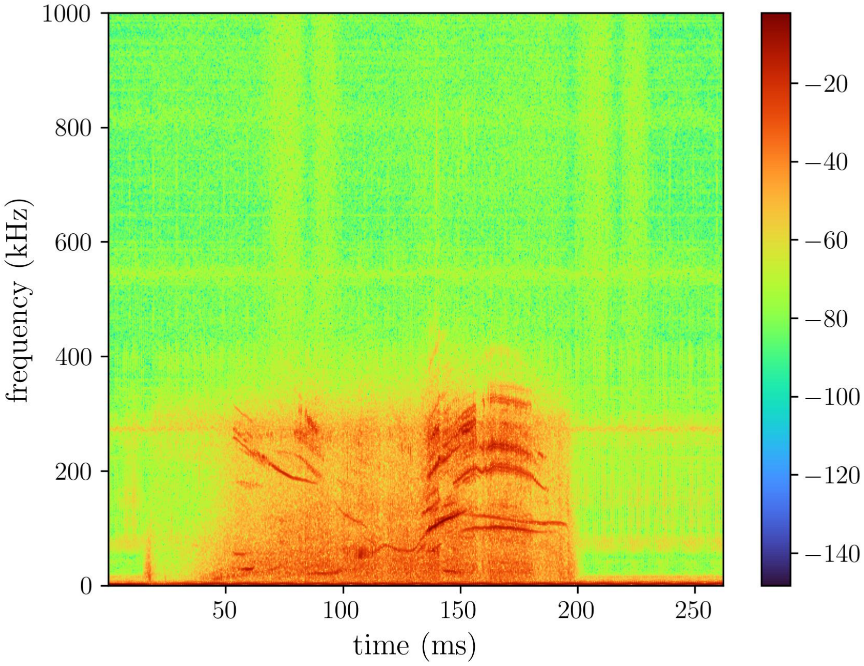


Figure 3.8.: Spectrogram of shot 38399, using MIR5C coil signal sampled at 2MHz, having complex mode activity between 0 and 400 kHz.

Other signals used in this thesis work have been: the electron density, the plasma current, and the magnetic energy (in Fig. 3.9). They are obtained from magnetic sensors (microwave interferometer, Rogowski coil, and the diamagnetic loop respectively.

The microwave interferometer is responsible for measuring the plasma density along a specific line of sight. This line passes by the plasma close to the magnetic axis with an inclination of 18.7° with the vertical axis. The correlations of this measurement with the frequency of the modes are key to identifying Alfvénic activity.
The plasma current is measured in kA using a Rogowski coil, This current is the contribution of 4 currents: one induced by NBI heating system, the current driven by oblique ECRH injection, and the bootstrap current or ripple currents in coils close enough to the plasma. Due to the proximity of the TJ-II's main field conductors to the plasma, an unwanted Ohmic contribution due to the ripple in the currents flowing through the conductors induces current fluctuations in the plasma [Mul+23].

A diamagnetic loop measures the energy stored in the plasma. It is important to mention that the diamagnetic loop can be affected by the plasma current, part of the current fluctuations will be present in the energy's time traces.

The sampling frequencies for the different measurements varies as follows: the Mirnov coil is sampled most frequently, between 1 and 2 MHz, whereas the magnetic energy is collected at the least frequency (≈ 10 kHz). Therefore, linear interpolation is needed to calculate relationships between variables.

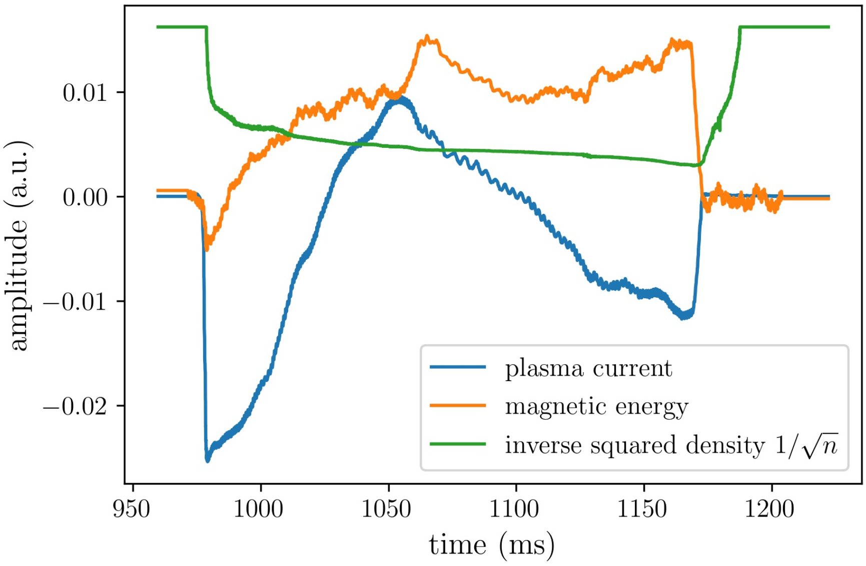


Figure 3.9.: TJ-II signals in a.u.: plasma current, magnetic energy and $1/\sqrt{n}$

The stellarator has two heating systems: two NBI (Neutral Bean Injection) units, and two ECRH (Electron Cyclotron Resonant Heating) beams. Fig.3.8 shows the spectrogram of magnetic fluctuations registered by the MIR5C pickup coil in a NBI heated plasma, the ECRH was not activated in this shot. When both NBI's are consecutively engaged, a complex mode activity is formed between 0 and 400 kHz along broadband frequency noise. It is important to mention, that not all modes that can be observed are necessarily Alfvénic, identifying the instability mechanisms might require additional study with plasma simulations and theoretical analysis. In addition, Alfvénic activity can be generated in plasma regions out of the interferometer's line of sight, with different density fluctuations. This can cause a misidentification of Alfvénic modes.

The Alfvénic mode activity in TJ-II has been widely observed with frequencies in the range of 50kHz to 400 kHz, some studies are [Jim+11; Mel+12; Mel+14; Mel+18; Eli+21; VSG17; SOL15; Cap+21; Ghi+24]. The Alfvénic activity can be observed along a rich variety of phenomena, which makes its identification difficult.

In the case of most plasma signals, the data is collected as a time series. The approach of this work for clustering time series is based on the observation that finding modes in a signal is equivalent to creating a new signal representation in which only the modes are presented. It is important to note that, the mentioned previous work employing unsupervised learning to mine fusion databases does not specifically address the problem of labeling Alfvénic activity by using sparse coding on raw signals.

3.4. The Elastic Random Mode Decomposition algorithm (ERMD)

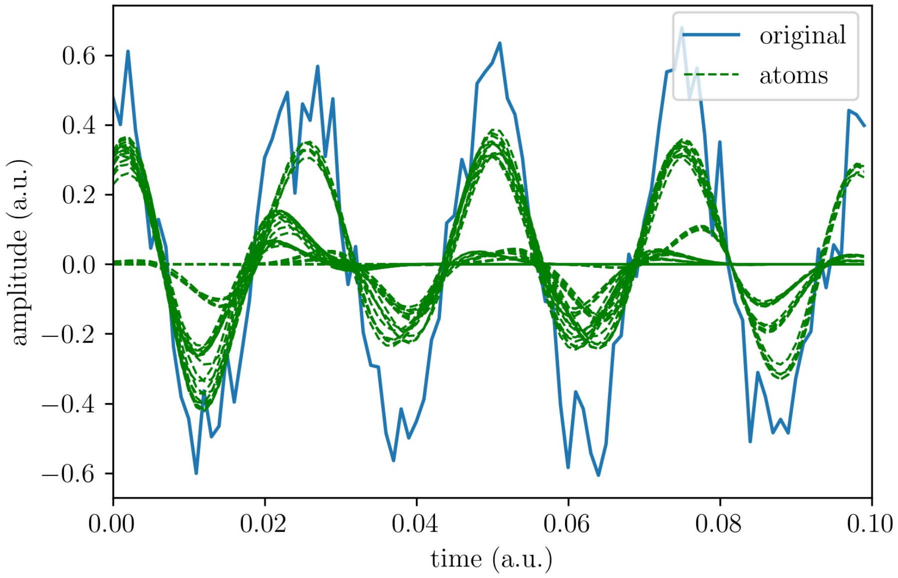


Figure 3.10.: Examples of atoms that minimize reconstruction error, corresponding code values c_i will be significantly greater than zero.

To construct the dictionary matrix **D**, we can use Gabor's atoms as in Eq.(3.1), assuming a uniform distribution of parameters (τ_i, f_i, ϕ_i) time, frequency, and phase.

By taking random samples of these distributions we can assign an atom to each column of the matrix $\mathbf{D}_{:,i}$. We use the notation $\mathbf{D}_{:,i}$ to refer to the i-th column of the matrix. Then, each atom ($\mathbf{D}_{:,i}$ column) constitutes a time series's feature. It should be noted that the number of atoms in the basis is not predetermined, and the variance of the parameters (τ_i, f_i, ϕ_i) has to be enough to represent the signal accurately.
This method for constructing the dictionary is derived from an algorithm called Sparse Random Mode Decomposition (SRMD) [RST22], explained in the previous section. The complete algorithm is available as open-source code. In Figs. 3.10-3.11-3.12 an application of the algorithm on a synthetic signal is shown. The atoms in Fig.3.10 have a similar frequency and phase to the original signal, the code values will adjust to set a proper amplitude for each atom to represent the sign. On the other side, atoms in Fig.3.11 cannot represent the local features of the time signal, SRMD will adjust their amplitudes to be close to zero. In Fig.3.12 we can see that very few atoms can reconstruct the right frequency, and increasing the number of atoms improves the capability of representation of the signal's features.

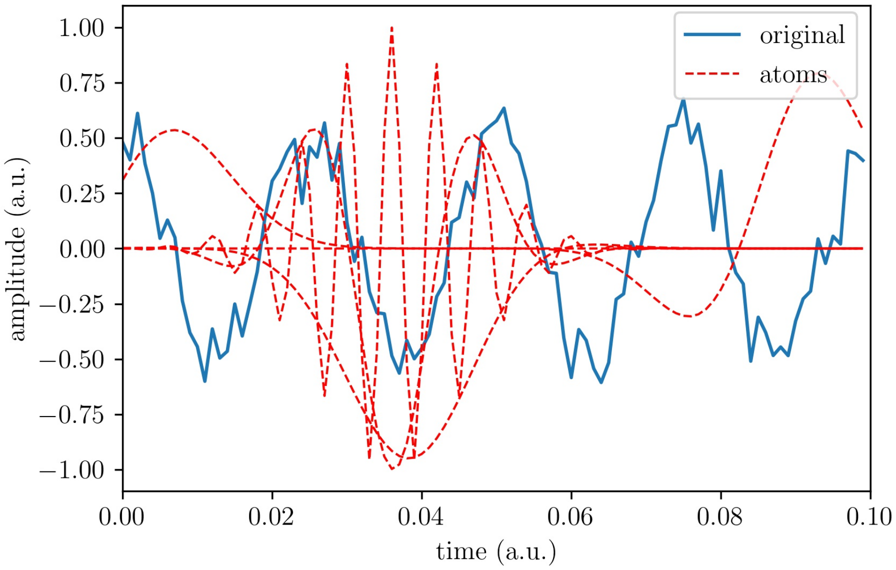


Figure 3.11.: Examples of atoms that do not minimize reconstruction error, respective code values c_i need to be close to zero.

3.4.1. A tailored optimization algorithm

Two needs lead us to introduce a tailored optimization algorithm: to improve the computation so that it can be used with high time-resolution plasma diagnostics, and to improve the performance of coding in large dictionaries.

In this subsection, we formulate the signal representation by starting with the most general constrained minimization problem and subsequently introducing step-by-step different alternative approaches to achieve sparsity in the codification. A detailed

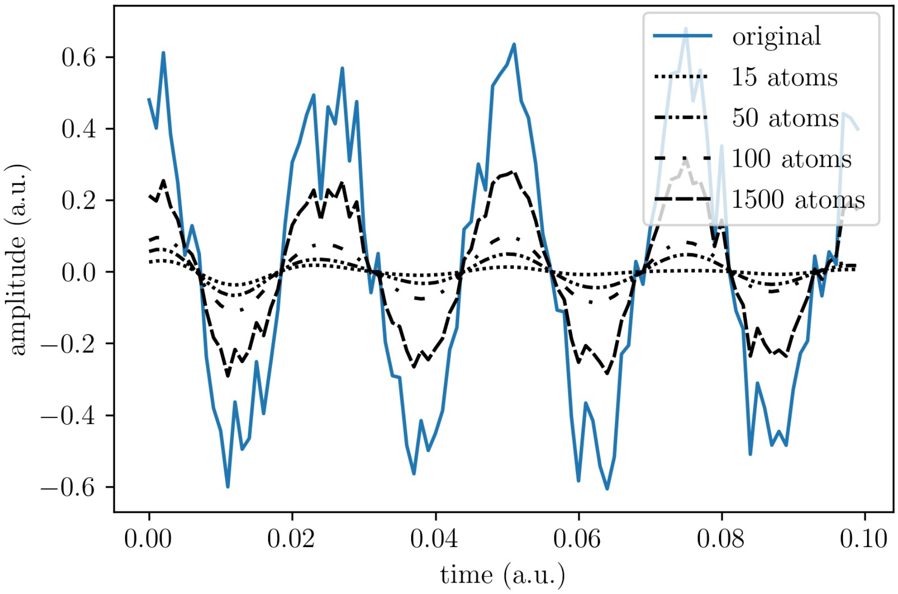


Figure 3.12.: Examples of reconstruction improvement by adding atoms to the code.

discussion can be found in Elad's book [Ela10].

It is assumed that the dictionary matrix has more columns than rows, so the system of equations $\mathbf{D}\cdot\boldsymbol{c}=\boldsymbol{s}$ is undetermined as there are more unknowns than equations. If the system is consistent, in the sense that $\boldsymbol{s}$ can be linearly generated by columns of $\mathbf{D}$, there will be infinite solutions. Therefore, to cope with infinite solutions we aim to find the solution $\mathbf{c}$ with minimum norm. The problem can be formulated as a constrained optimization problem $(P_{\mathscr{L}})$,

$$(P_{\mathscr{L}}) \equiv \min_{\boldsymbol{c}} \quad \mathscr{L}(\boldsymbol{c}) \quad \text{subject to} \quad \mathbf{s}=\mathbf{D}\cdot\boldsymbol{c}, \tag{3.5}$$

where the loss function $\mathscr{L}(\boldsymbol{c})$ controls the solutions we can obtain. Choosing $\mathscr{L}(\boldsymbol{c})$ to be the Euclidean squared norm $||c||_2^2$ is the common choice,

$$(P_2^2) \equiv \min_{\boldsymbol{c}} \quad ||\boldsymbol{c}||_2^2 \quad \text{subject to} \quad \mathbf{s}=\mathbf{D}\cdot\boldsymbol{c} \tag{3.6}$$

as it can be shown [Ela10] (page 4), the problem P_2^2 has a unique solution because the objective functional is convex and the solution can be obtained [Ela10] using the Moore-Penrose pseudoinverse matrix $\mathbf{X}^{+}$.

We can formulate an *unconstrained* version of the problem $(P_2^{2'})$ which has equivalent results, and it is stated as the minimization of the signal's mean squared reconstruction error (MSE), so choosing the cost function as $\mathscr{L}(\boldsymbol{c}) \equiv ||\mathbf{D}\cdot\boldsymbol{c}-\mathbf{s}||_2^2$ results in

$$(P_2^{2'}) \equiv \quad \min_{\boldsymbol{c}} \quad ||\mathbf{D}\cdot\boldsymbol{c}-\mathbf{s}||_2^2. \tag{3.7}$$

The unconstrained formulation can be more familiar to the reader because it is by

definition the ordinary least squared (OLS) estimator for linear regression. Remarkably, the unconstrained version does not enforce obtaining a solution of $\mathbf{s} = \mathbf{D} \cdot \boldsymbol{c}$, nor does it assume any property of the components of the solution vector $\boldsymbol{c}$. Above all, we can get a result that is as good as the optimization algorithm's performance.
However, the solution of the OLS estimator is not adequate for creating a basis, because generally all $\boldsymbol{c}$'s components $[c_1, c_2, ..., c_N]$ can be different than zero, and the explanatory power of the code is thereby lost. We need to ensure that as many code components $[c_1, c_2, ..., c_N]$ as possible vanish. This can be achieved using l_0 norm instead of l_2 in the constrained problem P_2^2. The norm l_0 is defined as the number of vector components different than zero, so minimizing l_0 and the MSE simultaneously can lead to a suitable basis. Hence, we can define the problem P_0 as

$$(P_0) \equiv \min_{\boldsymbol{c}} ||\boldsymbol{c}||_0 \quad \text{subject to} \quad \boldsymbol{s} = \mathbf{D} \cdot \boldsymbol{c}. \tag{3.8}$$

This problem is significantly difficult to solve since (P_0) is an $\mathscr{NP}$ hard problem [Til15], so trying to solve it directly is not a common practice. Nevertheless, we can relax the constraints if we use the ℓ_1 norm which is defined as the sum of the vector's components. Consequently, minimization of the ℓ_1 norm will shrink some vector components close to zero. This *optimization strategy* [CDS01] is called *basis pursuit* (BP) , it will be noted as P_1

$$(P_1) \equiv \min_{\boldsymbol{c}} ||\boldsymbol{c}||_1 \quad \text{subject to} \quad \boldsymbol{s} = \mathbf{D} \cdot \boldsymbol{c}. \tag{3.9}$$

Additionally, we need to take into account that most signals will not be pure and they will have noise. That can be modeled by adding an σ error parameter to the functional in the following way

$$(BPDN) \equiv \min_{\boldsymbol{c}} ||\boldsymbol{c}||_1 \quad \text{subject to} \quad ||\mathbf{D} \cdot \boldsymbol{c} - \mathbf{s}||_2^2 \leq \sigma. \tag{3.10}$$

The later version is called basis pursuit denoised (BPDN). Though it was first proposed [CDS01] in its unbound version (P_1'), it can be formulated as follows:

$$(P_1') \equiv \min_{\boldsymbol{c}} \left\{ ||\mathbf{D} \cdot \boldsymbol{c} - \mathbf{s}||_2^2 + \lambda_1 ||\boldsymbol{c}||_1 \right\}. \tag{3.11}$$

This problem is also found in machine learning known as *Lasso* regression.

Another alternative for regularization is using ℓ_2 norm (no squared). The functional takes the form

$$(P_2') \equiv \min_{\boldsymbol{c}} \left\{ ||\mathbf{D} \cdot \boldsymbol{c} - \mathbf{s}||_2^2 + \lambda_2 ||\boldsymbol{c}||_2 \right\} \tag{3.12}$$

in its unconstrained version. As in this case, the equivalent problem in machine learning is named *Ridge regression.* which is useful for avoiding shrinkage of the solution vector when data have multicollinearities [ZH05; Tib11].

The effect of ℓ_1 regularization on a group of correlated variables is the selection of one variable, discarding the rest, even if these discarded variables have explanation power or are part of the solution. If there are correlated features and a sparse solution that conserves groups of correlated variables is desired, adding ℓ_2 to the functional, like in

$$\begin{pmatrix} \textit{ELASTIC} \\ \textit{NET} \end{pmatrix} \equiv \min_{\boldsymbol{c}} \left\{ ||\mathbf{D}\cdot\boldsymbol{c}-\mathbf{s}||_2^2 + \lambda_1 ||\boldsymbol{c}||_1 + \lambda_2 ||\boldsymbol{c}||_2 \right\}, \tag{3.13}$$

is a good alternative. This problem is known in the machine learning community as elastic net regression.

About the decision of using constrained or unconstrained versions of the optimization problems, it usually resolves in favor of the unconstrained version, because more and faster algorithms are available with guaranteed performance and studied convergence. For instance, solvers [VF09; VF11] can be used to solve the basis pursuit problem with a random dictionary as with SRMD [RST22]. However, a major difficulty to be solved is the allocation of the matrix in memory. The features in matrix $\mathbf{D}$ must have at least the same number of samples as the signal. Moreover, because we have signals sampled with 1MHz with a pulse duration of 1 second and thousands of features are necessary to represent the modes accurately, the matrix size can easily exceed all available memory in the computing node (of the order of hundred terabytes).

A first valid idea to address this issue is to divide the signal into chunks and process each chunk sequentially on one CPU, or in a parallel process on several computing nodes. But this approach has one inconvenience, namely each signal chunk has a different signal-to-noise ratio, and the weights of vector $\boldsymbol{c}$ may be unevenly set along chunks. In addition, there are discontinuities in the reconstruction of the joins between chunks. The gradient accumulation proposed in this work tackles this issue.

The motivation for finding a sparse code for the dictionary is not only algebraic. Indeed, a closer look at the spectrograms with mode activity reveals that MHD mode signals are very sparse in the frequency domain. Imposing sparsity in the dictionary representation will help to encode and cluster the MHD activity. Finding a sparse code $\boldsymbol{c}$ such that $\mathbf{D}\cdot\boldsymbol{c} \approx \boldsymbol{s}$ constitutes an optimization problem. By changing the linear projection method of STFT with a non linear optimization, the flexibility to reproduce the signal is dramatically increased.

In [RST22], a Basis Pursuit De-Noised optimizer (BPDN) was used the obtain the code vector; the optimization problem is formulated as follows: $\boldsymbol{c} = \{\arg\min_{\boldsymbol{c}}\{||\boldsymbol{c}||_1\}$ subject to $||\mathbf{D}\cdot\boldsymbol{c}-\mathbf{s}||_2^2 \leq \sigma\}$. The norm $||\boldsymbol{c}||_1$ is responsible for promoting sparsity in the code $\boldsymbol{c}$ [ZH05]. However, the optimizers that solve BPDN require the allocation of the full dictionary matrix $\mathbf{D}$. This difficult the analysis of Mirnov signals for complete shots, which usually comprise several million samples. To solve this problem, we propose the use of a linear neural network to encode the atoms dictionary using the *elastic net* regularization [ZH05; Tib11] in the loss function.

We can use a single-layer neural network with linear activation function $\boldsymbol{y} = \mathbf{X}\cdot\boldsymbol{w} + \boldsymbol{b}$, and $\boldsymbol{b} = \mathbf{0}$. We train the network in such a way that the training is equivalent to solving our signal representation $\boldsymbol{s} = \mathbf{D}\cdot\boldsymbol{c}$. In other words, the neural network architecture is equivalent to the decoder equation 3.3. and training the neural network is equivalent to solving the optimization problem of the encoder equation using the elastic net regularization loss of equation 3.13.

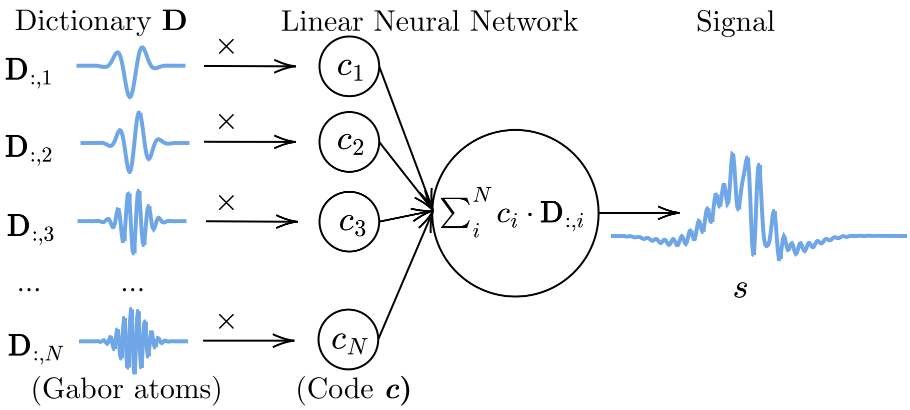


Figure 3.13.: Diagram of the proposed neural network architecture: dictionary columns $\mathbf{D}_{:,i}$ are inputs of the neural network with weights c_i.

It is worth noting the large asymmetry between coding and decoding: because the decoder equation is linear, whereas the encoder equation is nonlinear. Fig.3.13 illustrates schematically the architecture of the employed neural network. Even though the activation function is linear, nonlinearity is present in the Gabor atom, which acts as a nonlinear kernel. The deepness is replaced by the large length of the input layer. In this work, PyTorch [Pas+19] with Lightning modules have been used to implement this model. By using this neural network, we can benefit from parallel computing on GPUs.

Note that the encoder equation is different from the BPDN formulation, the loss function inside the encoder equation is known as *elastic net* [ZH05], there are two main differences though. First, the problem formulation is unconstrained (it does not need a noise parameter σ). Second, there are two regularization terms λ_1 and λ_2 associated with the modules of the code $\boldsymbol{c}$ in ℓ_1 norm $||\boldsymbol{c}||_1$ and ℓ_2 norm $||\boldsymbol{c}||_2$ respectively, while in BPDN only the norm ℓ_1 is used. The norm $||\boldsymbol{c}||_2$ is responsible for promoting the presence of correlated groups in the code $\boldsymbol{c}$ [ZH05].

In this work, the use of the elastic net is proposed based on a fundamental observation: the random atoms used in the construction of the random dictionary can be correlated. This idea is illustrated in Fig.3.14. It can be seen that atoms with the same frequency and phase but different amplitudes are highly correlated, while atoms out of phase are not correlated at all. Most importantly, atoms with connecting tails in phase have some correlation. The improvement of adding ℓ_1 regularization can be appreciated by comparing Fig.3.15 and Fig.3.16. It is important to emphasize that the atom's correlations can be exploited to promote the time continuity of the encoded modes, which is an essential physical feature of the signal.

Two major improvements in memory usage have been implemented. First, it is possible to use the atom formula given by Eq.3.1 and the random number generator seed to allocate dictionary atoms only when a signal's batch is allocated in the GPU. This is a breakthrough in memory optimization; we can refer to this method as *in-*

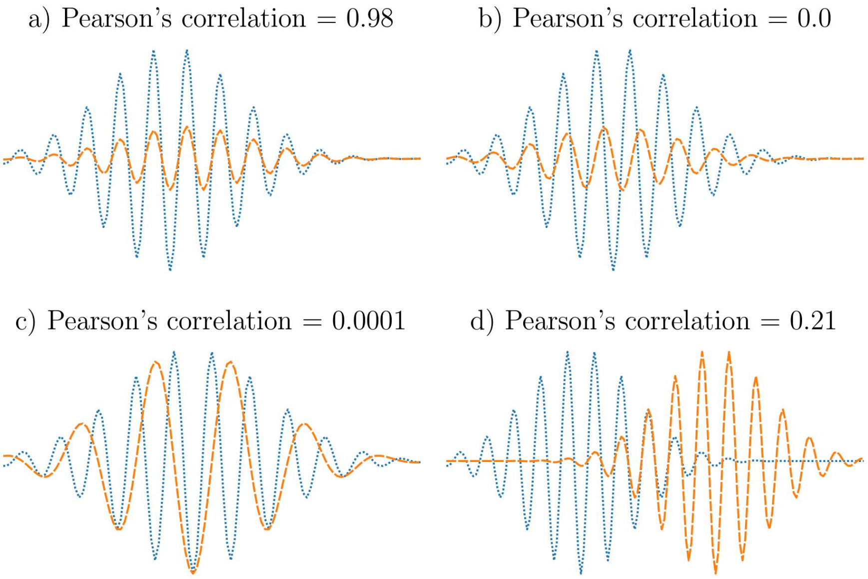


Figure 3.14.: Varying correlations between different pairs of atoms: Overlapping needed to reconstruct the signal can correlate.

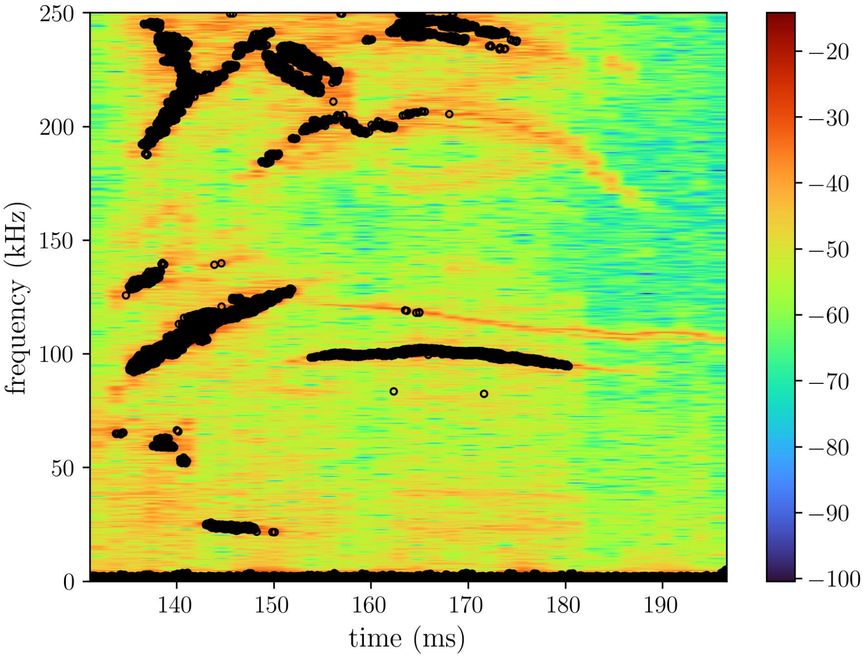


Figure 3.15.: Result of ℓ_1 encoding: setting $\lambda_1 \neq 0$ and $\lambda_2 = 0$ in Elastic Net regularization loss.

GPU dictionary construction. Second, we can adapt the gradient descent to optimize the weights of the full signal while working by batches. This technique is known as *gradient accumulation*. The gradient update is worked out for each batch until a given

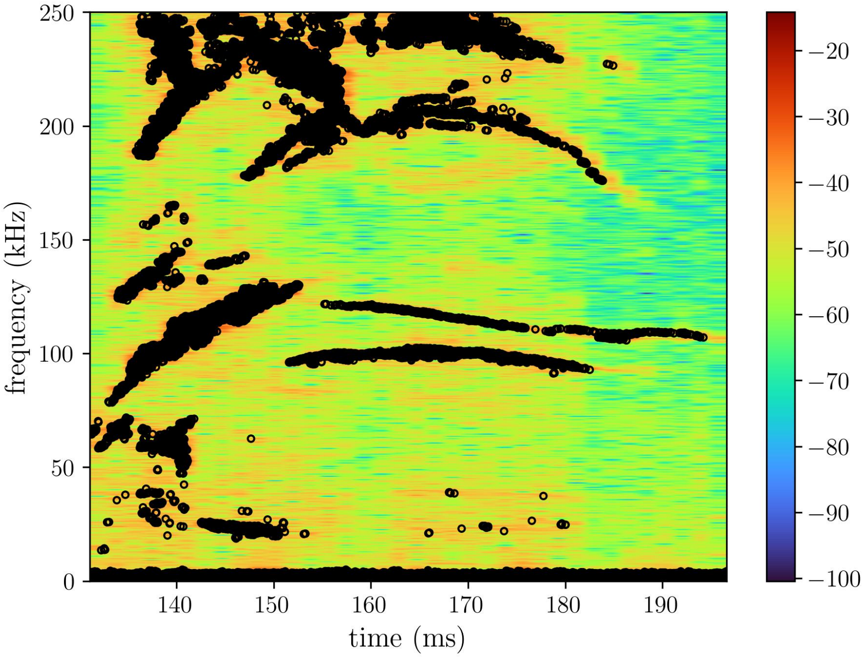


Figure 3.16.: Result of mixed ℓ_1 and ℓ_2 encoding: setting $\lambda_1 \neq 0$ and $\lambda_2 \neq 0$ in Elastic Net regularization loss.

number of batches is reached, then all updates are combined indicating the direction of the gradient descent. This method works assuming that the averaged gradient direction of all batches considered is in the right direction. Gradient accumulation allows the encoding of high-resolution signals, and to the best of our knowledge, it is used to encode signals for the first time in this work. It is worth mentioning that we used PyTorch's Adam optimizer with its default parameters.

Finally, a strong scaling test has been carried out to quantify the speed-up of the proposed encoding algorithm. The result can be seen in Figs. 3.17.a-3.17.b. The test was carried out using 8 NVIDIA Tesla P100 GPUs, each one with 16384 MB of memory. Execution times are averaged over 5 epochs, the error bars cannot be appreciated in the figure. The result shows that the time execution scales linearly with the number of atoms in the dictionary, as is evidenced in Fig.3.17. Also, the speed-up saturates over 8 GPUs due to data communication, as reported in Fig.3.17. Therefore, an epoch can be completed in less than 5 minutes. On the contrary, the encoding can be done on the CPU with the SPGL1 algorithm [VF09; VF11], dividing the signal into 10-15 chunks, running times take from 40 to 60 minutes, and chunks codes need to be re-weighted at the end. If we compare the performance with SPGL1's running times, the improvement achieved using the neural network is remarkable.

It is to be noted that the elastic net regularization has been used before for images in dictionary learning algorithms (when code and dictionary need to be learned simultaneously [Jon19; SLW16]). Recent works accelerated image encoding using a linear neural network [SSK23]. In addition, there are some previous works with deterministic

Gabor's dictionaries [Kel+12][FG24] but not with dictionaries of random features. The use of elastic net regularization in these publications consistently improves performance, when identifying correlated atoms (modes) is needed.

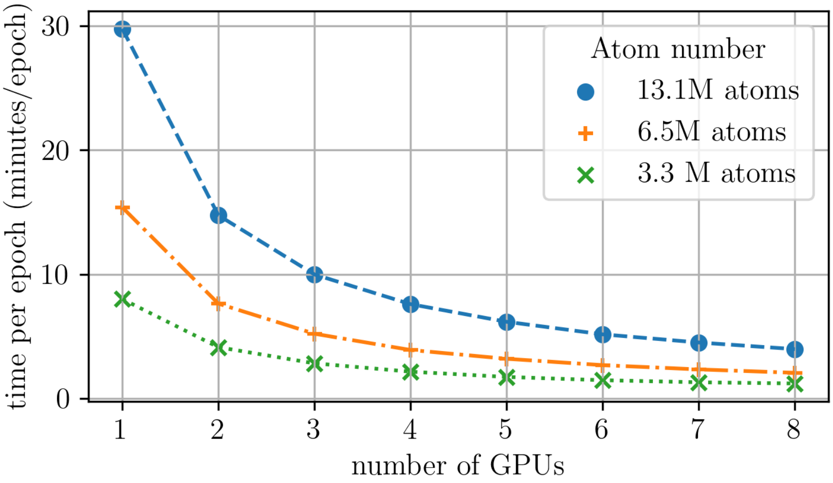


Figure 3.17.:**a** Strong Scaling Benchmark. Each experiment has been repeated 5 times, error bars can not be appreciated.

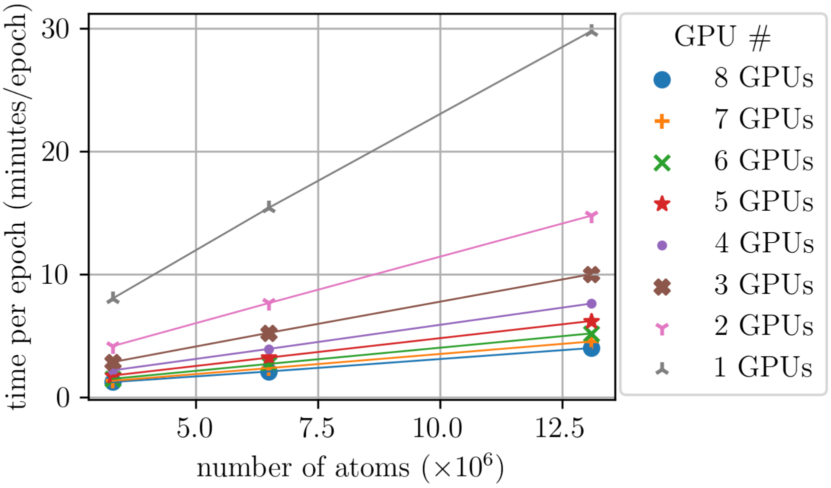


Figure 3.17.:**b** Linear scaling with number of atoms. Each experiment has been repeated 5 times, error bars can not be appreciated.

The algorithm can be used on GPUs. To process the signals in one GPU, it is recommended to select a reduced number of features and a smaller batch size. We show two different train losses in Fig.3.18 and Fig.3.19. In Fig.3.18, we can notice two epochs iterations, and in Fig.3.19 we can distinguish the point at which the gradient is updated by a big step at the end of each epoch. Notice how a bigger batch size stabilizes the learning, because a bigger size allows a better estimation of the direction of the gradient descent. If there is enough memory it is good practice to set the number of batches for gradient accumulation equal to the number of batches of the signal so each epoch is equivalent to a pass of all data through the GPU.

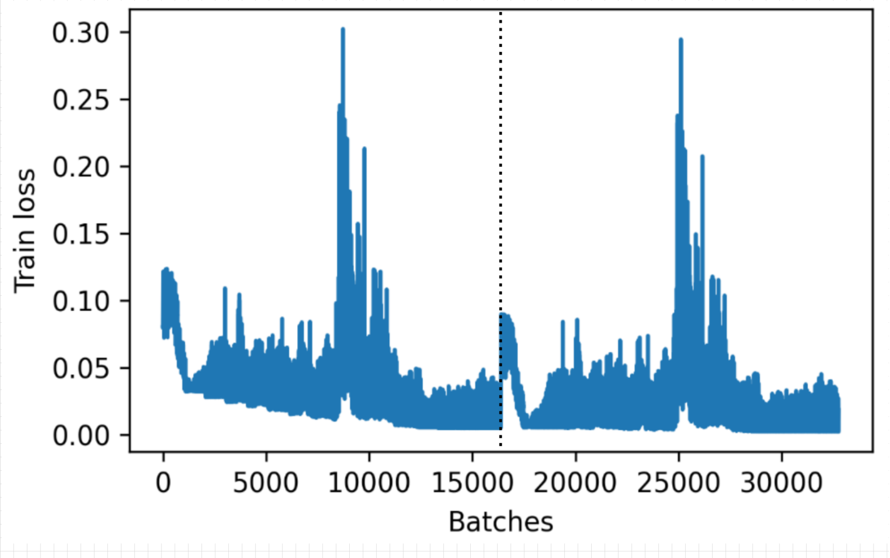


Figure 3.18.: Batch size of 32 samples, gradient accumulation of 1024 batches, 2 epochs. (1 GPU). Dashed vertical lines indicate the end of an epoch.

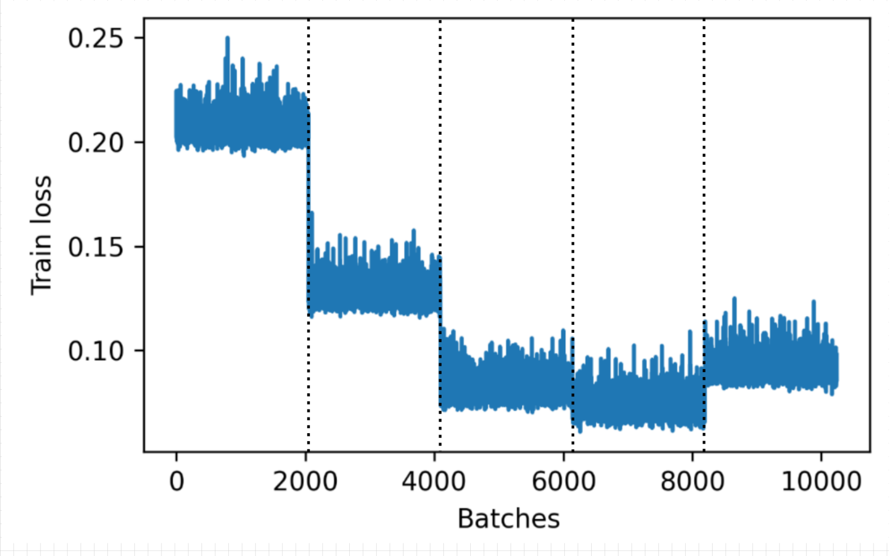


Figure 3.19.: Batch size of 2048 samples, gradient accumulation of 1 epoch, 5 epochs. (8 GPUs). Dashed vertical lines indicate the end of an epoch.

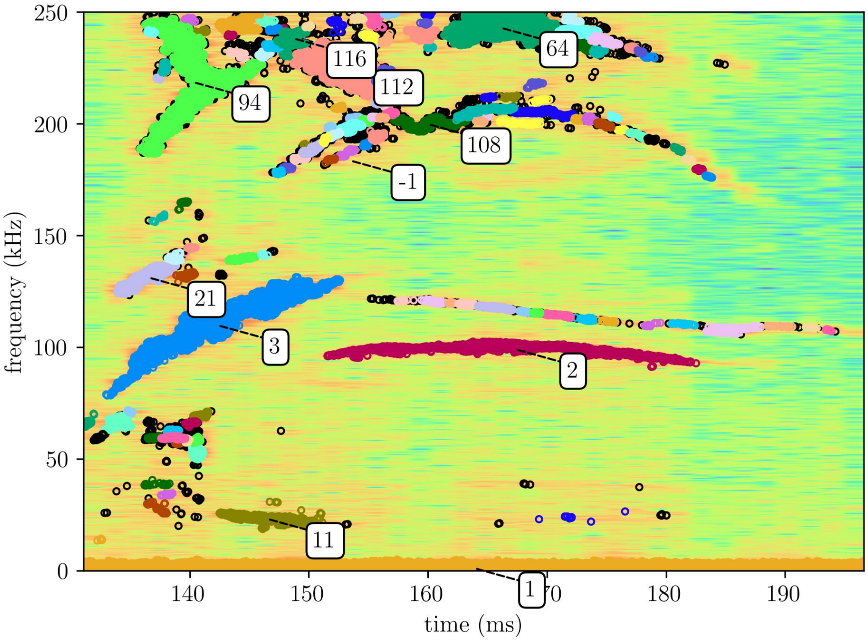


Figure 3.20.: Cluster result using HDBSCAN (Annotated the centers of 10 largest clusters and noise (-1).)

3.4.2. A parameter-free cluster method: HDBSCAN

The different atoms are grouped using an unsupervised cluster algorithm. The resulting clusters are identified as modes. In addition, improvements have been made in the clustering algorithm. The clustering algorithms based on density are adequate for our application as encoded atoms tend to concentrate on the time-frequency mode structures, and they are equipped to handle noise points and non-linearly separable clusters of arbitrary shapes and sizes. The clustering algorithm initially proposed in [RST22] for grouping the atoms in modes is DBSCAN [Est+96]. However, DBSCAN has two limitations. First, the most sensitive hyperparameter ϵ, which controls the maximum neighbor distance, varies between shots. Second, DBSCAN assumes constant density clusters, which is not true for the results of encoding. In contrast, hierarchical DBSCAN (HDBSCAN) [Cam+15] does not assume constant density and is capable of determining the varying density structure. Moreover, HDBSCAN does not have a sensitive hyperparameter to modify on each shot. For these reasons, HDBSCAN is more adequate than DBSCAN. An example of an HDBSCAN's result is shown in Fig.3.20.

3.4.3. ERMD algorithm and its hyperparameters

We propose the adoption of the name *Elastic Random Mode Decomposition* (ERMD) to refer to our new algorithm. ERMD is summarized as follows:

Algorithm 2: *Elastic Random Mode Decomposition* (ERMD) for Alfvénic mode classification

---

INPUTS: signal $\boldsymbol{s}$, time $\boldsymbol{t}$.
OUTPUT: code $\boldsymbol{c}$, best model code $\boldsymbol{c}_{\text{best}}$, atom's time position $\boldsymbol{\tau}$, frequencies $\boldsymbol{f}$, phases $\boldsymbol{\phi}$, cluster labels $\boldsymbol{l}$ and Alfvénic character $\boldsymbol{a}$. All with the same vector length.
PARAMETERS: $\{\lambda_1, \lambda_2, \sigma, N_{max}, N_f, B_s, \eta, max_{epoch}, Q, Q_c \; \text{min}_{size}, \rho_{thr}.\}$ See table (3.1).
STEPS:

1. Standardize signal $\frac{\boldsymbol{s}-\mu(\boldsymbol{s})}{\sigma(\boldsymbol{s})}$ and time $\frac{\boldsymbol{t}-\mu(\boldsymbol{t})}{\sigma(\boldsymbol{t})}$ arrays.

2. Apply high-pass filter (Optional: it can help to detect high-frequency modes)

3. Set a global seed for all GPUs, it is necessary because sampling features space $\{\tau_i, f_i, \phi_i\}$ has to be deterministic.

4. Set the number of atoms in the random dictionary N as a multiple of signal length $\boldsymbol{s}_l$, $N = N_f \times \text{lenght}(\boldsymbol{s})$. It is useful to set a maximum number of atoms N_{max} to avoid out-of-memory problems.

5. Repeat in parallel until reaching max_{epoch}:
    a) For each batch of time signals, load the data (segment of size B_s) to an assigned GPU and compute a segment of matrix $\mathbf{D}_{\text{batch}}$ in the assigned GPU. Use seed and atom equations to calculate the value of corresponding atoms in $\mathbf{D}_{\text{batch}}$ (in-GPU dictionary construction)

    $$\mathbf{D}_{\text{batch}} = [\boldsymbol{s}_i(\boldsymbol{t}_{\text{batch}})] = \left[exp\left(-\frac{(\boldsymbol{t}_{\text{batch}} - \tau_i)^2}{2\sigma}\right) \cdot \cos(2\pi f_i \boldsymbol{t}_{\text{batch}} + \phi_i) \right]. \tag{3.14}$$

    b) Forward pass for each batch: calculate elastic net regression loss

    $$\mathscr{L}_{\text{batch}} = ||\mathbf{D}_{\text{batch}} \cdot \boldsymbol{c} - \boldsymbol{s}_{\text{batch}}||_2^2 + \lambda_1 ||\boldsymbol{c}||_1 + \lambda_2 ||\boldsymbol{c}||_2, \tag{3.15}$$

    then calculate each batch gradient $g_{batch}(\boldsymbol{c})$ and store it.

    c) When the number of batches processed completes the epoch: accumulate gradient (the sum of all batches gradients determine the descend direction), and actualize code weights.

    $$\boldsymbol{c}_{\text{new}} = \boldsymbol{c}_{\text{old}} - \eta \sum_{\text{batches}} g_{\text{batch}}(\boldsymbol{c}) \tag{3.16}$$

    d) Keep the best code $\boldsymbol{c}_{\text{best}}$: model with best loss $\sum_{batches} \mathscr{L}_{\text{batch}}$ so far.

6. Return $\boldsymbol{c}$, $\boldsymbol{c}_{\text{best}}$, $\boldsymbol{\tau}$, $\boldsymbol{f}$, $\boldsymbol{\phi}$.

7. Code thresholding: group atoms code values $\boldsymbol{c}_{\text{best}}$ by quantiles, using the specified (number of quantiles Q) and keep those above the (quantile cut Q_c).

8. Cluster modes using HDBSCAN with desired Min. cluster size min_{size}. Return $\boldsymbol{l}$.

9. For each identified mode, except noise clusters (-1), there are two options:

  a) Calculate Pearson's correlation between mode frequencies and $1/\sqrt{n}$. If the correlation is greater than the correlation cut ρ_{thr}, label the mode as 1 (Alfvénic) otherwise 0 (No Alfvénic). Return $\boldsymbol{a}$.
  b) Use mutual information (MI) instead, this can help to retrieve more modes. (The details can be found in the next chapter). Return $\boldsymbol{a}$.

10. Store result $\boldsymbol{c}$, $\boldsymbol{c}_{\text{best}}$, $\boldsymbol{\tau}$, $\boldsymbol{f}$, $\boldsymbol{\phi}$, $\boldsymbol{l}$, $\boldsymbol{a}$.

The relevant hyperparameters and used values are summarized in the table 3.1. Among them, the most important parameters are the regularization terms λ_1 and λ_2, which adjust the promotion of sparsity and regularization. The values of λ_1 and λ_2 depend on the signal-to-noise ratio and they have an important effect on the mode retrieval. Unfortunately, there is no rule to adjust these parameters as they heavily depend on the data and the hyperparameter search space has multiple local minima. Using hyperparameter optimization is difficult, as there can be multiple objectives (minimize reconstruction error, minimize the number of modes, or minimize the number of epochs). Instead, it is advisable to first determine λ_1 with $\lambda_2 = 0$ and then add λ_2 as needed. All in all, it can be found values of λ_1 and λ_2 that work reasonably well with all shots. Nevertheless, to obtain the best result, avoiding the loss of modes, or fully eliminating broadband noise, it is necessary to fine-adjust λ_1 and λ_2.

The selection of the regularization parameters is influenced by the dictionary size. Quoting Zou and Hastie [ZH05], the creators of Elastic Net, "The elastic net is particularly useful when the number of predictors (p) is much bigger than the number of observations (n). By contrast, the Lasso is not a very satisfactory variable selection method in the $p \ggg n$ case." They also observed that Elastic Net only performs well when it is closely aligned with either Ridge regression or Lasso regression. This implies that the choice of λ_1 and λ_2 is biased towards Ridge or Lasso depending on the shape of the dictionary matrix, which is determined by the number of features and the signal length. These observations align with our experience. The reader can see that λ_2 is much bigger than λ_1 in table 3.1. In addition, the dictionary size also affects the learning rate choice.

In our study, we employed Bayesian multi-objective hyperparameter optimization to minimize the mean squared error (MSE) and the number of training epochs. To achieve a sparse reconstruction, we kept the learning rate resulting of this Bayesian search, but we had to manually fine-tune the obtained λ_1 and λ_2 until the atom selection matched the spectrograms. While grid hyperparameter searches with cross-validation are common practice to minimize reconstruction error and code vector energy [FG24], for our use case this approach is prohibitive. Therefore, we relied on a more manual parameter selection strategy.

Parameter	Value(s)
λ_1	0.00005
λ_2	0.001
Atom standard deviation σ	256 samples
Maximum number of atoms N_{max}	14×10^6
Feature factor N_f	20
Batch size B_s	32-64 samples
Learning rate η	0.00008
Maximum epochs $\max_{epoch}$	5
Number of quantiles Q	1000
Quantile cut Q_c	999
Minimum cluster size $\min_{size}$	10
Correlation threshold ρ_{thr}	0.9

Table 3.1.: ERMD Hyperparameters. This values are the selected for running the analysis in the 1291 shots of our database.

3.5. Results of unsupervised feature extraction of Alfvén activity using ERMD

To evaluate the performance of the algorithm, we run ERMD on 1000 shots with a fixed set of parameters. The hyperparameters were first selected by trial and error to obtain good performance on a selection of shots. About the execution times, it took 206.8 hours to run the 1000 shots, so this gives an average of 4.8 shots per hour approximately. The shots have different numbers of samples so this number can fluctuate from 3 to 7 shots per hour. In total, we accumulated 1291 shots in the database.

As an example of a typical result, the encoding and clustering of shot 23811 are shown in Fig.3.22. Three high-frequency modes are correctly identified, with the assigned numbers 0,2, and 3. Some noise atoms are correctly identified as noise by HDBSCAN. Finally, in Fig.3.23 the two higher-frequency modes are identified as Alfvénic, as they are strongly correlated with the $1/\sqrt{n}$ signal. However, part of the broadband noise structure is misidentified with fragmented low-frequency modes, this behavior is found to be common but does not affect the labeling of Alfvénic modes if the correlation threshold is set high. It is to be noted, how the amplitude of broadband noise increases when heating systems are activated, this can be seen comparing power inputs in Fig.3.21 with the spectrogram in Fig.3.22 and Fig.3.23.

Because each mode has a different density of atoms, the algorithm is capable of labeling crossing modes, as is shown in Fig.3.24. This task is very difficult for algorithms based on the segmentation of the spectrogram. Therefore, this is a remarkable advantage of this algorithm which is reinforced with the addition of the density signal $1/\sqrt{n}$ to the analysis. However, there is a trade-off: the density-based clustering also tends to divide one mode into different sub-modes. In consequence, ERMD results might not be reliable to measure mode duration Δt.

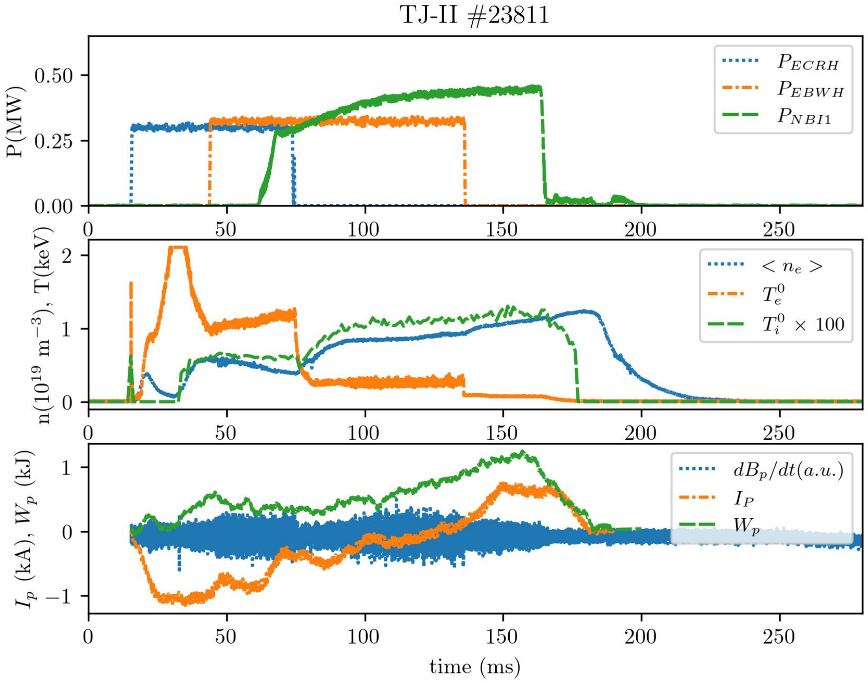


Figure 3.21.: Shot 23811 signals, from top to bottom: heating power, electron density $< n >$ and temperatures T_e^0, $T_i^0 \times 100$, and the MIR5C signal, I_p, and W_p.

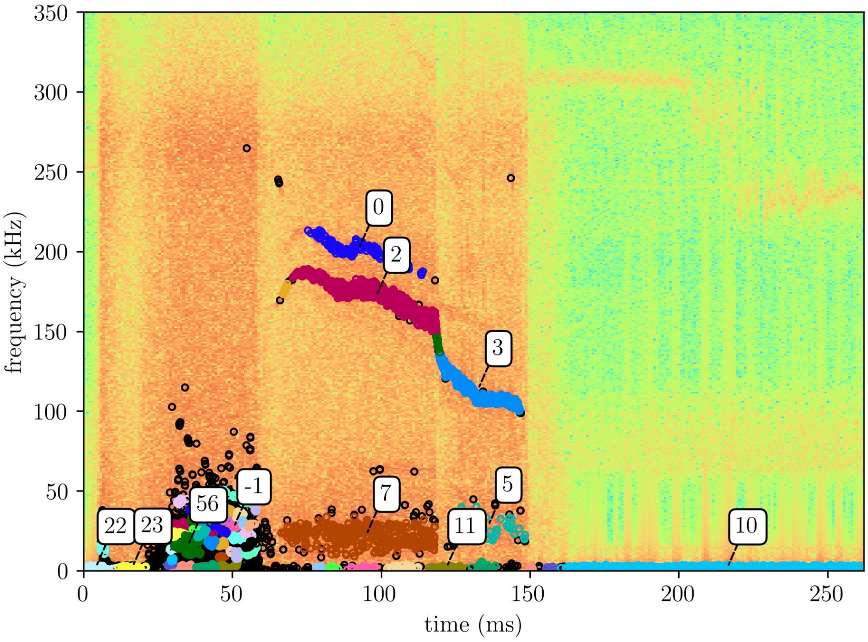


Figure 3.22.: Clustering result of shot 23811. The labels mark the center of the top 10 numerous clusters. Black circles (-1) are noise.

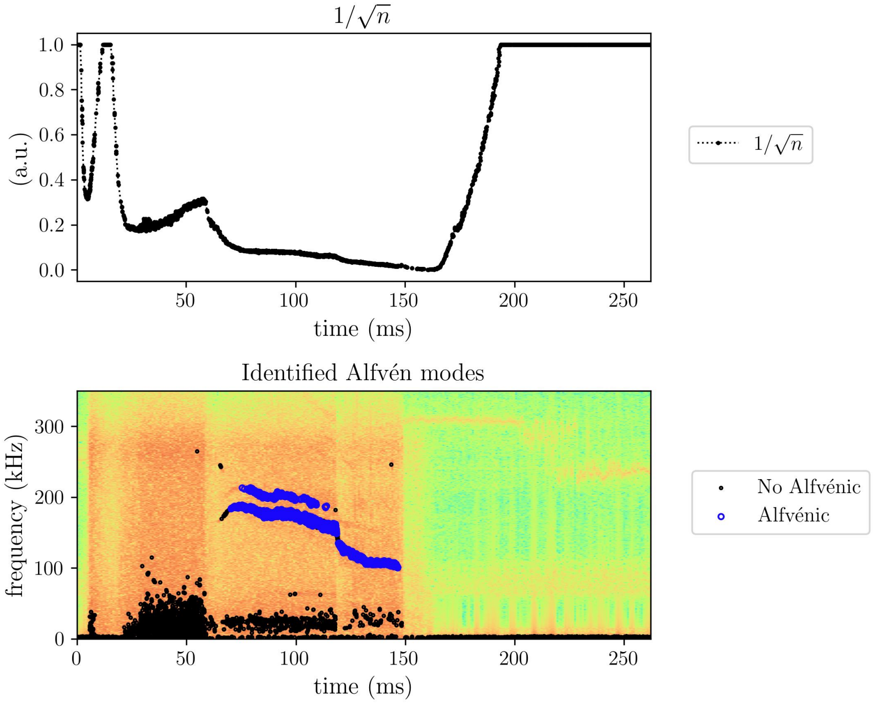


Figure 3.23.: Result of labeling shot 23811: modes identified as Alfvénic (big blue circles) follow $1/\sqrt{n}$.

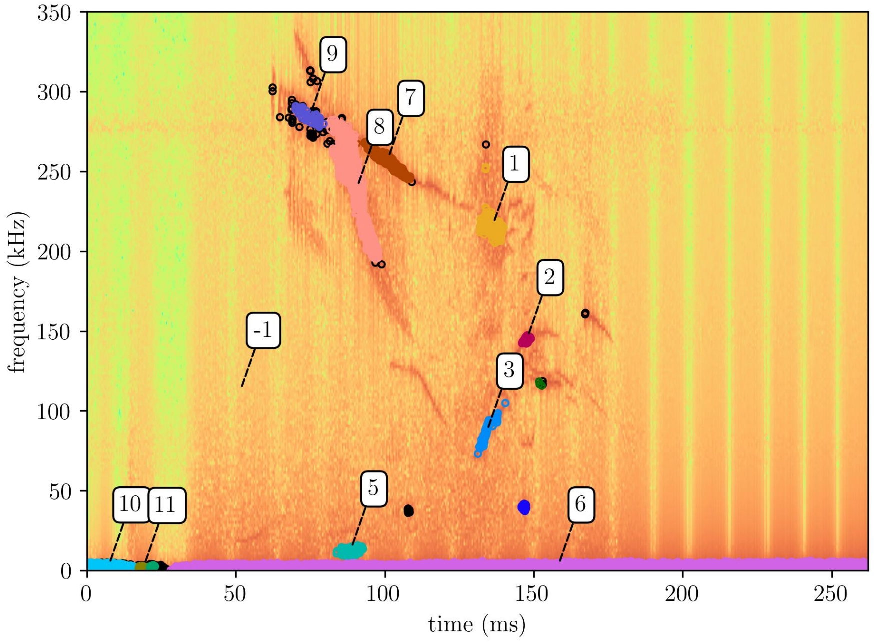


Figure 3.24.: Cluster results of shot 38393: ERMD handles crossing modes (7,8,9). Noise (-1) is distributed between low and high frequencies.

The frequency distribution of atoms in Alfvén modes is checked for agreement with the frequencies expected from the literature. As can be seen in Fig.3.25 and Fig.3.26. Alfvén frequencies range from 100 kHz to 400 kHz, overlapping with the mentioned interval 50kHz to 400kHz [Jim+11; Mel+12; Mel+14; Mel+18; Eli+21; VSG17; SOL15; Cap+21; Ghi+24]. Most Alfvénic modes appear between 50 ms and 200 ms, and the majority of modes (80%) in this set of shots is chirping down. The use of the algorithm is not exempted from difficulties, as some modes can be missed in this encoding (Fig.3.24). The risk of missing modes is in general low except for two cases: when there are several modes stacked in parallel frequencies or when the signal-to-noise ratio is very low. Because the algorithm is based on minimizing the signal reconstruction error with some regularization, these two errors will persist without fine-tuning the parameters for each shot. The reconstruction error is small in the case of modes with less amplitude, this makes the encoding difficult as other frequencies with greater amplitude are encoded first (lower frequencies have usually more amplitude). When the signal-to-noise ratio is very low, the regularization needs to have very shot-specific values to separate modes from broadband noise, like in some modes of Fig.3.24. We address this problem with a different approach in chapters 5 and 6.

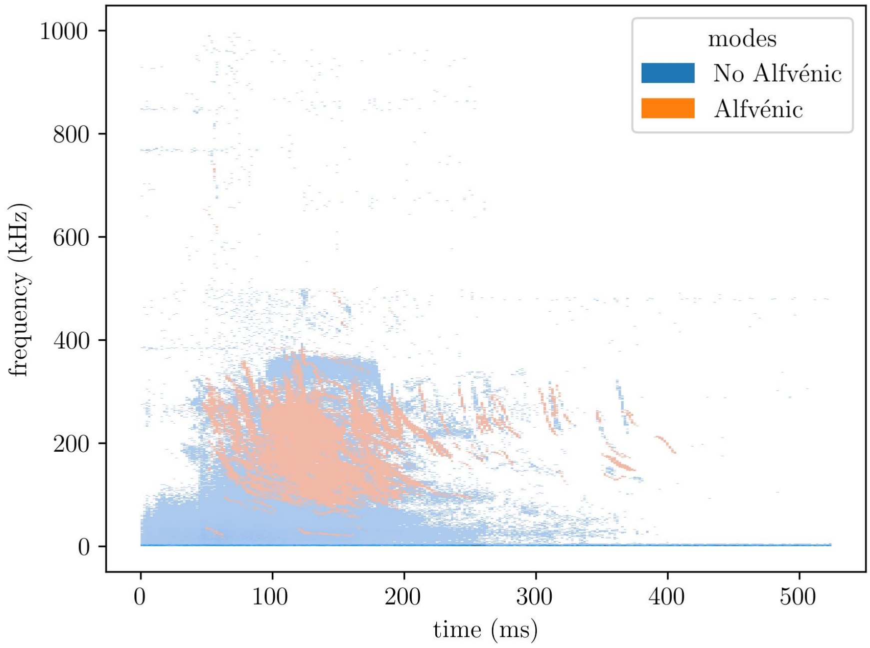


Figure 3.25.: Time-frequency distributions of Alfvénic mode atoms of this set of 1291 shots.

More difficulties have been found in the presence of sawteeth. These fast transient signals generate broad frequency spectral leakage, as it happens in STFFT. When minimizing the signal reconstruction error, many different atoms can fit these transients. Therefore, the sparsity assumption does not hold anymore and the regularization parameters should tuned to address this difficulty. This phenomenon could be tackled

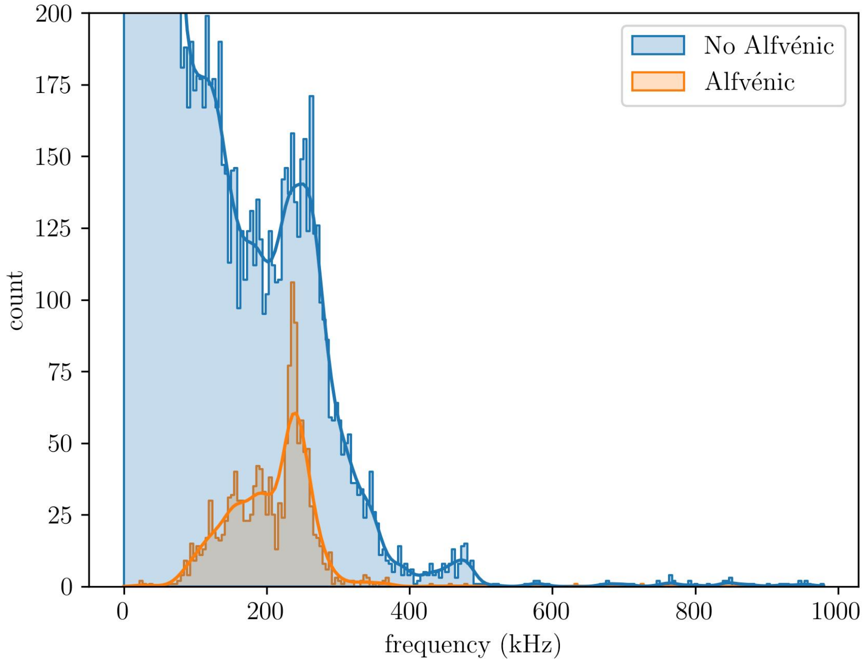


Figure 3.26.: Frequency distribution of Alfvénic mode activity in the analyzed shots.

in principle by adding "sawteeth atoms" with the shape of Gaussian's first derivatives to the dictionary, which recreates the transients. These difficulties could be addressed in future investigations.

In the next chapter we examine the time-frequency analysis performed in 1291 shots with the ERMD, and we use mutual information and clustering to detect relationships between mode activity and other plasma signals, creating a profile of the different types of modes observed, which we have called high-level patterns.

To conclude, we summarize the main advantages of the proposed algorithm are: (1) Robust mode detection, allowing for effective identification of complex modes in raw Mirnov signals. (2) Use of elastic net regularization, which enhances the algorithm's ability to handle correlated features. (3) Scalability to large datasets, the resulting table of atoms is highly convenient for further pattern recognition using multiple diagnostics. (4) High parallelization allowed by GPU optimizations enables efficient processing of high-resolution signals. (5) Improved clustering with HDBSCAN compared with preceding works which better handles varying densities and noise.

In addition, the main disadvantages of ERMD would be: (1) Complex parameter tuning, requiring careful adjustment of multiple hyper-parameters to achieve optimal results. (2) Computational cost, which is dependent on the availability and performance of GPU resources. (3) Sensitivity to noise in low signal-to-noise ratio scenarios, which may lead to challenges in accurate mode detection; and (4) Difficulty with fast transients, similar to challenges faced by the Fourier transform, where the algorithm may struggle to capture rapid signal changes effectively. However, the latter can be addressed in principle by changing the dictionary waveforms.

3.6. Summary

In this chapter, we first introduce the mode decomposition of Mirnov coil signals using a musical analogy in which interpreting plasma signals (finding MHD modes) has parallelism with musical transcription. For this purpose, we introduced SRMD as an algorithm for sparse coding of the signal with a dictionary of random Gabor atoms. The algorithms presented in this chapter are a combination of regularized linear regression and unsupervised learning.

The mode decomposition of the TJ-II plasma's signal is allowed by recognizing that sparsity is a good prior (sparse regression). However, this algorithm has limitations in speed and memory to be used in high-frequency sampled signals proper of fusion diagnostics.

Then, a novel algorithm called ERMD is introduced to cope with SRMD limitations. In order to do that, the optimization problem needs to be tailored for the use case of plasma diagnostics.

The signal reconstruction is enhanced by adding a multicollinearity term to the regularization (using Elastic-Net regression). The combination with optimization schedule techniques (gradient accumulation) from deep learning training, allows the encoding of codes bigger than before. We achieved improvements in speed and memory, therefore increasing the code's maximum length. In addition, a more suitable clustering technique (HDBSCAN) is proposed to avoid the tunning of DBSCAN's ϵ between shots. Finally, ERMD has been applied over 1291 shots with the same set of hyperparameters and the frequencies of the Alfvénic modes retrieved agree with the literature.

4. High-level pattern recognition

> *Joi: Mere data makes a man. A and C and T and G. The alphabet of you. All from four symbols. I am only two: 1 and 0.*
> *'K': Half as much but twice as elegant, sweetheart.*
>
> — Blade Runner 2049.

Summary

4.1. Introduction

Following the musical analogy of the last chapter's introduction: if we can recognize musical notes out of audio waveforms, how can we recognize melodies? Can we differentiate bass lines from the main motives played by different instruments? Or how can we detect chords and rhythms? Moreover, songs have different parts such as chorus, bridges, intro, or finals. Most concepts in musical theory serve the purpose of describing high-level patterns in music. As in music, fusion plasmas can exhibit oscillations and time-frequency structures leading to high-level patterns. The goal of this chapter is to find different kinds of mode structures determined by ERMD, this serves a double purpose of interpreting the result of ERMD when it is applied to a set of shots and explore pattern discovery algorithms.

Patterns are often hidden in the data. Indeed, there is a substantial difference between information and data. Data mining is the process of extracting knowledge from a dataset. This knowledge can be achieved by different means. Over years of research, petabytes of data are stored coming from thousands of shots, and a minimum level of data understanding is required to prepare the data to create successful machine learning models. The word mining often implies hard and dirty work, as the complexity of understanding what is in the dataset is highly time-consuming for the data analyst. In addition, there is a risk of falling into many pitfalls leading to false conclusions as the analyst explores a complete uncharted territory. If data is structured (i.e. tabular form, or hierarchical), visualization, or descriptive statistics can be employed to inspect variable distributions and relationships. However, interpreting the result of exploratory analysis requires domain expertise, which also requires learning time or expert aid. Best analysis often involves the creation of new variables that are relevant to the given domain, and that can only be done with domain knowledge and expertise. The creation of new variables that facilitate the analysis is called feature engineering, as we discussed in chapter 2.

If data is unstructured, for instance, images or text, the analysis gets even more complex. First, the number of pixels necessary to plot an image increases the number of dimensions, which can be orders of magnitude greater than a table. These dimensions need to be reduced. Second, objects present in an image create local relationships between the object's pixels, these pixel relationships are called features and they need to be extracted. One approach to extracting knowledge from the dataset consists of calculating similarities between images and then creating groups of similar images. This high-level pattern recognition is called clustering.

Relationships between variables can be used to create new features. There are many mathematical models to estimate relationships between two variables. In addition, caution must be held when estimating relationships. It is well known that correlation does not imply causality, but what is often ignored is that the relationship estimated by this model might not be even true. For instance, noise can often create spurious relationships. In other words, one can fit anything into a noisy dataset. Moreover, a correlation between two variables can be explained by an unknown third one with shared relationships with the other two. This third variable is known in statistics as a confounder.

Physics has a true advantage over other disciplines like biology, sociology, etc. The true relationship between variables can be explained by known physical mechanisms, based on first principles. In addition, experiments isolating variables can be used to prove hypotheses and validate mathematical models. However, finding correlations in data is often a common starting point for researching first principle laws. For instance, Kepler's laws were a starting point for Newton's gravitation, and scaling laws are used in fusion to design future reactors. Given the fact that isolating variables is such a demanding task in a complex system like fusion plasmas, searching patterns and relationships can be a seed for future physical discoveries.

As we mentioned before, the Alfvén waves' frequencies f follow a linear dependency with the inverse square root of density $1/\sqrt{n}$. Once modes are retrieved by the ERMD algorithm, we have a database of modes that can be mined for finding Álfvén instabilities.

In this chapter, we use feature engineering and clustering techniques to explore the result of applying ERMD to 1291 TJ-II shots. First, we discuss various statistical methods that can be used to determine if two continuous variables have a relationship (Pearson's correlation and Mutual information), and how we can find groups of observations in a dataset (discretization and clustering). The results of the clustering are interpreted with our knowledge of Alfvén activity.

4.2. Finding relationships between continuous random variables

Linear relationships can be identified using Pearson's correlation $r = cov(x, y)/(\sigma_x \sigma_y)$. Pearson's correlation model assumes a linear relationship (straight line) between variables X and Y. The correlation coefficient r varies between -1 and 1. Given two variables X and Y: the positive sign of the correlation coefficient r indicates a positive slope (when X increases, Y increases linearly), and the negative sign negative slope (when X increases, Y decreases linearly). It is to be noted that Pearson's correlation does not capture the slope of the data, but the strength of the relationship and the directionality of this relationship, which roughly matches the slope of a straight line fitted to the data.

However, assumptions made for Pearson model need to be true. We assume the data X Y lies in a straight line to begin with. In addition, the variables X and Y have to be normally distributed, with approximately equal variance along the regression line, and without outliers. In other words, Pearson's correlation is particularly sensitive to outliers and noise. If the mentioned assumptions are true, and Pearson's value $r(f, 1/\sqrt{n})$ is close to 1, we can say that we have identified an Alfvénic mode. Pearson's correlation can handle noise if the noise is evenly distributed along a straight line with a normal distribution (homoscedasticity) and there are no outliers. However, it is important to note that these conditions are not true for all clustered modes, especially for noisy modes determined by ERMD.

For n observations of random variables X, Y with continuous distributions, Pearson's correlation can be calculated as

$$r = \frac{cov(X, Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2 \sum_{i=1}^{n}(Y_i - \bar{Y})^2}}. \tag{4.1}$$

Pearson's model is easily interpretable, but there are other options to estimate relationships between two variables: like Spearman's correlation (useful with non-linear monotonic data) and mutual information regression (which captures non-linear relationships), among others.

The Spearman's correlation is calculated using Peason's model but transforming the variables X, Y in ranks $R(X)$ and $R(Y)$ first. The rank is calculated by assigning a natural number to each value of X with increasing order. The same with Y. For instance, given the small dataset $X = \{1, 3, 5, 7\}$ and $Y = \{9, 3, 1, 5\}$: $R(X) = \{1, 2, 3, 4\}$ and

$R(Y) = \{4,2,1,3\}$. Therefore the Spearman's rank correlation can be defined by

$$r_S = \frac{cov(R(X),R(Y))}{\sigma_{R(X)}\sigma_{R(Y)}}. \tag{4.2}$$

Spearman's correlation has several advantages over Pearson's correlation. First, it is less sensitive to outliers than Pearson's. Moreover, it is capable of capturing non-linear relationships, as long as X and Y increase or decrease monotonically. And it can be used with ordinal data, including categorical data that can be ordered.

Having two sets of data X and Y, we can say that if we can relate X with the output Y, it is because we have some "shared information" relating X and Y. Certainly, if the relationship is linear, having the information of parameters m, and b can explain the relationship of X and Y via the parametric model $y = mx + b$. Remarkably, the straight line parameters m, and b can code the data X and Y with very small errors in some cases.

But, what is information exactly? Everybody has an intuition about its meaning. However, it was not until Claude Shanonn's seminal paper [Sha48] that a solid foundation of the concept was established. The concept of information is unambiguously defined for discrete variables. Given a random variable X with a discrete probability distribution $p(x) = \{p(x_1),...p(x_m)\}$, its information I is the average entropy $H(X)$ [Sto22]

$$I \equiv H(X) = \sum_{i=1}^{m} p(x_i)\log_2 \frac{1}{p(x_i)} \text{ (bits)}. \tag{4.3}$$

It is important to mention, that most information theory literature notation assumes $\log_2 \equiv \log$, unless indicated otherwise. If natural logarithm is used in Eq. 4.3 instead of $\log_2$ the unit for information is called nats.

The intuition behind entropy is that a high-entropy system has higher levels of disorder or unpredictability than a system with lower entropy. In information theory, a source that produces high-entropy signals generates more information because its outputs are less predictable. While a low-entropy source produces more predictable (and thus less informative) signals, it can still carry information. This is why information is also called a measure of surprise, with "surprise" defined as the logarithm (base 2) of the inverse probability $\log_2(1/p(x_i))$ of an event. The rarer the event, the greater the surprise and the higher the information content.

A transmission channel's capacity to carry information depends on its physical characteristics, such as bandwidth and noise level, not on entropy. However, the efficiency with which a channel is used depends on matching the source's entropy to the channel's capacity.

The information unit is the bit. One bit is the information needed to choose between two equally probable options [Sto22]. The simplest example to understand the information definition given by Eq. 4.3 is the coin tossing case. A random variable X which has values $\{0,1\}$ representing $\{\text{heads},\text{tails}\}$ can describe the coin tosses. Effectively, if the coin is unbiased, the phenomenon follows a Bernoulli distribution and $p(x_{heads}) = p(x_{tail}) = 1/2$, so the surprise is $\log_2(1/0.5) = \log_2(2) = 1$ bit. Moreover, 1 bit is the memory needed to store the result of one flipping (fair) coin using binary digits.

If we have collected 100 samples from this distribution, we might have 50 heads and 50 tails. Hence, the average entropy can be calculated $H(X) = [50\log_2 p(x_{heads}) + 50\log_2 p(x_{tails}]/100 = 0.5\cdot\log_2(1/0.5) + 0.5\cdot\log_2(1/0.5) = 1$ bit per coin flip. It can be easily demonstrated [Sto22], that the information given from the events of a fair coin is higher than the information given by tossing a biased coin. Because the biased coin is more predictable than the fair coin.

It is worth discussing briefly the thermodynamical entropy concept as it differs from information entropy (or Shanonn's entropy) but they are closely related. Formally, thermodynamic entropy S is defined as the natural logarithm of the number of microstates W, times the Boltzmann constant k_B: $S \equiv k_B \ln(W)$ Joules/degree. The parallelism with Shanonn's information is even clearer if we use Gibbs' generalization [Sto22]:

$$S = k_B \sum_{i=1}^{m} p(x_i) \ln \frac{1}{p(x_i)} \text{ (Joules/Kelvin)}, \tag{4.4}$$

where m is the number of macroscopically distinguishable physical configurations[Sto22] and $p(x_i)$ is the probability of each microstate. If each microstate corresponds to one macrostate $m = W$, then all microscopic states have to be equally probable, it follows that $p(x_i) = 1/W$. For example, an 20-sided die would have 20 equally probable macrostates (outcomes), and 20 microstates.

The results enabled by these concepts are even more important than the definition. Given certain physical processes: we can demonstrate using the thermodynamical entropy that the amount of energy that can be released from a system, depends on the availability of a low entropy source. In particular, as the second law of thermodynamics declares, spontaneous processes always result in entropy increasing. Whereas the information theory tells us the limit of uncorrupted information that a transmission channel can transmit (channel capacity theorem), the thermodynamical entropy determines the limits of work that thermal machines can perform. Therefore every machine that needs to interchange energy through heat has an efficiency limit. Moreover, the explicit relationship between thermodynamical entropy and information tells us how much information a system can acquire, and how much energy the information costs: to acquire one bit of information a system must expend at least, $0.693 k_B T$ joules of energy (also known as Landauer limit [Lan61; Sto22]).

We have defined the information entropy for discrete variables, but defining an absolute information entropy for continuous random variables can be problematic as it ends up diverging (the reader can find the details in [Sto22]). We can intuitively explain why it is complicated: we need to define absolute entropy for real numbers, as real numbers are infinitely dense, the entropy can grow arbitrarily up to infinity. However, *relative information* between two continuous random variables distributions can be defined without falling into infinities if we use conditional probabilities.

We can define Mutual Information using a simple linear regression model between a feature variable X (or predictor) and a target variable Y (dependent variable). In a realistic case, the input feature information (input entropy $H(X)$) contains some noise $H(X|Y)$ (X given Y) and some information $I(Y,X)$ (Y and X) about the target. In addition, the target value also contains noise $H(Y|X)$ (Y given X) and information $I(X,Y)$ (X and Y) about the predictor variable.

If the features X have information about the target values Y, they can be predicted (and vice versa). Therefore $I(X,Y) = I(Y,X)$, and knowing the values of Y also allows the prediction of values of X. The relationship between conditional entropies $H(X|Y)$ and $H(Y|X)$ and joint entropy $H(X,Y)$ is expressed as follows

$$\begin{aligned} MI(X,Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X). \end{aligned} \tag{4.5}$$

It follows that the noise $H(X|Y) < H(X)$ and $H(Y|X) < H(Y)$, this is depicted in Fig. 4.1.

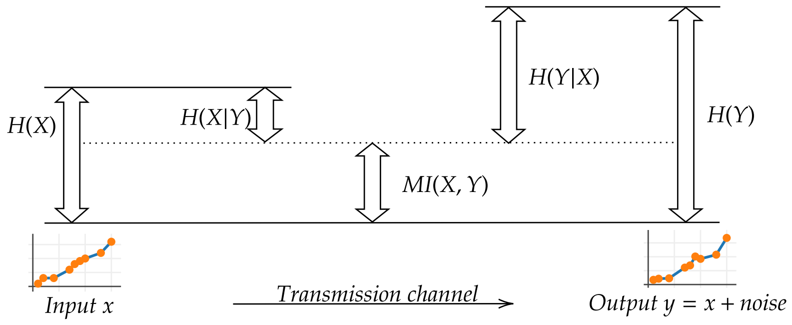


Figure 4.1.: Relation of Mutual Information with conditional and joint entropy sets. (Adapted from [Sto22])

It can be shown, after some algebraic manipulation, that MI is the intersection over the union of $H(X)$ and $H(Y)$, with total entropy $H(X,Y)$ as it is commonly illustrated in Fig. 4.2.

$$\begin{aligned} MI(X,Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X) - H(Y) - H(X,Y) \\ &= H(X,Y) - H(X|Y) - H(Y|X). \end{aligned} \tag{4.6}$$

A practical definition of MI that we can compute involves the Kullback-Leibler (KL) divergence. The Kullback-Leibler (KL) divergence, which measures the similarity (or “distance”) between two probability distributions p(X) and q(X) is defined as

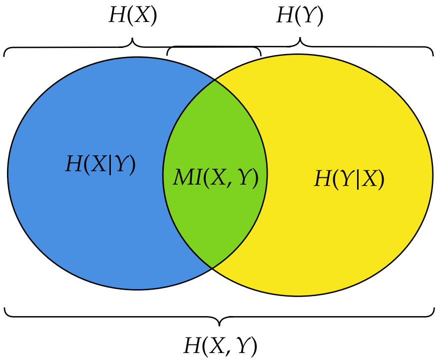


Figure 4.2.: Relation between different entropy quantities of X and Y.

[Mur22](pages 213ff)

$$\mathrm{D}_{KL}(p(X)||q(X)) = \int_x p(x)\log\left(\frac{p(x)}{q(x)}\right)\mathrm{d}x. \tag{4.7}$$

The KL divergence between $\mu(x,y)$ and joint distribution $\mu_x(x)\mu_y(y)$ (as product of the marginal densities $\mu_x(x)$, $\mu_y(y)$) can be estimated using [KSG04] the formula

$$\mathrm{MI}(X;Y) = \mathrm{D}_{KL}(\mu(x,y)||\mu(x)\mu(y)) = \int\int \mu(x,y)\log\left(\frac{\mu(x,y)}{\mu(x)\mu(y)}\right)\mathrm{d}x\,\mathrm{d}y. \tag{4.8}$$

It will be 0, if x and y are independent variables, i.e. $\mu(x,y) = \mu_x(x)\mu_y(y)$(separable). In other words, KL divergence will be 0 if both variables do not have a relationship.

For the mathematical details of a complete Bayesian interpretation of mutual information, please refer to [Sto22], page 149. We can say, if $p(X|y)$ $(y \in Y)$ is the posterior distribution and $p(X)$ is the prior distribution, the mutual information between X and Y is the expected KL-divergence between the posterior and prior distributions.

Without digging into the details, the integration of continuous variables is performed by binning the data and other methods [KSG04][Ros14], we have used scikit-learn implementation [Ped+11] for mutual information regression, which uses non-parametric entropy estimation.

It is important to note that, the output of scikit-learn implementations for estimating MI are not given in bits unit but nats (from natural logarithm). To convert nats to bits, divide nats values by ln(2). From now on, unless otherwise indicated, we consider that the result is in nats.

The different behavior between Pearson's, Spearman's and MI can be analyzed using the Anscombe's-quartet datasets in Fig.4.3. For each dataset, the different values of Pearson's, Spearman's and MI have been calculated. Since we are using the Scikit-learn implementations of MI, the so-called Mutual information coefficient, and the Mutual information regression values are included. The former is used for categorical variables (though it admits real-valued data) whereas the latter is used for real-valued data.

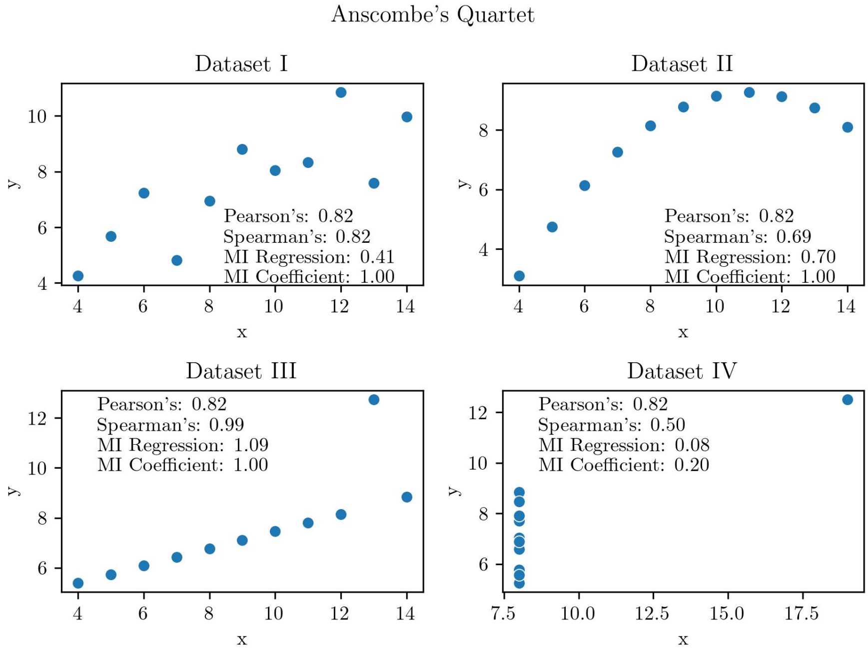


Figure 4.3.: Anscombe's-quartet datasets, showing different values of Pearson's correlation, Spearman's, Mutual Information regression, and Mutual information coefficient

The 4 datasets in Anscombe's-quartet show different cases: (I) linearly related data with Gaussian noise, (II) nonlinear data (quadratic), (III) linearly related data with one outlier, and non-related data with one outlier (IV).

Examining the values of the different measures of correlation we can see that Pearson's value is the same for all cases I-4. The fact that Pearson's is insensitive to these changes is problematic, as in datasets (II) and (IV) there is no linear relationship at all. The Spearman's values show better behavior.

For this example, MI values are better behaved when there is no relationship between the variables. The values of the MI coefficient are insensitive as well (remember that the MI coefficient is designed for categorical data). We conclude that only MI regression shows different responses to the different datasets and its value is reduced when the relationship between the two variables is not clear. In order to illustrate the behavior of these relationship estimators under the amount of data available we did the following experiment. For a given straight line modeled data by y=mx+c+N(μ=0,σ=1), and we randomly selected 3, 4,...100 data points. Like it is shown in Fig.4.4 for n=3, 5, 10, 20.

Then the values for the different estimators are calculated to show how they evolve when the amount of data points is increasing, the result is shown in Fig.4.5. It can be appreciated how the values of Pearson's and Spearman's fluctuate close to one until they stabilize. The value of the MI coefficient increases indefinitely when data

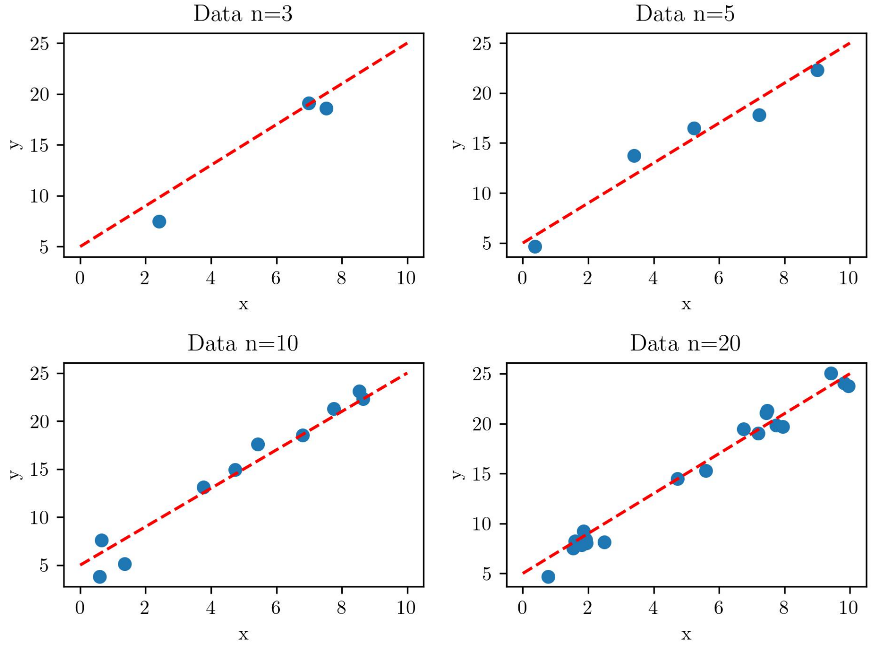


Figure 4.4.: Random samples from y=mx+c+N(μ=0,σ=1) with increasing data size (n= 3, 5, 10, 15).

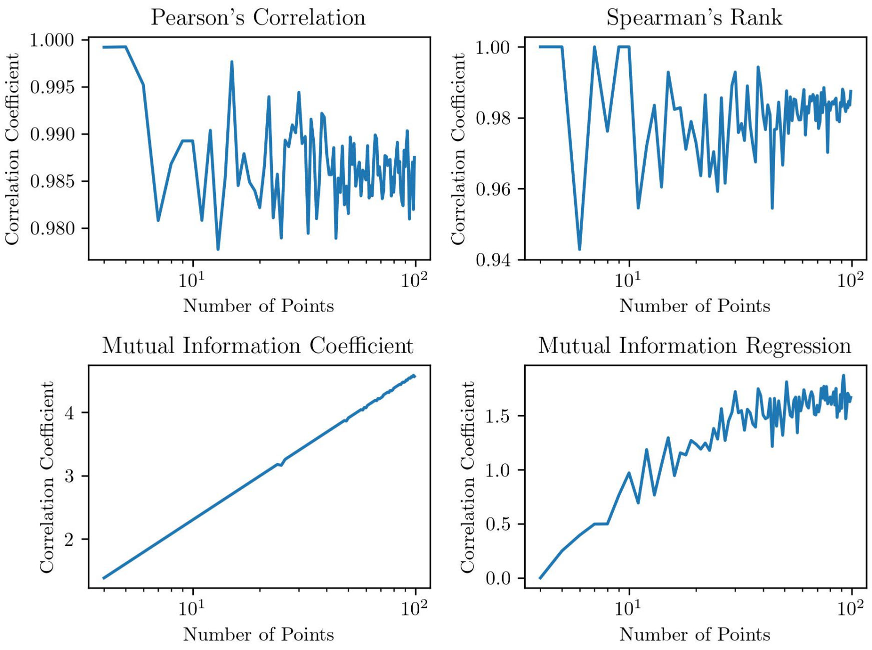


Figure 4.5.: Behavior of correlation measure with increasing data size for methods: Pearson's, Spearman's, and Mutual Information coefficient, and Mutual Information Regression.

is added, we remember that the MI coefficient should not be used with continuous data. On the contrary, MI increases with fluctuations from values close to 0 when the number of data points is small, and then slowly stabilizes. In this case, the MI estimator is biased to 0 when data is scarce. This bias with few data points is not always desirable. However, this bias is very convenient when using data extracted from ERMD modes, as it will discard modes with very few Gabor atoms.

This estimator bias is caused because the values of X and Y can vary in a big range, and when the data is binned the resulting sampling is very sparse. Therefore, if the data volume is not densely sampled enough to show a relationship, then the resulting information value is underestimated.

It is important to note, that in other scenarios different than our experiment, the bias can overestimate the true relationship between variables. For doing this experiment, data points are uniformly sampled from the real distribution. Instead, if the sampling is not uniform the MI estimated can be higher than the true relationship, which would be inconvenient for our use case.

MI is known as “generalized correlation” [Mur22](pages 213ff) as it is capable of capturing linear and nonlinear dependencies on continuous and categorical data, with better resistance to outliers, and fewer assumptions. However, MI does have limitations though. It does not distinguish between positive and negative correlation (negative or positive slope sign). In addition, the the MI value depends on the chosen method of estimation, and this is an active area of research (compared with Pearson’s and Spearman’s whose estimation is well established).

It is commonly known that correlation does not imply causation; however, correlation does not even imply correlation in many circumstances. As we could see in Anscombe’s quartet discussion and our experiment with varying data sizes. Estimating the relationship between variables is a complex task that should be done with caution, because in a noisy dataset, always is possible to fit a linear model. A good practice is to try and compare different methods.

In this work, we have used Pearson’s correlation and Mutual information to determine the Alfvénic character, and we explored the possibility of mixing the two methods. Then, MI is used to explore general relations between the mode activity and the behavior of plasma signals.

It is important to remark that there is no perfect or better method for estimating relationships between variables (“There is no free lunch”). Choosing a method depends on the data, and use case. However, we can conclude that MI captures the more general (and deeper) relationship between random variables, which is always useful in mining big datasets without many prior assumptions. Besides, Pearson’s correlation is a good method when the linear relationship is known to exist for the given dataset. In table 4.1 the advantages and disadvantages of the methods have been summarized.

A few final words of caution are needed when determining the relationship between variables that include different groups. If the variables have different groups, the correlation value can vary and even be contradictory when the separation between groups is not considered. This phenomenon is known as Simpson’s paradox and it is best illustrated with an example. In Fig.4.6 a synthetic dataset with 3 different

Method	Strengths	Weaknesses
Pearson's	Easy to interpret (values between -1 and 1) Widely used and understood.	Sensitive to outliers Assumes linearity (only captures linear relationships). Not suitable for non-numeric data.
Spearman's	Less sensitive to outliers than Pearson's correlation. Applicable to ordinal categories.	Weaker interpretation of strength/direction compared to Pearson's. Does not capture strength of nonmonotonic relationships as well as MI.
Mutual Information	Captures any kind of statistical dependence (linear or non-linear). Less sensitive to outliers than Pearson's correlation. Works with non-numeric data.	Harder to interpret (unitless value - no direction). Can be computationally expensive (for high-dimensional data) Bias with binning methods.

Table 4.1.: Comparison between methods for estimating correlation

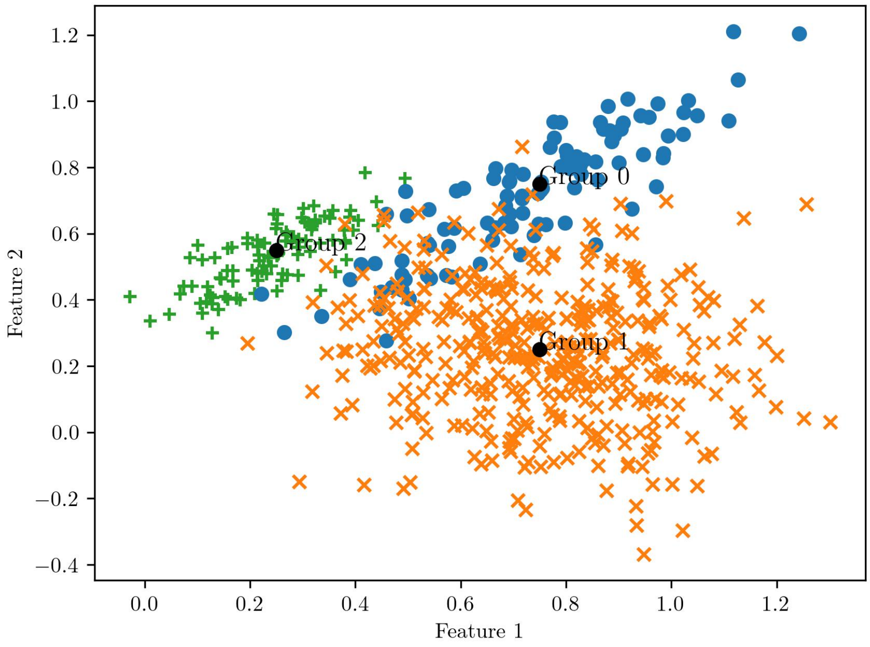


Figure 4.6.: Synthetic dataset with 3 Gaussian blobs non-linearly separable

Gaussian blobs is created, they have different densities and orientations, and the true centers of the groups are marked with black dots.

The grouping in Fig.4.6 is not known a priori, therefore a cluster analysis is needed to estimate the shape of the mixed groups (k-means for this example). Often the variables in the dataset are not enough to separate the groups, the result of clustering in Fig.4.7 shows some misclassified points between group borders. However, the cluster result is accurate enough to estimate the correlation for each group.

If the correlation is calculated using all points, a negative value $\rho = -0.16$ is obtained (the corresponding fitted line is displayed in Fig.4.7 with a red dashed line). In contrast, the correlation values for groups 0 and 2 have positive signs and values of 0.88 and 0.078 respectively (represented by the blue and green solid lines). This contradictory

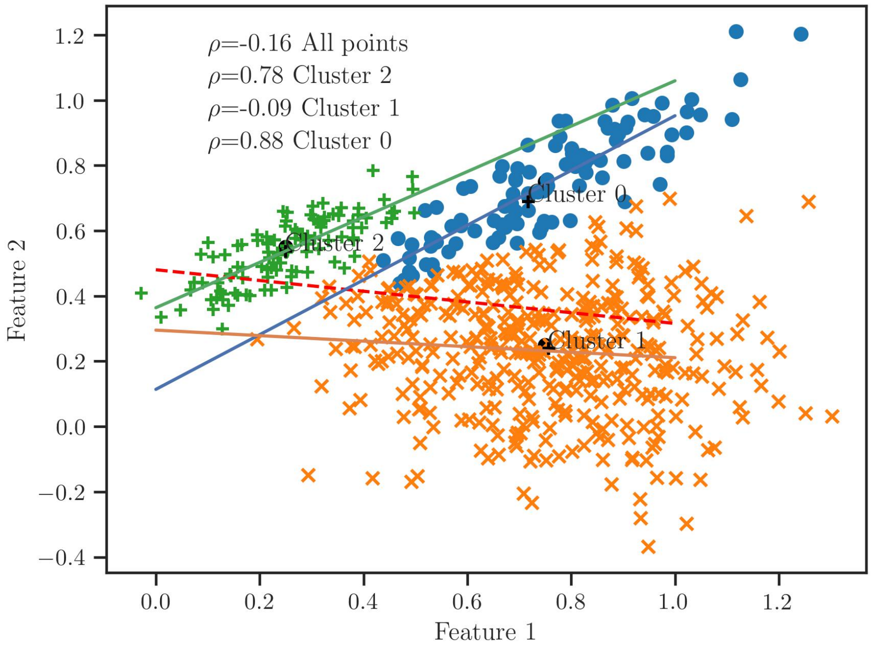


Figure 4.7.: Result of Clustering and Variable relationship using Pearson's correlation. The dashed line represents the correlation of all groups.

result (change of sign) is known as Simpson's paradox. We can conclude if the cluster analysis does not perform correctly (in other words, if the modes are mixed) there is a danger of obtaining spurious correlations.

4.3. Determining the modes Alfvénic character

As mentioned, Alfvén waves frequency follows the evolution of $1/\sqrt{n}$. In consequence, if there are changes in plasma density there will be changes in mode frequencies. Then, Pearson's correlation or mutual information can be used to determine which modes are Alfvénic.

As we have shown in Anscombe's datasets, Pearson's correlation is very sensitive to outliers and noise. Moreover, by the results shown in the last chapter, we can see that the clustered modes often have outliers. Therefore, the normality assumption and equal noise variance may not hold true, causing problems when using Pearson's model. To avoid false correlations the threshold value can be set high ≈0.9, but Alfvén modes can be lost.

In addition, the plasma conditions can be modified very fast, in this case, Alfvén's instabilities frequency follows the evolution of $1/\sqrt{n}$, with some non-linear distortion. Another effect that might distort the relationship is the fact that the measured density is integrated along one line of sight, and the changes in density might be localized outside this line of sight.

Modes with very few data points (less than 10 points) especially appear in low frequencies with multiple “slopes”, when part of the background noise is encoded by SRMD. This can cause many modes to be determined as Alfvénic just by chance (by random alignment with $1/\sqrt{n}$ fluctuations).

To summarize, the potential problems are outlier sensitivity, non-linear distortion, and false positives caused by modes with very few data points. To cope with these problems, we make us of mutual information (MI) to determine if the frequencies are related to the density variation.

Mutual information ranges from 0 to ∞, 0 meaning full statistical independence, then the higher the value the stronger the dependence. However, as MI does not have a sign, it cannot capture the directionality (or “slope sign”) of the relationship. Consequently, MI can not distinguish between Alfvénic ($f \propto 1/\sqrt{n}$) and Anti-Alfvénic ($f \propto -1/\sqrt{n}$.), so in this work, we propose a mixed approach.

The directionality can be estimated by different methods, for instance working out the average frequency-slope of a mode $\Delta f = f_{max} - f_{min}$ $\Delta t = t_{max} - t_{min}$ and taking the sign of $\Delta f / \Delta t$. Given that we have already calculated Pearson’s correlation, and we have its sign: we take the sign from Pearson’s correlation, to determine the Alfvénic character (or Anti-Alfvénic).

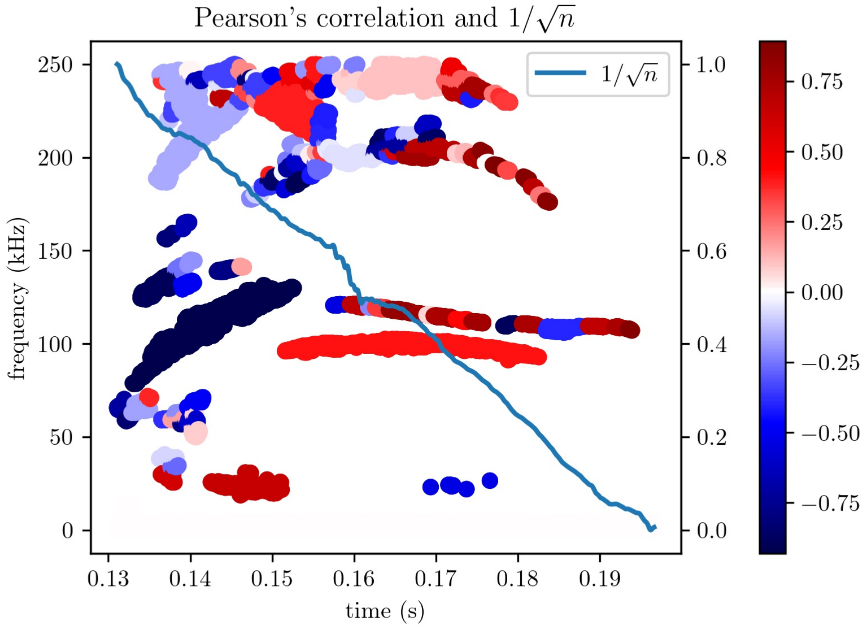


Figure 4.8.: Pearson correlation of modes frequencies and density

As we can see in figures 4.8 and 4.9 the results of using MI and Pearson’s correlation are comparable. In both cases, at time 0.15 s we identify a possible Alfvén mode (dark-red) crossing other modes at ≈225 kHz approximately, and another one at ≈35 kHz. However, at time 0.17 s, subtle differences can be appreciated. For instance, at a time 0.17 s and ≈100 kHz, there is a mode that shows mild correlation but high mutual information. This can be caused by the bias of MI when there are a high number of observations, it tends to overestimate the relationship between variables. There is

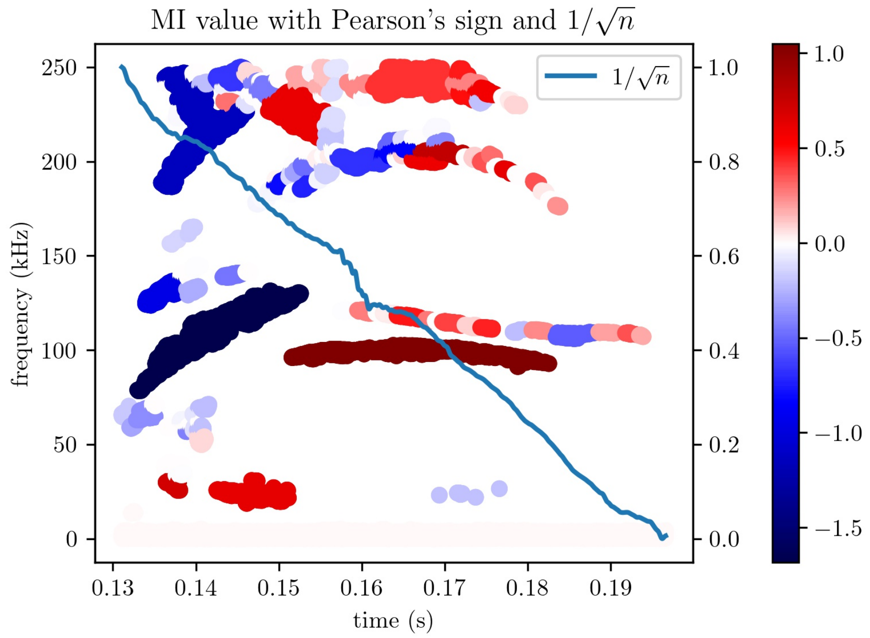


Figure 4.9.: MI of modes frequencies and density.

some level of relationship expected only by chance. When the number of observations is small, the opposite might occur. MI could underestimate relationships.

As discussed in the experiment varying the data size, since MI is more robust to noise and needs more data to estimate a relation, it can avoid establishing a relationship with few and noisy observations. This can be appreciated on the long mode at ≈120 kHz from 0.15 to 0.19 s in Fig. 4.9, which is broken into smaller density groups by the encoding. It can be seen in this case that Pearson's values are higher than MI, but there is not a clear relationship with $1/\sqrt{n}$.

This example from shot 38399 was selected to show how complex can be encoding and finding Álfvén eigenmodes, in other shots this relationship is clearer. Selecting a different value cut to consider modes Alfvénic significantly varies the result. Therefore, to compare both methods, we show the result for the same threshold value of 0.9 in Fig.4.10 and Fig.4.11.

In total, when using Pearson's correlation method, 1315 modes are retrieved. While using the MI with sign method recovers 1433 Alfvénic modes, 8.9% more modes. This can be appreciated in Fig.4.10 and Fig.4.11. This result gives us an average of Alfvénic mode 1.1 per shot. On the other hand, if we examine the negative class we have 35472 (Pearson's) and 35354 (MI) non-Alfvénic modes respectively. As we can see, the dataset is heavily imbalanced towards the negative class, only 3-4 % of the mode structures are Alfvénic. Given this imbalance of classes, an 8.9% more Alfvénic modes is a relevant result. Because each mode counts for training models in imbalanced datasets. Therefore, because of its better performance in handling noisy false positives, and keeping a higher atom count in the frequency interval, the MI method was selected to

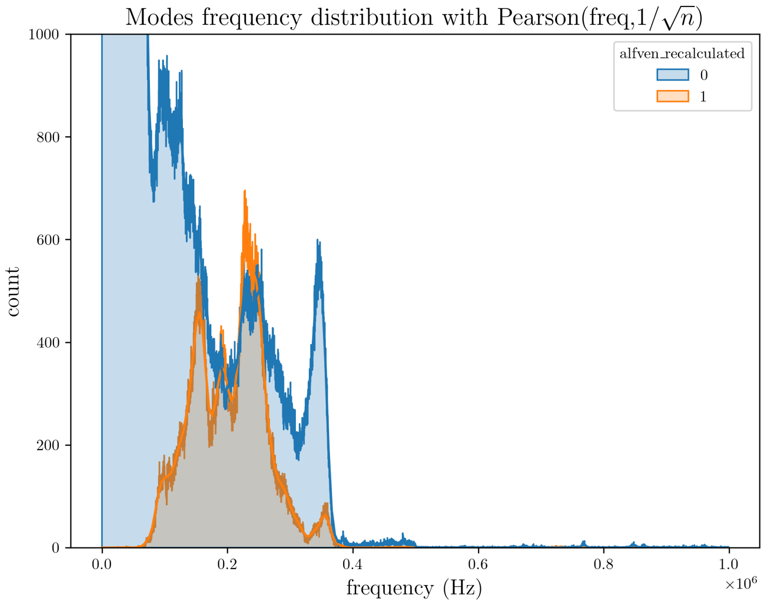


Figure 4.10.: Distributions of Alfvénic atoms frequencies using Pearson's for labeling

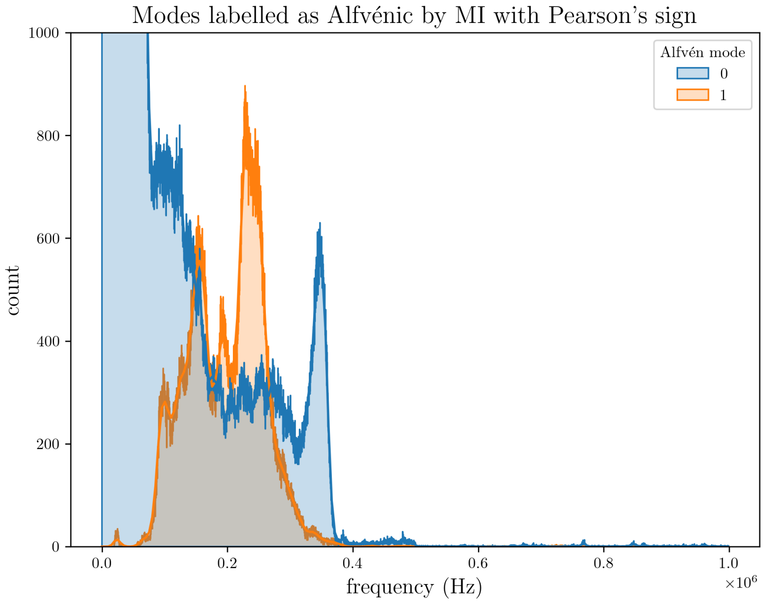


Figure 4.11.: Distributions of Alfvénic mode frequencies using MI for labeling

continue the analysis.

4.4. Finding relationships between categorical variables

Before estimating relationships between mode frequency and other plasma signals in the created dataset of 1291 shots, (next section 4.5). We need to explain how to estimate relationships between categorical variables using MI and indicator functions.

It is possible to create categorical variables to analyze the relationships between continuous variables. Transforming continuous variables into categorical is a common practice in business intelligence. It serves as an additional descriptive effort of the results of ERMD.

The analysis method is illustrated with a toy example first. In Fig. 4.12 a Gaussian distribution of points is shown. To transform x and y to their categoric versions respectively, we create a threshold value of 0.5 for both variables. This divides the dataset into 4 regions indicated by black dashed lines. Hence, the first category has a value of 1 if x is greater than 0.5, and it is valued as 0 when the first category is less than 0.5, the same can be done with the second category. This type of function is known as the indicator function, which is noted as $\mathbf{1}_{x>0.5}$ and $\mathbf{1}_{y>0.5}$ for each feature.

Once we have two categorical variables, contingency tables (or crosstabulations, or cross-tables) can be calculated, as shown in Fig. 4.12. The table has been colored with a heatmap. The contingency table just resumes all possibilities counting the number of samples (absolute frequency) for each one.

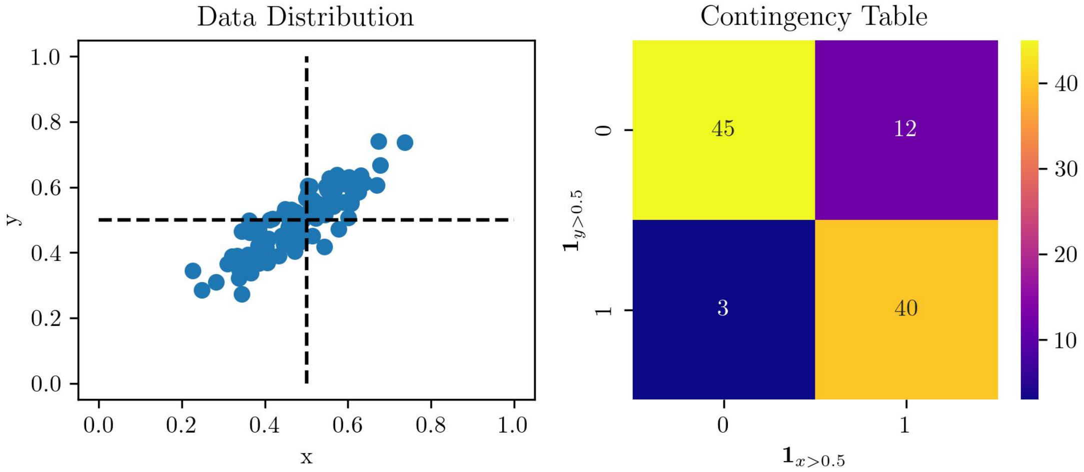


Figure 4.12.: Example of a contingency table for a data distribution with correlated variables

The numbers in the table of Fig . 4.12, show the count of data points belonging to the 4 possibilities. The normalized contingency table can be calculated as well, dividing by 100 (the total number of data points in this case is $N = 40 + 12 + 3 + 45 = 100$), then we obtain the relative frequency of dataset points with the respective values of the categories. If we interpret these relative frequencies as probabilities, we can see that two corners are more probable than the others. This asymmetry indicates a relationship between the categories.

In order, to have a quantification we can run a χ^2 test or we can estimate the MI, using the KL-divergence which in the case of categorical variables, the integrals in Eq. 4.8 are replaced by a sum like

$$\mathrm{MI}(X;Y) = \mathrm{D}_{KL}(p(i,j)||p(i)p(j)) = \sum_i \sum_j p(i,j)\log_2\left(\frac{p(i,j)}{p(i)p(j)}\right). \tag{4.9}$$

	$x <= 0.5$	$x > 0.5$	Row Sum
$y > 0.5$	45	3	48
$y <= 0.5$	12	40	52
Column Sum	57	43	100

The marginal probabilities can be calculated with the rows and columns summations: $N_{1,:}$ first row, second row $N_{2,:}$, first column $N_{:,1}$ and second column $N_{:,2}$.

$$N_{1,:} = 45 + 3 = 48, \quad N_{2,:} = 12 + 40 = 52$$

$$N_{:,1} = 45 + 12 = 57, \quad N_{:,2} = 3 + 40 = 43$$

Then, the joint and marginal probabilities are:

$$P(y > 0.5, x <= 0.5) = \frac{45}{100} = 0.45, \quad P(y > 0.5, x > 0.5) = \frac{3}{100} = 0.03$$

$$P(y <= 0.5, x <= 0.5) = \frac{12}{100} = 0.12, \quad P(y <= 0.5, x > 0.5) = \frac{40}{100} = 0.4$$

$$P(y > 0.5) = \frac{48}{100} = 0.48, \quad P(y <= 0.5) = \frac{52}{100} = 0.52$$

$$P(x <= 0.5) = \frac{57}{100} = 0.57, \quad P(x > 0.5) = \frac{43}{100} = 0.43.$$

Note that if we use the indicator function notation we would have:

$$P(y <= 0.5, x <= 0.5) \equiv P(\mathbf{1}_{y>0.5} = 0, \mathbf{1}_{x>0.5} = 0) \sim P(0,0),$$

$$P(y <= 0.5, x > 0.5) \equiv P(\mathbf{1}_{y>0.5} = 0, \mathbf{1}_{x>0.5} = 1) \sim P(0,1),$$

$$P(y > 0.5, x <= 0.5) \equiv P(\mathbf{1}_{y>0.5} = 1, \mathbf{1}_{x>0.5} = 0) \sim P(1,0),$$

$$P(y > 0.5, x > 0.5) \equiv P(\mathbf{1}_{y>0.5} = 1, \mathbf{1}_{x>0.5} = 1) \sim P(1,1).$$

Now, by using Eq. 4.9 we can estimate the MI for the indicator functions:

$$\begin{aligned}I(X;Y) = P(y > 0.5, x <= 0.5)\log_2\left(\frac{P(y > 0.5, x <= 0.5)}{P(y > 0.5)P(x <= 0.5)}\right)\\ + P(y > 0.5, x > 0.5)\log_2\left(\frac{P(y > 0.5, x > 0.5)}{P(y > 0.5)P(x > 0.5)}\right)\\ + P(y <= 0.5, x <= 0.5)\log_2\left(\frac{P(y <= 0.5, x <= 0.5)}{P(y <= 0.5)P(x <= 0.5)}\right)\\ + P(y <= 0.5, x > 0.5)\log_2\left(\frac{P(y <= 0.5, x > 0.5)}{P(y <= 0.5)P(x > 0.5)}\right).\end{aligned} \tag{4.10}$$

By substituting the values we finally obtain:

$$\begin{aligned}I(X;Y) = 0.45\log_2\left(\frac{0.45}{0.48 \times 0.57}\right)\\ + 0.03\log_2\left(\frac{0.03}{0.48 \times 0.43}\right)\\ + 0.12\log_2\left(\frac{0.12}{0.52 \times 0.57}\right)\\ + 0.4\log_2\left(\frac{0.4}{0.52 \times 0.43}\right) = 0.419 \quad \text{bits.}\end{aligned} \tag{4.11}$$

It is to be noted, that MI is not defined when some joint probability is 0, as $log(0)$ is undefined. However, it is common practice to consider the limit $\lim_{x\to 0} x\log(x) = 0$, consequently ignoring these terms to perform the calculations. This way MI can be estimated when some value in the contingency matrix is 0. In addition, we can distinguish now that KL-divergence is not symmetrical, marginal distributions equal to 0 create more serious problems as the $\lim_{x\to 0} x\log(x) = \infty$, if the event is impossible, the surprise has to be infinity.

The reader might have realized already that the threshold selection introduces a bias. This bias phenomenon is exactly what occurs when continuous variables are binned to work out the integrals in Eq. 4.8. In this toy example, we have binned the variables x, y in 2 bins respectively, we could take more bins but the algebra would be too tedious to serve as a demonstration. If we use the mutual information regression designed for continuous variables on the data x, y we obtain a value of 0.699 nats (or 1.008 bits).

4.5. Estimating relationships between mode frequency and other plasma signals

Revisiting the plasma signals, for each mode we can define indicator functions that revealing if the mode frequency is correlated with the density $MI(f, 1/\sqrt{n})$, current $MI(f, I_p)$, or the magnetic energy $MI(f, W)$.

To be more confident that a relationship exists we can select a threshold value close to 1 (0.9). Then for each mode, we have binary categories: is the mode frequency correlated with the density $1/\sqrt{n}$? (Yes or No); Is the mode frequency correlated with the current I_p? (Yes or No); Is the mode frequency correlated with the energy W? (Yes or No).

The mathematical notation to express each indicator function is: $\mathbf{1}_{MI(f,1/\sqrt{n})>0.9}$(modes), $\mathbf{1}_{MI(f,I_p)>0.9}$(modes) and $\mathbf{1}_{MI(f,W)>0.9}$(modes), in a shorter notation; $\mathbf{1}_{1/\sqrt{n}}$, $\mathbf{1}_{I_p}$, $\mathbf{1}_W$.

The resulting contingency table (Table. 4.2) for the modes features, shows the multivariate frequency distribution of the Alfvénic mode class, quantifying the dependence on these new categorical variables, and interpreting relative frequencies as a probability if the table is normalized.

It is to be noted that, all modes labeled as noise by HDBSCAN (-1) are discarded. Then, MI is used to establish the relationship between the modes frequency chirp f and other diagnostic signals like I_p, W, $1/\sqrt{n}$. Once MI is calculated, if the value is greater than 0.9 the indicator variables $\mathbf{1}_{1/\sqrt{n}}$, $\mathbf{1}_{I_p}$ and $\mathbf{1}_W$, are set to 1, otherwise to 0: for all atoms in the mode. Finally, the table needs to be grouped to count modes instead of atoms if necessary. The contingency table can be calculated indicating the relative frequency of having an Alfvénic mode related to the given plasma signals.

If we study the result in Table 4.2 we can count 1433 Alfvénic modes in total, and 35354 non Alfvénic. This analysis incorporates all frequencies, and due to the need for numerous low-frequency components to describe the signal, 35,111 low-frequency modes are unrelated to any variables. We note that if there is no correlation with $\frac{1}{\sqrt{n}}$, no Alfvénic mode count is observed, as expected. Meanwhile, 515 Alfvénic modes are related to variations of the density exclusively. Remarkably, 533 Alfvénic modes are related to changes in plasma energy as well, and 308 Alfvénic modes are related to changes in all variables; density, plasma energy, and current simultaneously.

It is worth mentioning that the χ^2 statistical test can be used to find evidence of the relationships of variables in the cross-table, rejecting the null hypothesis in all combinations. However, this dependence is obvious if we observe the large difference between the values of the first and second columns. We can conclude that there is a strong relationship between the Alfvén class variable and all the plasma signals considered.

$\mathbf{1}_{1/\sqrt{n}}$	$\mathbf{1}_{I_p}$	$\mathbf{1}_W$	Alfvén=0	Alfvén=1
0	0	0	35111	0
0	0	1	44	0
0	1	0	78	0
0	1	1	22	0
1	0	0	46	515
1	0	1	13	533
1	1	0	26	77
1	1	1	14	308

Table 4.2.: Alfvénic modes' contingency table

If the low frequencies are included, $\mathbb{P}(\text{Alfvén} = 0 | \mathbf{1}_{1/\sqrt{n}}, \mathbf{1}_{I_p}, \mathbf{1}_W) = 0.95$, but the remaining 5% is distributed over 15 possibilities, which difficults the interpretation of the result.

Next, if we examine the Table 4.3 of relative frequencies (in %), ≈6% of modes over 100 kHz are Alfvénic modes related to the plasma energy, and 6.4% of modes are Alfvénic modes related to the plasma current as well.

$\mathbf{1}_{1/\sqrt{n}}$	$\mathbf{1}_{I_p}$	$\mathbf{1}_W$	f(Alfvén=0) (%)	f(Alfvén=1) (%)
0	0	0	80.168150	0.000000
0	0	1	0.519288	0.000000
0	1	0	0.927300	0.000000
0	1	1	0.272008	0.000000
1	0	0	0.408012	5.984174
1	0	1	0.123640	6.441642
1	1	0	0.272008	0.952028
1	1	1	0.160732	3.771019

Table 4.3.: Normalized contingency table for modes over 100kHz in (%)

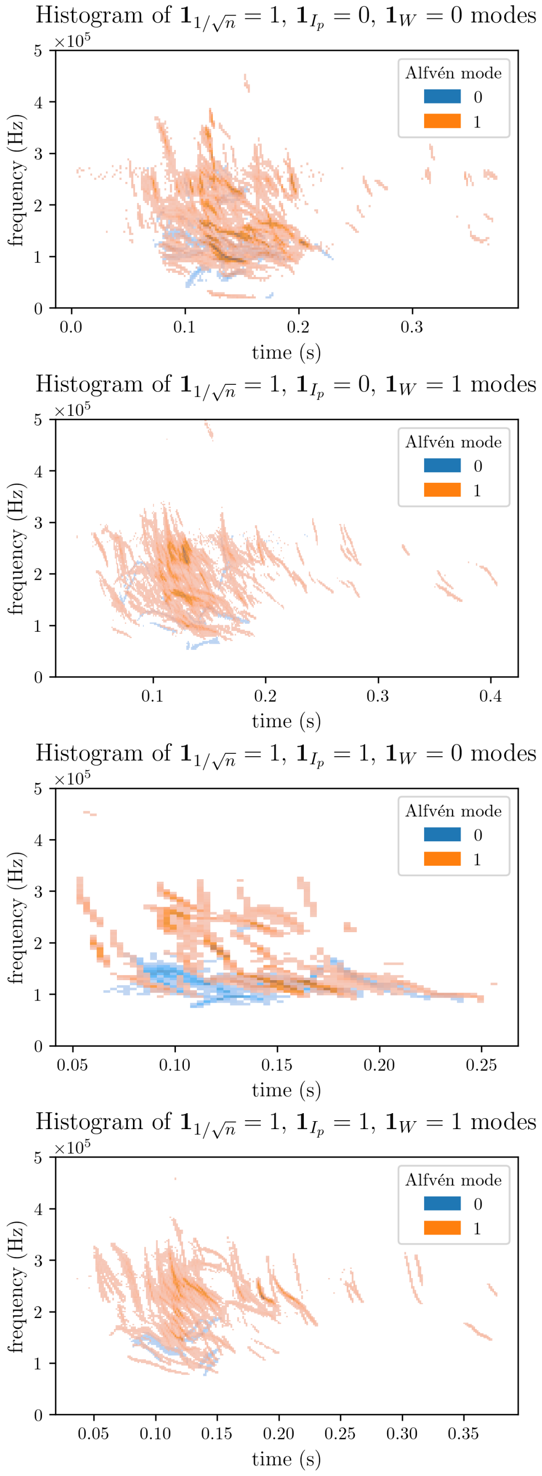
By using MI to construct the categorical variables, linear and nonlinear relationships between variables can be detected. However, when using MI to establish relationships, the positive or negative relation is not captured. Capturing nonlinear relationships is more general and therefore more important than knowing the sign of the relationship when studying modes in all shots. Moreover, the sign of the relationship can be calculated later.

We can conclude that there is a strong nonlinear dependence between mode frequency and the plasma signals, because most Alfvénic modes in this dataset are strongly driven when the heating systems are connected. If the plasma is denser, it can potentially store more energy and drive more current.

4.6. Profiling TJ-II modes

Once the atoms representing the Mirnov coil signal have been conveniently extracted and clustered into modes, a second clustering can be applied to extract higher-level patterns. The plasma signals used in addition to the frequency of the modes are the current I_p, the magnetic energy W, and the inverse square root of density $1/\sqrt{n}$. An example of these signals is plotted in Fig.3.21. Profiling each cluster will help to understand the result of the ERMD run, and the nature of the collected modes. The clustering method selected is K-means, which differs from HDBSCAN. K-means is based on minimizing within-cluster variance, it does not use density like HDBSCAN, and it has great scalability with the number of samples [AV07].

If we project the modes obtained by ERMD into a lower dimensional embedding, using the *T-distributed Stochastic Neighbor Embedding* (T-SNE) algorithm [MH08],



Histogram of $\mathbf{1}_{1/\sqrt{n}} = 1,\ \mathbf{1}_{I_p} = 0,\ \mathbf{1}_W = 0$ modes

Histogram of $\mathbf{1}_{1/\sqrt{n}} = 1,\ \mathbf{1}_{I_p} = 0,\ \mathbf{1}_W = 1$ modes

Histogram of $\mathbf{1}_{1/\sqrt{n}} = 1,\ \mathbf{1}_{I_p} = 1,\ \mathbf{1}_W = 0$ modes

Histogram of $\mathbf{1}_{1/\sqrt{n}} = 1,\ \mathbf{1}_{I_p} = 1,\ \mathbf{1}_W = 1$ modes

Figure 4.13.: Modes of contingency table 4.3

we can mark the cluster numbers and visualize which groups are similar by their separation distance and neighborhood.

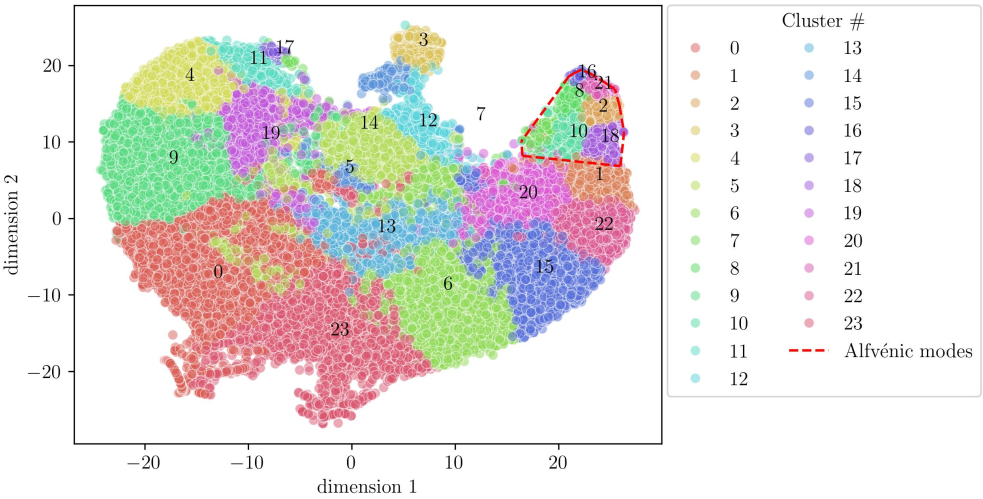


Figure 4.14.: T-SNE embedding of mode features. Clusters 2, 21, 16, 8, 10, and 18 are mostly Alfvénic modes the rest can be noise or a mix of different modes.

The features used for the projections are the same as the features used for clustering: the mutual information (MI) of the mode's frequencies with the density MI(f,$1/\sqrt{n}$), current MI(f,I_p), and energy of the plasma MI(f,W). In stellarators, and particularly in TJ-II, small changes in plasma current can have a measurable impact on the spectrum of shear Alfvén waves and therefore, evaluating the correlations between changes in mode frequency and plasma current is necessary. In addition, the modes's frequency chirp $\Delta f = f_{\max} - f_{\min}$, the variation of density $\Delta(1/\sqrt{n}) = (1/\sqrt{n})_{\max} - (1/\sqrt{n})_{\min}$, the variation of current $\Delta I_p = I_{p\,\max} - I_{p\,\min}$, and plasma energy $\Delta W = W_{\max} - W_{\min}$ are included to give directionality and magnitude of parameters change.

The result of T-SNE is shown in Fig.4.14 all detected modes are represented in a two-dimensional space. The numbers of the labels are positioned in the plot by taking the mean values of dimension 1 and dimension 2 in each group. Finally, the class label that results from the K-means clustering is used to assign a color to the projection points and interpret the projection result.

The interpretation of the cluster analysis is summarized in the table 4.4. For the sake of clarity, we provide some examples of clusters in Fig.4.15. We can observe that the clustering algorithm is capable of separating the Alfvénic class from other types of modes with the given variables. The region of Alfvénic modes is highlighted with a dashed red line polygon, in the upper right corner of Fig.4.14. Groups 8, 16, 21, and 2 are mostly composed of Alfvénic modes, whereas clusters 10, 18, and 7 have a mixed composition, as their position is close to the border of the polygonal region.

The groups neighboring cluster 0 in the opposite region to the Alfvénic class are noisy. We consider noise broadband frequency structures mislabeled in some cases by ERMD. In addition, we consider as noise the very low-frequency structures misidentified by modes that can be seen in Fig.3.22, which also occurs when broadband noise is present. The MI values of these groups are close to 0, as they do not have

Cluster Number	Interpretation
8, 16, 21	Strongly driven Alfvénic modes occurring when heating is engaged
2	Alfvénic modes correlated with magnetic energy
7	Mix of Alfvénic and Non-Alfvénic modes around 100 kHz correlated with current
18	Mix of Alfvénic and Non-Alfvénic modes (probably False negatives)
10	Mix of Alfvénic and Non Alfvénic modes
17	Anti-Alfvénic modes strongly correlated with current
1	Non Alfvénic modes and mid-frequency noise
4-5, 9, 11-13, 19-20, 22-23	Noise + low-frequency modes (No-Alfvénic)
0, 3, 6, 12, 14-15	Broadband and low-frequency noise produced by heating systems

Table 4.4.: Clusters interpretation

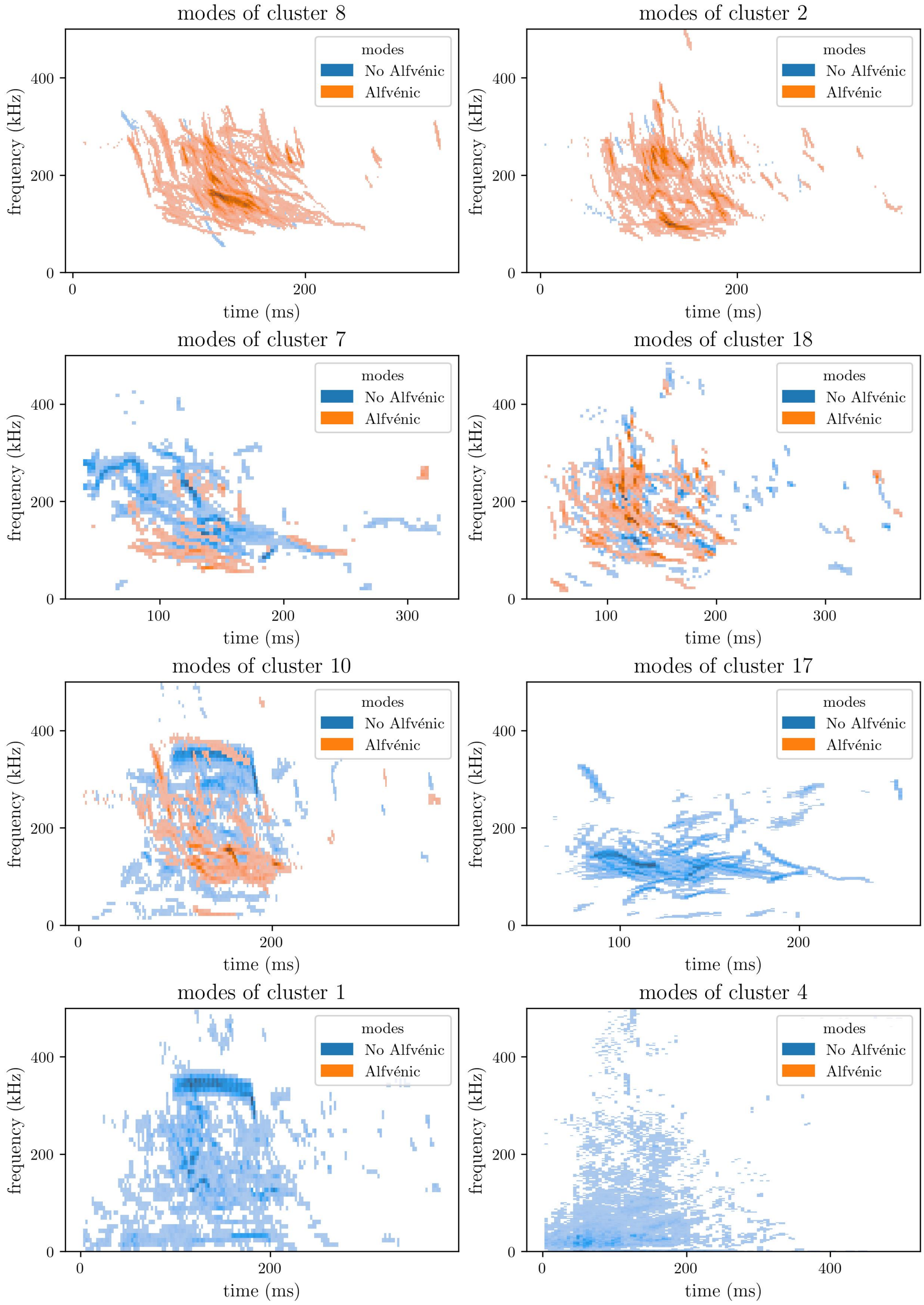


Figure 4.15.: Time-frequency histograms of mode distributions from different clusters.

any relationship with fluctuating plasma signals, which helps discard errors of the algorithm.

If we look at the time-frequency distributions of modes in Fig.4.15 we can appreciate variate behavior. Note that the intensity of color gets very high in some values due to the presence of repeated shots. Moreover, even if the time variable has not been used, the modes appear to be grouped around specific times at the beginning of the pulse when NBI is engaged. We can observe as well that the less Alfvénic activity, the more probable the presence of noise. Though some noise might be present, the identification of Alfvénic activity is satisfactory. In addition, some clusters appear to have different natures, like clusters 7 and 17, in which modes are correlated with the current, in this case, modes appear to have lower frequencies.

It is of interest that this clustering technique can be used for looking for patterns in shot databases and eventually finding unexplored plasma activity. However, a detailed physical interpretation of each cluster requires further analysis based on complementary simulations and first physical principles. For instance, once the presence of modes is detected it is possible to run equilibrium reconstruction and stability analysis to determine the physical nature of the modes. This could add labels like HAE, TAE, GAE or MIAE to the clusters groups. The mode identification problem in TJ-II from the physics point of view has been addressed in [VSG17; SOL15; Cap+21].

It is worth mentioning that the MI information can be replaced by Pearson's correlation or even combined, and the results of the cluster are comparable. Still, we prioritize the use of MI as it is capable of capturing nonlinear relationships, and it is more robust to outliers than Pearson's. The interested reader can find an alternative to clustering in Appendix C. It is possible to create categorical features using MI and constructing contingency tables to analyze the mode's relationship with other plasma signals.

Other mode variables can be created and added to the clustering. However, we follow the parsimony principle, i.e. adding variables until the result is interpretable, and then stop including features. The same principle is followed when selecting the number of clusters, i.e. the number of clusters is increased until the granularity is fine enough to describe Alfvénic subgroups.

4.7. Discussion

Comparing the results of the clustering with the results of the contingency tables, by examining the histograms of Fig.4.13, we can find a resemblance with the results of K-means clustering explained in the previous section. In fact, some modes of cluster 2 appear to be correlated with the density only; modes correlated with energy and density appear in clusters 21 and 8 and non-Alfvénic modes of cluster 17 are strongly correlated with the current. However, establishing a complete equivalence between the cluster analysis and the contingency tables of categorical features it is not possible, as we arbitrarily selected a criterion of the category creation.

The treatment of the full variance in the cluster analysis appears more general than the use of contingency tables. In other words, the cluster analysis has finer granularity because the values of MI are not discretized and we included directionality with Δf, ΔI_p, ΔW and $\Delta(1/1\sqrt{n})$.

For future statistical experiments, the MI values between the frequencies and plasma signals can be binned in more categories (for example low, medium, high) instead of using only indicator functions. In addition, other variables like the mode amplitude could be investigated.

The manual creation of features (new variables) is a common practice in the industrial application of machine learning. In many cases, performance gains are better obtained by creating features product of domain knowledge than by optimizing the hyperparameters of more complex models.

We conclude this chapter with a last remark: establishing a relationship between variables with statistics is a complex task that needs to be carried out with caution, and often conclusions are limited. In a noisy dataset, it is always possible to find correlations. However, as a byproduct of this effort, we could jointly visualize the results of analyzing a big dataset of plasma modes, and separate noise from true modes.

4.8. Summary

In this chapter, we discussed how can we estimate if a relationship (correlation) exists between random variables. First, we review the basics of statistics and information theory to explain the trade-offs of using Pearson's correlation and mutual information.

The first goal is to determine which modes retrieved by ERMD are Alfvénic. We explored the possibilities of using a mixed approach, using Pearson's correlation sign and MI strength to determine Alfvenic character.

The second goal was to investigate the relationship between plasma signals like density, current, and magnetic energy with the frequencies of the modes retrieved by ERMD in a TJ-II dataset. In order to do that, we followed two approaches: analysis of contingency tables (previous creation of indicator functions), and profiling of the modes retrieved by ERMD by using a clustering analysis. Thanks to this study we found alternative ways of visualizing MHD activity in a dataset.

5. Spectral feature extraction with signal analysis: Fourier, Wavelet and Hough transform

> *"A picture is worth a thousand words"*
>
> — popular

> *"... a formula is worth a thousand pictures"*
>
> — Dijkstra 1996

Summary

5.1. Introduction

For years, experimental physicist looked at their computer screens to identify plasma instabilities. The researchers scanned the spectrograms for very specific patterns. Therefore, a natural approach to automating the identification of MHD activity would be to use computer vision to recognize the same patterns, imitating this human behavior.

Working with images has several advantages. For instance, time-frequency representations facilitate the study of time signals. In time series patterns the information is so localized that we need to create spaces with additional dimensions to investigate them, passing from 1D to a 2D representation. This contrasts with the usual effort of trying to reduce dimensions.

Images differ significantly from raw time series in their statistical properties. For example, a colored image has three RGB channels, resulting in 5 dimensions (x,y,R,G,B). The number of possible black and white images in a 256x256 pixel space is 2^{65536}, a number with 19729 digits, much greater than the estimated number of atoms in the universe ($\approx 10^{80}$). Moreover, images pose unique challenges such as object recognition, filtering, and generation, each requiring specialized algorithms tailored to the visual domain.

Common tasks in computer vision include manipulating exposure, color, size, and geometry; filtering and restoration (denoising); and detecting edges, ridges, contours, or other geometric shapes. More advanced problems involve object detection, texture analysis, image classification, segmentation, and thresholding. In recent years the development of deep convolutional neural networks has changed the field, achieving unprecedented performance and advanced applications. Although image complexity is very high, having many applications to business and daily life applications, they have been researched and developed quite extensively.

A shared family of algorithms between time signal analysis and image analysis are the Fourier transform and the wavelet transform since they can be extended to 2D easily. An image can be understood as a collection of 1D signals along a horizontal direction and vertical direction.

The fact that plasma phenomena have multiple scales is partially reflected in the spectrogram. MHD modes usually are macroscale perturbations of the plasma, and microscale phenomena translate into broadband noise and other artifacts. Wavelets can be useful in this task since they are designed to handle the presence of different scales in data, allowing the separation of distinct features.

In the previous chapters, we worked to model the signal adding sparsity of the modes as an important model assumption. The sparsity property allows the identification of mode structures embedded in broadband noise. Nevertheless, we need to translate this assumption to an equivalent statement for pixel space. Wavelets can help with this task since sparse mode features will appear with low spatial frequency in the image space, or equivalently: modes will have a larger scale (lower frequency) than noise in the image space.

Because many computer vision algorithms are created with real-time applications in mind, their computational requirements can be very low. GPUs are only mandatory for the training of deep learning models. Having models without GPU requirements can be helpful in the situation of mining big databases.

In this chapter, we explore the application of Fourier transforms, wavelets, and classic computer vision algorithms to identify and remove noise and other unwanted features from JET spectrograms. This preprocessing step is crucial for the subsequent identification of time-frequency structures, which will be discussed in the following chapters.

5.2. Signal analysis for feature extraction of spectrograms

In this section, we briefly revisit the signal processing and computer vision (CV) algorithms introduced in Chapter 2, with a focus on their implementation for feature extraction in spectrograms.

Before continuing, it is important to remark that magnetohydrodynamic (MHD) instabilities like energetic particles (EP) driven Alfvén eigenmodes [Hei08; GPT14], can be observed with different diagnostic systems: magnetic sensors [Sha+13] (Mirnov coils [MS74]), electron cyclotron emission (ECE) [Van+06; Fre+16], soft X-ray measurements [Naz+97], or Far infrared interferometry (FIR) [Sha+06]. Frequently, the end product of the diagnostic analysis is often a spectrogram.

To analyze these signals, we begin by calculating the spectrograms using the discrete short-time Fourier transform (STFT), $P_S f(u,\xi) = |STFT\{f(u,\xi)\}|^2$, as defined by Eq. 2.27. We employ the Scipy implementation with the following settings: a window length of 2048 samples for the FFT, spectral amplitude on a dB scale, the Tukey window function $g(t-u)$, and the "turbo" colormap—a rainbow colormap proposed by Google AI Research for its perceptual uniformity and enhanced value dissimilarities. The remaining parameters use the default library settings. Before applying the processing pipeline, we resize the images to 1024 by 1024 pixels.

Although we defined the DWT as a projection over an orthogonal dictionary basis in Eq. 2.21, discrete wavelets can also be visualized as pyramids of multiscale and multidirectional filters. These filters are extensively used in image processing to remove noise and extract features. By applying the discrete wavelet transform (DWT), an image can be decomposed into four sub-images: an approximation image, vertical details, horizontal details, and diagonal details. The approximation images are used iteratively to continue the decomposition at the next level, as illustrated in Fig. 5.1. An example of spectrogram decomposition is shown in Fig. 5.2. This thesis work used the PyWavelets implementation [Lee+19].

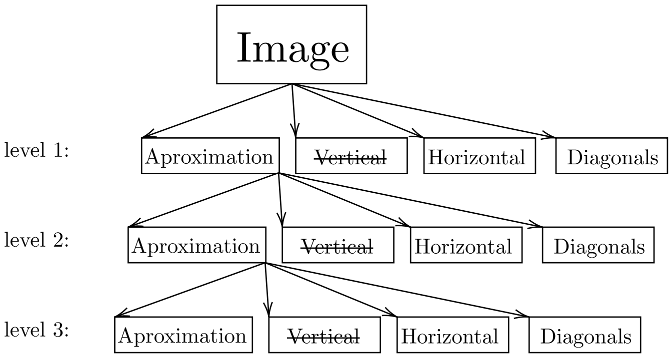


Figure 5.1.: Cartoon showing wavelet cascade of filters; Approximation, Vertical details, Horizontal Details, Diagonal details

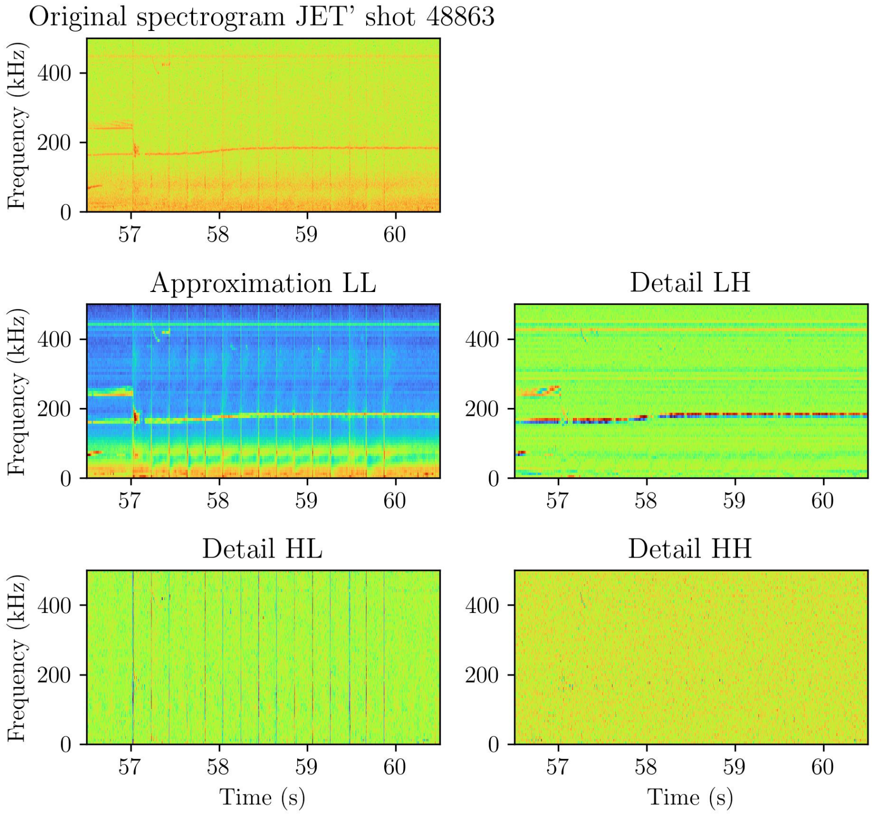


Figure 5.2.: Example of a JET' spectrogram (top) and its third level wavelet decomposition (bottom) Approximation image, horizontal details, vertical details and diagonal details.

The approximation image is essentially the result of applying a low-pass filter to both the rows and columns of the image (hence the notation LL in the literature). The vertical details image is obtained by applying a low-pass filter to the rows and a high-pass filter to the columns (LH), while the horizontal details image (HL) and diagonal details image (HH) complement the decomposition. In this work, the Haar wavelet is employed for these transformations.

The discrete wavelet transform (DWT) is particularly valuable because MHD instabilities, observable in spectrograms, tend to have low spatial frequencies, meaning they exhibit low repetition rates across the spectrogram image. In contrast, signal noise usually appears as high spatial frequencies.

The frequency content of an image can also be analyzed by extending the discrete Fourier transform to two dimensions. This allows for the detection of periodic patterns in the frequency domain. In this thesis, the primary application of the 2D Fourier transform is the manipulation of images in the frequency space, particularly through the filtering of specific spectral components. For example, filtering out certain frequencies can effectively remove periodic patterns [HS]. The 2D Fourier transform implementations used in this work are available in Python's NumPy library.

The discrete wavelet decomposition of images is a powerful tool for denoising and feature extraction [Oro; GKF02; San04; BRH20; KU18; TZ21]. However, the number of directions might be too coarse in some scenarios. For instance, a vertical and horizontal multiscale pattern could not be enough to describe curved ridges filling the image space. A similar situation can occur with MHD activity on spectrograms. Variants of the DWT like the contourlet transform [DV05] or the curvelet transform [Can+06; MP10] were introduced to cope with the lack of directionality of DWT.

The idea of using filter banks in the fast discrete wavelet transform can also be applied to the 2D Fourier space of images. By tessellating the Fourier space into windows, we can create a multi-scale, multi-directional filter bank. The fast Curvelet transform of an image $f(x_1, x_2)$ is defined by Eq. 2.28. However, because translations in the same direction are symmetrical, we can define curvelets informally as a decomposition over scales and angles:

$$\begin{aligned} &\text{Curvelet transform}\{f(x_1, x_2)\} \\ &= \sum_{scales} \sum_{angles} \langle f, \phi_{scale,angle} \rangle \phi_{scale,angle}, \end{aligned} \tag{5.1}$$

where $\langle f, \phi_{scale,angle} \rangle$ is the coefficient corresponding to the “wedge” $\phi_{scale,angle}$ [Can+06].

This work uses CurveLab implementation, copyrighted by the authors [Can+06], and it is available for academic use. The Python library Curvelops allows curvelops integration with the rest of the workflow.

In this thesis, we used the Hough transform [DH72; GMK99] to detect straight lines $y = mx + b$ in images (section 5.4). Without discussing all the details: for each pixel point, multiple straight lines represented by $x\cos(\theta) + y\sin(\theta) = \rho$ are considered. Each one is situated at a distance ρ from the origin and the perpendicular to each candidate line forms a θ angle with x-axis. The image is then transformed into a 2D space (Hough space) in which all straight lines are transformed into sine curves. The accumulation of sinusoids in a point of Hough space indicates the presence of a straight line characterized by ρ and θ (Fig.5.5). It is important to mention that some edge detection (or ridge detection) is needed before running the Hough transform. Otherwise, accumulation points are not clear and the straight-line detection might not be efficient enough. For this reason, the Canny edge detector [Can86] or Ridge detection combined with thresholding have been used in this case.

All these CV algorithms are available in the scikit-image library [Van+14]. Rather than focusing on hyperparameter tuning, we compared different algorithm performances using the default parameters, aiming to identify the ones that best suited our data features.

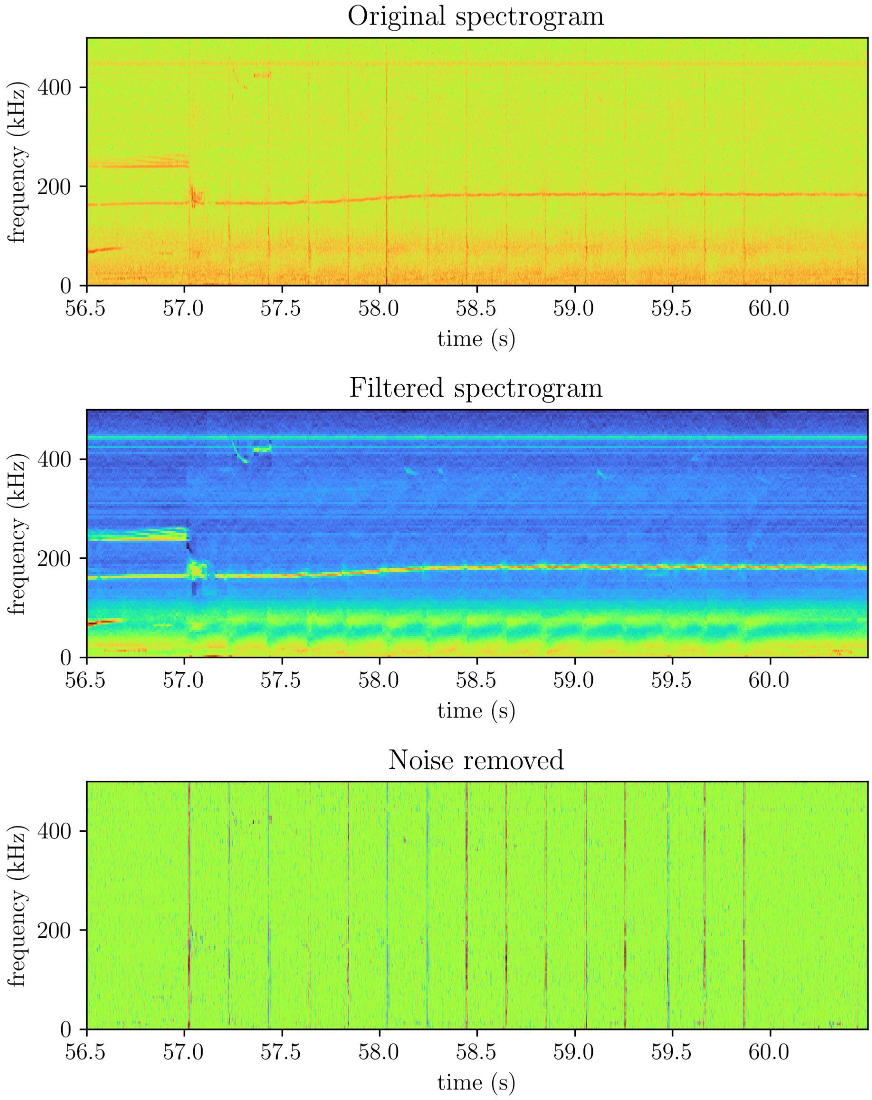


Figure 5.3.: Example of broadband spectral noise removal. Clips caused by pellets injections are removed from the spectrogram, leaving MHD activity. TAEs and EAEs are observed.

5.3. Broadband noise removal using discrete wavelet transforms

In JET shot in Fig.5.3 we can see various toroidal Alfvén eigenmodes over 200 kHz and between 56.5 and 57 s approximately. Then 15 periodic pellet injections fuel the plasma (they can be distinguished by the vertical lines in the spectrogram). The pellet injections fundamentally increase plasma density and reduce its temperature. It can be seen in the first pellet injection just after 57 (s), how they affect the TAE modes. As the density increases abruptly, the TAE frequency $f_{\text{TAE}} \propto 1/\sqrt{n}$ decreases, and then the TAE disappears soon afterward. A similar effect can be observed on an elliptical Alfvén eigenmode at ≈420 kHz and ≈57.5s. More deeply, the effect of pellet injection is a modification of the Alfvén continuum spectrum, which is studied [Oli+19] in shots similar to the one in Fig.5.3.

Time series signals from JET magnetic diagnostics are characterized by abundant spikes and discontinuities. It is well known that the Fourier transform struggles with non-differentiable or non-stationary time series, resulting in Gibbs artifacts in the signal reconstruction and spectral leakage in the frequency domain. A classic example is the Fourier transform of a square signal, where ripples (known as Gibbs phenomenon) are present at the edges of the reconstructed square wave. The Fourier decomposition of sines and cosines inherently struggles to accurately recreate this type of signal. A more dramatic example is the Fourier transform of a unit impulse (Dirac's delta $\delta(\tau = 0)$), which theoretically requires an infinite range of frequencies to reconstruct the pulsed signal. Due to these challenges, JET's time-frequency spectrograms often exhibit abundant broadband noise.

The character of spectrograms noise is highly anisotropic, due to the spectral leakage of the STFFT on each time window. This broadband noise is predominantly located in the vertical details sub-image. Therefore, by removing the vertical details of the image, we can significantly reduce the noise in the spectrograms. When the pixels in the vertical detail sub-images (across all levels) are set to zero and the inverse DWT is applied, broadband spectral noise is substantially diminished. Fig.5.3 demonstrates this process, including the effect of pellet injection events in this particular example. The steps are summarized in Algorithm 3.

The approximation and horizontal details subimages are left unchanged. In addition, some high-frequency noise is present in the diagonal subimages, so they have set a soft thresholding (all pixel values in diagonal subimages are multiplied by $\alpha_D = 0.5$).

Algorithm 3: Remove broadband noise, clics and pops from spectrogram

```
Input: Time series $\boldsymbol{s}$
Parameters: Levels deep: N=8
Vertical threshold: $\alpha_V = 0$,
diagonal threshold: $\alpha_D = 0.5$
Output: Reassigned spectrogram $S_{xx(mod)}$
Steps:
1 Obtain spectrogram matrix $S_{xx}$ using STFT($\boldsymbol{s}$).
2 Workout DWT tree.
3 Set $\alpha_V = 0$ in all vertical details images in levels 1:N (hard thresholding).
4 Set $\alpha_D = 0.5$ in all diagonal details images in level 1:N (soft thresholding).
5 Use inverse DWT to generate a filtered spectrogram.
```

It is important to remark that we are modeling noise through the selection of one direction and the selection of the parameters, therefore it is necessary to check these prior assumptions while using this algorithm. So far we have assumed that all the information contained in the vertical direction is noise or undesired artifacts. However, this is not the case when rapid sweeping frequencies of physical origin like fishbones or energetic particle modes. The algorithm removes these types of modes and the information removed can not be discarded.

Given this case, it is convenient to examine the removed image and evaluate how much information has been removed. Then several approaches are valid: (1) if all vertical modes are removed, it is necessary to continue the analysis on the vertical components, this way vertical and horizontal components can be segmented separately and finally merged together; (2) use soft thresholding to ($\alpha_V = 0.5$ or greater) to adjust the subtracted fraction of the vertical component on proceed with segmentation (these modes are strongly driven and contrast is good to segment them); (3) use the approximation image of level 1 or 2 instead, because keeping one of the first approximation sub-images can also result in a spectrogram with an improved signal-to-noise ratio, though omitting details sub-images could produce modes with coarser details.

The maximum number of levels that can be calculated depends on the number of pixels in the image. More pixels allow for a greater number of levels. Given an optimal time-frequency uncertainty, determined by the window size in the STFFT, one way to increase the number of pixels in the spectrogram is by allowing more overlap in time windows. The maximum number of levels can be determined by the appearance of border effects, and we calculate as many levels as possible.

5.4. Straight line pattern detection

Some signals observed in spectrograms appear as straight lines with specific orientations. In many instances, these signals are not caused by plasma instabilities. For example, the TAE antenna creates a frequency sweeping pattern, which can be seen in Fig.5.4. Shot 92416, an afterglow experiment, has been studied in detail [Fit+22]. To remove the straight line patterns present in this shot, we must generalize the method used for pellet signals to accommodate other orientations beyond the vertical.

```
Algorithm 4: Straight line detection using Hough transform
  Input: $s$ diagnostic signal
  Parameters: N=100000, $\theta_1 = -\pi/2$, $\theta_N = +\pi/2$, $\boldsymbol{\theta_{test}} = [\theta_1, ...\theta_N]$
  Output: $lines$
  Steps:
  Obtain spectrogram using STFT($\boldsymbol{s}$)
  Apply Canny edge detector/Ridge detectors to enhance features
  Use scikit-image implementation of Hough transform (hough_line($\boldsymbol{\theta_{text}}$)) or
    probabilistic_Hough(threshold=85, line_length=12, line_gap=6) for Fig.5.7.
```

It is to be expected that the pellet's injection time or requested TAE antenna frequencies are available in the shot database. However, side lobes (harmonics) of the TAE

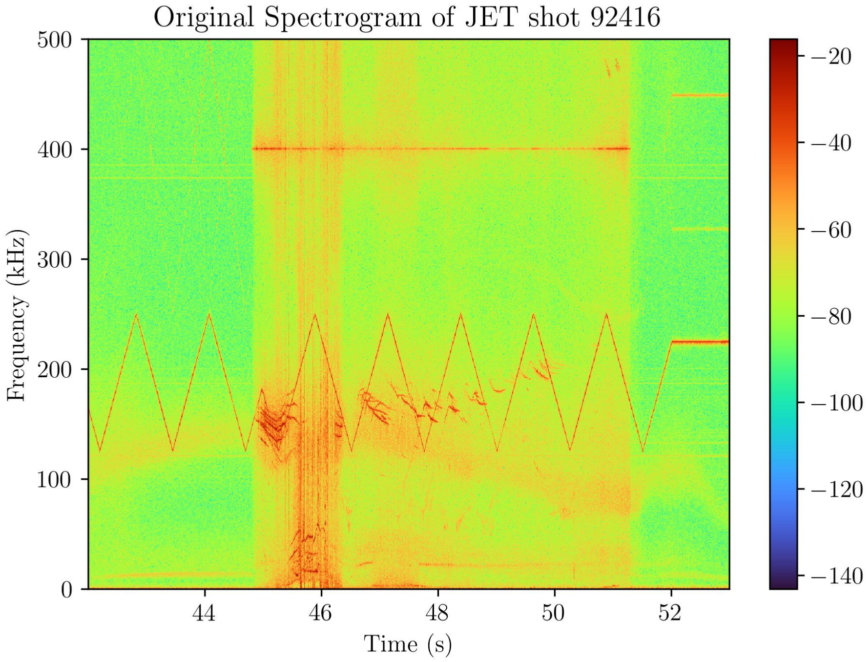


Figure 5.4.: Spectrogram of JET shot 92416.

antenna signal may also appear.

Before applying the Hough transform (Algorithm 4), it is necessary to enhance the edges in the spectrogram image. The Canny edge detector is a standard choice, but better results can often be obtained using ridge detectors, which are discussed in the next chapter. The Hough transform then expresses the image in its own space. The main hyperparameter for the Hough transform is the number of test angles $N = 100000$ that need to be sampled between $\pm\pi/2$. The more angles tested, the better the result, though this comes at the cost of increased computational time. When overlapping sinusoids converge at a point in Fig.5.5, it indicates the presence of a straight line in the image space. Notice how some curves converge to points at ±90 degrees (corresponding to horizontal lines in images) and around ±10 degrees (corresponding to the antenna sweep).

The result of the straight-line detection is shown in Fig.5.6. The angles detected are centered around 0.39 ± 1.63, 77.59± 2.63, 83.1± 1.63, 95.97± 2.01, 103.32± 2.61, and 179.61 ± 1.65 (in degrees). It is also worth noting that the probabilistic Hough transform [GMK99] can be used to incorporate priors such as line length, angular orientation, or separation between lines. The results of applying this technique are shown in Fig.5.7. This result extends the further applications of the Hough transform.

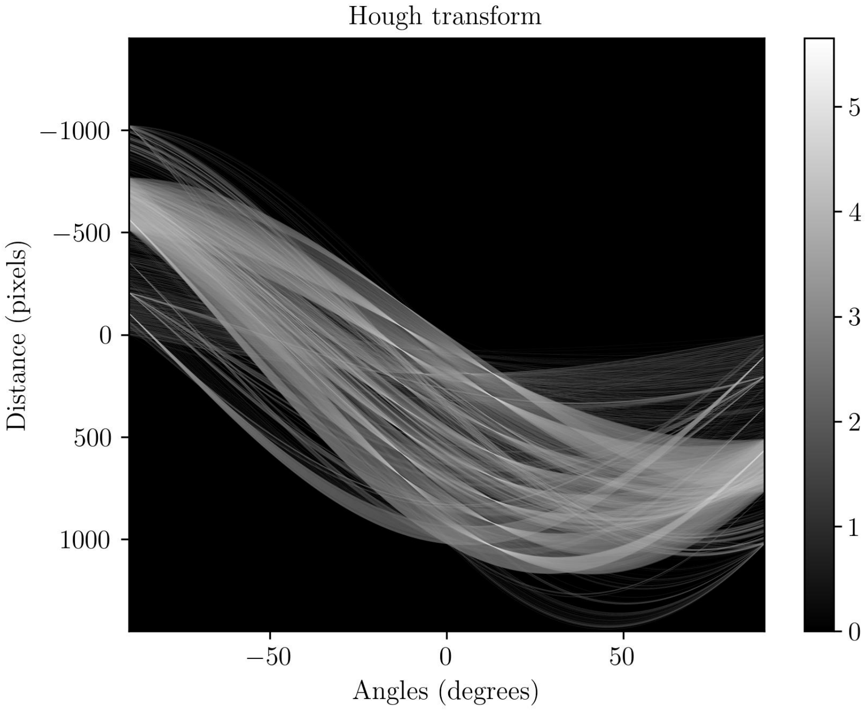


Figure 5.5.: Result of Hough transform. The accumulation of sinusoids passing through a point indicates the presence of a straight line in the image space.

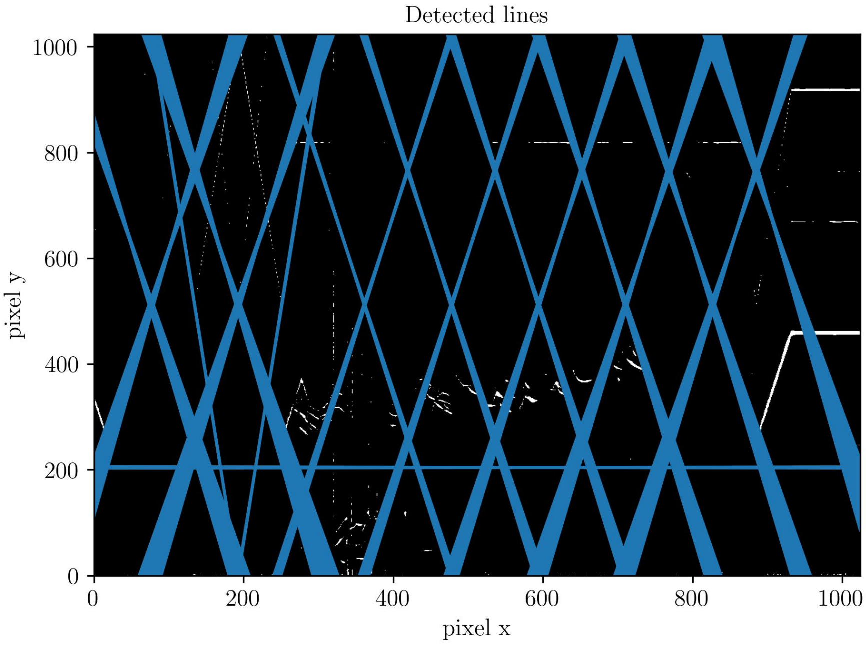


Figure 5.6.: Result of Hugh transform analysis showing the straight lines detected.

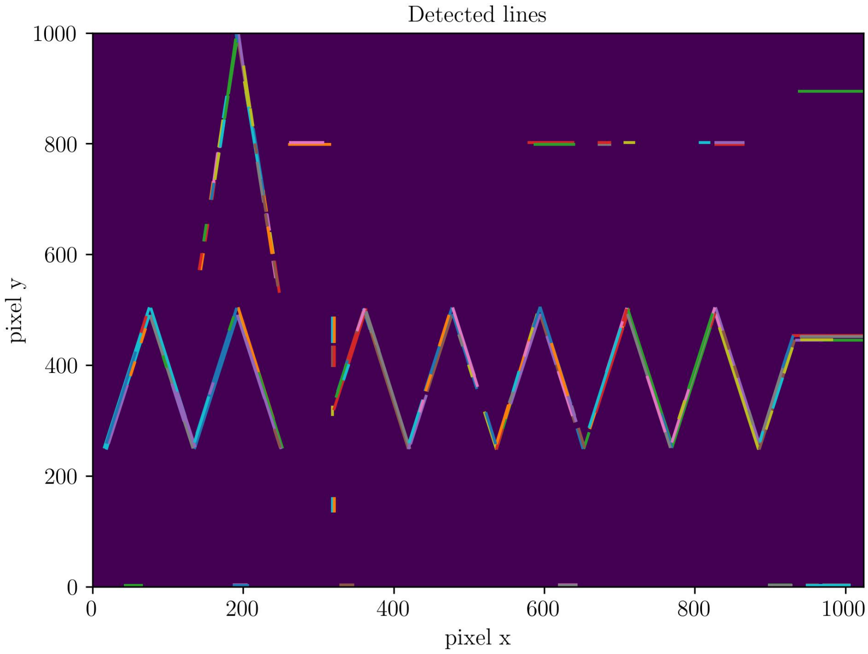


Figure 5.7.: Result of probabilistic Hugh transform. Prior length is 12 pixels.

5.5. Removal of oriented patterns with 2DFFT

To proceed with the removal of oriented patterns, we start with the straight lines detected using the Hough transform. Our first approach would be to transform the spectrogram's image $f(x_1, x_2)$ to the Fourier space $F(k_1, k_2)$. The steps are summarized in the algorithm (5).

In the Fourier space, the image information is expressed as spatial frequencies along two directions. We can then design a mask over the magnitude spectrum to remove frequencies along the desired directions. It is important to note that the FFT2D rotates the straight line angles by $\pi/2$ or 90^o, as detected with the Hough transform. For instance, lines detected at 77.59^o generate features at 77.59^o-90^o = 12.4^o in the Fourier space. The supplementary angles of $\pm 12.4^o$ are 12.4^o+180^o=192.4^o and -12.4^o+180^o=167.6^o.

Therefore, the mask angles used in this example are

$$\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4] = [12.4^o, -12.4^o, 192.4^o, 167.6^o] \tag{5.2}$$

. Then, the angular mask widths are set to

$$\boldsymbol{\Delta\theta} = [\Delta\theta_1, \Delta\theta_2, \Delta\theta_3, \Delta\theta_4] = [8^o, 8^o, 8^o, 8^o]. \tag{5.3}$$

In addition, the radial mask is set so it reaches the end of the Fourier 2D space, so the mask cut-off is

$$\boldsymbol{\delta} = [\delta_1, \delta_2, \delta_3, \delta_4] = [1024, 1024, 1024, 1024]. \tag{5.4}$$

When the inverse Fourier transform is applied to the masked spectra, the resulting image is free from features along these angles. The inverse 2D FFT produces a complex number matrix, so the real part (or the absolute value) of this matrix must be taken to plot the spectrogram. Taking the real part preserves the original dynamic range of the spectrogram (colors) while effectively removing the TAE antenna signal, as shown in Fig.5.8. It is important to note that this process may alter the amplitude values (observe the color bar scale in Fig.5.8). Taking the absolute value yields a similar result, but it may produce a negative image, which can change the color range in Fig.5.8.

To summarize, Fig.5.8 shows the original spectra, the 2D FFT magnitude spectrum with a filter mask, the resulting spectrogram, and the features removed from the images: the TAE antenna signal is successfully removed from the spectrogram. Remarkably, removing these directions preserves most of the information from the original signal, allowing sawteeth or MHD activity to remain observable in the removed features. However, modes aligned with these directions may be attenuated or disappear. As mentioned earlier, if vertical modes are present, further processing of the vertical image may be necessary. In the case of 2DFFT, it is also possible to redesign the mask to create a directional band-pass filter that allows lower frequencies to pass through.

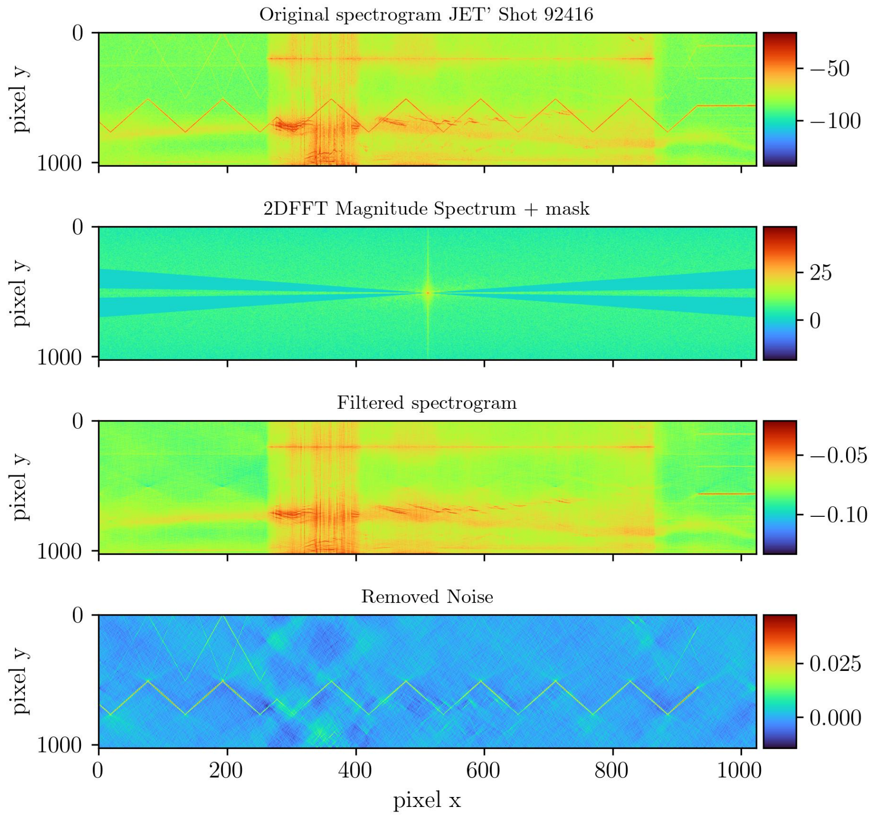


Figure 5.8.: Use of 2D FFT to remove TAE antenna sweeps: Original image, Mask over 2D FFT, Filtered image, and noise removed.

```
Algorithm 5: Remove straight line patter using 2D FFT high pass directional filters
  Input: Time series $\boldsymbol{s}$
  Parameters:
  Mask angles: $\boldsymbol{\theta} = [\theta_1, ...\theta_N]$.
  Angle width: $\boldsymbol{\Delta\theta} = [\Delta\theta_1, ...\Delta\theta_N]$.
  Mask cut-off: $\boldsymbol{\delta} = [\delta_1, ...\delta_N]$
  Output: Reassigned spectrogram $S_{xx(mod)}$
  Steps:
  Obtain spectrogram matrix $S_{xx}$ using STFT($\boldsymbol{s}$).
  Resize image $S_{xx}$ to have squared size: $1024 \times 1024$.
  Transform resized image to frequency space.
  Apply Mask to frequency space (truncated circle sectors centered at $\theta_i$, width $\Delta\theta_i$, and maximum radius $\delta_i$).
  Inverse transform the masked image to obtain the filtered image $S_{xx(mod)}$.
  Return $\mathrm{Re}\{S_{xx(mod)}\}$
```

If algorithms (3) and (5) are combined, the resulting image offers an improved contrast highlighting MHD features of the signal. In Fig.5.9 most of the broadband noise and the complete TAE antenna signal are removed. However, this approach is not without limitations. If the MHD activity aligns with the TAE antenna sweeps, we might lose these modes. For instance, two elliptic alfvén eigenmodes (EAE)s disappear, these small modes are situated at 51s and 475 kHz approximately (decreasing in frequency). Moreover, sawteeth signals are still crossing the MHD modes.

Depending on the application, designing the mask may require some trial and error to select the appropriate parameters. In the 2D FFT, low frequencies are located at the center. To avoid covering this region, the mask consists of radial circle sectors centered at angles $\boldsymbol{\theta}$ with respective apertures $\boldsymbol{\Delta\theta}$. The mask begins at a radial cut-off distance $\boldsymbol{\delta}$ from the origin, so low spatial frequencies are not filtered. Moreover, it has been found useful to resize the spectrogram to a square image beforehand. This helps create a symmetric mask (the radial mask apertures $\boldsymbol{\Delta\theta}$ are centered around the given angles $\boldsymbol{\theta}$).

We found that this algorithm is remarkably useful for removing straight lines crossing the whole image, like the ones commonly present in spectrograms from interferometers. Once the mask is designed for this use case ($\boldsymbol{\theta} = 0$ or 90 and $\boldsymbol{\Delta\theta} = 1$) the fine-tuning of parameters is optional. An example of this application is presented in the following chapter.

Manipulating the Fourier space is also crucial for understanding the curvelet transform. Given the success of using discrete wavelets in Algorithm 3, we extended this approach by employing curvelets.

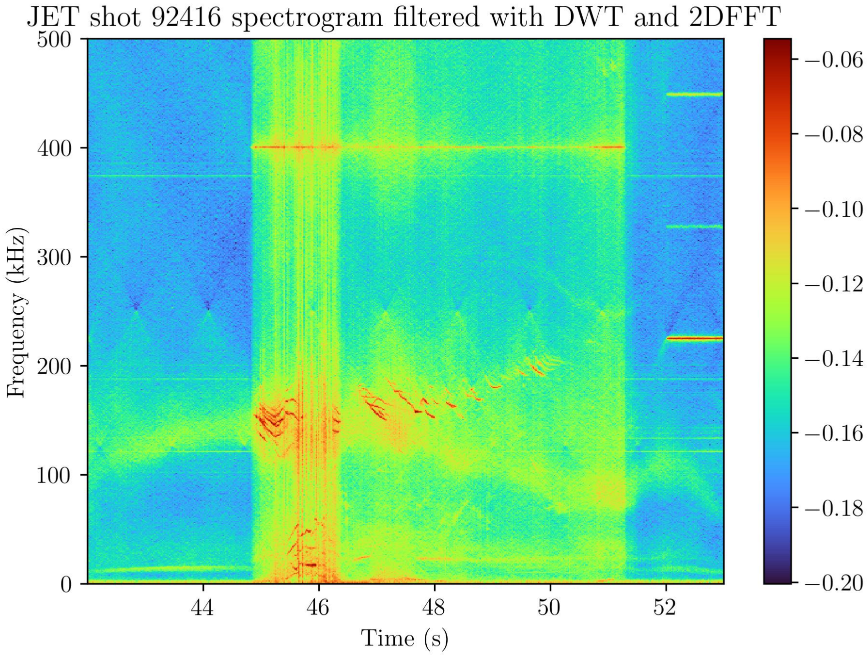


Figure 5.9.: Result of combining Algorithm 3 and 5. On the left the original spectrogram is shown, in the right picture the result of using both algorithms.

5.6. Multiscale-multidirection pattern matching using curvelets

The curvelet transform [Can+06] is a multidirectional generalization of the DWT, better suited for images with curved edges and multiple directional features, like the MHD modes in our spectrograms. To remove noise and unwanted components in a selected angular direction, we set the corresponding sub-images to 0 and then apply the inverse curvelet transform.

In Fig.5.10, a polar representation of the 2D Fourier space is shown. The 2D Fourier space is divided into a tessellation of “wedges” [Can+06]. Each wedge corresponds to a sub-image with specific scale and directional properties. The wedges closer to the center represent sub-images with large-scale (low-frequency) features. The colored wedges indicate the regions selected for removal before applying the inverse Curvelet transform. These directions include vertical, horizontal, and the antenna’s sweep directions. The scales to be removed in these directions exclude the two largest scales (the low frequencies represented by the central wedge and the first concentric wedges).

The result of using Algorithm 6 is shown in Fig.5.11. Curvelets effectively remove anisotropic noise and improve the contrast of the spectrogram. The TAE antenna signal and the ICRH subharmonics are also successfully removed from the spectrogram, along with the elimination of sawteeth. The wedge tessellation is configured by selecting the number of scales and wedges corresponding to the angles detected with the Hough transform. For more details on the curvelet transform, readers may refer to

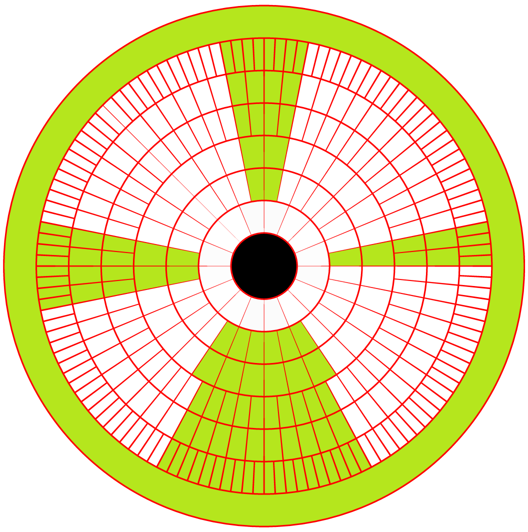


Figure 5.10.: Cartoon of curvelet's wedges in polar coordinates. Colored wedges are set to zero before doing the inverse Curvelet transform

Algorithm 6: Removing directional patterns with Curvelets

```
Input: $\boldsymbol{s}$
Parameters: $\sigma = 1, \boldsymbol{\theta} = [\theta_1, ...\theta_N]$
Output: $S_{xx(mod)}$
Steps:
STFT($\boldsymbol{s}$) to obtain spectrogram
Detect straight line patterns (optional).
Apply curvelet transform.
Set to 0 (hard thresholding) wedges corresponding to directions $\boldsymbol{\theta}$
Apply inverse curvelet transform.
```

Ma's review paper [MP10]. In Appendix A, original and processed images are plotted in Fourier space (Cartesian coordinates), highlighting the difference between a simple mask and the curvelets.

In Fig.5.12, we examine the information removed by the curvelet algorithm (6). It contains the removed TAE antenna signal, the ICRH signal, and a couple of lost EAE modes. The sawteeth pattern, which is of interest for studying edge localized modes (ELMs), can be analyzed separately. By applying Algorithm 5 to the curvelet's result, removing the vertical components of the image, we obtain a separate spectrogram of the sawteeth signals, as shown in Fig.5.13, lower panel. Although the image resolution in this example (1024 pixels) may not be sufficient for precise measurements, increasing the number of pixels can resolve this issue.

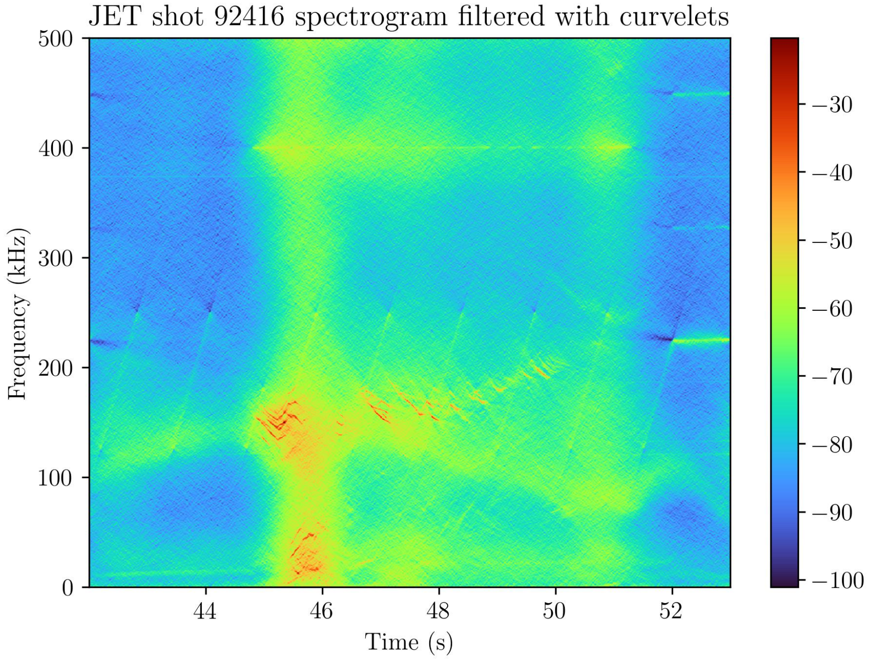


Figure 5.11.: Result of applying curvelet transform algorithm to remove TAE antenna: Algorithm (6)

Figure 5.12.: Features extracted using curvelets

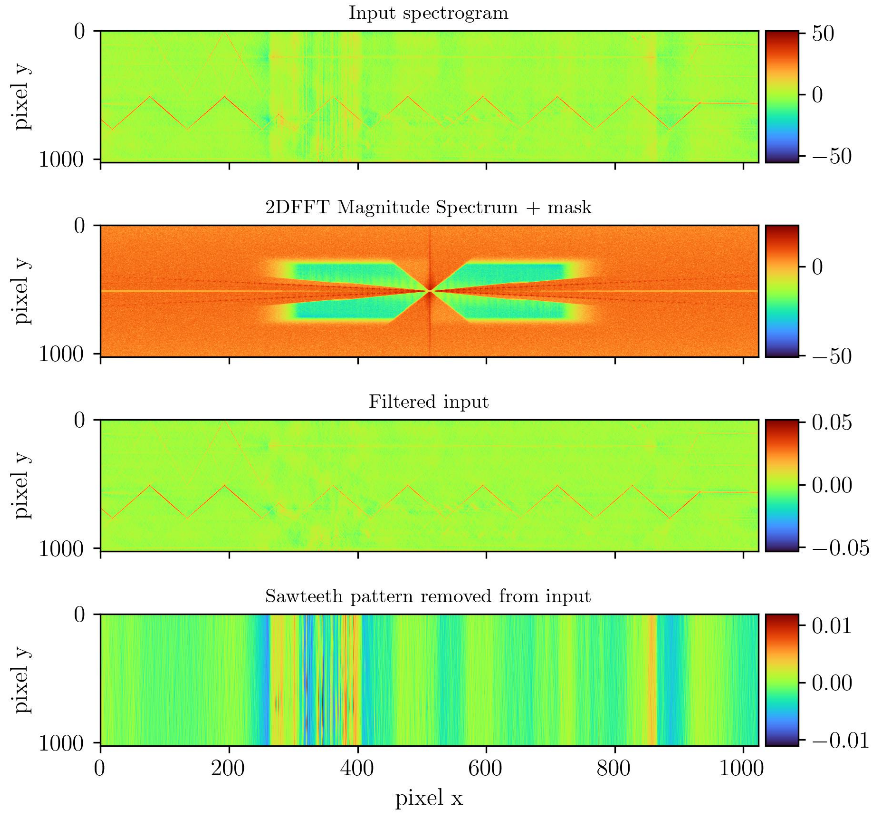


Figure 5.13.: Sawteeth extracted from Noise residuals using FFT2D

5.7. Summary

In this chapter, we presented a set of image spectral algorithms applied for the first time to spectrograms of magnetic diagnostics. The algorithms developed here can remove unwanted image components, thereby enhancing specific spatial features of spectrograms. The first development involved the introduction of DWT to remove broadband noise and pellet injections from spectrograms. Subsequently, the Hough transform was introduced to detect straight-line patterns, which helped identify the angles of directions to be removed. We found that 2DFFT could manipulate spectrogram features by masking the image in frequency space. We then extended DWT to multiple directions using the Curvelet transform and concluded by demonstrating how these methods can be applied to a complex spectrogram of shot 92416, showing how MHD activity can be separated from sawteeth oscillations and other features. These algorithms serve as a preprocessing step for the segmentation algorithms explained in the next chapter.

6. Mode Segmentation with Computer Vision

> *"Never send a human to do a machine's job"*
>
> — Agent Smith (The Matrix, 1999)

Summary

6.1. Introduction

The segmentation of an image $I(x, y)$ can be defined (page 131 of Ref. [AR05]) as the identification of a finite set of regions, or segments $(R_1, R_2, R_j, \ldots, R_N)$ such that:

1. $I(x, y) = R_1 \cup R_2 \cup \cdots \cup R_N$.
2. $R_i \cap R_j = \emptyset, \forall\, i \neq j$.
3. $P(R_i) = \text{True}, \forall\, i$.
4. $P(R_i \cup R_j) = \text{False}, \forall\, i \neq j$.

where the definition property P determines the result of the segmentation.

The first property ensures complete coverage of the image by all segments, while the second guarantees non-overlapping segments. The third property imposes a specific criterion that each region must satisfy, and the fourth prevents merging distinct regions by maintaining that combining them would violate the segmentation criterion. Notably, this definition of segmentation allows for different partitions of the image without assuming region connectivity.

In chapter 3, we illustrated the mode decomposition of a signal with a musical analogy. Expanding on this concept, we now explore how segmentation of spectrograms, like the one in Fig.3.2 can be used for modal decomposition.

In Fig.6.1 a 3D version of spectrogram Fig.3.2 is presented. Additionally, a “sea level” has been set so individual musical notes can be identified. After setting a threshold, we can find connected components, or “islands”, which identify musical notes of score 3.1. The segmentation result and the subsequent identification of separated components can be examined in Fig.6.2. The property P that governs the segmentation problem is the presence or absence of a musical note in a spectrogram’s pixel.

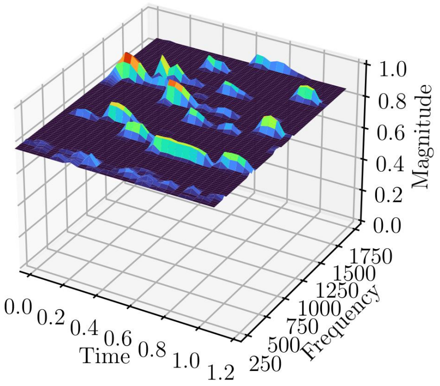


Figure 6.1.: 3D view of thresholded spectrogram (Fig.3.2)

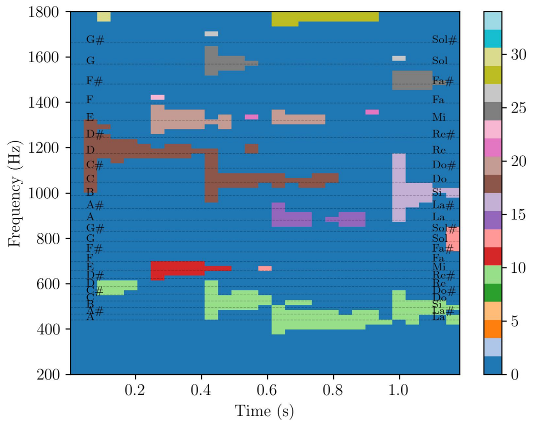


Figure 6.2.: Notes labeled by segmentation of thresholded spectrogram (Fig.3.2)

This CV approach offers direct advantages. First, it is much faster than the dictionary encoding approach presented in chapter 3. Additionally, the spectral time-frequency representation relies only on STFT, which is the algorithm of choice to analyze signals in the fusion community. Because the STFT is a projection over a dictionary method, it is not necessary to run complex and costly optimizations. However, it has also disadvantages. First, the threshold selection is not trivial, as it might vary between different time and frequency regions and it is very sensitive to noise. Moreover, to separate modes or musical notes we assume that they are regions of connected pixels, which might not be always true, especially with crossing modes. We can also use the feature extraction methods introduced in the last chapter to cope with noise and crossing signals.

Regarding fusion diagnostic spectrograms, we can consider two basic regions in a spectrogram image: regions with MHD activity and those without it. Consequently, the property P can be defined as the presence of an MHD mode in a spectrogram's pixel. Then, there are then two distinct regions: R_1, with MHD activity ($P(R_1) = 1$), and R_0, representing the background ($P(R_0) = 0$). We assume that if the segmentation is correct, non-connected regions of MHD activity constitute different time-frequency modes. Then, it is very easy to separate different modes using pixel connectivity.

In the first part of this chapter, we introduce ridge detection as a feature-enhancement method necessary to solve the threshold difficulty. In addition, different methods for the automatic selection of threshold are also discussed. The second part explores how segmentation can be combined with the feature extraction techniques from the previous chapter. The last part explores the use of deep learning for segmentation of spectrograms using fine tuning of state of the art models: SAM, Detectron2 and YOLOv8.

6.2. Contour-based segmentation of MHD spectrograms

According to Lei's book ([LN23], page 5) there are 4 types of segmentation: (1) pixel-based, (2) contour-based, (3) region-based, and (4) deep learning based.

First, the pixel-based segmentation algorithm includes global thresholding or clustering methods like k-means, which classify pixels based on similar pixel intensity (or color) but often they ignore spatial relationships. The musical example in figures 6.1 and 6.2 is a pixel-based segmentation method. This type of method works well with colored images, and with features with the same pixel intensity, like the spectrogram of a ragtime's pianola recording. However, they have difficulties distinguishing MHD activity from the background, as the background is not homogeneous.

Second, contour-based segmentation focuses on detecting object boundaries using methods like edge detection followed by a threshold. This is the approach followed in this chapter as we aim to distinguish ridges in the image.

Next, region-based segmentation grows regions from seed points or splits and merges regions based on feature consistency. Two common examples of region-based segmentation algorithms are the "watershed" methods [NS94; BLM14] and other superpixels algorithms [SHL18; Ach+12; NP14]. These methods are too coarse, merging areas with MHD activity and the background.

Finally, neural network–based segmentation uses deep learning models like CNNs, offering superior feature extraction but facing challenges in data requirements, model robustness, and inference speed, as discussed by Lei [LN23], page 5ff. This is the approach followed in the next chapter.

As mentioned in chapter 2, calculus is used for edge and ridge detection. The edge detection based on the first derivative is called first-order edge detection, page 162 of Ref. [NA12]. For example, the Sobel edge detector is based on the gradients of the image. The first derivative has a maximum along an edge position. In contrast, the second-order edge detection, based on the second derivative, has a zero crossing in the presence of an edge.

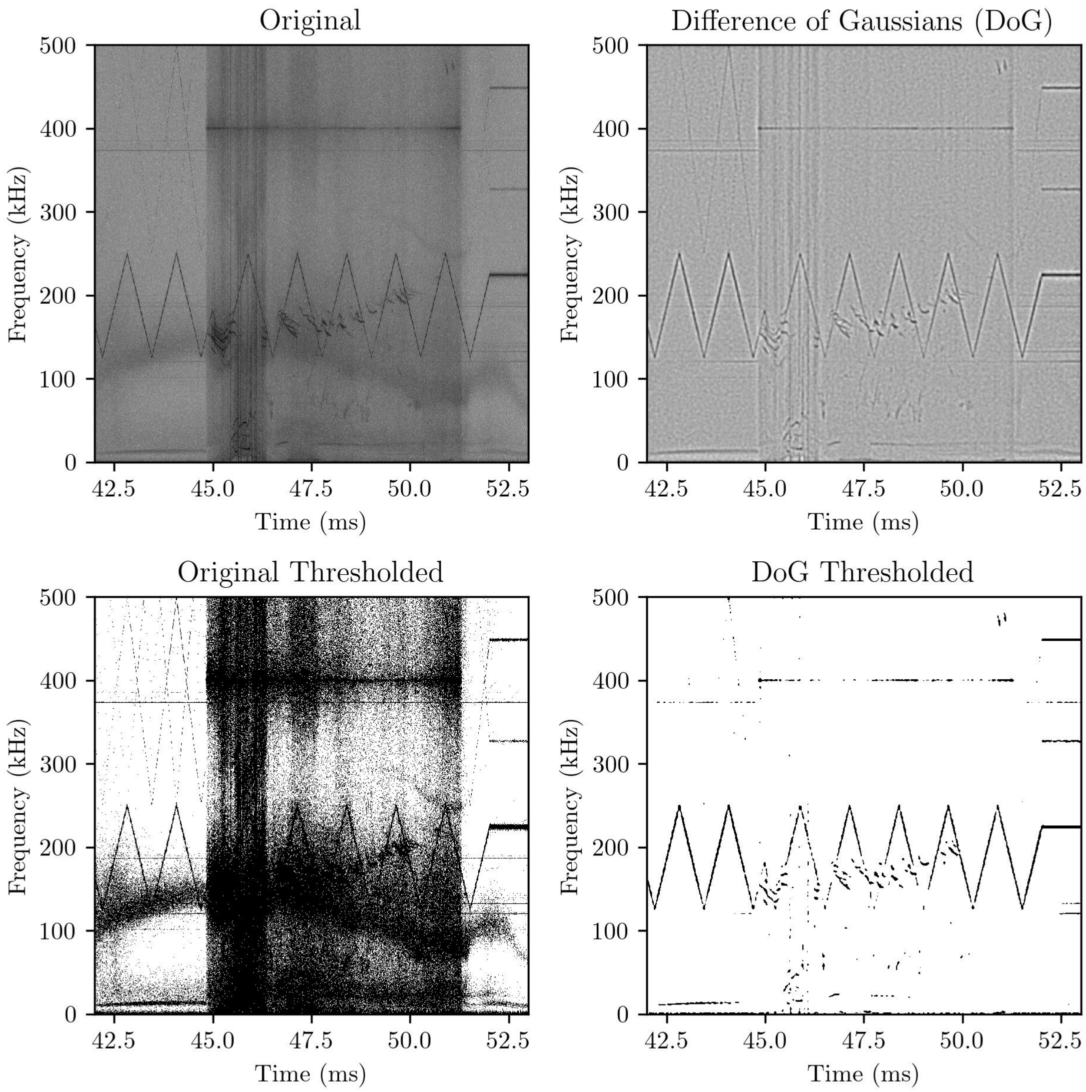


Figure 6.3.: Comparision of thresholding with and without feature enhancement. The DoG is used to detect pixel fluctuations

To take into account the directional properties of an image $I(x, y)$, the gradient $\nabla_{\mathbf{u}} I(x, y)$ along the direction of unit vector $\mathbf{u}$ are used for the first order edge detection. The directions for vectors $\mathbf{u}$ are usually the $\mathbf{x}$ (vertical) and $\mathbf{y}$ (horizontal) directions in the image $I(x, y)$. The orientation of an edge for each pixel can be calculated using $\tan^{-1}(\nabla_{\mathbf{y}} I(x, y)/\nabla_{\mathbf{x}} I(x, y))$.

The Laplacian $\nabla^2 I(x, y)$ is used for the second-order edge detection. The zero crossings of the Laplacian reveals an edge. In contrast with the gradient, the Laplacian is a magnitude, not a vector. In addition, the Laplacian is invariant to rotations of the image, a very useful property for extracting features.

Both gradient-based and Laplacian-based estimators are commonly used because they are easy to discretize and implement. However, these estimators are very sensitive to image noise [DMT16], page 70. To improve their robustness against noise, the image is usually smoothed using a Gaussian kernel before calculating the derivatives.

A Gaussian filter is a very simple process, taking an image and convolving it with a Gaussian $G(x,y,\sigma) = \frac{1}{2\pi\sigma}\exp\big((x^2+y^2)/2\sigma^2\big)$ results in an “blurred” image $G(x,y,\sigma) * I(x,y)$, with smooth details and edges. The main parameter of a Gaussian filter is the Gaussian σ, which determines the Gaussian spreading (scale) and consequently the intensity of the blurring effect. The “blurring” effect is caused because the Gaussian kernel acts like a low pass filter. It is important to note that using a Gaussian kernel is a necessary step to remove noise fluctuations at scales irrelevant to the objects of interest.

The low pass effect of a Gaussian filter also acts as a scale selection mechanism. A similar “low pass” effect can be also obtained by wavelet decompositions discussed in previous chapters (by keeping approximation images). Indeed, using a Gaussian filter with a σ between 1 and 2 pixels also improves the contrast of spectrograms because noise fluctuations are smoothed out. The effect is comparable to using DWT introduced last chapter to remove broadband noise in spectrograms.

The Laplacian of Gaussian operator (also known as Marr-Hildreth operator[MH80]) is constructed taking the Laplacian of the image convolved with a Gaussian $\nabla^2(G(x,y,\sigma) * I(x,y)) = \nabla^2(G(x,y,\sigma)) * I(x,y)$. It can be shown that the LoG is [NA12]

$$\mathrm{LoG}(x,y,\sigma) = \nabla^2(G(x,y,\sigma)) = \frac{1}{\sigma^2}\left(\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right)e^{-\frac{x^2+y^2}{2\sigma^2}}. \tag{6.1}$$

The 3D shape of this operator is known as “Mexican hat”, which can be also approximated by the difference of two Gaussians (DoG), as illustrated in Fig.6.4 (in 2D). The convolution with this operator is equivalent to applying a bandpass filter that omits low and high frequencies. Indeed, the difference between two low-pass filters of different scales forms a bandpass filter. This can be demonstrated by applying the Fourier transform to a Mexican hat curve (or the DoG curve), as on page 168 of Ref. [NA12].

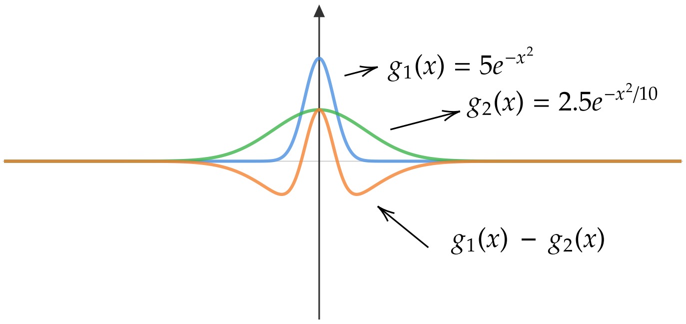


Figure 6.4.: Difference of Gaussians: The DoG

The DoG operator [Mar10; Lin94; WKO12] is mathematically defined as

$$\mathrm{DoG}(x,y,\sigma_1,\sigma_2) = \frac{1}{\sigma_1^2} e^{-\frac{x^2+y^2}{2\sigma_1^2}} - \frac{1}{\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}}. \tag{6.2}$$

The ratio of σ_1 and σ_2 is set to a value between 1 and 2 for a good edge detection, [AR05] page 143. The DoG operator has a positive response in the presence of increasing edges, or ridges. And negative response in the presence of a decreasing edge or a negative ridge (or a valley). In other words, it does not make a difference between edges and ridges. However, in the case of our spectrograms, the edges have a smaller scale than the ridges.

The DoG is commonly used as an artistic effect [WKO12] to simulate human drawings or photocopies from a given image. Although this may seem unrelated to fusion research, it becomes relevant when manual mode labeling requires sketching over spectrograms. The pencil effect, a DoG-based technique, could be particularly useful for automatically labeling mode structures in fusion spectrograms.

It is also possible to generalize the DoG to multiple scales to N multiple scales $\boldsymbol{\sigma} = [\sigma_1, ..., \sigma_N]$, applying the DoG repeatedly with different σ values in pairs. Then, there are alternative ways of selecting or aggregating the features from different scales, as discussed in [Sze22] page 191ff. This “stacking” of filters of different scales is known as the DoG scale space (page 242 of Ref. [BB13]), and it can be used to extract complex features from different image scales.

After the edges of spectrograms are enhanced, the binarization of an image $I(x,y)$ can be performed by the selection of a global threshold T (page 169 of Ref. [Kle14]), mathematically

$$J(x,y) = \begin{cases} 0 & \text{if } I(x,y) < T \\ 1 & \text{otherwise} \end{cases}. \tag{6.3}$$

The threshold T parameter can be tuned manually. However, this is not convenient for analyzing different shots in large databases. However, there are global automatic thresholding algorithms that can calculate T for a given image $J(x,y)$. In Fig. 6.3 we can see the result of applying the dog with Otsu’s [Ots79] automatic global thresholding algorithm to the original image (Fig. 6.3 left), compared with the result of thresholding after the edge detection (Fig. 6.3 right).

The binarized mode structures are clearly identifiable after feature enhancement. This remarks the necessity of using some kind of feature enhancement, such as edge detection or ridge detection, before binarization to ensure modes are properly separated from the background. Otherwise, modes are lost or mixed with the background as we can see in Fig. 6.3 (left).

6.3. Ridge detection algorithms for spectrograms

The DoG can detect ridges well if the scales selected are close to the ridge’s size. However, if the scales of DoG are smaller than the ridges in the image, there is no way to differentiate between edges and ridges, as the DoG captures all changes of curvature. Therefore, it is necessary to further develop the model if the structures that we want to detect have a ridge structure. These algorithms were developed for the analysis of biological images taken by microscopy. Under the microscope, vessels, neurites (part of neurons), some types of bacteria, and many other objects of study have a ridge structure. Sometimes is necessary to label and count them. For example, when studying the growth of certain bacteria, it is also necessary to measure their size variation. Consequently, detecting only the edges of the image structures is insufficient.

The ridge filters used in this thesis are Sato's [Sat+98] ridge detector, Meijering's [Mei+04], Frangi's [Fra+98] and Hessian's [Ng+15] results. All these ridge detectors use the image Hessian matrix as part of the algorithm.

The second-order directional derivatives are used to capture complex directional structures, [DMT16] pag 74. The Hessian matrix $H(x_0, y_0)$ at a pixel point (x_0, y_0) is composed of the second-order partial derivatives:

$$H(x_0, y_0) = \begin{pmatrix} \frac{\partial^2 f(x_0,y_0)}{\partial x^2} & \frac{\partial^2 f(x_0,y_0)}{\partial x \partial y} \\ \frac{\partial^2 f(x_0,y_0)}{\partial x \partial y} & \frac{\partial^2 f(x_0,y_0)}{\partial y^2} \end{pmatrix} \tag{6.4}$$

This matrix is smoothed by a kernel G, leading to the smoothed Hessian matrix:

$$H(x_0, y_0) = \begin{pmatrix} (G_{xx} * I(x,y))(x_0, y_0) & (G_{xy} * I(x,y))(x_0, y_0) \\ (G_{yx} * I(x,y))(x_0, y_0) & (G_{yy} * I(x,y))(x_0, y_0) \end{pmatrix} \tag{6.5}$$

The dominant orientation of a ridge at (x_0, y_0) is determined by finding the eigenvector corresponding to the largest eigenvalue of the Hessian matrix. Specifically, the eigenvector u_2 that maximizes $u^T H u$ subject to $\|u\| = 1$ provides the ridge direction. The eigenvalues λ_1 and λ_2 of the Hessian matrix give information about the curvature in orthogonal directions, with the eigenvector associated with the smallest eigenvalue λ_1 indicating the direction orthogonal to the ridge, as explained in [DMT16] page 74.

Different authors modify the Hessian method slightly because there are different ways of interpreting the eigenvalue information. Consequently, there are different criteria to determine the presence of a ridge. For example, Frangi defines a "vesselness" V(x,y) [Fra+98] which is the likelihood of the pixel eigenvalues representing a vessel's ridge

$$V(x_0, y_0) = \begin{cases} 0 & \text{if } \lambda_1 > 0 \\ \exp\left(-\frac{\lambda_1^2}{2\beta_1^2}\right)\left(1 - \exp\left(-\frac{\lambda_2^2}{2\beta_2^2}\right)\right) & \text{otherwise} \end{cases} \tag{6.6}$$

where β_1 and β_2 are sensitive parameters.

It is also possible to add a sensitive parameter α [Mei+04] directly to the Hessian matrix, [Mei+04]. representing f_{ij} as $(G_{ij} * I(x,y))(x_0, y_0)$ we have a modified Hessian H'

$$H'(x_0, y_0) = \begin{pmatrix} f_{xx} + \alpha f_{yy} & (1-\alpha) f_{xy} \\ (1-\alpha) f_{yx} & f_{yy} + \alpha f_{xx} \end{pmatrix} \tag{6.7}$$

Meijering then defines a “neuriteness” $\rho(x,y)$ [Mei+04] which is the likelihood of the eigenvalues representing a ridge (neurites from microscopic images).

$$\rho(x_0,y_0) = \begin{cases} 0 & \text{if } \lambda(x,y) \geq 0 \\ \frac{\lambda(x_0,y_0)}{\lambda(x_0,y_0)_{\min}} & \text{if } \lambda(x_0,y_0) < 0 \end{cases} \tag{6.8}$$

where $\lambda(x_0,y_0)$ is the larger magnitude of the two egenvalues [Mei+04], and $\lambda(x_0,y_0)_{\min}$ is the minimum eigenvalue in all image.

Sato [Sat+98] proposed an alternative measure of “vesselness”, and Li et al [Ng+15] used a hybrid approach using both Hessian and directional gradients. The bottom line is that these ridge detectors use image curvature information, in contrast to other segmentation methods, with different criteria for “ridgeness”.

These ridge operators can be extended to multiple scales, or a set of $\boldsymbol{\sigma} = [\sigma_1,...,\sigma_N]$. The ridge filtering is repeated for each σ_i, keeping the maximum value of “ridgeness” combining features from different scales. As explained in [Fra+98], the response of the ridge filter will be maximum at a scale that approximately matches the size of the ridge to detect.

The global automatic thresholding algorithms used in this thesis are [Ots79] and Lneuritenessi’s [LL93; LT98], for comparison, we also show the result of Yen’s method [YCC95; San04], ISODATA method [RC78; San04], triangle’s method [ZRL77], minimum’s threshold [Gla93; PM66] and mean threshold. It is not necessary to dig into their model assumptions at least we need to explain the result.

The result of the binarization varies depending on the combination of the ridge filter, scales selected, and the thresholding method result. In Fig.6.7, the first row shows the image with enhanced ridges. The original image and the DoG result are included for comparison. In addition, the results for different scales are calculated for each algorithm, the σ parameter is indicated. The second, third and fourth rows show the result of binarizing the image with different threshold methods: Yen’s, Otsu’s and Mean methods respectively. In addition, the value of T is indicated for comparison between threshold methods.

The bigger the differences between background and foreground (contrast) the more successful results the segmentation. When binarizing a spectrogram, most of the pixels should appear white, as the modes and TAE antenna signal occupy a small fraction of the image area. In contrast, if most of the pixels are black that means that the value T is too low, causing noise to be mistaken for significant signals. It may also indicate that the performance of the feature enhancement is not good enough to separate background from foreground.

First, regarding the scales, two options have been used for this comparison: one-pixel scale $\sigma = [1]$, and four scales of width $\sigma = [1,2,3,4]$ pixels. If we compare both in Fig. 6.5 and Fig.6.6 and first row of Fig.6.7, we can conclude that the multiscale analysis is able to recover more features from the image spectrogram. Note how the ridges of the TAE antenna signal and the modes are more intense when multiple scales are used.

We can appreciate in Fig.6.7 that binarizing the original image (without feature enhancement) results in a failure to identify modes, mixing the MHD signal with the background. Also, the so-called Hessian method, based on Ref. [Ng+15] and implemented by scikit-image developers does not perform well, detecting most noise in the spectrogram background.

Comparing thresholding methods, we can observe that in the fourth row of Fig.6.7, the Mean's threshold method selects a value of T that does not separate background and foreground. In addition, Otsu's method in the third appears very noisy except when it is combined with Frangi's ridge filter. Meanwhile, it can be seen in the second row of Fig.6.7 that Yen's method performs well while separating background from foreground, except when the original image or the Hessian filter is used. That also means that the DoG, Meijering's, Sato's and Frangi ridge detectors can be used for processing spectrograms, if the right σ and T parameters are chosen.

To better compare different threshold methods with an example. The results of segmenting a spectrogram with Meijering's method can be seen in Fig. 6.8. It can be appreciated straightaway that the minimum, mean and triangle methods do not work, because they do not detect any signal (T is too high) or because they capture too much noise (T is too low). In contrast, the Isodata, Li's and Yen's methods give comparable results, varying slightly the amount of background recovered.

We can conclude this exercise by comparing Ridge filters and thresholding methods with the following observations. First, we trace time-frequency signals and separate them from the spectrogram background successfully. Moreover, we have several combinations of methods available. Second, it is necessary to remove vertical and horizontal patterns as well as the TAE antenna signal, before or after the binarization is completed.

There might be cases in which there are not any undesired signals. Maybe because the TAE antenna or ICRH are not operating, or the spectrograms came from another machine different than JET without these features. In those cases, the presented algorithm 7 can be applied directly, for example, Fig.6.20 later in this chapter.

Algorithm 7: Segmentation of fusion spectrograms

```
Input: $\boldsymbol{s}$
Parameters: $\boldsymbol{\sigma} = [\sigma_1, ...\sigma_N]$
Output: $S_{xx(binarized)}$
Steps:
STFT($\boldsymbol{s}$) to obtain spectrogram
Apply a Ridge detector with $\sigma = [1, 2, 3, 4]$ using Sato's method or equivalent.
Calculate threshold $T$ using Yen's method or equivalent.
Binarize the image I(x,y) using $T$
```

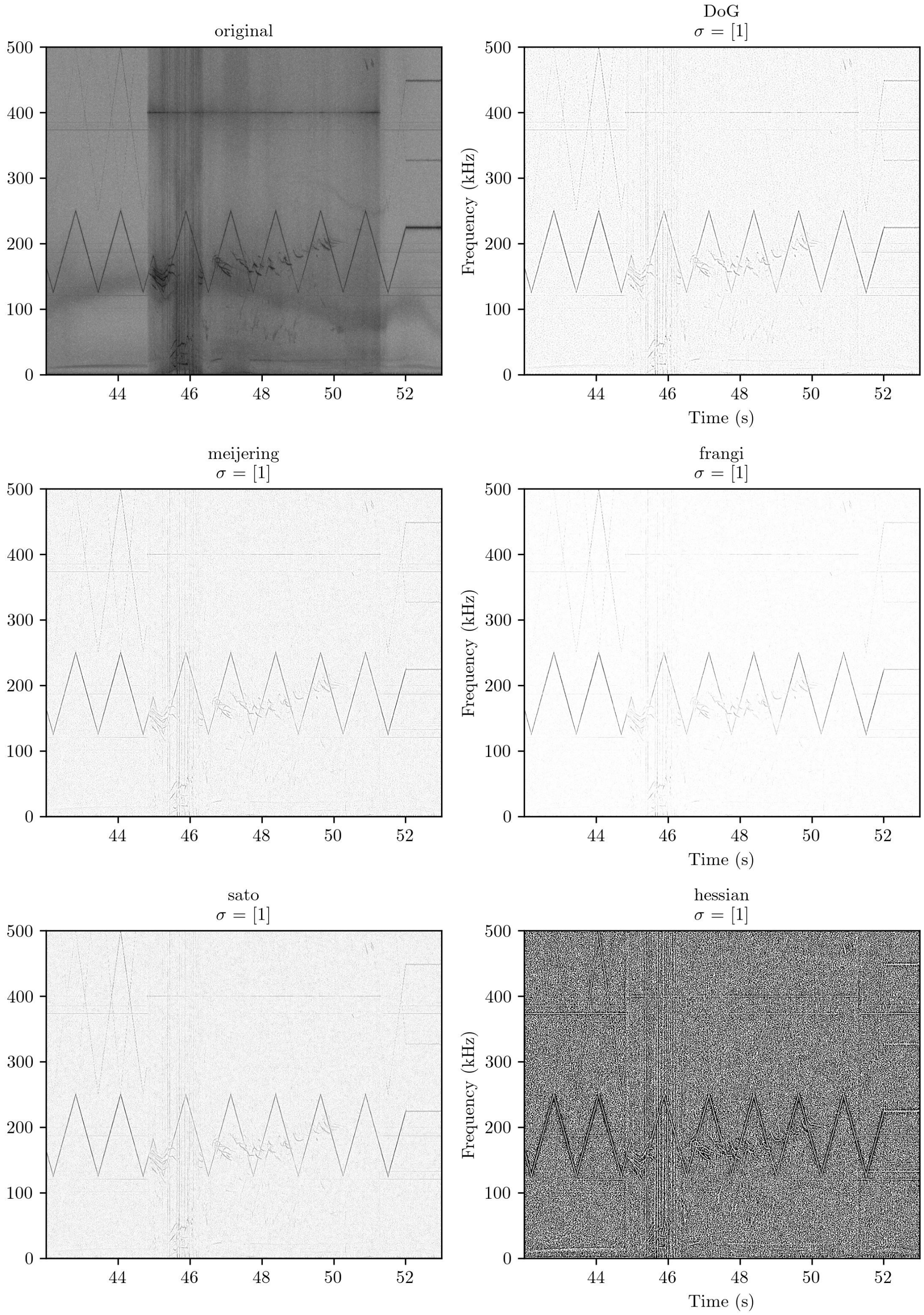


Figure 6.5.: Ridge filter comparison $\sigma = 1$

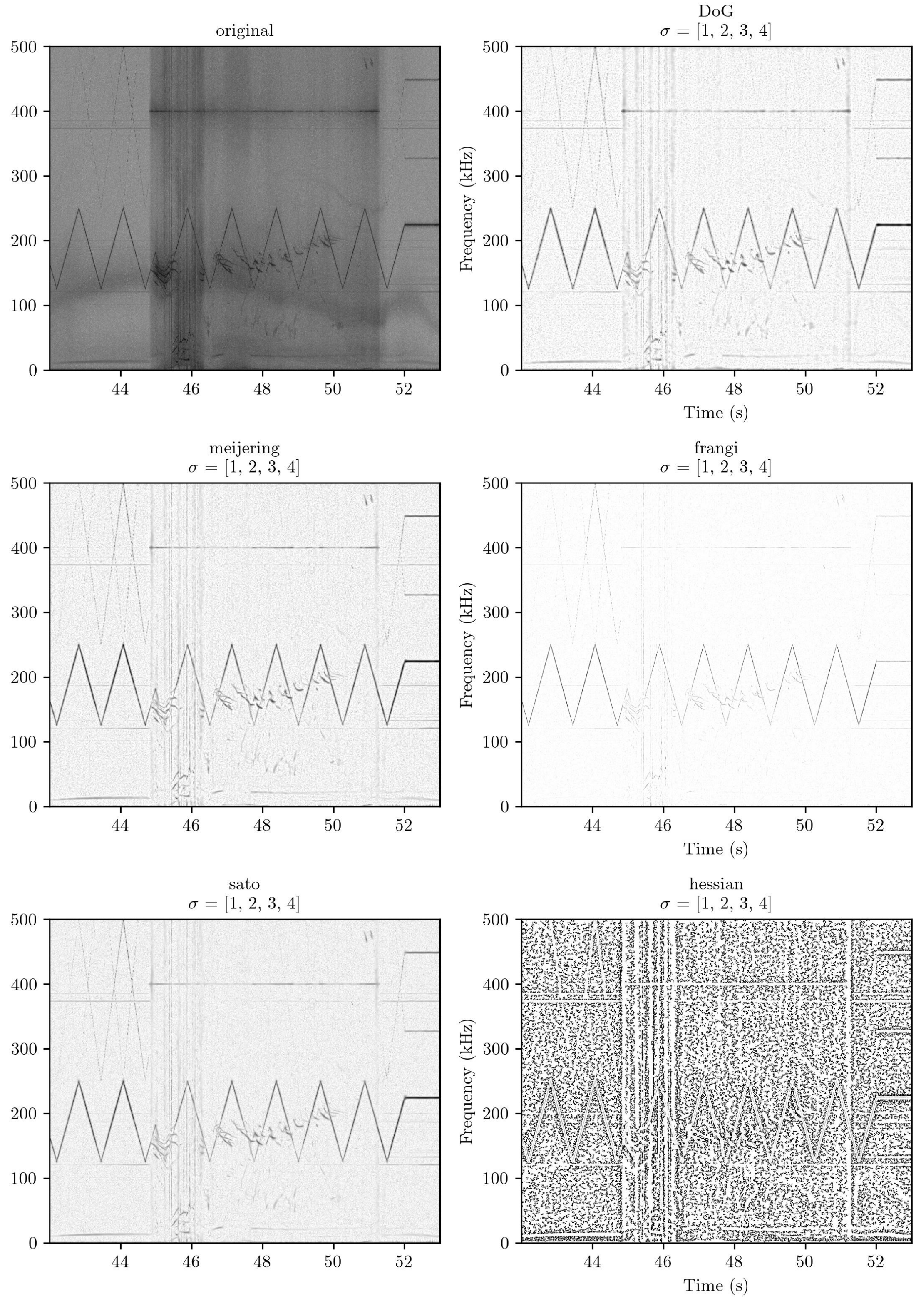


Figure 6.6.: Multi-scale Ridge filter comparison

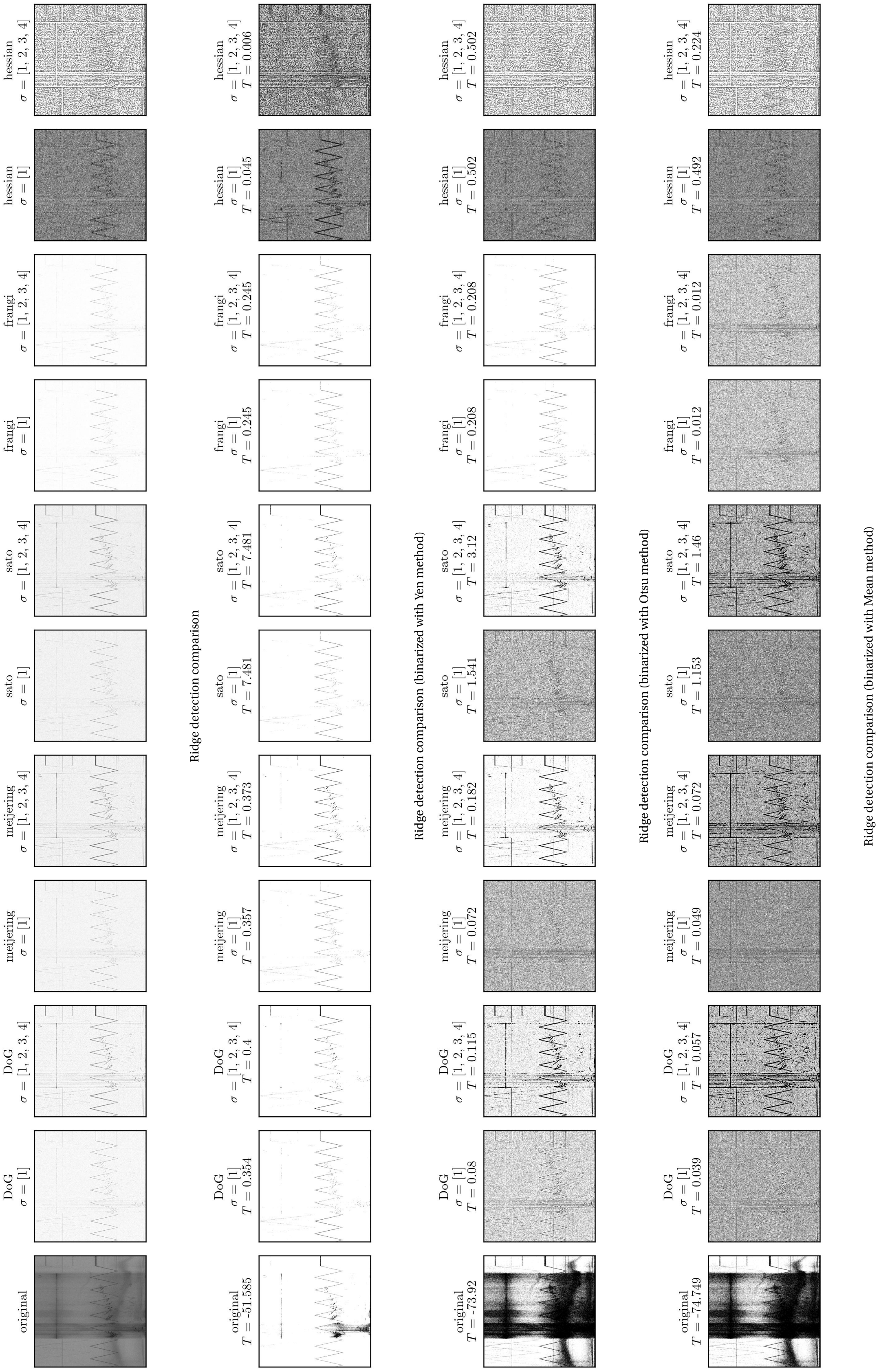


Figure 6.7.: Examples of different ridge detection algorithms on JET spectrogram

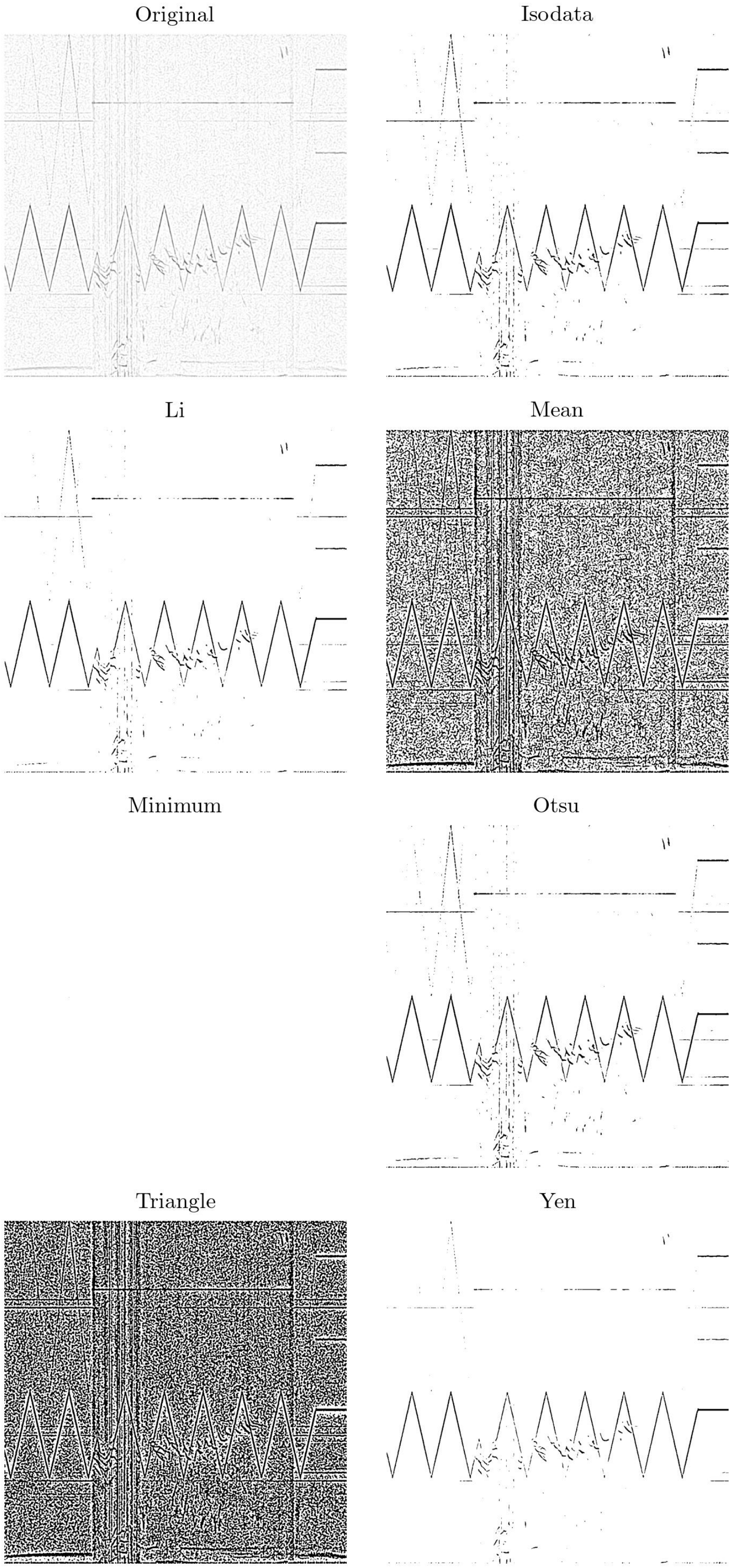


Figure 6.8.: Comparision of thresholding algorithm results with Meijering enhanced image

6.4. Automatic mode labeling using feature extraction and automatic thresholding

Now we can inspect the behavior of the ridge filters and threshold methods with the result of Fig.5.11 in which we used curvelets to remove broadband noise and the TAE antenna signal. Repeating the exercise of the last section, we can observe in Fig. 6.9, Fig. 6.10, and Fig. 6.11, that the equivalent results for separating modes from the background are obtained with the DoG, Meijering's, and Sato's and Frangi's methods with multiple scales $\sigma = [1, 2, 3, 4]$. In this case, only the thresholds calculated by Yen's methods work. We can see that the threshold values of the first row of Fig.6.11 are significantly higher than the other values calculated by Otsu and the mean method. Additional evidence of the effectiveness of Yen's method for images processed with curvelets can be found in Fig.6.12.

After improving contrast and having removed other undesired signal components. It is a reasonable assumption that connected areas of image pixels are MHD modes, which can be detected using a connectivity filter [WOS05; FG96].

Two neighboring pixels with the same value are linked. In a 2-connectivity filter (full), diagonal linking is allowed, whereas a 1-connectivity filter permits only vertical and horizontal linking. To give some examples, in a 3×3 image, among all combinations, five connected pixels on a binarized image can be

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \tag{6.9}$$

In order to compare the results of labeling with different ridge operators. Given the result of the spectrogram processed with curvelets in Fig.5.11, all multiscale ridge operators are applied to enhance the mode pixels. Then, Yen's automatic thresholding is used to binarize the image. Once the pixel values are 0 or 1, the Scikit-Image implementation of [Van+14] is used to label separated elements in the binary images. The result is shown in Fig.6.13 and Fig.6.14.

These labeling algorithms not only assign a natural number and a bounding box to each mode, but they also calculate very useful properties for each pixel region. For instance, the region's pixel areas can be used to discard most modes that are mislabeled as noise. We can consider noisy regions as those regions with less than 9 pixels connected (the size of one fully connected pixel). Moreover, low eccentricity can reveal noisy points mislabeled as modes. Some of the basic properties available in Scikit-Image region properties [Van+14] are included in table 6.1.

The correct number of modes in Fig. 6.11 and Fig.6.14 is uncertain. It lay between 40 and 60, the uncertainty depending on the criteria of the person counting the modes. All ridge detectors except Hessian's have a similar performance. However, Sato's algorithm ridges have fewer noisy points mislabeled than the others, justifying its selection for our final CV pipeline in the algorithm 8.

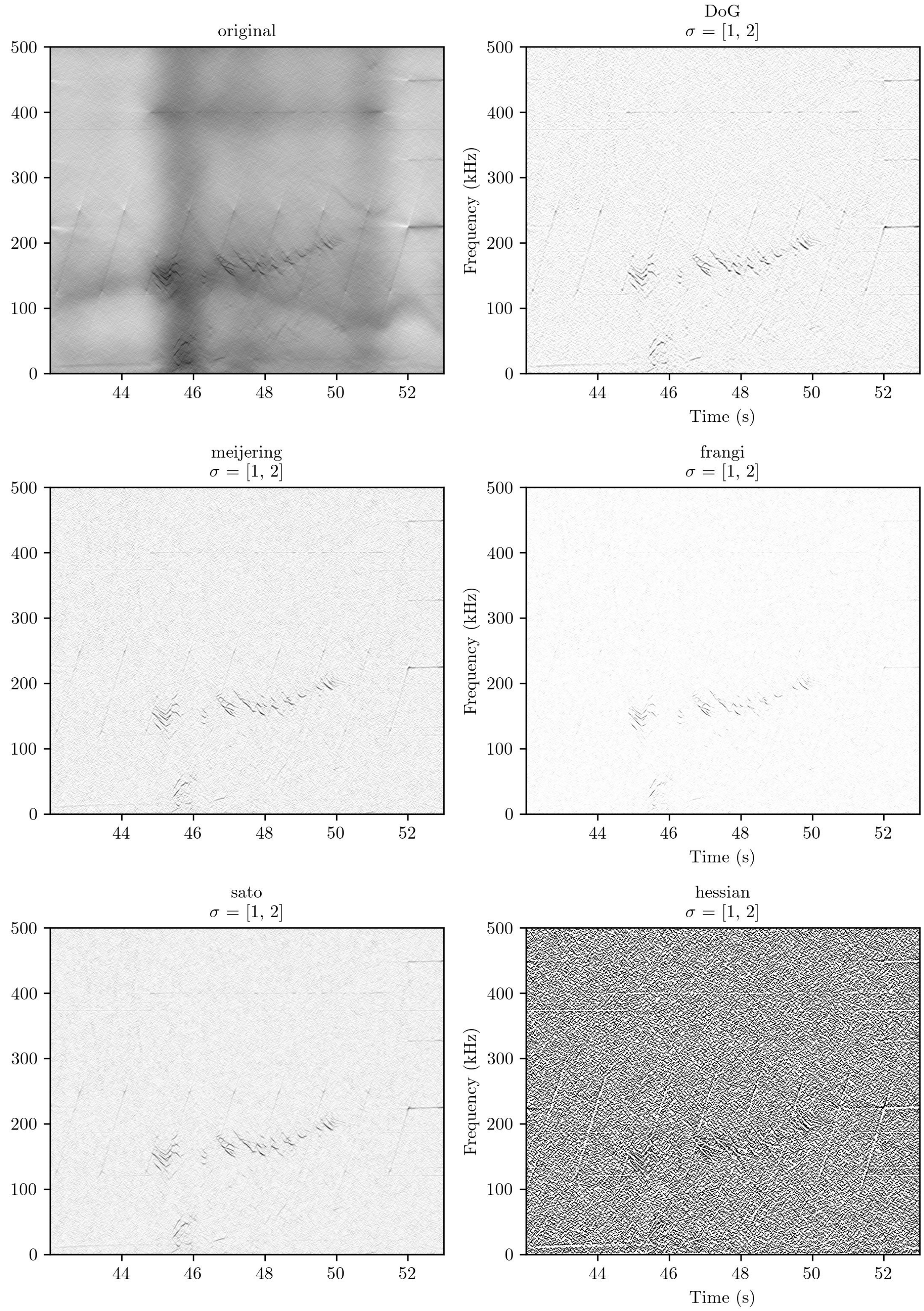


Figure 6.9.: Ridge filter comparison $\sigma = [1, 2]$

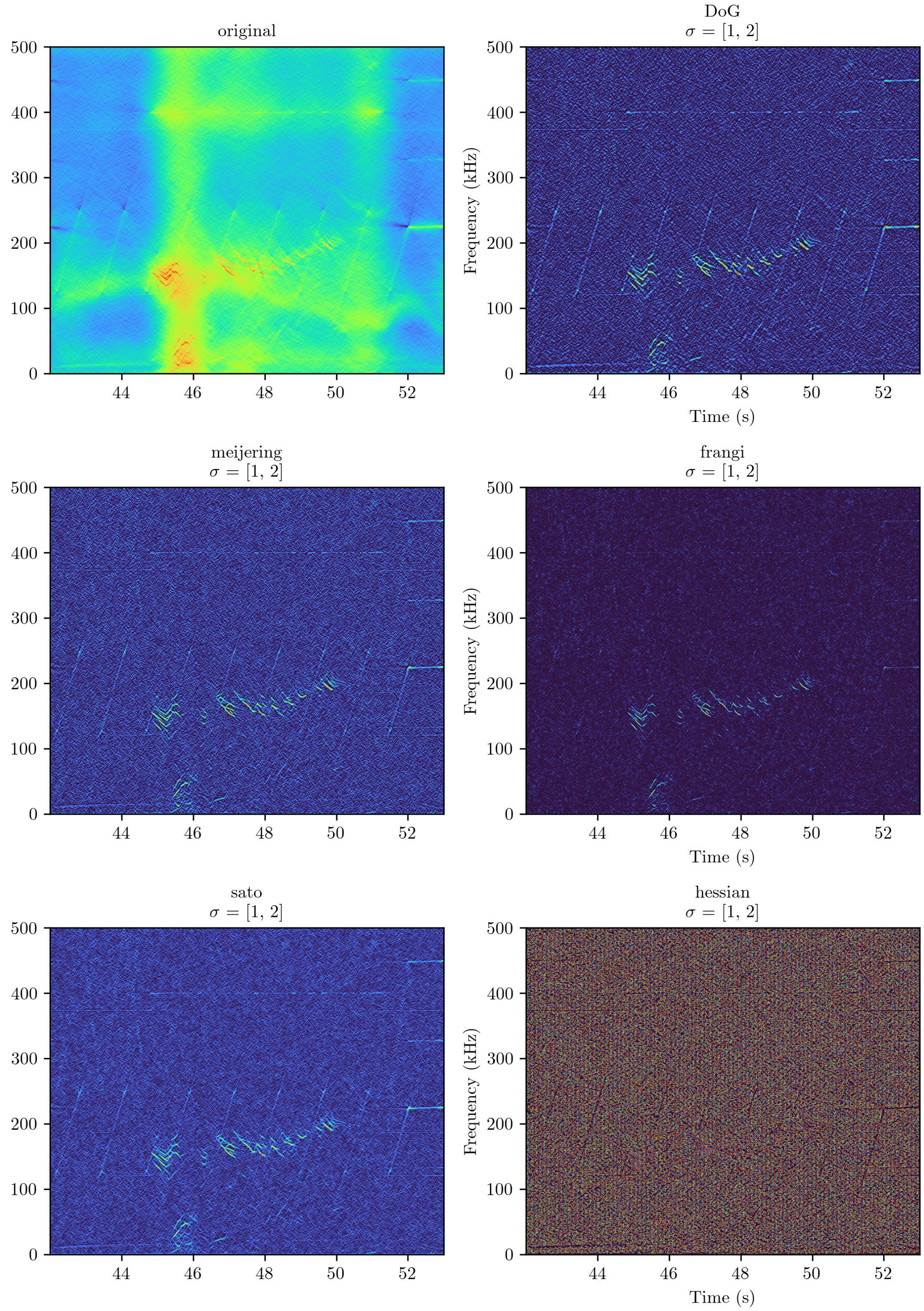


Figure 6.10.: Ridge filter comparison $\sigma = [1, 2]$ on false color (“turbo” palette).

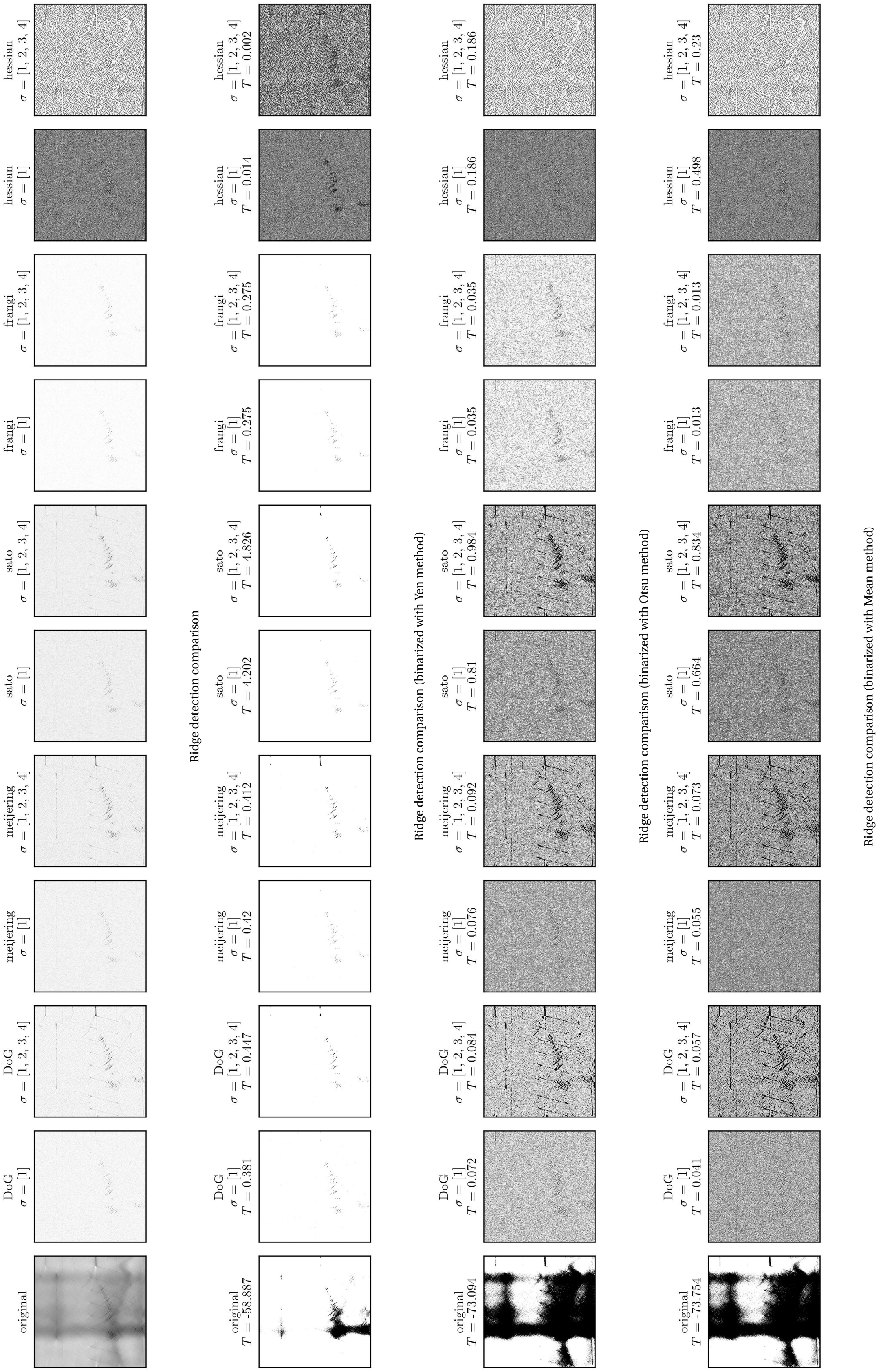


Figure 6.11.: Examples of different ridge detection algorithms on JET spectrogram processed with curvelets

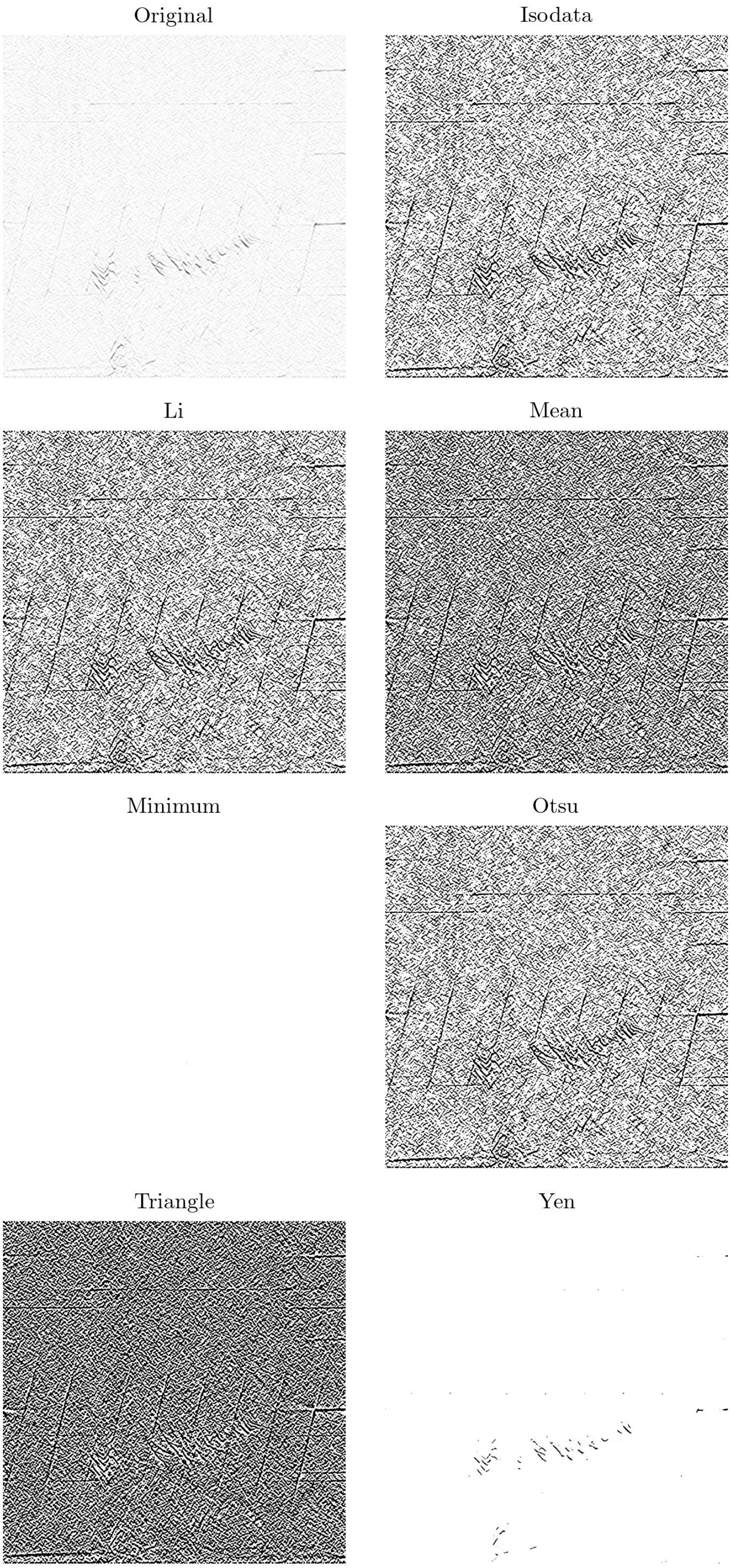


Figure 6.12.: Comparision of thresholding algorithm results with Meijering enhanced image

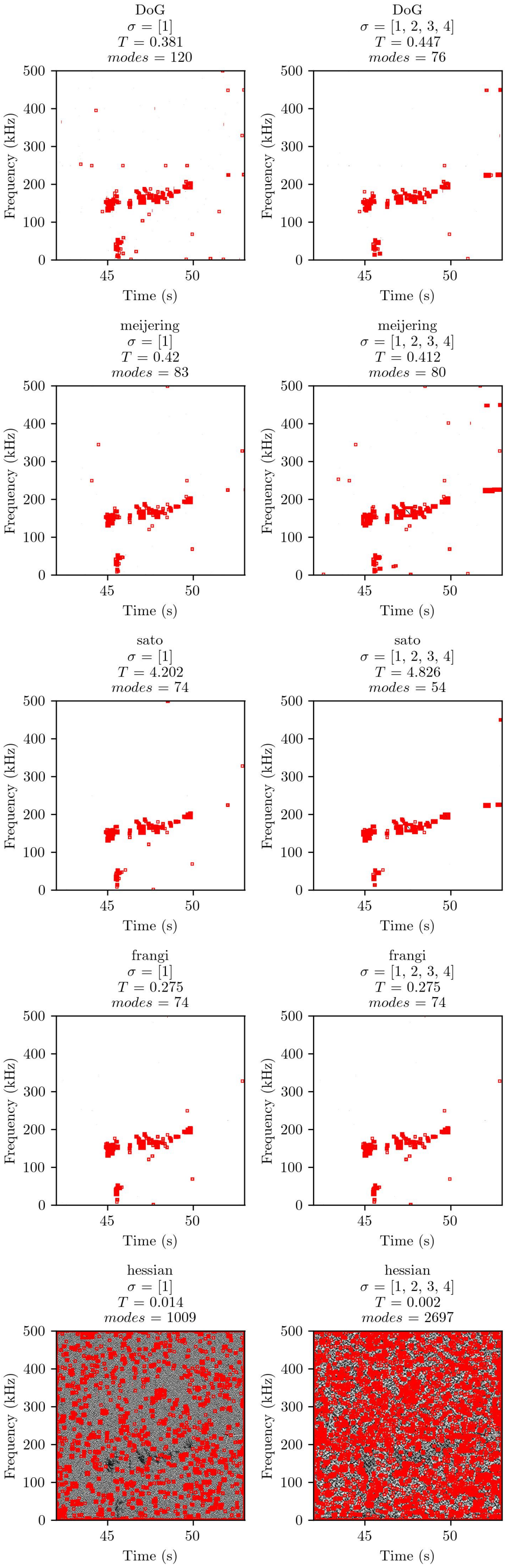


Figure 6.13.: Modes labeled with different methods (red boxes) over curvelet's spectrogram

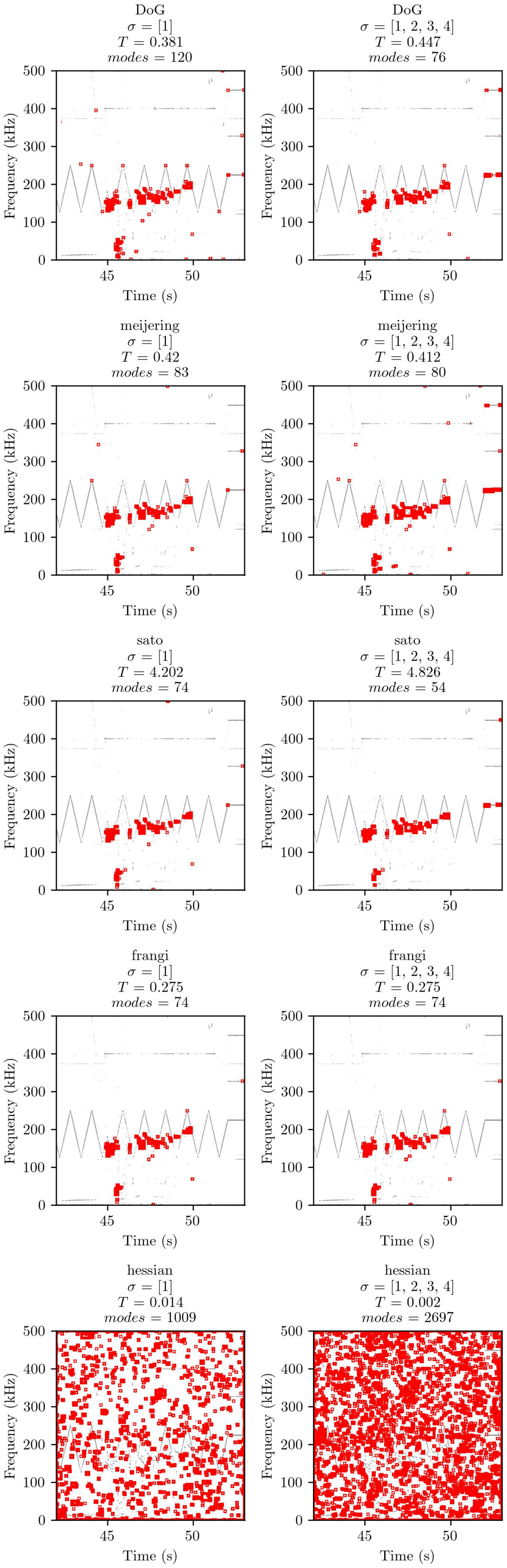


Figure 6.14.: Modes labeled with different methods (red boxes) over original spectrogram

Property	Description
Area	Number of pixels in the region
Bounding Box (bbox)	Coordinates of the smallest rectangle containing the region
Centroid	Center of mass of the region
Convex Area	Area of the smallest convex polygon that can enclose the region
Eccentricity	Measure of how elongated the region is
Equivalent Diameter	Diameter of a circle with the same area as the region
Extent	Ratio of the region's area to the area of its bounding box
Orientation	Angle between the region's major axis and the horizontal axis
Perimeter	Length of the region's boundary
Solidity	Ratio of the area of the region to the area of its convex hull

Table 6.1.: Some properties included in scikit-image representation

It is worth summarizing the intrinsic limitations of every segmentation algorithm, including the ones presented in this work: missing modes (false negatives if ground truth exit), identifying noise as modes (false positives), merging crossing structures or fragmentizing (separating) continuous structures (because of fluctuations in pixel intensity). Ridge detectors assume that the mode signal has a ridge structure, eliminating details like extended shadows or large blobs.

Since no ground truth exists in our case, the algorithms presented in this chapter are unsupervised learning techniques. Consequently, they require human interpretation. However, if the results are satisfactory, these classic CV algorithms can serve as a ground truth for DL algorithms. Especially because the results of using a CV pipeline like the algorithm (8) have potentially more consistency than humans labeling spectrograms manually.

We can consider eliminating possible noisy regions using the region's properties. Those regions with less than 9 pixels connected (the size of one pixel fully linked with surroundings) can be removed in the last step in the algorithm (8). To give an example of their size, they are marked in Fig. 6.16 with red boxes. The 54 modes labeled with Sato's filter are reduced to 45. This parameter can vary depending on the number of pixels in the spectrogram.

The application of algorithm (8) to a Mirnov coil signal of JET shot 92416 results in Fig.6.15. We can conclude that individual modes are satisfactorily labeled with minor errors like a few modes missed and only three false positives out of 45 regions of pixels detected. In addition, some modes of very low signal-to-noise ratio and mostly aligned with the removed directions are missed.

To summarize all steps from the calculation of the spectrogram, feature extraction, to labeling. The pipeline combination of algorithms introduced in the algorithm (8).

It is to be noted that, an application with a graphical user interface could be used to discard these minor labeling errors. An application called "Wavystar" with a GUI has been implemented, available under request in JET. This application implements algorithms discussed in this work, and other analysis capabilities of transitory signals not treated in this work.

Algorithm 8: Complete pipeline for automatic labeling of MHD activity

```
Input: Time series $\boldsymbol{s}$
Parameters: $\sigma = 1$, $\sigma_{\text{scales}} = [1,2]$, $\boldsymbol{\theta} = [\theta_1, ...\theta_N]$, $\boldsymbol{\varsigma}$=9 (optional)
Output: Reassigned spectrogram $S_{xx(mod)}$
Steps:
Calculate the spectrogram STFT($\boldsymbol{s}$);
Detect straight line patterns or specify angles manually.
Apply curvelet transform (or use Algorithm 5 and skip steps 4 and 5).
Set to 0 (hard thresholding) wedges corresponding to directions $\boldsymbol{\theta}$
Apply inverse curvelet transform, take the absolute value and invert the image.
Apply Sato's ridge operator with $\sigma_{\text{scales}} = [1,2]$. (Or equivalent)
Smoothing: Gaussian($\sigma$=1).
Yen's automatic thresholding (binarize the image).
Mark up pixel-connected areas (for labeling MHD activity), using a
  full-connectivity filter
Removed noisy labels: modes with less than $\boldsymbol{\varsigma}$=9 pixels. (Optional)
```

It is also important to remark that identifying the instability name for each time-frequency structure is an ill-posed problem as the spectrogram alone does not possess enough information. However, the methods introduced in this thesis can be enhanced by combining information from other diagnostics or using robust theory-based results from physics. For instance, it is easy to add 3D phase information from other magnetic sensors via mode number analysis or coherence of the signals, this would help to separate crossing modes and label some instabilities. Moreover, physical theory or simulations can complete the analysis. For example, the Alfvén spectra can be calculated by contributing with useful prior frequencies corresponding to different instabilities. Further MHD numerical calculation can identify time-frequency properties of the plasma, inferring equilibrium information or kinetic effects.

In addition, it is to be noted that an application with a graphical user interface could be used to discard these minor labeling errors. For instance, a tool called *Wavystar* with a GUI has been implemented, available under request in JET. This application implements algorithms discussed in this work, and other analysis capabilities of transitory signals not treated in this thesis.

We conclude this section by mentioning that the computational resources needed to run these algorithms are very low. A common laptop (11th Gen Intel(R) Core(TM) i7-1185G7, 3.00GHz) can run these algorithms in a few seconds without special memory requirements (16-32 GB RAM). Also, the presented algorithms have few hyper-parameters. These facts make them suitable for mining big databases.

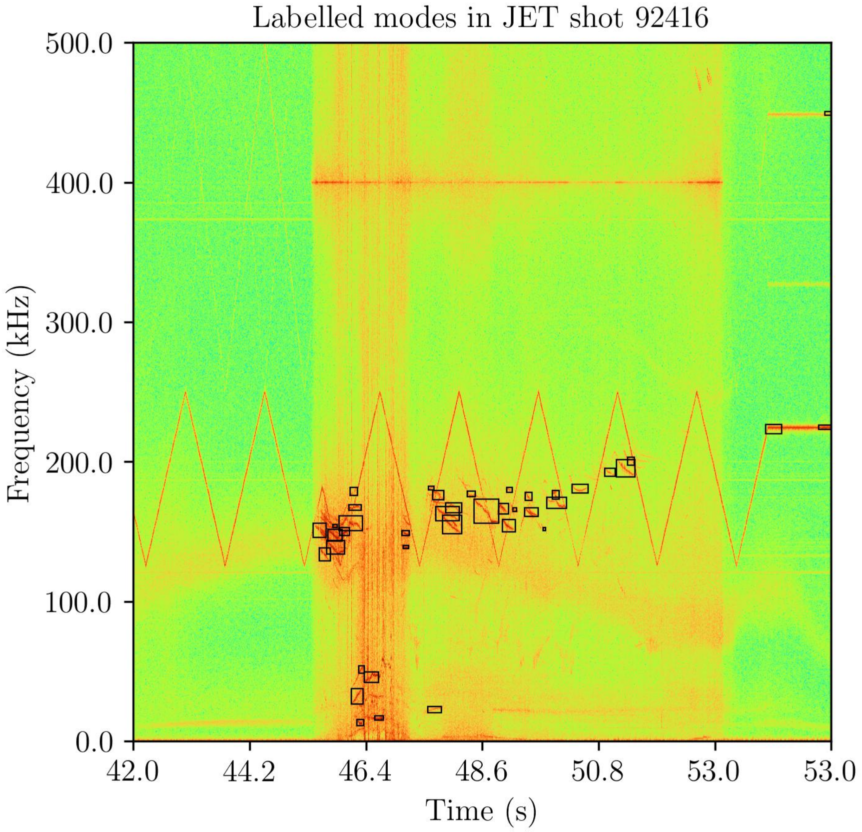


Figure 6.15.: Result of automatic labeling over original spectrogram. In total, 43 modes are segmented and numbered correctly. Note that only modes 0 and 1 correspond to part of the TAE antenna signal.

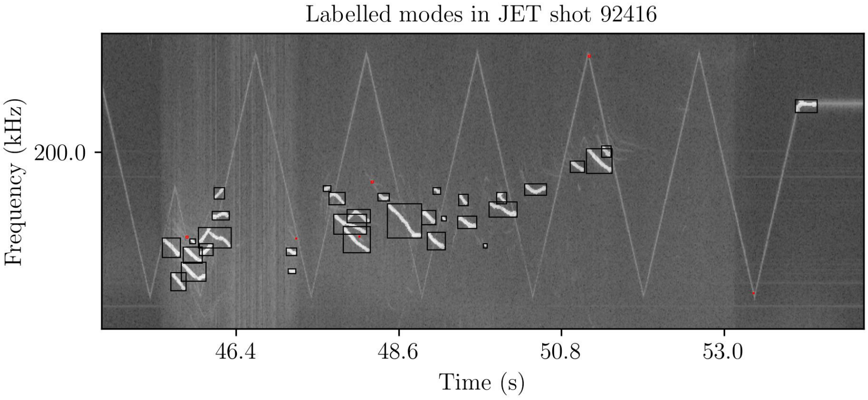


Figure 6.16.: Detail of Result of automatic labeling: red squares are modes with less than 9 pixels (discarded), black squares are modes (detected).

6.5. Application to interferometer spectrograms

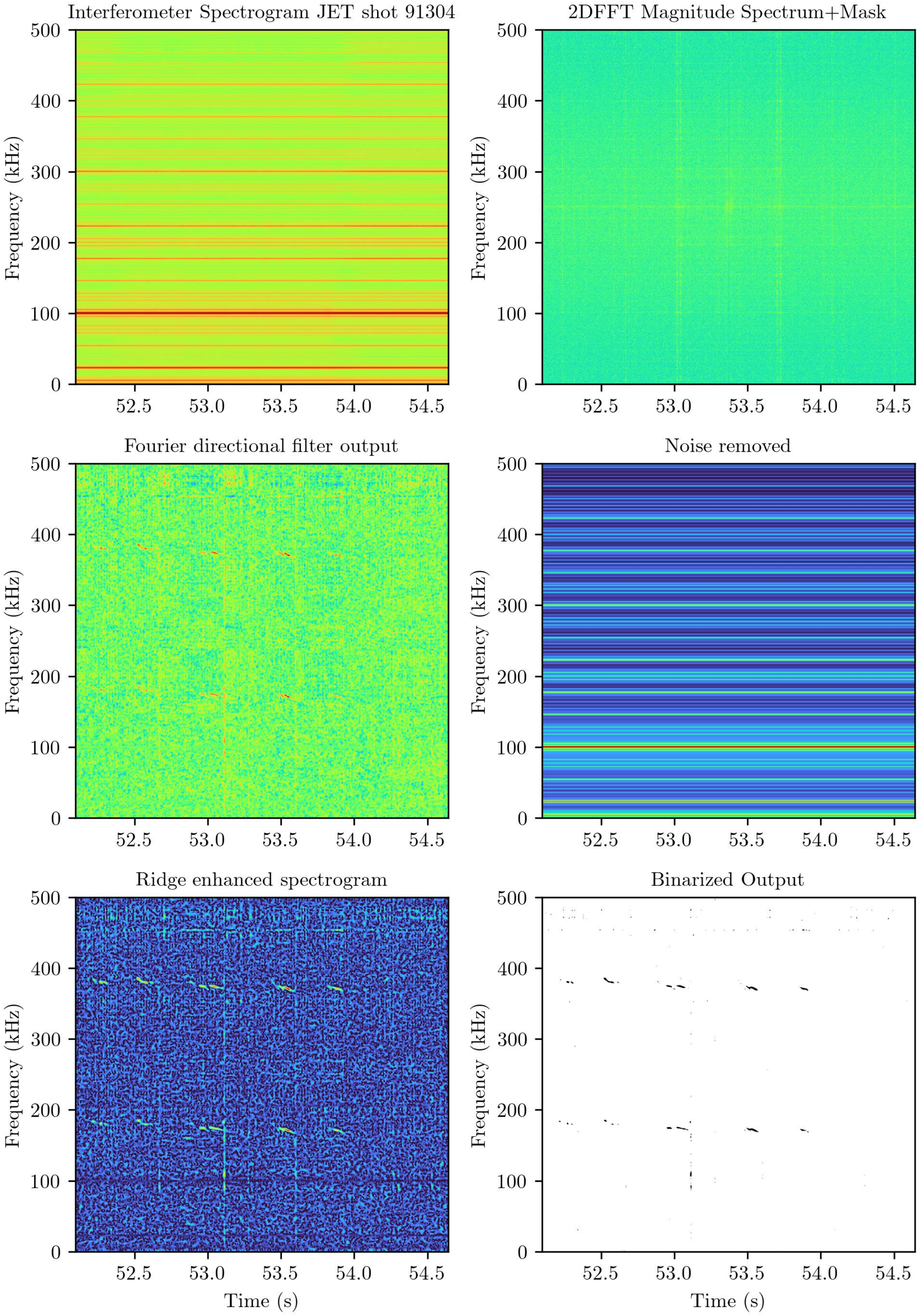


Figure 6.17.: Algorithms of feature extraction applied to the Fast infrared FIR interferometry signal showing modes present in the plasma core of JET shot 91304.

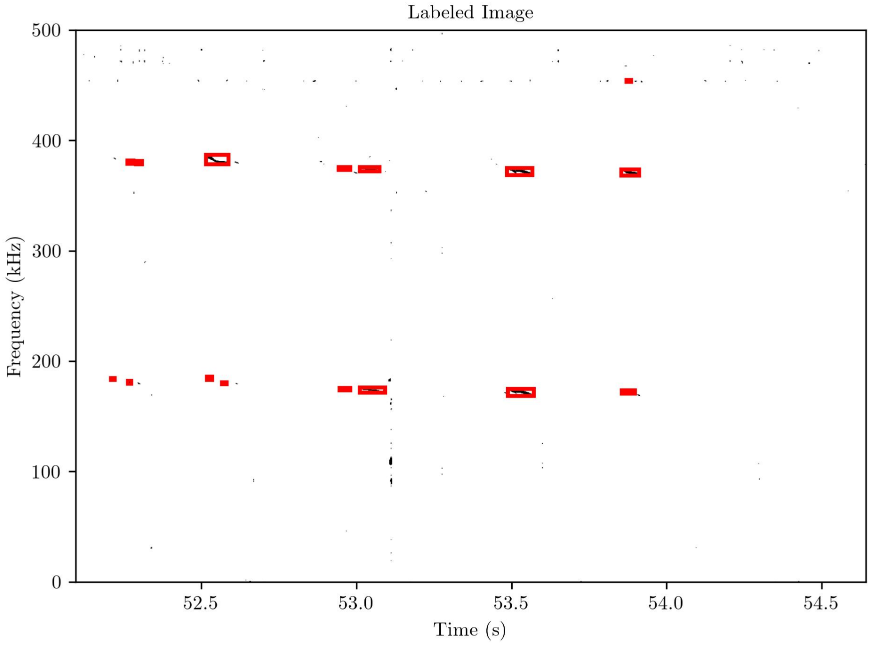


Figure 6.18.: Results of labeling using box properties (area>10 pixels, eccentricity>0.6, and horizontal orientation) to filter out noise.

Laser interferometers are a very useful tool for plasma diagnostics. Their working principle is very simple, a far infrared laser beam is split into two equal-length beams, with one of them passing through the plasma. After passing through the plasma, they are recombined creating an interference pattern that reveals a phase difference between incoming beams, which is proportional to the integrated electron density across the plasma path $\Delta\phi \propto \int n_e dl$ as detailed in Wesson's book [WC11].

Interferometers can be designed so multiple lines of sight can pass through the plasma. This is very useful for example to detect instabilities in the core that are not present in the plasma edge and vice versa. The signal sampling can be very high so is possible to create spectrograms.

Depending on the machine's design, more or less optical components are mounted in the vessel. The mechanical oscillations can create spurious patterns in the spectrogram. In addition, there can be other unidentified sources of noise or electric interference that can affect the resulting spectrogram. In Fig. 6.17 (first panel) we can see a spectrogram of JET's fast FIR interferometer (KG1F) channel 3. We can see that, at least without zooming the image is not possible to see any modes.

In the previous chapter, we introduced how to create masks in Fourier space, we can use a very thin mask in Fourier space to remove horizontal lines in the spectrogram (low). This mask is so narrow in the vertical direction (1 pixel width in Fig. 6.17 second panel) that it is barely visible. This kind of filter is known as a notch reject filter (page 344 of Ref. [GW18]).

In the second row of Fig. 6.17 it can be seen the result of filtering the spectrogram, and the removed noise from the image. Now it is possible to distinguish some modes structures present in the plasma core, just below 200 and 400 kHz. In the third row of Fig. 6.17 the result of thresholding the spectrogram

Finally, the labeling algorithm has been applied to the binary image identifying the modes in Fig.6.18. To filter out remaining noisy structures, the region properties have been use to only include pixel regions with area>10 pixels (big enough), eccentricity>0.6 (elongated enough), and with $\|\theta\| > 0.1\pi$ (horizontal enough).

It is worth mentioning that thanks to a recent collaboration with Princeton Plasma Physics Laboratory using DIII-D data. The presented result has been reproduced with the DIII-D's CO2 interferometer. Although permission to display this data in this thesis work has not been requested.

We can conclude that this CV pipeline, algorithm 9, can help to detect instabilities in spectrograms that are not visible to the bare eye. This method for removing noise from spectrograms has been presented for the first time in this thesis work, and it could help in analyzing interferometer data from different fusion devices.

Algorithm 9: Segmentation of interferometer spectrograms

```
Input: $\boldsymbol{s}$
Parameters: $\boldsymbol{\sigma} = [\sigma_1, ...\sigma_N]$
Output: $S_{xx(binarized)}$
Steps:
STFT($\boldsymbol{s}$) to obtain spectrogram
Apply notch filter (Algorithm 5 with $\theta = [+\pi/2, -\pi/2]$, $\Delta\theta = 1$, $\delta$ = image size) to
  remove horizontal pattens.
Apply a Ridge detector with $\sigma = [1,2,3,4]$ using Sato's method or equivalent.
Calculate threshold $T$ using Yen's method or equivalent.
Binarize the image I(x,y) using $T$
Label connected regions.
Filter out noisy regions, using area, eccentricity and orientation.
```

6.6. Segmentation of a shot spectrogram with fishbones

Under certain scenarios, especially when heating power is high, there is an interaction between the energetic particles generated and the $n = 1$, $m = 1$ surface, generating a kink perturbation, page 400 of Ref. [WC11]. This perturbation rapidly increases oscillation frequency, while the magnitude of perturbation quickly grows and diminishes. The time-voltage signal drawn in Mirnov coils resembles a fishbone, naming the perturbation. This cycle of rapids frequency burst repeats quasiperiodically, creating almost vertical mode patterns in spectrograms.

In the algorithms introduced in the last chapter. We assumed that all the information contained in the vertical direction was noise or undesired artifacts. It is important to remark that if we are modeling noise and undesired features in a spectrogram through the selection of one direction, it is necessary to check this prior assumption while using the algorithms introduced in this thesis. This is not the case when rapid sweeping frequencies of physical origin like fishbones or energetic particle modes are present. The algorithm removes this type of modes and the information removed should not be discarded.

If the algorithms for feature extraction or noise removal are being used *before segmentation*, it is convenient to examine the result and the removed-noise image. Then evaluate how much information has been removed from the original spectrogram. If MHD activity has been removed, several approaches are valid: (1) if all vertical modes are removed, it is necessary to continue the analysis on the vertical components, this way vertical and horizontal components can be segmented separately and finally merged together; (2) use soft thresholding to ($\alpha_V = 0.5$ or greater) to adjust the subtracted fraction of the vertical component on proceed with segmentation (this modes are strongly driven and contrast is good to segment them); (3) use the approximation image of level 1 or 2 instead, because keeping one of the first approximation sub-images can also result in a spectrogram with an improved signal-to-noise ratio, though omitting details sub-images could produce modes with coarser details.

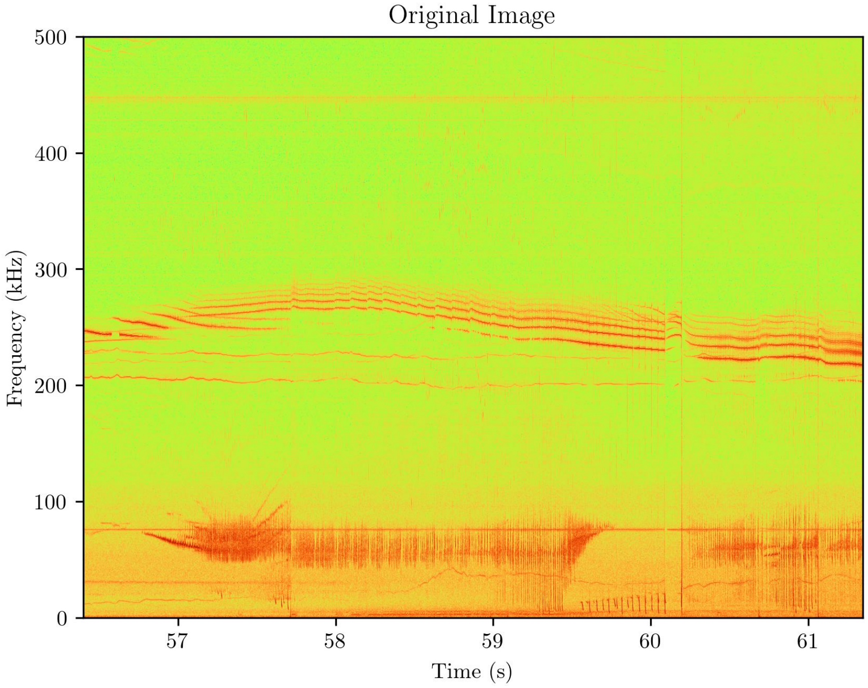


Figure 6.19.: JET shot 54300 with abundant Fishbone activity

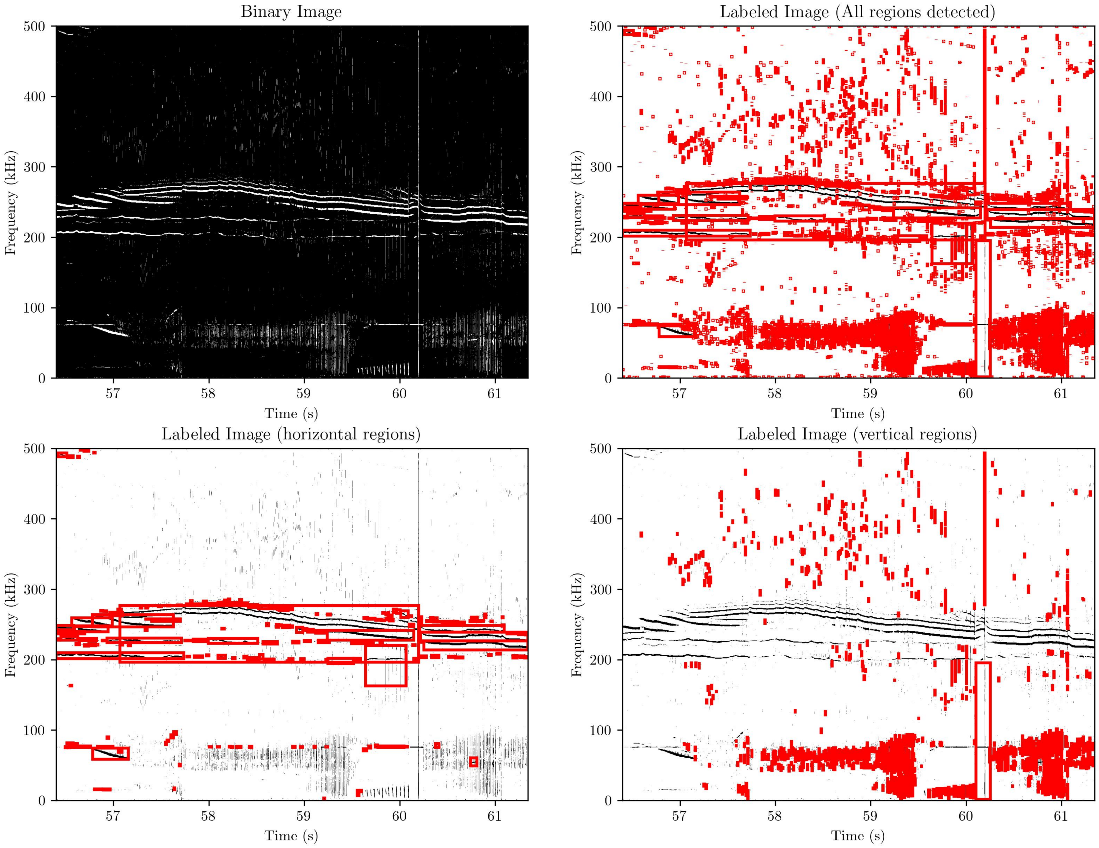


Figure 6.20.: Results of labeling of JET shot 54300 (red-boxes around detected modes), region properties (area-orientation-eccentricity) have been used to separate quasi-stationary modes from fast time-frequency chirping activity, and noise.

A simpler approach, if the segmentation is applied first (algorithm 7), consists of using the region properties *after segmentation* to separate and filter mode stricture the result.

JET shot 54300 in Fig. 6.19 is an example of a shot with fishbones. This shot is studied in [Zon+09], where rapid frequency sweeping modes are visible. To preprocess this shot, a combination of techniques introduced in this thesis can be used. However, if the segmentation is applied first (algorithm 7), the region properties can be used to filter the result *after segmentation*.

In Fig.6.20, the region orientation is used to separate the result of segmentation in vertical and horizontal-oriented structures. In addition, small pixel regions and roughly circular regions are discarded. Remarkably, most horizontal and vertical regions are separately succesfully and many regions detected that appear to be noise can be discarded. However, some crossing regions create bigger bounding boxes, in which the algorithm struggles to separate longer mode structures. Future work could look to tackle this limitation.

An alternative to treat this problem would be to preprocess the spectrogram with the spectral algorithms. To begin with, DWT could be used to remove some noise, then continuous horizontal straight lines should be removed by the 2DFFT mask, and finally, the ridge and threshold operators could be applied to the vertical and horizontal images separately so both results are merged together at the end.

It is also important to note that adding mode number information (or coherent phase) to each pixel region could help to separate crossing MHD time-frequency signatures. As discussed in Ref. [KAC22] time-frequency signatures can become non-overlapping in the time-frequency-spatial domain.

We can conclude that ridge detection and thresholding could be used to analyze spectrograms of shots with fishbones or rapid frequency bursts. However, the analysis needs to account for the different pattern orientations in the preprocess before the segmentation, or after the segmentation.

6.7. Detection of MHD modes using transfer learning and deep learning

Finally, we revisit TJ-II data analysis performed in [Bus+21] using computer vision algorithms to demonstrate the use of transfer learning when only very few amount of data is available. Transfer learning is the process in which pre-trained models are fine-tuned to work with a specific dataset.

The dataset used in this study consists of 500 spectrogram images from the TJ-II stellarator manually annotated to mark regions of MHD mode activity [Bus+21]. The images are RGB, though all the information is contained in a single channel, and they capture frequencies up to 500 kHz. The dataset was divided into training (70%), validation (15%), and test (15%) sets, we keep same partitions so we can make a fair comparison. Spectrograms were calculated from 1 ms intervals, providing 256 Fourier modes per image. The diversity of discharges ensures the dataset's representativeness, including both successful and failed plasma experiments (empty spectrograms).

The authors of Ref. [Bus+21] used three encoder architectures—VGG, MobileNet, and a simple Vanilla CNN—paired with two decoder architectures, FCN8 and FCN32, for image segmentation. After the neural network predicts mode locations, a blob detection algorithm is used to clean up the noisy outputs and identify relevant modes. The effectiveness of the model is evaluated using Intersection over Union (IoU) and Area Under the Curve (AUC) metrics. The best Jaccard index obtained was 0.427, and the AUC was 0.599, indicating a reasonable mode detection capability. The work of Bustos et al. [Bus+21] serves as our state-of-the-art benchmark.

The pre-trained models used in our work are: SAM [Kir+23] (Segment Anything Model), Detectron2 [Wu+19; 24b] (Facebook's open-source framework for computer vision), and YOLOv8 [Ult] (an open-source CV framework from Ultralytics). These frameworks include state-of-the-art models for segmentation and panoptic segmentation. They are open source and easy to use (an important source to learn how to use them was Ref. [Bha]).

SAM is Facebook's open-source model for instance and panoptic segmentation as well as prompt-based segmentation. The unique feature of the SAM model is that it was trained with 1 billion images to work with image prompts. It does not have labels; the human user draws boxes, points, or writes text to indicate the regions to segment. Its architecture is based on vision transformers [Dos+21], and for this application, we repurposed the mask generator to identify the MHD activity.

The Detectron2 model used in this experiment is the Mask R-CNN (Mask Region-based Convolutional Neural Network [Gir+14; 24c]). It was trained with the COCO dataset, which contains 80 classes of objects. It performs panoptic segmentation, which is simultaneous instance and semantic segmentation (it determines the regions and the class name of the regions).

YOLO stands for "you only look once" [Red+16]. This model is state-of-the-art for object detection and instance segmentation in real time. In contrast to SAM and Detectron2, which are optimized for performance, YOLO is optimized for speed, capable of performing panoptic segmentation at 30 fps. YOLOv8's architecture is based on a Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) [Rei+24], and it is designed to process image feature extraction and labeling in one unified step.

Both Detectron2 and YOLO can easily implement data augmentation (DA) to improve model performance. Data augmentation is a technique used to increase the size of the training dataset by applying transformations to the images. These image modifications are not stored; instead, the augmentations are performed online. For Detectron2, we used random brightness changes, random flips, and random crops. For YOLO, we used the default segmentation techniques, as well as "copy-paste" [Ghi+20] and "mixup" with a 50% probability.

The number of epochs was 10 for SAM, approximately 743 for Detectron2, and 100 for YOLO, respectively. For Detectron2, the maximum number of iterations is 1000 wich translates to 743 epochs: $N\_EPOCHS = \frac{\text{MAX\_ITER}\times\text{BATCH\_SIZE}}{\text{TOTAL\_NUM\_IMAGES}} = \frac{1000\times256}{345} \approx 743$.

To fine tune the mask decoder of SAM, we needed to select a loss function apropiate for segmentation. We found that using a weighted loss compossed of DICE loss [Sud+17] and FOCAL loss [Lin+18] offers the best result. The weights of the loss function were optimized using a Bayesian optimization framework called OPTUNA [Aki+19], the result is shown in appendix E. This way, we found that FOCAL loss is more relevant than DICE loss for this application, as it is designed to handle imbalanced datasets. Finally we set the weights to be 0.05 for DICE loss and 0.95 for FOCAL loss.

The evaluation metrics for binary segmentation are detailed in Appendix A, Table 4. We aimed to improve the Jaccard index, also known as Intersection over Union (IoU). However, the different metrics reveal various aspects of the model's performance. It is also important to improve precision and recall, which indicate the model's ability to recognize the positive and negative classes. The normalized values are displayed, as most metrics are based on the confusion matrix. The F1 score is also a good indicator of performance, especially since the positive and negative classes are unbalanced (there are more pixels without objects than with objects). As a result of this imbalance, we can observe that accuracy is insensitive to changes in the model's performance.

Metric	Bustos et al.	SAM	**Detectron2**	YOLO	Detectron2 (No DA)	YOLO (No DA)	CV
accuracy	0.991	0.988	0.988	0.988	0.988	0.988	0.987
Jaccard	0.427	0.486	**0.498**	0.478	0.472	0.496	0.238
ROC AUC	0.599	0.857	0.825	0.83	0.833	0.838	**0.876**
f1	(N/A)	0.637	**0.654**	0.635	0.626	0.651	0.362
precision	(N/A)	0.631	**0.693**	0.652	0.629	0.662	0.262
recall	(N/A)	0.721	0.655	0.673	0.673	0.682	**0.811**
tn	**0.996**	0.993	0.946	0.924	0.992	0.994	0.893
fp	0.004	0.007	0.054	0.076	0.008	**0.006**	0.107
fn	0.421	0.308	0.342	0.330	**0.005**	0.323	0.213
tp	0.579	0.692	0.658	0.670	**0.995**	0.676	0.787

Table 6.2.: Comparison of Bustos et al [Bus+21]. results with SAM, Detectron2, YOLOv8 (with and without DA), and CV pipeline. The best results in each row are indicated in bold. Overall, Detectron2 with DA obtained the best performance.

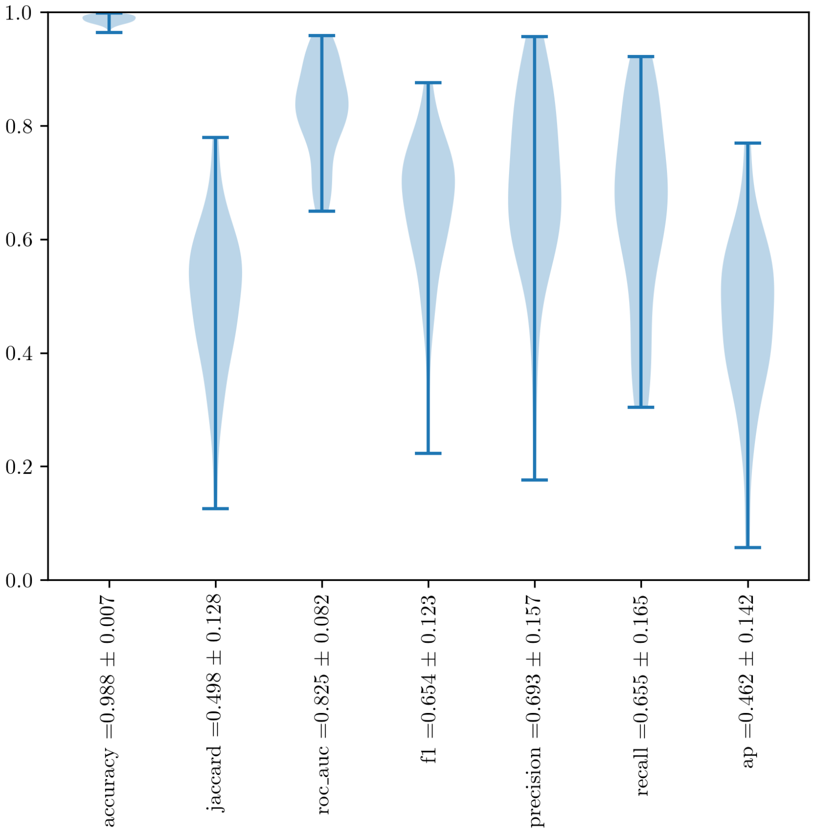


Figure 6.21.: Metric distributions for the best model in the test set (Detectron2 with data augmentation).

The results of the different experiments in the test set are presented in Table 6.2. We included experiments with and without DA for Detectron2 and YOLO. In addition, a CV pipeline has been designed for this dataset: (2D FFT for removing noise (horizontal and vertical lines) - Sato's ridge detector - Yen method for thresholding - (lower 10% of the frequency range is removed)).

In Fig. 6.21, the distributions of the metrics for the best model in the test set are shown. Moreover, Appendix D and Table 7 include the values of the metrics along with their standard deviations. It is crucial to acknowledge that performance metrics exhibit an error distribution function with multiple modalities. This implies that, within the error distributions, distinct groups of spectrograms can be identified, and that the segmentation of some image groups is more effective than that of others.

The results show that transfer learning using Detectron2 achieves a new state-of-the-art segmentation of MHD activity: a Jaccard index of 0.491 ± 0.130 on the test dataset (the previous best Jaccard index was 0.427 [Bus+21]).

There are some caveats though. The Jaccard Index can be improved by increasing the area of the modes detected, meaning that missing small and thin modes does not impact the Jaccard Index. In other words, the Jaccard Index can be improved by detecting the larger modes first, even if smaller ones are missed. This is why monitoring the F1 score is important for assessing model performance, which is also the case for the Detectron2 model.

In addition, it is also good to visually inspect the results of the inference for each model. In Figures 6.22, 6.23, 6.24, 6.25, we compare one example of TJ-II shot 38446 for SAM, Detectron2, YOLO, and the CV pipeline, respectively. In all figures, the input of the model and the ground truth as provided by the human labelling are on the left column, while the output over the spectrogram and the binary mask are displayed on the right. In the case of SAM, a probability map is plotted instead.

There is variability in the output of the different models. We can appreciate that the deep learning models have learned to join modes of different intensities, as the human did when labeling the data. SAM is the only model whose output has some noisy points as a result of fluctuations in the probability map. The regions determined by Detectron2 and the CV pipeline have particularly smooth contours. Furthermore, some modes that were not labeled by the human are also detected.

The results of the CV pipeline are not as good as the deep learning models. This is because to handle shots without modes, only Yen's thresholding can be used, which sets a very high threshold. Note that the CV pipeline has the best recall but the worst precision. This implies that the model can lose pixels from the mode structures, as can be seen in Fig. 6.25. However, we should note that future research could aim to improve this outcome. As illustrated in the results of Fig. 6.26, there is potential for improvement in the CV pipeline.

Therefore, using directly the same parameters (and components) for the CV pipeline on image dataset is detrimental to the model's performance. However, if we compare the modes detected in Fig. 6.26 with those in SAM (Fig. 6.22), we can see that both the CV pipeline and SAM are capable of detecting modes that were not labeled by humans. The differences in criteria between different data labelers can also introduce a source of error when training and evaluating these models, making model comparisons less straightforward. With this result, we can affirm that SAM and the CV pipelines could be very useful tools for labeling datasets in a semi-supervised way, with superior consistency to human labeling.

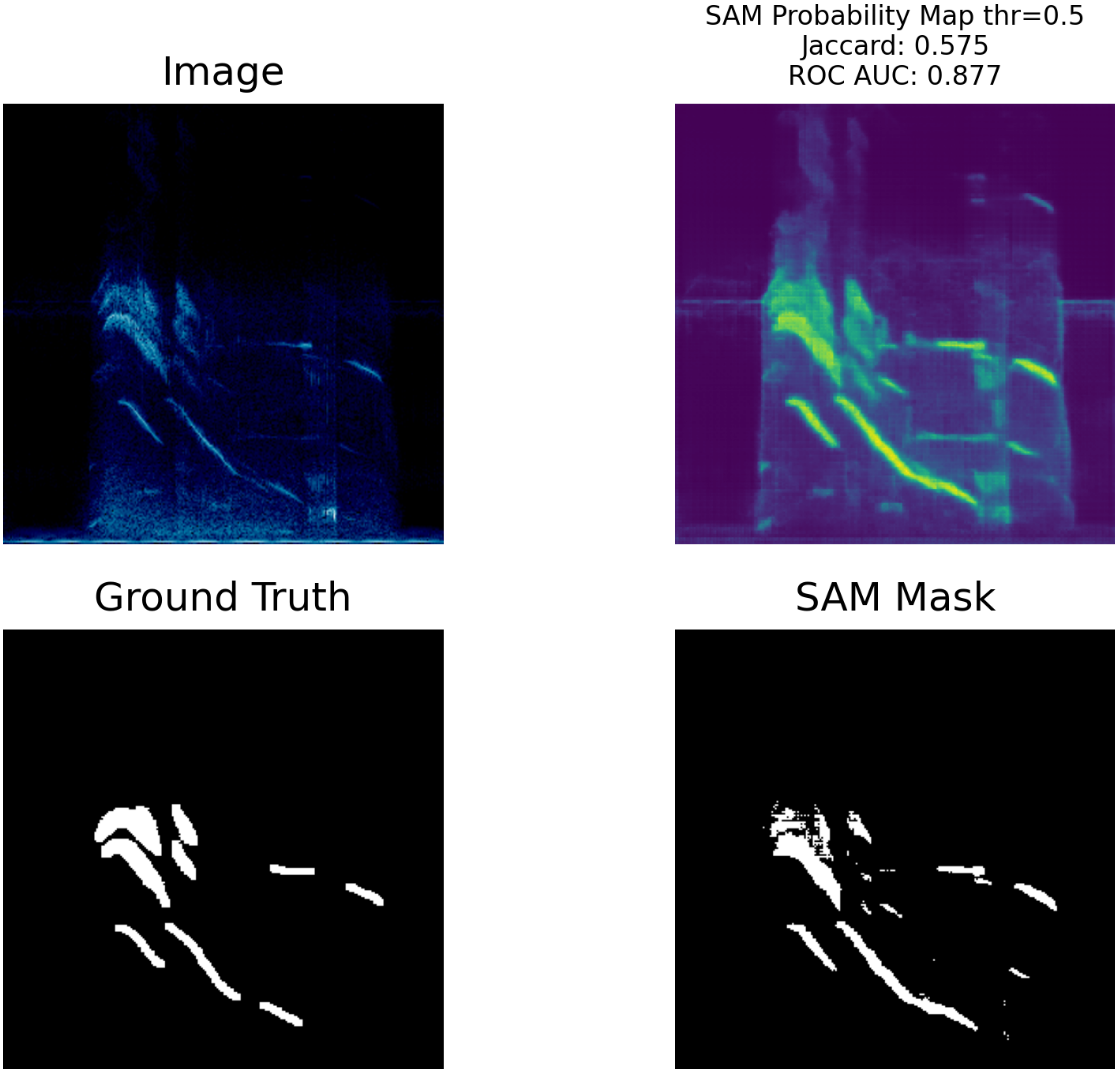


Figure 6.22.: Results of SAM on TJ-II shot 38446: On the right columns, the results of SAM inference are shown. SAM uses a probability map to segment the objects in images.

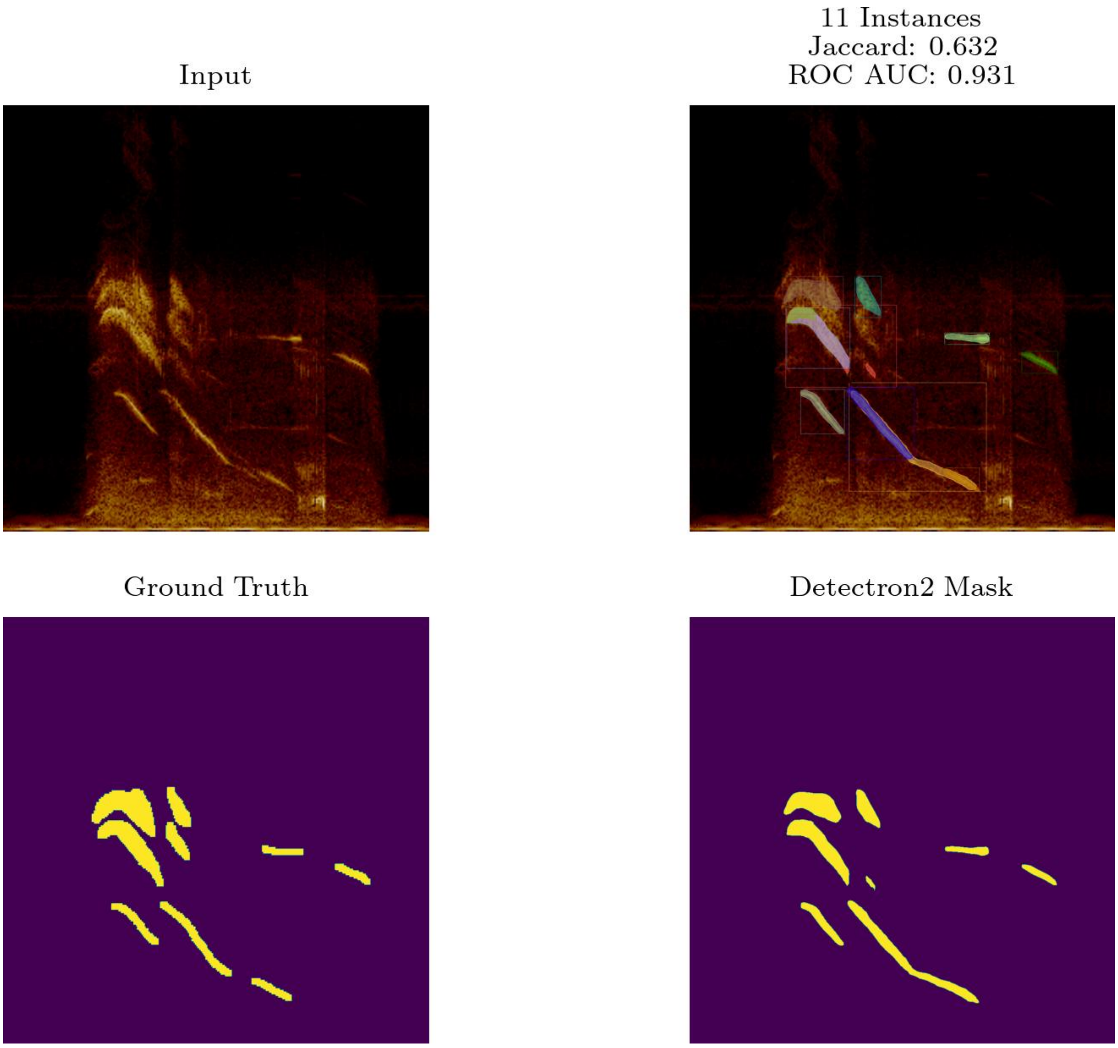
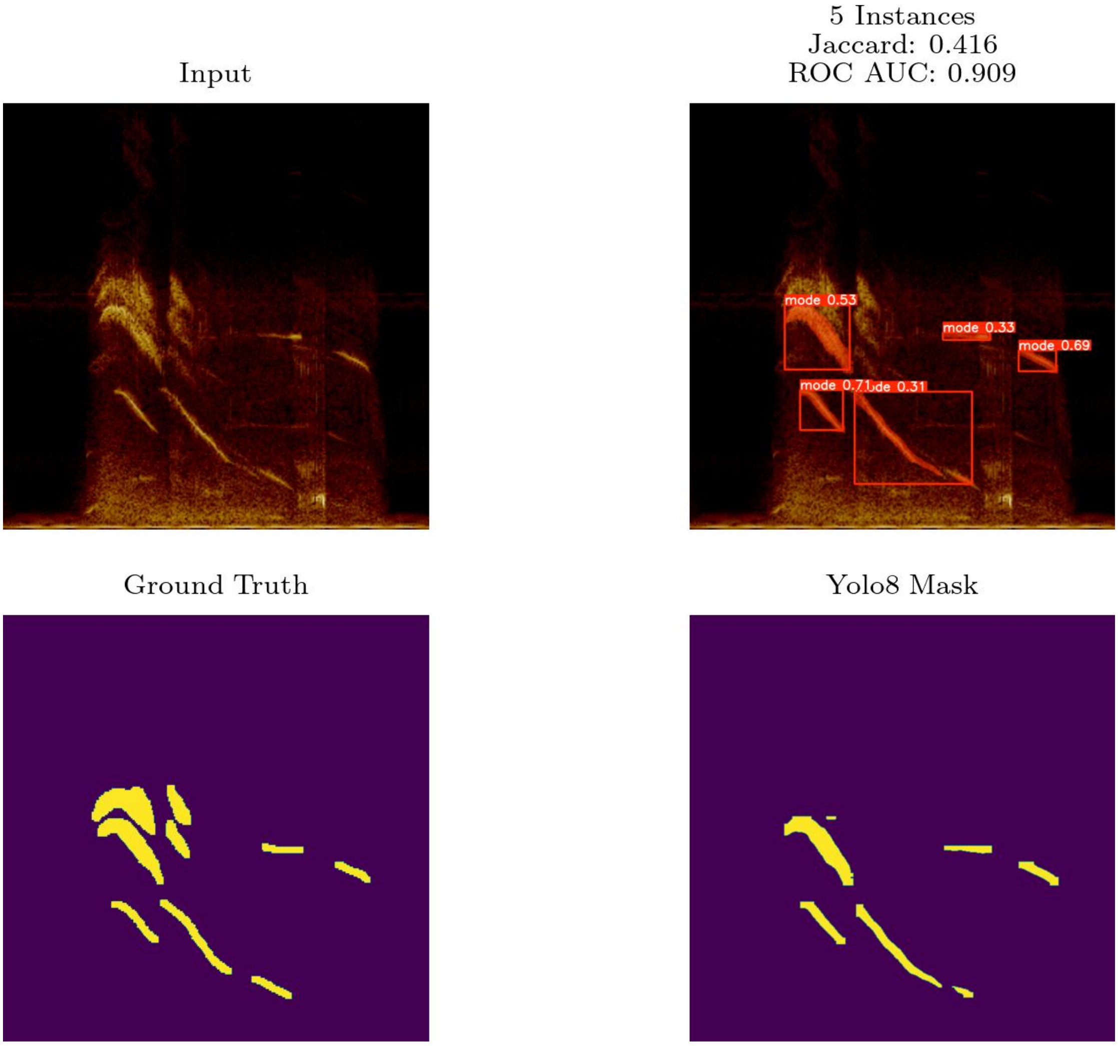


Figure 6.23.: Results of Detectron2 on TJ-II shot 38446: 11 modes are detected (some of them overlap).



Input

5 Instances
Jaccard: 0.416
ROC AUC: 0.909

Ground Truth

Yolo8 Mask

Figure 6.24.: Results of YOLOv8 on TJ-II shot 38446: 5 modes are detected, missing some in this case. A probability is shown in the bounding boxes of the masks generated.

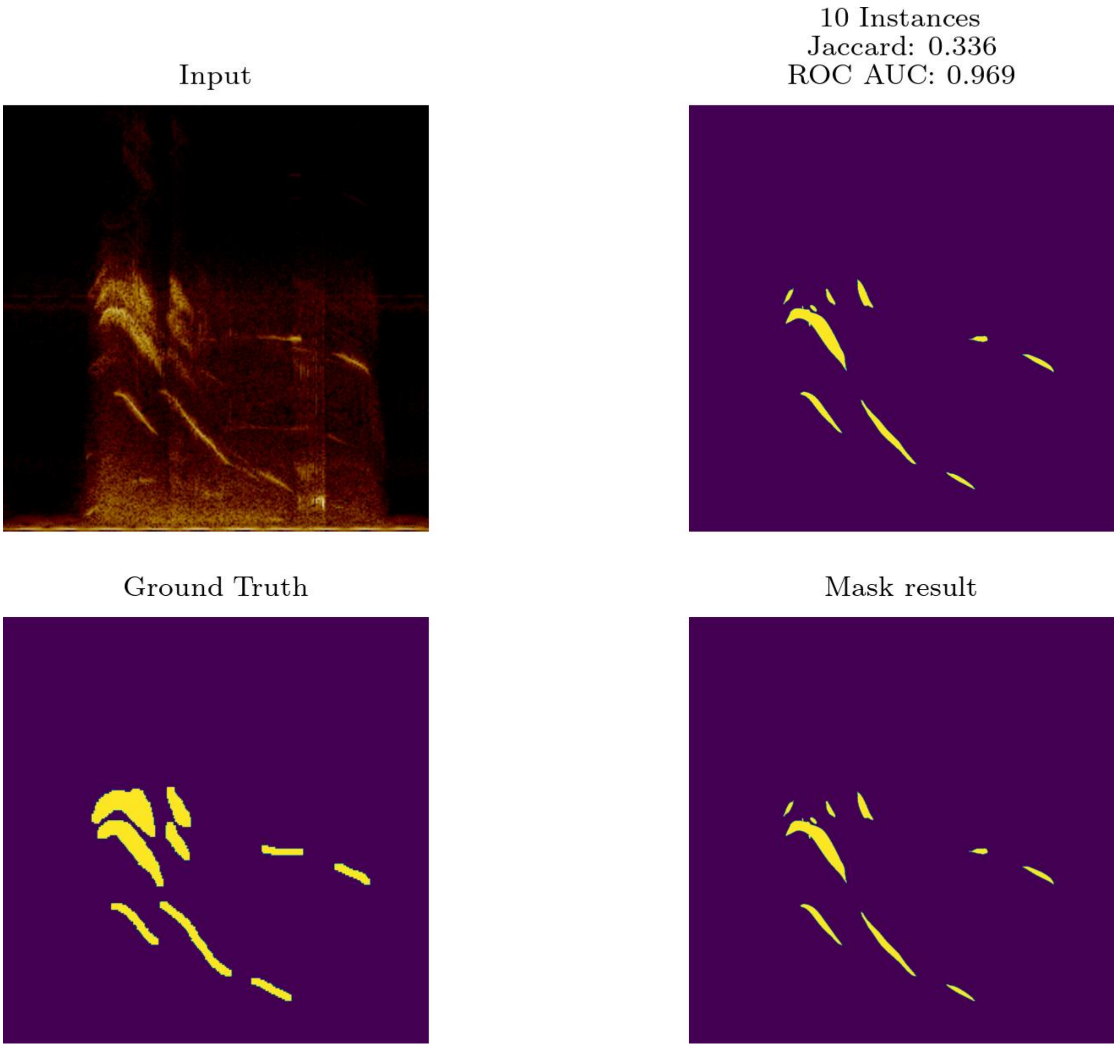


Figure 6.25.: Results of the CV pipeline on TJ-II shot 38446: 10 modes are detected, close to Detectron2 results. However, the area of the retrieved modes is smaller, which is detrimental.

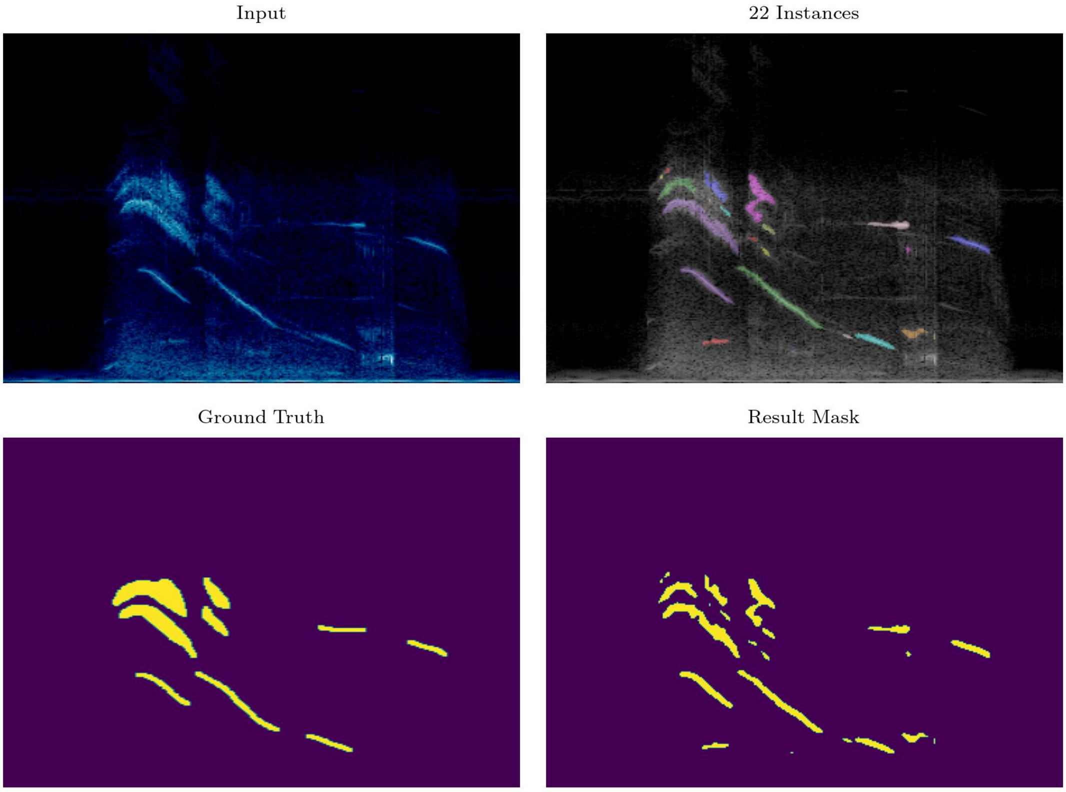


Figure 6.26.: Results of the CV pipeline (fine-tuned) on TJ-II shot. This pipeline uses Meijering's ridge detector and Otsu thresholding only. The result is comparable with SAM. Note the increased detail when labeling.

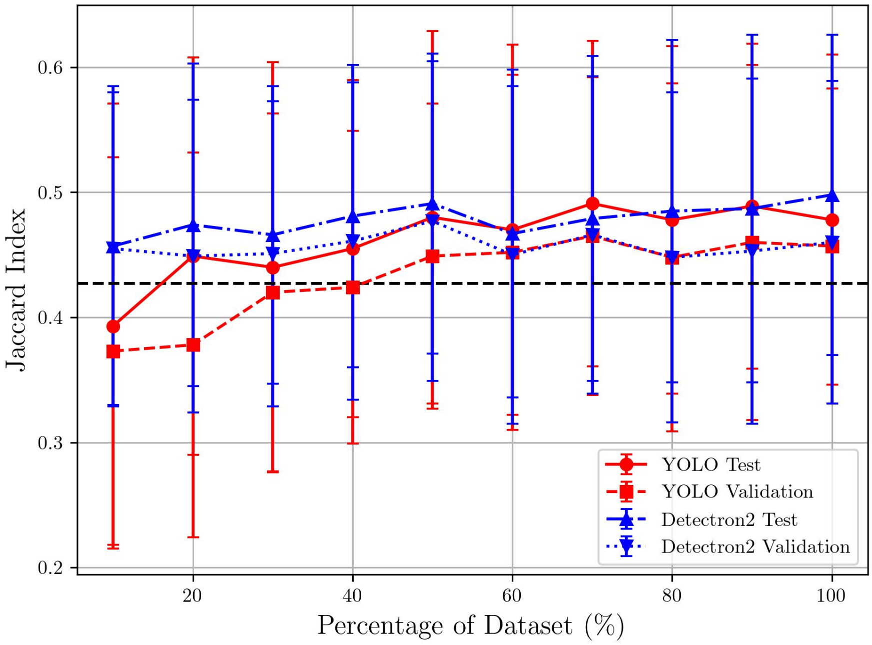


Figure 6.27.: Results of Detectron2 and YOLOv8 (on the test and validation dataset) with varying percentages of the training dataset. Data augmentation is used.

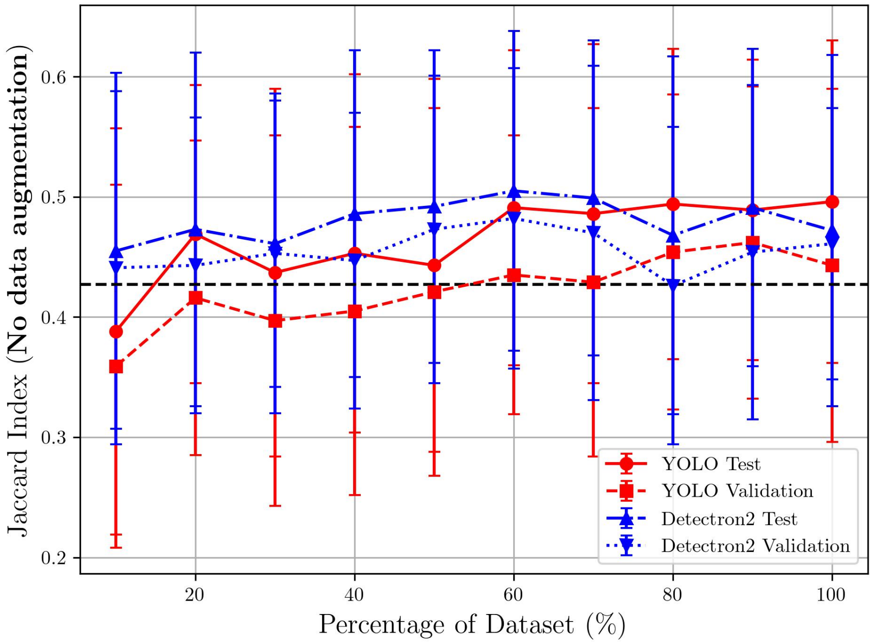


Figure 6.28.: Results of Detectron2 and YOLOv8 (on the test and validation dataset) with varying percentages of the training dataset. No data augmentation is used.

Data with specific physics like Alfvén activity in recently commisionned machines may be scarce, which motivates the use of models with limited data. We apply transfer learning with a decreasing percentage of the training dataset. The results are shown in Figures 6.27 and 6.28. We can see how the Jaccard index decreases slightly as the percentage of the training dataset decreases. However, the performance remains reasonable.

Our reference benchmark [Bus+21] with a Jaccard index of 0.427 is indicated as a horizontal dashed black line. We can see that the effect of data augmentation is subtle but reduces the error bars of the Jaccard index. Additionally, we can see that the results of YOLOv8 are comparable, although Detectron2 has less variance in the error bars. However, YOLOv8 has the advantage of having the fastest inference time, only 8 ms for this dataset. If the spectrogram could be plotted in real-time, this architecture could be used for real-time detection in fusion devices with longer shots.

The minimum data percentage needed to achieve a Jaccard index greater than 0.427 corresponds to 5% of the original dataset (not displayed in the figures), which is approximately 18 spectrogram images in the training dataset (for Detectron2), with a result of Jaccard index = 0.438 ± 0.13. This demonstrates that very few labeled data are needed to train these models. This result is of the most interest for applying machine learning mode detection in new fusion devices, where the amount of labeled data is scarce. However, the behavior in the case of multi-class classification (type of modes) remains to be tested.

6.8. Summary

In this chapter, we introduced the use of segmentation for labeling MHD activity. The application of computer vision techniques, particularly ridge detection and thresholding, for segmenting spectrograms has been explored. These methods allow us to distinguish between regions of interest in spectrograms, such as MHD activity, and the background noise. We compared various segmentation approaches, including different combinations of ridge filters and global thresholding algorithms with feature extraction and denoising techniques presented in Chapter 2.

This chapter emphasizes the importance of feature enhancement techniques like ridge detection, which improves the accuracy of segmentation by enhancing mode structures before applying thresholding. Different ridge detection algorithms, including Sato's, Meijering's, and Frangi's methods, were tested on spectrograms, with Yen's method proving effective for thresholding after feature enhancement.

We also examined the application of these methods to specific cases, such as interferometer spectrograms and shots with fishbone activity. The removal of periodic patterns in interferometer spectrograms can reveal hidden features and help to localize MHD activity. For preprocessing shots with fishbones, we discussed several possibilities. Then, for post-processing after segmentation, a practical example was given.

The comparison of the CV pipeline with deep learning models gives another perspective on model performance. The CV pipeline is a good alternative when the amount of data is scarce, and computational resources are limited, and it can be used to label the data in a semi-supervised way. The deep learning models outperform the CV pipeline in terms of precision and recall because small adjustments need to be made for each image.

The results of the deep learning models improve our previous state-of-the-art benchmark, with Detectron2 achieving the best performance. However, the CV pipeline and SAM are capable of detecting modes that were not labeled by humans. This demonstrates the limitations of human labeling and the potential of these models to label datasets in a semi-supervised way with superior consistency compared to human labeling.

Finally, we explored the possibility of using deep learning models with limited data. The results show that the performance of the models remains reasonable even with a small percentage of the training dataset. This is particularly relevant for applying machine learning mode detection in new fusion devices where the amount of labeled data is scarce. If this can be achieved with the YOLOv8 model, modes might be detected in spectrograms with a lag of only a few milliseconds, in addition to the time required for plotting the spectrogram. This information could be utilized in other control systems or algorithms.

Conclusion

The results of this thesis work demonstrate the potential of unsupervised machine learning (ML) and signal processing techniques to automate the detection and classification of plasma modes in fusion experiments.

In chapters 3 and 4, we successfully illustrated the practical use of sparse coding in plasma instabilities detection. Overall, the results shown using our new proposed algorithm called ERMD are promising. However, it has some practical limitations, mostly the computational cost, and the hyperparameter tuning. Nevertheless, ERMD allows us to work on the time domain with raw signals, it can extract the Alfvénic pattern in a TJ-II sample of shots, and it is capable of outperforming its antecessor SRMD in less computing time.

This novel approach required algorithmic innovations as well. The major contributions to the sparse coding field are (1) the use of an in-GPU dictionary construction of Gabor's atoms, (2) the use of accumulated gradient descend for parallelizing the optimization algorithm in multiple GPUs with small signal batches, (3) the improvement of mode identification by using a variable density based clustering HDBSCAN, and (5) the acknowledge of the multicollinearity in random atoms dictionaries by using elastic net regularization to improve mode decomposition.

The use of MI information to capture linear and nonlinear relationships between variables is another innovation in mode detection, that provides more robustness against noise and outliers. Moreover, the clustering technique used for profiling the modes retrieved by ERMD has proven to be useful in summarizing mode activity in the 1291 shots. It complements the ERMD algorithm as it can be used to remove noisy clusters from the ERMD result.

Mirnov signals have a very high sampling rate, up to 2 MHz, which is the main factor slowing down the ERMD runs. Nevertheless, the algorithm can label more than 700 shots in a week, with consistency that cannot be provided by a human team. Though the technique is useful for inter-shot analysis, a ML surrogate model of ERMD could in principle detect Alfvén activity using raw Mirnov coil signals in real-time. Subsequent work might investigate the training of machine learning models using the created database for the detection of the Alfvénic class or the clustering sub-classes. In addition, researching the use of dictionary learning (learning dictionary elements while encoding) opens new possibilities to process plasma signals.

ERMD is adapted to analyze high-resolution, non-stationary time signals in any field. Future applications can be as variate as mode decomposition of heart electrocardiograms, seismic analysis, animal calls, or musical analysis. Owing to the broad application of ML and signal processing, our algorithm could impact other scientific disciplines as well.

In chapters 5 and 6, we have shown the potential of signal analysis and computer vision (CV) algorithms to filter noise and identify key features in spectrograms.

The main innovations introduced in this work are the use of wavelets, curvelets, along simple CV algorithms: to filter (reassign) the spectrograms of magnetic sensors. The results are satisfactory and the proposed algorithms are fully interpretable: the addition of removed noise and filtered spectrogram returns the original signal. Moreover, these algorithms do not require GPUs and the computational time is low. Although some parameter adjustment is needed, the algorithms show enough flexibility to be adjusted to data from different fusion devices. Which make them a good candidate for labeling spectrograms between shots.

The main concepts that allowed these results are two: (1) noise present in spectrograms is highly anisotropic, and (2) the plasma phenomena observed in spectrograms are multi-scale. Therefore, removing features in a certain scale and direction maintains a high fraction of the original information. This method could be particularly useful for the processing of spectrograms from interferometers, or the reflectometers, where the noise is present in the form of horizontal lines.

Nowadays, the state of the art in computer vision algorithms is the use of deep CNNs and other neural network architectures. However, they require large amounts of curated data. The application of classic computer vision algorithms has not been extensively researched in Fusion applications, this constitutes a gap of knowledge that this work aims to reduce. The results allow a flexible method that is capable of systematically labeling spectrogram data. The application of these classic algorithms can contribute to creating curated datasets, improving the performance of future deep learning algorithms.

Future work can be to incorporate mode numbers (masking the result of a mode analysis), and labels for different types of instabilities observed: Alfvén eigenmodes and others. This new analysis tool opens the door to new physical studies, such as investigating the relationship between AE and turbulence, or searching for scaling laws of MHD activity in large data sets.

The detection of different types of instabilities, with time-frequency-mode number information could be crucial to determine equilibrium properties through MHD spectroscopy, as the instabilities reveal equilibrium information that might be hard to retrieve using reconstruction codes. Finally, the database obtained can be used to train DL networks for real-time mode detection or prediction.

Finally, we demonstrated the use of transfer learning with state-of-the-art segmentation models. The accuracy and inference speed of YOLOv8 improves previous research and opens the possibility of using it for real-time detections if diagnostic spectrograms are plotted in real-time. Also, its capability to perform well with less data makes it a suitable solution for newly commissioned fusion devices where the number of shots is scarce.

Hopefully, the results presented in this work will contribute to the automation of fusion diagnostics, contributing in the future to the realization of real-time monitoring and control in fusion reactors by giving systematic way of label time-frequency data.

The difficulty of labeling systematically large amounts of data motivated the development of the algorithms introduced in this thesis. A principle of parsimony and the development of explainable algorithms have always driven the development of this research, leaving more "black-box" approaches such as deep learning for the final stage while prioritizing feature engineering and classical machine learning approaches (approaches I and II of Fig. 2.7).

The analysis of the raw time series was motivated by the fact that future online mode detectors may work with the raw signals from the Mirnov coils. Therefore, direct labelling of the raw signal is convenient in this use case. First, the sparsity assumption of the mode structures was considered, resulting in the development of ERMD.

In addition, to cope with plasma diagnostics' multimodality, a mode clustering method involving feature engineering with Mutual Information was envisioned. It has been shown that the proposed method is capable of recognizing real physical patterns and artifacts of the ERMD processing in a large database.

Although several aspects of the computation efficiency were improved, exploring faster and more efficient methods was necessary. The cumbersome parameter selection of ERMD also motivated finding a method with fewer free parameters.

Then, the use of a computer vision pipeline on spectrograms (directional filters-ridge-threshold-label) provided a fast and practical method for labeling data in a semi-supervised way. Finally, the use of transfer learning with deep learning segmentation models complemented the computer vision pipeline offering a fully adaptive and free-parameter method if enough labeled data is provided.

The methods proposed in this thesis identify time-frequency structures in single time traces from one individual Mirnov coil signal, or one individual spectrogram channel in an unsupervised way. The only alternative with similar performance is the manual selection of pixel's mode structures or training deep learning models with human labeled data [Bus+21].

Previous mode identification uses the phase information from different channels to workout mode numbers which can be used to identify time-frequency structures. However, this approach may suffer from the aliasing between different mode numbers and phase noise. As mentioned in section 2.5, also phase coherence between different signals can be used to enhance the mode trace. For these reasons, it is to be expected that future implementation of phase information could further improve the mode segmentation by computer vision.

As a final remark, given that labeling large databases of modes is a necessary step before training ML models (for real-time instability identification and prediction in fusion devices), this thesis work has the potential to impact many data science projects in fusion research.

Bibliography

[Abu+22] H. Abu-Shawareb et al. “Lawson Criterion for Ignition Exceeded in an Inertial Fusion Experiment”. en. In: *Physical Review Letters* 129.7 (Aug. 2022), p. 075001. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.129.075001. URL: https://link.aps.org/doi/10.1103/PhysRevLett.129.075001 (visited on 09/02/2024) (cit. on p. 5).

[Ach+12] R. Achanta et al. “SLIC Superpixels Compared to State-of-the-Art Superpixel Methods”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (Nov. 2012), pp. 2274–2282. ISSN: 0162-8828, 2160-9292. DOI: 10.1109/TPAMI.2012.120. URL: http://ieeexplore.ieee.org/document/6205760/ (visited on 08/19/2024) (cit. on p. 142).

[AR05] Tinku Acharya et al. *Image processing: principles and applications*. eng. Hoboken, NJ: Wiley-Interscience, 2005. ISBN: 978-0-471-71998-4 (cit. on pp. xi, 140, 145).

[Aki+19] Takuya Akiba et al. “Optuna: A Next-generation Hyperparameter Optimization Framework”. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019 (cit. on p. 169).

[Alf42] H. Alfvén. “Existence of Electromagnetic-Hydrodynamic Waves”. en. In: *Nature* 150.3805 (Oct. 1942). Number: 3805 Publisher: Nature Publishing Group, pp. 405–406. ISSN: 1476-4687. DOI: 10.1038/150405d0. URL: https://www.nature.com/articles/150405d0 (visited on 02/20/2024) (cit. on p. 6).

[ACt13] D Alves et al. “Kalman filter methods for real-time frequency and mode number estimation of MHD activity in tokamak plasmas”. en. In: *Plasma Physics and Controlled Fusion* 55.10 (Oct. 2013), p. 105003. ISSN: 0741-3335, 1361-6587. DOI: 10.1088/0741-3335/55/10/105003. URL: https://iopscience.iop.org/article/10.1088/0741-3335/55/10/105003 (visited on 03/05/2024) (cit. on p. 65).

[Are08] Félix Ares. *El robot enamorado: una historia de la inteligencia artificial*. spa. Primera edición. OCLC: 827791421. Barcelona: Ariel, 2008. ISBN: 978-84-344-5369-2 (cit. on p. 45).

[AV07] David Arthur et al. “k-means++: the advantages of careful seeding”. In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. SODA ’07. USA: Society for Industrial and Applied Mathematics, Jan. 2007, pp. 1027–1035. ISBN: 978-0-89871-624-5. (Visited on 02/26/2024) (cit. on pp. 38, 116).

[Azi12] E A Azizov. “Tokamaks: from A D Sakharov to the present (the 60-year history of tokamaks)”. en. In: (2012) (cit. on p. 3).

[Bah+24] Yasaman Bahri et al. “Explaining Neural Scaling Laws”. In: *Proceedings of the National Academy of Sciences* 121.27 (July 2024). arXiv:2102.06701 [cond-mat, stat], e2311878121. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.2311878121`. URL: `http://arxiv.org/abs/2102.06701` (visited on 09/23/2024) (cit. on p. 44).

[BLM14] Richard Barnes et al. “Priority-flood: An optimal depression-filling and watershed-labeling algorithm for digital elevation models”. en. In: *Computers & Geosciences* 62 (Jan. 2014), pp. 117–127. ISSN: 00983004. DOI: `10.1016/j.cageo.2013.04.024`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0098300413001337` (visited on 08/19/2024) (cit. on pp. 60, 142).

[Bha] S Bhattiprolu. *Python for microscopists Youtube channel and Code*. en. URL: `https://github.com/bnsreenu/python_for_microscopists/blob/master/%20and%20https://www.youtube.com/channel/UC34rW-HtPJulxr5wp2Xa04w` (visited on 09/16/2024) (cit. on p. 168).

[Bis16] C.M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer New York, 2016. ISBN: 9781493938438. URL: `https://books.google.fr/books?id=kOXDtAEACAAJ` (cit. on p. 37).

[BB04] José A. Bittencourt et al. *Fundamentals of plasma physics*. eng. 3. ed. New York Berlin Heidelberg Hong Kong London Milan Paris Tokyo: Springer, 2004. ISBN: 978-1-4419-1930-4 978-0-387-20975-3 (cit. on pp. 10, 15).

[Bla+] B D Blackwell et al. “Observations of Alfvénic MHD Activity in the H-1 Heliac”. en. In: () (cit. on p. 66).

[BRH20] Khawla Bnou et al. “A wavelet denoising approach based on unsupervised learning model”. en. In: *EURASIP Journal on Advances in Signal Processing* 2020.1 (Dec. 2020), p. 36. ISSN: 1687-6180. DOI: `10.1186/s13634-020-00693-4`. URL: `https://asp-eurasipjournals.springeropen.com/articles/10.1186/s13634-020-00693-4` (visited on 02/28/2024) (cit. on p. 127).

[BSA15] V. Bolón-Canedo et al. *Feature Selection for High-Dimensional Data*. Artificial Intelligence: Foundations, Theory, and Algorithms. Springer International Publishing, 2015. ISBN: 9783319218588. URL: `https://books.google.es/books?id=QgO0CgAAQBAJ` (cit. on pp. 42, 43, 218).

[Boo21] Allen H. Boozer. “Stellarators as a fast path to fusion”. en. In: *Nuclear Fusion* 61.9 (Sept. 2021), p. 096024. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ac170f`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ac170f` (visited on 09/02/2024) (cit. on pp. 5, 22).

[BK99] Duarte Borba et al. “CASTOR-K: stability analysis of Alfvén eigenmodes in the presence of energetic ions in tokamaks”. In: *Journal of Computational Physics* 153.1 (1999), pp. 101–138 (cit. on p. 28).

[BK17] Edward Bormashenko et al. “Rotating and rolling rigid bodies and the “hairy ball” theorem”. en. In: *American Journal of Physics* 85.6 (June 2017), pp. 447–453. ISSN: 0002-9505, 1943-2909. DOI: `10.1119/1.4979343`. URL: `https://pubs.aip.org/ajp/article/85/6/447/1057102/Rotating-and-rolling-rigid-bodies-and-the-hairy` (visited on 09/09/2024) (cit. on p. 11).

[BCB07] Sébastien Bourguignon et al. “SparSpec: a new method for fitting multiple sinusoids with irregularly sampled data”. In: *Astronomy & Astrophysics* 462.1 (2007). Publisher: EDP Sciences, pp. 379–387 (cit. on p. 65).

[BS02] C. M. Braams et al. *Nuclear fusion: half a century of magnetic confinement fusion research*. eng. OCLC: 1107880260. Place of publication not identified: CRC Press, 2002. ISBN: 978-0-367-80151-9 (cit. on p. 3).

[Bre01] Leo Breiman. “Random Forests”. en. In: *Machine Learning* 45.1 (Oct. 2001), pp. 5–32. ISSN: 1573-0565. DOI: `10.1023/A:1010933404324`. URL: `https://doi.org/10.1023/A:1010933404324` (visited on 09/23/2024) (cit. on p. 38).

[Bre+17] Leo Breiman et al. *Classification and Regression Trees*. New York: Chapman and Hall/CRC, Oct. 2017. ISBN: 978-1-315-13947-0. DOI: `10.1201/9781315139470` (cit. on p. 38).

[BK22] Steven L Brunton et al. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022 (cit. on pp. 37, 68).

[BPK16] Steven L. Brunton et al. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. en. In: *Proceedings of the National Academy of Sciences* 113.15 (Apr. 2016), pp. 3932–3937. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.1517384113`. URL: `https://pnas.org/doi/full/10.1073/pnas.1517384113` (visited on 03/06/2024) (cit. on p. 36).

[BB13] Wilhelm Burger et al. *Principles of Digital Image Processing: Advanced Methods*. 1st ed. 2013. Undergraduate Topics in Computer Science. London: Springer London : Imprint: Springer, 2013. ISBN: 978-1-84882-919-0. DOI: `10.1007/978-1-84882-919-0` (cit. on p. 145).

[BA87] Peter J. Burt et al. “The Laplacian Pyramid as a Compact Image Code”. en. In: *Readings in Computer Vision*. Elsevier, 1987, pp. 671–679. ISBN: 978-0-08-051581-6. DOI: `10.1016/B978-0-08-051581-6.50065-9`. URL: `https://linkinghub.elsevier.com/retrieve/pii/B9780080515816500659` (visited on 09/23/2024) (cit. on p. 53).

[Bus+21] A. Bustos et al. “Automatic identification of MHD modes in magnetic fluctuation spectrograms using deep learning techniques”. en. In: *Plasma Physics and Controlled Fusion* 63.9 (July 2021). Publisher: IOP Publishing, p. 095001. ISSN: 0741-3335. DOI: `10.1088/1361-6587/ac08f7`. URL: `https://dx.doi.org/10.1088/1361-6587/ac08f7` (visited on 02/20/2024) (cit. on pp. xiii, 65, 168, 170, 171, 178, 182).

[CMS13] Ricardo J. G. B. Campello et al. “Density-Based Clustering Based on Hierarchical Density Estimates”. en. In: *Advances in Knowledge Discovery and Data Mining*. Ed. by Jian Pei et al. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2013, pp. 160–172. ISBN: 978-3-642-37456-2. DOI: `10.1007/978-3-642-37456-2_14` (cit. on p. 38).

[Cam+15] Ricardo J. G. B. Campello et al. “Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection”. In: *ACM Transactions on Knowledge Discovery from Data* 10.1 (July 2015), 5:1–5:51. ISSN: 1556-4681. DOI: `10.1145/2733381`. URL: `https://doi.org/10.1145/2733381` (visited on 02/26/2024) (cit. on pp. ix, 88).

[Can+06] Emmanuel Candès et al. “Fast Discrete Curvelet Transforms”. en. In: *Multiscale Modeling & Simulation* 5.3 (Jan. 2006), pp. 861–899. ISSN: 1540-3459, 1540-3467. DOI: `10.1137/05064182X`. URL: `http://epubs.siam.org/doi/10.1137/05064182X` (visited on 02/28/2024) (cit. on pp. xi, 127, 136).

[CD05] Emmanuel J. Candès et al. “Continuous curvelet transform”. en. In: *Applied and Computational Harmonic Analysis* 19.2 (Sept. 2005), pp. 198–222. ISSN: 10635203. DOI: `10.1016/j.acha.2005.02.004`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1063520305000205` (visited on 09/23/2024) (cit. on p. 56).

[Can86] John Canny. “A Computational Approach to Edge Detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (Nov. 1986), pp. 679–698. ISSN: 0162-8828. DOI: `10.1109/TPAMI.1986.4767851`. URL: `https://ieeexplore.ieee.org/document/4767851` (visited on 03/12/2024) (cit. on pp. 60, 127).

[Cap+21] Á. Cappa et al. “Stability analysis of TJ-II stellarator NBI driven Alfvén eigenmodes in ECRH and ECCD experiments”. In: *Nuclear Fusion* 61.6 (June 2021), p. 066019. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/abf74b`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/abf74b` (visited on 03/04/2024) (cit. on pp. 78, 94, 121).

[Che11] Francis F. Chen. *An indispensable truth: how fusion power can save the planet*. OCLC: ocn668191396. New York: Springer, 2011. ISBN: 978-1-4419-7819-6 (cit. on p. 3).

[Che18] Francis F. Chen. *Introduction to plasma physics and controlled fusion*. eng. Third edition, corrected publication. Cham: Springer, 2018. ISBN: 978-3-319-22308-7 (cit. on pp. 8, 9).

[CDS01] Scott Shaobing Chen et al. "Atomic Decomposition by Basis Pursuit". en. In: *SIAM Review* 43.1 (Jan. 2001), pp. 129–159. ISSN: 0036-1445, 1095-7200. DOI: `10.1137/S003614450037906X`. URL: `http://epubs.siam.org/doi/10.1137/S003614450037906X` (visited on 02/20/2024) (cit. on p. 81).

[CCC85] C.Z Cheng et al. "High-n ideal and resistive shear Alfvén waves in tokamaks". en. In: *Annals of Physics* 161.1 (Apr. 1985), pp. 21–47. ISSN: 00034916. DOI: `10.1016/0003-4916(85)90335-5`. URL: `https://linkinghub.elsevier.com/retrieve/pii/0003491685903355` (visited on 09/12/2024) (cit. on p. 26).

[24a] *CIEMAT's FusionWiki*. 2024. URL: `https://wiki.fusion.ciemat.es/wiki/Main_Page` (visited on 09/10/2024) (cit. on p. 32).

[Col23] John Loughran Colvin. "Automated tracing of divergent ridges in tokamak magnetic spectra". en. In: *Master thesis* (2023) (cit. on p. 66).

[CH67] T. Cover et al. "Nearest neighbor pattern classification". In: *IEEE Transactions on Information Theory* 13.1 (Jan. 1967). Conference Name: IEEE Transactions on Information Theory, pp. 21–27. ISSN: 1557-9654. DOI: `10.1109/TIT.1967.1053964`. URL: `https://ieeexplore.ieee.org/document/1053964` (visited on 09/23/2024) (cit. on p. 38).

[CR08] Nello Cristianini et al. "Support Vector Machines". en. In: *Encyclopedia of Algorithms*. Ed. by Ming-Yang Kao. Boston, MA: Springer US, 2008, pp. 928–932. ISBN: 978-0-387-30162-4. DOI: `10.1007/978-0-387-30162-4_415`. URL: `https://doi.org/10.1007/978-0-387-30162-4_415` (visited on 09/23/2024) (cit. on p. 38).

[Dha12] W. D. D'haeseleer. *Flux coordinates and magnetic field structure: a guide to a fundamental tool of plasma theory*. eng. OCLC: 1340935531. Berlin: Springer-Verlag, 2012. ISBN: 978-3-642-75597-2 (cit. on p. 14).

[Dau88] Ingrid Daubechies. "Orthonormal bases of compactly supported wavelets". en. In: *Communications on Pure and Applied Mathematics* 41.7 (Oct. 1988), pp. 909–996. ISSN: 0010-3640, 1097-0312. DOI: `10.1002/cpa.3160410705`. URL: `https://onlinelibrary.wiley.com/doi/10.1002/cpa.3160410705` (visited on 09/23/2024) (cit. on p. 53).

[Day+95] P. Dayan et al. "The Helmholtz machine". eng. In: *Neural Computation* 7.5 (Sept. 1995), pp. 889–904. ISSN: 0899-7667. DOI: `10.1162/neco.1995.7.5.889` (cit. on p. 36).

[DMT16] Winnok H. De Vos et al., eds. *Focus on Bio-Image Informatics*. en. Vol. 219. Advances in Anatomy, Embryology and Cell Biology. Cham: Springer International Publishing, 2016. ISBN: 978-3-319-28547-4 978-3-319-28549-8. DOI: 10.1007/978-3-319-28549-8. URL: http://link.springer.com/10.1007/978-3-319-28549-8 (visited on 08/09/2024) (cit. on pp. 144, 146).

[Deg+22] Jonas Degrave et al. "Magnetic control of tokamak plasmas through deep reinforcement learning". en. In: *Nature* 602.7897 (Feb. 2022), pp. 414–419. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-021-04301-9. URL: https://www.nature.com/articles/s41586-021-04301-9 (visited on 03/06/2024) (cit. on p. 66).

[DDC88] Elizabeth R DeLong et al. "Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach". In: *Biometrics* (1988), pp. 837–845 (cit. on p. 216).

[DV05] M.N. Do et al. "The contourlet transform: an efficient directional multiresolution image representation". In: *IEEE Transactions on Image Processing* 14.12 (Dec. 2005). Conference Name: IEEE Transactions on Image Processing, pp. 2091–2106. ISSN: 1941-0042. DOI: 10.1109/TIP.2005.859376. URL: https://ieeexplore.ieee.org/document/1532309 (visited on 02/28/2024) (cit. on p. 127).

[DJ+94] David L Donoho et al. "Ideal denoising in an orthonormal basis chosen from a library of bases". In: *Comptes rendus de l'Académie des sciences. Série I, Mathématique* 319.12 (1994), pp. 1317–1322 (cit. on p. 58).

[Dos+21] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV]. URL: https://arxiv.org/abs/2010.11929 (cit. on p. 169).

[DH72] Richard O. Duda et al. "Use of the Hough transformation to detect lines and curves in pictures". en. In: *Communications of the ACM* 15.1 (Jan. 1972), pp. 11–15. ISSN: 0001-0782, 1557-7317. DOI: 10.1145/361237.361242. URL: https://dl.acm.org/doi/10.1145/361237.361242 (visited on 03/12/2024) (cit. on pp. x, 127).

[Edd20] A. S. Eddington. "The Internal Constitution of the Stars". en. In: *Nature* 106.2653 (Sept. 1920), pp. 14–20. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/106014a0. URL: https://www.nature.com/articles/106014a0 (visited on 09/03/2024) (cit. on p. 3).

[Ela10] Michael Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. en. New York, NY: Springer New York, 2010. ISBN: 978-1-4419-7010-7 978-1-4419-7011-4. DOI: 10.1007/978-1-4419-7011-4. URL: https://link.springer.com/10.1007/978-1-4419-7011-4 (visited on 02/20/2024) (cit. on pp. 51, 80).

[Eli+21] L. G. Eliseev et al. “Experimental observation of the geodesic acoustic frequency limit for the NBI-driven Alfvén eigenmodes in TJ-II”. en. In: *Physics of Plasmas* 28.7 (July 2021), p. 072510. ISSN: 1070-664X, 1089-7674. DOI: 10.1063/5.0049225. URL: https://pubs.aip.org/pop/article/28/7/072510/594585/Experimental-observation-of-the-geodesic-acoustic (visited on 03/04/2024) (cit. on pp. 78, 94).

[Est+96] Martin Ester et al. “A density-based algorithm for discovering clusters in large spatial databases with noise”. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD’96. Portland, Oregon: AAAI Press, Aug. 1996, pp. 226–231. (Visited on 02/27/2024) (cit. on p. 88).

[FRS01] Karl Pearson F.R.S. “LIII. On lines and planes of closest fit to systems of points in space”. EN. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* (Nov. 1901). Publisher: Taylor & Francis Group. DOI: 10.1080/14786440109462720. URL: https://www.tandfonline.com/doi/abs/10.1080/14786440109462720 (visited on 09/23/2024) (cit. on p. 38).

[fac24] facebookresearch. *Llama 3.1*. en. 2024. URL: https://www.llama.com/ (visited on 09/23/2024) (cit. on p. 47).

[24b] *facebookresearch/detectron2*. original-date: 2019-09-05T21:30:20Z. Sept. 2024. URL: https://github.com/facebookresearch/detectron2 (visited on 09/16/2024) (cit. on p. 168).

[24c] *facebookresearch/maskrcnn-benchmark*. original-date: 2018-10-24T17:34:50Z. Sept. 2024. URL: https://github.com/facebookresearch/maskrcnn-benchmark (visited on 09/16/2024) (cit. on p. 169).

[FG24] Mahmoud Fakhry et al. “Elastic net regularization and gabor dictionary for classification of heart sound signals using deep learning”. en. In: *Engineering Applications of Artificial Intelligence* 127 (Jan. 2024), p. 107406. ISSN: 09521976. DOI: 10.1016/j.engappai.2023.107406. URL: https://linkinghub.elsevier.com/retrieve/pii/S0952197623015907 (visited on 02/26/2024) (cit. on pp. 86, 90).

[Fas+02] A. Fasoli et al. “MHD spectroscopy”. en. In: *Plasma Physics and Controlled Fusion* 44.12B (Nov. 2002), B159. ISSN: 0741-3335. DOI: 10.1088/0741-3335/44/12B/312. URL: https://dx.doi.org/10.1088/0741-3335/44/12B/312 (visited on 09/23/2024) (cit. on p. 28).

[FMR22] Diogo R Ferreira et al. “Explainable deep learning for the analysis of MHD spectrograms in nuclear fusion”. en. In: *Machine Learning: Science and Technology* 3.1 (Mar. 2022), p. 015015. ISSN: 2632-2153. DOI: 10.1088/2632-2153/ac44aa. URL: https://iopscience.iop.org/article/10.1088/2632-2153/ac44aa (visited on 12/27/2024) (cit. on p. 66).

[FG96] Christophe Fiorio et al. “Two linear time Union-Find strategies for image processing”. en. In: *Theoretical Computer Science* 154.2 (Feb. 1996), pp. 165–181. ISSN: 03043975. DOI: `10.1016/0304-3975(94)00262-2`. URL: `https://linkinghub.elsevier.com/retrieve/pii/0304397594002622` (visited on 04/11/2024) (cit. on pp. xii, 153).

[Fit+22] M. Fitzgerald et al. “Toroidal Alfvén eigenmode stability in JET internal transport barrier afterglow experiments”. en. In: *Nuclear Fusion* 62.10 (Oct. 2022), p. 106001. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ac84ee`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ac84ee` (visited on 03/05/2024) (cit. on p. 130).

[Fra+98] Alejandro F. Frangi et al. “Multiscale vessel enhancement filtering”. In: *Medical Image Computing and Computer-Assisted Intervention — MICCAI’98*. Ed. by William M. Wells et al. Vol. 1496. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 130–137. ISBN: 978-3-540-65136-9 978-3-540-49563-5. DOI: `10.1007/BFb0056195`. URL: `http://link.springer.com/10.1007/BFb0056195` (visited on 02/29/2024) (cit. on pp. xii, 146, 147).

[Fre+16] S. J. Freethy et al. “Measurement of turbulent electron temperature fluctuations on the ASDEX Upgrade tokamak using correlated electron cyclotron emission”. en. In: *Review of Scientific Instruments* 87.11 (Nov. 2016), 11E102. ISSN: 0034-6748, 1089-7623. DOI: `10.1063/1.4958908`. URL: `https://pubs.aip.org/rsi/article/87/11/11E102/357932/Measurement-of-turbulent-electron-temperature` (visited on 04/10/2024) (cit. on p. 125).

[Fre14] Jeffrey P. Freidberg. *Ideal MHD*. Cambridge: Cambridge University Press, 2014. ISBN: 978-1-107-00625-6 (cit. on pp. 14, 15, 18, 23).

[GMK99] C. Galamhos et al. “Progressive probabilistic Hough transform for line detection”. In: *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*. Fort Collins, CO, USA: IEEE Comput. Soc, 1999, pp. 554–560. ISBN: 978-0-7695-0149-9. DOI: `10.1109/CVPR.1999.786993`. URL: `http://ieeexplore.ieee.org/document/786993/` (visited on 03/12/2024) (cit. on pp. x, xi, 127, 131).

[Gar+23] Alvin V. Garcia et al. “Alfvén eigenmode detection using Long-Short Term Memory Networks and CO $_2$ Interferometer data on the DIII-D National Fusion Facility”. In: *2023 International Joint Conference on Neural Networks (IJCNN)*. Gold Coast, Australia: IEEE, June 2023, pp. 1–8. ISBN: 978-1-66548-867-9. DOI: `10.1109/IJCNN54540.2023.10191760`. URL: `https://ieeexplore.ieee.org/document/10191760/` (visited on 03/05/2024) (cit. on p. 65).

[Gar90] H.J. Gardner. “Modelling the behaviour of the magnetic field diagnostic coils on the W VII-AS stellarator using a three-dimensional equilibrium code”. In: *Nuclear Fusion* 30.8 (Aug. 1990), pp. 1417–1424. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/30/8/002`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/30/8/002` (visited on 09/13/2024) (cit. on p. 18).

[Ghi+20] Golnaz Ghiasi et al. *Simple Copy-Paste is a Strong Data Augmentation Method for Instance Segmentation*. arXiv:2012.07177 [cs] version: 1. Dec. 2020. DOI: `10.48550/arXiv.2012.07177`. URL: `http://arxiv.org/abs/2012.07177` (visited on 09/19/2024) (cit. on p. 169).

[Ghi+24] A.G. Ghiozzi et al. “Modeling of frequency-sweeping Alfvén modes in the TJ-II stellarator”. In: *Nuclear Fusion* 64.3 (Mar. 2024), p. 036005. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ad1c93`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ad1c93` (visited on 03/04/2024) (cit. on pp. 78, 94).

[Gir+14] Ross Girshick et al. *Rich feature hierarchies for accurate object detection and semantic segmentation*. arXiv:1311.2524 [cs]. Oct. 2014. DOI: `10.48550/arXiv.1311.2524`. URL: `http://arxiv.org/abs/1311.2524` (visited on 09/23/2024) (cit. on p. 169).

[Gla93] C.A. Glasbey. “An Analysis of Histogram-Based Thresholding Algorithms”. en. In: *CVGIP: Graphical Models and Image Processing* 55.6 (Nov. 1993), pp. 532–537. ISSN: 10499652. DOI: `10.1006/cgip.1993.1040`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1049965283710400` (visited on 08/10/2024) (cit. on pp. xii, 147).

[GP04] J. P. Goedbloed et al. *Principles of magnetohydrodynamics: with applications to laboratory and astrophysical plasmas*. Cambridge, UK ; New York: Cambridge University Press, 2004. ISBN: 978-0-521-62347-6 978-0-521-62607-1 (cit. on pp. 14, 15).

[GKP10] Johan P. Goedbloed et al. *Advanced magnetohydrodynamics: with applications to laboratory and astrophysical plasmas*. eng. 1. publ. Cambridge: Cambridge Univ. Press, 2010. ISBN: 978-0-521-87957-6 978-0-521-70524-0 (cit. on p. 15).

[Goe81] JP Goedbloed. “Conformal mapping methods in two-dimensional magnetohydrodynamics”. In: *Computer Physics Communications* 24.3-4 (1981), pp. 311–321 (cit. on p. 18).

[GW18] Rafael C. Gonzalez et al. *Digital image processing*. eng. Fourth edition, global edition. New York, NY: Pearson, 2018. ISBN: 978-1-292-22304-9 (cit. on p. 164).

[GBC16] Ian Goodfellow et al. *Deep learning*. eng. OCLC: 1183962587. Cambridge, Massachusetts: The MIT Press, 2016. ISBN: 978-0-262-33737-3 (cit. on pp. 37, 50).

[GPT14] N.N. Gorelenkov et al. “Energetic particle physics in fusion research in preparation for burning plasma experiments”. en. In: *Nuclear Fusion* 54.12 (Dec. 2014), p. 125001. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/54/12/125001`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/54/12/125001` (visited on 04/10/2024) (cit. on pp. 28, 125).

[GKF02] A Gyaourova et al. *Undecimated Wavelet Transforms for Image De-noising*. en. Tech. rep. UCRL-ID-150931, 15002085. Nov. 2002, UCRL–ID–150931, 15002085. DOI: `10.2172/15002085`. URL: `http://www.osti.gov/servlets/purl/15002085-KoMQnk/native/` (visited on 02/28/2024) (cit. on p. 127).

[Haa10] Alfred Haar. “Zur Theorie der orthogonalen Funktionensysteme: Erste Mitteilung”. de. In: *Mathematische Annalen* 69.3 (Sept. 1910), pp. 331–371. ISSN: 0025-5831, 1432-1807. DOI: `10.1007/BF01456326`. URL: `http://link.springer.com/10.1007/BF01456326` (visited on 09/23/2024) (cit. on p. 52).

[24d] *Hairy ball theorem*. en. Page Version ID: 1219128203. Apr. 2024. URL: `https://en.wikipedia.org/w/index.php?title=Hairy_ball_theorem&oldid=1219128203` (visited on 04/19/2024) (cit. on p. 11).

[HBP14] S.R. Haskey et al. “Clustering of periodic multichannel timeseries data with application to plasma fluctuations”. en. In: *Computer Physics Communications* 185.6 (June 2014), pp. 1669–1680. ISSN: 00104655. DOI: `10.1016/j.cpc.2014.03.008`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0010465514000885` (visited on 03/05/2024) (cit. on p. 66).

[HTF09] Trevor Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009 (cit. on p. 37).

[HTW20] Trevor Hastie et al. *Statistical learning with sparsity: the lasso and generalizations*. eng. First issued in paperback. Monographs on statistics and applied probability 143. Boca Raton London New York: CRC Press, 2020. ISBN: 978-0-367-73833-4 978-1-4987-1216-3 (cit. on p. 51).

[HS] Seyede Mahya Hazavei et al. “A New Method for Removing the Moire’ Pattern from Images”. en. In: () (cit. on p. 126).

[HM03] R. D. Hazeltine et al. *Plasma confinement*. Mineola, N.Y: Dover Publications, 2003. ISBN: 978-0-486-43242-7 (cit. on pp. 15, 17, 18).

[Hei08] W. W. Heidbrink. “Basic physics of Alfvén instabilities driven by energetic particles in toroidally confined plasmas”. en. In: *Physics of Plasmas* 15.5 (May 2008), p. 055501. ISSN: 1070-664X, 1089-7674. DOI: `10.1063/1.2838239`. URL: `https://pubs.aip.org/pop/article/15/5/055501/`

1015770/Basic-physics-of-Alfven-instabilities-driven-by (visited on 04/10/2024) (cit. on pp. vi, 25, 27–29, 125).

[Hel+12] P Helander et al. "Stellarator and tokamak plasmas: a comparison". en. In: *Plasma Physics and Controlled Fusion* 54.12 (Dec. 2012), p. 124009. ISSN: 0741-3335, 1361-6587. DOI: 10.1088/0741-3335/54/12/124009. URL: https://iopscience.iop.org/article/10.1088/0741-3335/54/12/124009 (visited on 09/02/2024) (cit. on p. 22).

[Hel14] Per Helander. "Theory of plasma confinement in non-axisymmetric magnetic fields". en. In: *Reports on Progress in Physics* 77.8 (Aug. 2014), p. 087001. ISSN: 0034-4885, 1361-6633. DOI: 10.1088/0034-4885/77/8/087001. URL: https://iopscience.iop.org/article/10.1088/0034-4885/77/8/087001 (visited on 09/02/2024) (cit. on pp. 18, 22).

[Hen+20] Tom Henighan et al. *Scaling Laws for Autoregressive Generative Modeling*. arXiv:2010.14701 [cs]. Nov. 2020. DOI: 10.48550/arXiv.2010.14701. URL: http://arxiv.org/abs/2010.14701 (visited on 09/23/2024) (cit. on p. 44).

[Her+96] J.V Hernandez et al. "Neural network prediction of some classes of tokamak disruptions". en. In: *Nuclear Fusion* 36.8 (Aug. 1996), pp. 1009–1017. ISSN: 0029-5515. DOI: 10.1088/0029-5515/36/8/I05. URL: https://iopscience.iop.org/article/10.1088/0029-5515/36/8/I05 (visited on 12/27/2024) (cit. on p. 66).

[Hes+17] Joel Hestness et al. "Deep Learning Scaling is Predictable, Empirically". In: (2017). Publisher: [object Object] Version Number: 1. DOI: 10.48550/ARXIV.1712.00409. URL: https://arxiv.org/abs/1712.00409 (visited on 03/05/2024) (cit. on p. 65).

[HW83] S. P. Hirshman et al. "Steepest-descent moment method for three-dimensional magnetohydrodynamic equilibria". en. In: *The Physics of Fluids* 26.12 (Dec. 1983), pp. 3553–3568. ISSN: 0031-9171. DOI: 10.1063/1.864116. URL: https://pubs.aip.org/pfl/article/26/12/3553/809269/Steepest-descent-moment-method-for-three (visited on 09/13/2024) (cit. on p. 18).

[HVM86] S.P. Hirshman et al. "Three-dimensional free boundary calculations using a spectral Green's function method". en. In: *Computer Physics Communications* 43.1 (Dec. 1986), pp. 143–155. ISSN: 00104655. DOI: 10.1016/0010-4655(86)90058-5. URL: https://linkinghub.elsevier.com/retrieve/pii/0010465586900585 (visited on 09/13/2024) (cit. on p. 18).

[HS97] Sepp Hochreiter et al. "Long short-term memory". In: *Neural computation* 9.8 (1997). Publisher: MIT Press, pp. 1735–1780 (cit. on p. 47).

[HK00] Arthur E. Hoerl et al. “Ridge Regression: Biased Estimation for Nonorthogonal Problems”. In: *Technometrics* 42.1 (2000). Publisher: [Taylor & Francis, Ltd., American Statistical Association, American Society for Quality], pp. 80–86. ISSN: 0040-1706. DOI: 10.2307/1271436. URL: https://www.jstor.org/stable/1271436 (visited on 09/23/2024) (cit. on p. 48).

[HA07] M J Hole et al. “Fourier decomposition of magnetic perturbations in toroidal plasmas using singular value decomposition”. en. In: *Plasma Physics and Controlled Fusion* 49.12 (Dec. 2007), pp. 1971–1988. ISSN: 0741-3335, 1361-6587. DOI: 10.1088/0741-3335/49/12/002. URL: https://iopscience.iop.org/article/10.1088/0741-3335/49/12/002 (visited on 03/06/2024) (cit. on p. 66).

[Hot33] H. Hotelling. “Analysis of a complex of statistical variables into principal components.” en. In: *Journal of Educational Psychology* 24.6 (Sept. 1933), pp. 417–441. ISSN: 1939-2176, 0022-0663. DOI: 10.1037/h0071325. URL: https://doi.apa.org/doi/10.1037/h0071325 (visited on 09/23/2024) (cit. on p. 38).

[HGK+91] GTA Huysmans et al. “Isoparametric bicubic Hermite elements for solution of the Grad-Shafranov equation”. In: *International Journal of Modern Physics C* 2.01 (1991), pp. 371–376 (cit. on p. 18).

[Igo14] V Igochine. *Active control of magneto-hydrodynamic instabilities in hot plasmas.* New York: Springer, 2014. ISBN: 978-3-662-44221-0 (cit. on pp. 13, 25, 28–30).

[Jal+21] Azarakhsh Jalalvand et al. “Alfvén eigenmode classification based on ECE diagnostics at DIII-D using deep recurrent neural networks”. en. In: *Nuclear Fusion* 62.2 (Dec. 2021). Publisher: IOP Publishing, p. 026007. ISSN: 0029-5515. DOI: 10.1088/1741-4326/ac3be7. URL: https://dx.doi.org/10.1088/1741-4326/ac3be7 (visited on 02/20/2024) (cit. on p. 65).

[Jal+22] Azarakhsh Jalalvand et al. “Alfvén eigenmode classification based on ECE diagnostics at DIII-D using deep recurrent neural networks”. In: *Nuclear Fusion* 62.2 (Feb. 2022), p. 026007. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/1741-4326/ac3be7. URL: https://iopscience.iop.org/article/10.1088/1741-4326/ac3be7 (visited on 03/05/2024) (cit. on p. 65).

[Jam+14] G. James et al. *An Introduction to Statistical Learning: with Applications in R.* Springer Texts in Statistics. Springer New York, 2014. ISBN: 9781461471370. URL: https://books.google.es/books?id=at1bmAEACAAJ (cit. on pp. vi, 37, 38, 43, 44).

[Jar10] Stephen Jardin. *Computational methods in plasma physics*. eng. OCLC: 680039941. Boca Raton, FL: CRC Press/Taylor & Francis, 2010. ISBN: 978-1-4398-1095-8 (cit. on p. 15).

[Jim+11] R. Jiménez-Gómez et al. "Alfvén eigenmodes measured in the TJ-II stellarator". en. In: *Nuclear Fusion* 51.3 (Feb. 2011), p. 033001. ISSN: 0029-5515. DOI: `10.1088/0029-5515/51/3/033001`. URL: `https://dx.doi.org/10.1088/0029-5515/51/3/033001` (visited on 02/20/2024) (cit. on pp. 78, 94).

[Jon19] Mohsen Joneidi. *Functional Brain Networks Discovery Using Dictionary Learning with Correlated Sparsity*. en. arXiv:1907.03929 [eess, q-bio, stat]. July 2019. URL: `http://arxiv.org/abs/1907.03929` (visited on 03/10/2024) (cit. on p. 85).

[Kap+20] Jared Kaplan et al. *Scaling Laws for Neural Language Models*. en. arXiv:2001.08361 [cs, stat]. Jan. 2020. URL: `http://arxiv.org/abs/2001.08361` (visited on 09/23/2024) (cit. on p. 44).

[Kap+22a] Alan A. Kaptanoglu et al. "Exploring data-driven models for spatiotemporally local classification of Alfvén eigenmodes". In: *Nuclear Fusion* 62.10 (Oct. 2022), p. 106014. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ac8a03`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ac8a03` (visited on 03/05/2024) (cit. on pp. 65, 66).

[Kap+22b] Alan A. Kaptanoglu et al. "Permanent-Magnet Optimization for Stellarators as Sparse Regression". en. In: *Physical Review Applied* 18.4 (Oct. 2022), p. 044006. ISSN: 2331-7019. DOI: `10.1103/PhysRevApplied.18.044006`. URL: `https://link.aps.org/doi/10.1103/PhysRevApplied.18.044006` (visited on 03/05/2024) (cit. on p. 75).

[Kap+23] Alan A. Kaptanoglu et al. "Sparse regression for plasma physics". en. In: *Physics of Plasmas* 30.3 (Mar. 2023), p. 033906. ISSN: 1070-664X, 1089-7674. DOI: `10.1063/5.0139039`. URL: `https://pubs.aip.org/pop/article/30/3/033906/2881938/Sparse-regression-for-plasma-physics` (visited on 03/05/2024) (cit. on p. 75).

[Kas80] G. V. Kass. "An Exploratory Technique for Investigating Large Quantities of Categorical Data". In: *Applied Statistics* 29.2 (1980), p. 119. ISSN: 00359254. DOI: `10.2307/2986296`. URL: `https://www.jstor.org/stable/10.2307/2986296?origin=crossref` (visited on 09/23/2024) (cit. on p. 38).

[KST19] Julian Kates-Harbeck et al. "Predicting disruptive instabilities in controlled fusion plasmas through deep learning". en. In: *Nature* 568.7753 (Apr. 2019), pp. 526–531. ISSN: 0028-0836, 1476-4687. DOI: `10.1038/s41586-019-1116-4`. URL: `https://www.nature.com/articles/s41586-019-1116-4` (visited on 03/06/2024) (cit. on p. 66).

[Kel+12] J. W. Kelly et al. “Sparse linear regression with elastic net regularization for brain-computer interfaces”. en. In: *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. San Diego, CA: IEEE, Aug. 2012, pp. 4275–4278. ISBN: 978-1-4577-1787-1 978-1-4244-4119-8. DOI: `10.1109/EMBC.2012.6346911`. URL: `http://ieeexplore.ieee.org/document/6346911/` (visited on 02/26/2024) (cit. on p. 86).

[Ker+98] W Kerner et al. “CASTOR: Normal-mode analysis of resistive MHD plasmas”. In: *Journal of computational physics* 142.2 (1998), pp. 271–303 (cit. on p. 28).

[KAC22] Nabeel Ali Khan et al. “An Efficient and Accurate Multi-Sensor IF Estimator Based on DOA Information and Order of Fractional Fourier Transform”. en. In: *Entropy* 24.4 (Mar. 2022), p. 452. ISSN: 1099-4300. DOI: `10.3390/e24040452`. URL: `https://www.mdpi.com/1099-4300/24/4/452` (visited on 03/06/2024) (cit. on p. 168).

[KU18] Maxim Kimlyk et al. “Image denoising using discrete wavelet transform and edge information”. In: *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. Moscow: IEEE, Jan. 2018, pp. 1823–1825. ISBN: 978-1-5386-4339-6 978-1-5386-4340-2. DOI: `10.1109/EIConRus.2018.8317461`. URL: `http://ieeexplore.ieee.org/document/8317461/` (visited on 02/28/2024) (cit. on p. 127).

[KB17] Diederik P. Kingma et al. *Adam: A Method for Stochastic Optimization*. arXiv:1412.6980 [cs]. Jan. 2017. DOI: `10.48550/arXiv.1412.6980`. URL: `http://arxiv.org/abs/1412.6980` (visited on 09/23/2024) (cit. on p. 46).

[KW19] Diederik P. Kingma et al. “An Introduction to Variational Autoencoders”. In: *Foundations and Trends® in Machine Learning* 12.4 (2019). arXiv:1906.02691 [cs, stat], pp. 307–392. ISSN: 1935-8237, 1935-8245. DOI: `10.1561/2200000056`. URL: `http://arxiv.org/abs/1906.02691` (visited on 09/23/2024) (cit. on p. 36).

[Kir+23] Alexander Kirillov et al. *Segment Anything*. arXiv:2304.02643 [cs]. Apr. 2023. DOI: `10.48550/arXiv.2304.02643`. URL: `http://arxiv.org/abs/2304.02643` (visited on 09/16/2024) (cit. on pp. xiii, 168).

[Kle+08] A Klein et al. “A sparsity-based method for the analysis of magnetic fluctuations in unevenly-spaced Mirnov coils”. en. In: *Plasma Physics and Controlled Fusion* 50.12 (Dec. 2008), p. 125005. ISSN: 0741-3335, 1361-6587. DOI: `10.1088/0741-3335/50/12/125005`. URL: `https://iopscience.iop.org/article/10.1088/0741-3335/50/12/125005` (visited on 03/05/2024) (cit. on pp. 65, 75).

[Kle14] Reinhard Klette. *Concise Computer Vision: An Introduction into Theory and Algorithms*. en. Undergraduate Topics in Computer Science. London: Springer London, 2014. ISBN: 978-1-4471-6319-0 978-1-4471-6320-6. DOI: `10.1007/978-1-4471-6320-6`. URL: `https://link.springer.com/10.1007/978-1-4471-6320-6` (visited on 08/08/2024) (cit. on p. 145).

[Koh82] Teuvo Kohonen. "Self-organized formation of topologically correct feature maps". en. In: *Biological Cybernetics* 43.1 (Jan. 1982), pp. 59–69. ISSN: 1432-0770. DOI: `10.1007/BF00337288`. URL: `https://doi.org/10.1007/BF00337288` (visited on 12/27/2024) (cit. on p. 43).

[KSG04] Alexander Kraskov et al. "Estimating mutual information". In: *Physical Review E* 69.6 (June 2004). Publisher: American Physical Society, p. 066138. DOI: `10.1103/PhysRevE.69.066138`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.69.066138` (visited on 02/27/2024) (cit. on p. 103).

[Kra16] Bartosz Krawczyk. "Learning from imbalanced data: open challenges and future directions". In: *Progress in Artificial Intelligence* 5.4 (2016), pp. 221–232 (cit. on p. 39).

[Kri16] Scott Krig. *Computer Vision Metrics*. en. Cham: Springer International Publishing, 2016. ISBN: 978-3-319-33761-6 978-3-319-33762-3. DOI: `10.1007/978-3-319-33762-3`. URL: `http://link.springer.com/10.1007/978-3-319-33762-3` (visited on 08/08/2024) (cit. on p. 64).

[KSH17] Alex Krizhevsky et al. "ImageNet classification with deep convolutional neural networks". en. In: *Communications of the ACM* 60.6 (May 2017), pp. 84–90. ISSN: 0001-0782, 1557-7317. DOI: `10.1145/3065386`. URL: `https://dl.acm.org/doi/10.1145/3065386` (visited on 07/25/2024) (cit. on pp. 36, 47).

[Lan61] R. Landauer. "Irreversibility and Heat Generation in the Computing Process". In: *IBM Journal of Research and Development* 5.3 (July 1961), pp. 183–191. ISSN: 0018-8646, 0018-8646. DOI: `10.1147/rd.53.0183`. URL: `http://ieeexplore.ieee.org/document/5392446/` (visited on 06/03/2024) (cit. on p. 101).

[Lao+22] L. L. Lao et al. "Application of machine learning and artificial intelligence to extend EFIT equilibrium reconstruction". en. In: *Plasma Physics and Controlled Fusion* 64.7 (June 2022). Publisher: IOP Publishing, p. 074001. ISSN: 0741-3335. DOI: `10.1088/1361-6587/ac6fff`. URL: `https://dx.doi.org/10.1088/1361-6587/ac6fff` (visited on 09/23/2024) (cit. on p. 18).

[Lau+07] Ph Lauber et al. "LIGKA: A linear gyrokinetic code for the description of background kinetic and fast particle effects on the MHD stability in tokamaks". In: *Journal of Computational Physics* 226.1 (2007), pp. 447–465 (cit. on p. 28).

[Law57] J D Lawson. “Some Criteria for a Power Producing Thermonuclear Reactor”. In: *Proceedings of the Physical Society. Section B* 70.1 (Jan. 1957), pp. 6–10. ISSN: 0370-1301. DOI: `10.1088/0370-1301/70/1/303`. URL: `https://iopscience.iop.org/article/10.1088/0370-1301/70/1/303` (visited on 09/03/2024) (cit. on p. 4).

[Lee+19] Gregory Lee et al. “PyWavelets: A Python package for wavelet analysis”. In: *Journal of Open Source Software* 4.36 (Apr. 2019), p. 1237. ISSN: 2475-9066. DOI: `10.21105/joss.01237`. URL: `http://joss.theoj.org/papers/10.21105/joss.01237` (visited on 02/28/2024) (cit. on p. 125).

[LN23] Tao Lei et al. *Image segmentation: principles, techniques, and applications.* eng. Hoboken, NJ: Wiley, 2023. ISBN: 978-1-119-85900-0 (cit. on p. 142).

[LL93] C.H. Li et al. “Minimum cross entropy thresholding”. en. In: *Pattern Recognition* 26.4 (Apr. 1993), pp. 617–625. ISSN: 00313203. DOI: `10.1016/0031-3203(93)90115-D`. URL: `https://linkinghub.elsevier.com/retrieve/pii/003132039390115D` (visited on 08/10/2024) (cit. on pp. xii, 147).

[LT98] C.H. Li et al. “An iterative algorithm for minimum cross entropy thresholding”. en. In: *Pattern Recognition Letters* 19.8 (June 1998), pp. 771–776. ISSN: 01678655. DOI: `10.1016/S0167-8655(98)00057-9`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0167865598000579` (visited on 08/10/2024) (cit. on pp. xii, 147).

[Lin+18] Tsung-Yi Lin et al. *Focal Loss for Dense Object Detection.* arXiv:1708.02002 [cs] version: 2. Feb. 2018. DOI: `10.48550/arXiv.1708.02002`. URL: `http://arxiv.org/abs/1708.02002` (visited on 09/20/2024) (cit. on pp. 169, 223).

[Lin94] Tony Lindeberg. “Scale-space theory: a basic tool for analyzing structures at different scales”. en. In: *Journal of Applied Statistics* 21.1-2 (Jan. 1994), pp. 225–270. ISSN: 0266-4763, 1360-0532. DOI: `10.1080/757582976`. URL: `https://www.tandfonline.com/doi/full/10.1080/757582976` (visited on 08/10/2024) (cit. on p. 144).

[Liu+23] Yiheng Liu et al. “Summary of ChatGPT-Related research and perspective towards the future of large language models”. In: *Meta-Radiology* 1.2 (Sept. 2023), p. 100017. ISSN: 2950-1628. DOI: `10.1016/j.metrad.2023.100017`. URL: `http://dx.doi.org/10.1016/j.metrad.2023.100017` (cit. on p. 47).

[MP10] Jianwei Ma et al. “The Curvelet Transform”. In: *IEEE Signal Processing Magazine* 27.2 (Mar. 2010), pp. 118–133. ISSN: 1053-5888. DOI: `10.1109/MSP.2009.935453`. URL: `http://ieeexplore.ieee.org/document/5438971/` (visited on 02/28/2024) (cit. on pp. 127, 137).

[MH08] Laurens van der Maaten et al. “Visualizing Data using t-SNE”. In: *Journal of Machine Learning Research* 9.86 (2008), pp. 2579–2605. ISSN: 1533-7928. URL: `http://jmlr.org/papers/v9/vandermaaten08a.html` (visited on 02/26/2024) (cit. on pp. 43, 116).

[Mal09] S Mallat. *A Wavelet Tour of Signal Processing*. en. Elsevier, 2009. ISBN: 978-0-12-374370-1. DOI: `10.1016/B978-0-12-374370-1.X0001-8`. URL: `https://linkinghub.elsevier.com/retrieve/pii/B9780123743701X00018` (visited on 02/20/2024) (cit. on pp. 51–53, 56, 67, 74).

[MZ93] S.G. Mallat et al. “Matching pursuits with time-frequency dictionaries”. In: *IEEE Transactions on Signal Processing* 41.12 (Dec. 1993). Conference Name: IEEE Transactions on Signal Processing, pp. 3397–3415. ISSN: 1941-0476. DOI: `10.1109/78.258082`. URL: `https://ieeexplore.ieee.org/abstract/document/258082` (visited on 02/20/2024) (cit. on pp. 58, 71).

[Mal89] Stephane G. Mallat. “Multiresolution approximations and wavelet orthonormal bases of L2(R)”. en. In: *Transactions of the American Mathematical Society* 315.1 (1989), pp. 69–87. ISSN: 0002-9947, 1088-6850. DOI: `10.1090/S0002-9947-1989-1008470-5`. URL: `https://www.ams.org/tran/1989-315-01/S0002-9947-1989-1008470-5/` (visited on 09/23/2024) (cit. on p. 53).

[MSC21] A. Marinoni et al. “A brief history of negative triangularity tokamak plasmas”. en. In: *Reviews of Modern Plasma Physics* 5.1 (Dec. 2021), p. 6. ISSN: 2367-3192. DOI: `10.1007/s41614-021-00054-0`. URL: `https://link.springer.com/10.1007/s41614-021-00054-0` (visited on 09/02/2024) (cit. on p. 3).

[MH80] D Marr et al. “Theory of edge detection”. en. In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 207.1167 (Feb. 1980), pp. 187–217. ISSN: 0080-4649, 2053-9193. DOI: `10.1098/rspb.1980.0020`. URL: `https://royalsocietypublishing.org/doi/10.1098/rspb.1980.0020` (visited on 08/10/2024) (cit. on p. 144).

[Mar10] David Marr. *Vision: a computational investigation into the human representation and processing of visual information*. OCLC: ocn472791457. Cambridge, Mass: MIT Press, 2010. ISBN: 978-0-262-51462-0 (cit. on p. 144).

[Mei+04] E. Meijering et al. “Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images”. en. In: *Cytometry Part A* 58A.2 (Apr. 2004), pp. 167–176. ISSN: 1552-4922, 1552-4930. DOI: `10.1002/cyto.a.20022`. URL: `https://onlinelibrary.wiley.com/doi/10.1002/cyto.a.20022` (visited on 02/29/2024) (cit. on pp. xii, 146, 147).

[Mel+12] A. V. Melnikov et al. “Alfvén eigenmode properties and dynamics in the TJ-II stellarator”. en. In: *Nuclear Fusion* 52.12 (Nov. 2012). Publisher: IOP Publishing and International Atomic Energy Agency, p. 123004. ISSN: 0029-5515. DOI: 10.1088/0029-5515/52/12/123004. URL: https://dx.doi.org/10.1088/0029-5515/52/12/123004 (visited on 02/20/2024) (cit. on pp. 65, 78, 94).

[Mel+14] A.V. Melnikov et al. “Effect of magnetic configuration on frequency of NBI-driven Alfvén modes in TJ-II”. In: *Nuclear Fusion* 54.12 (Dec. 2014), p. 123002. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/0029-5515/54/12/123002. URL: https://iopscience.iop.org/article/10.1088/0029-5515/54/12/123002 (visited on 03/04/2024) (cit. on pp. 78, 94).

[Mel+18] A.V. Melnikov et al. “Detection and investigation of chirping Alfvén eigenmodes with heavy ion beam probe in the TJ-II stellarator”. In: *Nuclear Fusion* 58.8 (Aug. 2018), p. 082019. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/1741-4326/aabcf8. URL: https://iopscience.iop.org/article/10.1088/1741-4326/aabcf8 (visited on 03/04/2024) (cit. on pp. 78, 94).

[Mey86] Yves Meyer. “Principe d’incertitude, bases hilbertiennes et algebres d’operateurs”. In: *Séminaire Bourbaki* 662 (1986) (cit. on p. 53).

[MM04] Yves Meyer et al. *Wavelets and operators*. eng. Transferred to digital print. Cambridge studies in advanced mathematics 37. Cambridge: Cambridge Univ. Press, 2004. ISBN: 978-0-521-45869-6 978-0-521-42000-6 (cit. on p. 53).

[Mik+97] AB Mikhailovskii et al. “Optimization of computational MHD normal-mode analysis for tokamaks”. In: *Plasma Physics Reports* 23 (1997) (cit. on p. 28).

[MS74] S V Mirnov et al. “THE NATURE OF THE LARGE-SCALE INSTABILITIES IN THE TOKAMAK”. en. In: *Soviet Physics JETP* 33.6 (1974), p. 1134 (cit. on p. 125).

[Mir19] S.V. Mirnov. “Tokamak evolution and view to future”. en. In: *Nuclear Fusion* 59.1 (Jan. 2019), p. 015001. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/1741-4326/aaee92. URL: https://iopscience.iop.org/article/10.1088/1741-4326/aaee92 (visited on 09/02/2024) (cit. on p. 3).

[Mit97] T.M. Mitchell. *Machine Learning*. McGraw-Hill international editions - computer science series. McGraw-Hill Education, 1997. ISBN: 978-0-07-042807-2. URL: https://books.google.fr/books?id=xOGAngEACAAJ (cit. on p. 37).

[Mor+16] R. Moreno et al. “Disruption Prediction on JET during the ILW Experimental Campaigns”. en. In: *Fusion Science and Technology* 69.2 (Apr. 2016), pp. 485–494. ISSN: 1536-1055, 1943-7641. DOI: `10.13182/FST15-167`. URL: `https://www.tandfonline.com/doi/full/10.13182/FST15-167` (visited on 12/27/2024) (cit. on p. 66).

[Mor18] Edward Morse. *Nuclear Fusion*. en. Graduate Texts in Physics. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-98170-3 978-3-319-98171-0. DOI: `10.1007/978-3-319-98171-0`. URL: `http://link.springer.com/10.1007/978-3-319-98171-0` (visited on 09/04/2024) (cit. on p. 3).

[Mul+23] S. Mulas et al. “Validating neutral-beam current drive simulations in the TJ-II stellarator”. In: *Nuclear Fusion* 63.6 (June 2023), p. 066026. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/acca92`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/acca92` (visited on 02/20/2024) (cit. on p. 77).

[Mur+08] A. Murari et al. “Prototype of an adaptive disruption predictor for JET based on fuzzy logic and regression trees”. en. In: *Nuclear Fusion* 48.3 (Mar. 2008), p. 035010. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/48/3/035010`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/48/3/035010` (visited on 12/27/2024) (cit. on p. 66).

[Mur+09] A. Murari et al. “Unbiased and non-supervised learning methods for disruption prediction at JET”. en. In: *Nuclear Fusion* 49.5 (May 2009), p. 055028. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/49/5/055028`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/49/5/055028` (visited on 12/27/2024) (cit. on p. 66).

[Mur+24] Andrea Murari et al. “A control oriented strategy of disruption prediction to avoid the configuration collapse of tokamak reactors”. en. In: *Nature Communications* 15.1 (Mar. 2024), p. 2424. ISSN: 2041-1723. DOI: `10.1038/s41467-024-46242-7`. URL: `https://www.nature.com/articles/s41467-024-46242-7` (visited on 12/27/2024) (cit. on p. 66).

[Mur22] Kevin P. Murphy. *Probabilistic Machine Learning: An introduction*. MIT Press, 2022. URL: `probml.ai` (cit. on pp. 37, 40, 41, 44, 103, 106).

[NS94] Laurent Najman et al. “Watershed of a continuous function”. en. In: *Signal Processing* 38.1 (July 1994), pp. 99–112. ISSN: 01651684. DOI: `10.1016/0165-1684(94)90059-0`. URL: `https://linkinghub.elsevier.com/retrieve/pii/0165168494900590` (visited on 08/19/2024) (cit. on pp. 60, 142).

[Nar92] C Nardone. “Multichannel fluctuation data analysis by the singular value decomposition method. Application to MHD modes in JET”. In: *Plasma Physics and Controlled Fusion* 34.9 (Sept. 1992), pp. 1447–1465. ISSN: 0741-3335, 1361-6587. DOI: `10.1088/0741-3335/34/9/001`. URL: `https://iopscience.iop.org/article/10.1088/0741-3335/34/9/001` (visited on 03/30/2024) (cit. on p. 66).

[Naz+97] R. Nazikian et al. “Alpha-Particle-Driven Toroidal Alfvén Eigenmodes in the Tokamak Fusion Test Reactor”. en. In: *Physical Review Letters* 78.15 (Apr. 1997), pp. 2976–2979. ISSN: 0031-9007, 1079-7114. DOI: `10.1103/PhysRevLett.78.2976`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.78.2976` (visited on 04/10/2024) (cit. on p. 125).

[NP14] Peer Neubert et al. “Compact Watershed and Preemptive SLIC: On Improving Trade-offs of Superpixel Segmentation Algorithms”. In: *2014 22nd International Conference on Pattern Recognition*. Stockholm, Sweden: IEEE, Aug. 2014, pp. 996–1001. ISBN: 978-1-4799-5209-0. DOI: `10.1109/ICPR.2014.181`. URL: `http://ieeexplore.ieee.org/document/6976891/` (visited on 08/19/2024) (cit. on p. 142).

[Ng+15] Choon-Ching Ng et al. “Automatic Wrinkle Detection Using Hybrid Hessian Filter”. en. In: *Computer Vision – ACCV 2014*. Ed. by Daniel Cremers et al. Vol. 9005. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 609–622. ISBN: 978-3-319-16810-4 978-3-319-16811-1. DOI: `10.1007/978-3-319-16811-1_40`. URL: `https://link.springer.com/10.1007/978-3-319-16811-1_40` (visited on 02/29/2024) (cit. on pp. xii, 146–148).

[NA12] Mark S. Nixon et al. *Feature extraction & image processing for computer vision*. eng. 3rd ed. Oxford: Academic Press, 2012. ISBN: 978-0-12-396549-3 (cit. on pp. 142, 144).

[OMa+20] Niall O’Mahony et al. “Deep Learning vs. Traditional Computer Vision”. en. In: *Advances in Computer Vision*. Ed. by Kohei Arai et al. Vol. 943. Series Title: Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2020, pp. 128–144. ISBN: 978-3-030-17794-2 978-3-030-17795-9. DOI: `10.1007/978-3-030-17795-9_10`. URL: `http://link.springer.com/10.1007/978-3-030-17795-9_10` (visited on 03/05/2024) (cit. on p. 59).

[OMM13] Michal Odstrcil et al. “Comparison of Advanced Machine Learning Tools for Disruption Prediction and Disruption Studies”. en. In: *IEEE Transactions on Plasma Science* 41.7 (July 2013), pp. 1751–1759. ISSN: 0093-3813, 1939-9375. DOI: `10.1109/TPS.2013.2264880`. URL: `http://ieeexplore.ieee.org/document/6531635/` (visited on 12/27/2024) (cit. on p. 66).

[ON21] Nati Ofir et al. *Classic versus deep learning approaches to address computer vision challenges*. en. arXiv:2101.09744 [cs]. Oct. 2021. URL: `http://arxiv.org/abs/2101.09744` (visited on 04/04/2024) (cit. on p. 59).

[Oli+19] H.J.C. Oliver et al. "Modification of the Alfvén wave spectrum by pellet injection". en. In: *Nuclear Fusion* 59.10 (Oct. 2019), p. 106031. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ab382b`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ab382b` (visited on 04/09/2024) (cit. on p. 129).

[OF97a] Bruno A Olshausen et al. "Sparse coding with an overcomplete basis set: A strategy employed by V1?" In: *Vision research* 37.23 (1997), pp. 3311–3325 (cit. on pp. 56, 61).

[OF97b] Bruno A. Olshausen et al. "Sparse coding with an overcomplete basis set: A strategy employed by V1?" en. In: *Vision Research* 37.23 (Dec. 1997), pp. 3311–3325. ISSN: 00426989. DOI: `10.1016/S0042-6989(97)00169-7`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0042698997001697` (visited on 02/20/2024) (cit. on p. 75).

[Oro23] Naveed Oroumchian. "ECEI Spectrogram Denoising and Feature Extraction using a Multi-Wavelet Convolutional Neural Network Model". en. In: *Master thesis* (2023). DOI: `MerrittID:ark:/13030/m5h52mx2`. URL: `https://escholarship.org/uc/item/2jc5q2sg` (cit. on p. 66).

[Oro] Naveed Oroumchian. "ECEI Spectrogram Denoising and Feature Extraction using a Multi-Wavelet Convolutional Neural Network Model". en. In: () (cit. on p. 127).

[Ots79] Nobuyuki Otsu. "A Threshold Selection Method from Gray-Level Histograms". In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (Jan. 1979). Conference Name: IEEE Transactions on Systems, Man, and Cybernetics, pp. 62–66. ISSN: 2168-2909. DOI: `10.1109/TSMC.1979.4310076`. URL: `https://ieeexplore.ieee.org/document/4310076` (visited on 02/28/2024) (cit. on pp. xii, 145, 147).

[Pas+19] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. URL: `https://papers.nips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html` (visited on 02/26/2024) (cit. on pp. ix, 83).

[Pau+02] G Pautasso et al. "On-line prediction and mitigation of disruptions in ASDEX Upgrade". en. In: *Nuclear Fusion* 42.1 (Jan. 2002), pp. 100–108. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/42/1/314`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/42/1/314` (visited on 12/27/2024) (cit. on p. 66).

[Pea+69] N. J. Peacock et al. “Measurement of the Electron Temperature by Thomson Scattering in Tokamak T3”. en. In: *Nature* 224.5218 (Nov. 1969), pp. 488–490. ISSN: 0028-0836, 1476-4687. DOI: `10.1038/224488a0`. URL: `https://www.nature.com/articles/224488a0` (visited on 09/04/2024) (cit. on p. 5).

[Ped+16] T. Sunn Pedersen et al. “Confirmation of the topology of the Wendelstein 7-X magnetic field to better than 1:100,000”. en. In: *Nature Communications* 7.1 (Nov. 2016), p. 13493. ISSN: 2041-1723. DOI: `10.1038/ncomms13493`. URL: `https://www.nature.com/articles/ncomms13493` (visited on 09/02/2024) (cit. on p. 5).

[Ped+11] Fabian Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12.85 (2011), pp. 2825–2830. ISSN: 1533-7928. URL: `http://jmlr.org/papers/v12/pedregosa11a.html` (visited on 02/20/2024) (cit. on p. 103).

[Pin96] S. D. Pinches. “Nonlinear interaction of fast particles with Alfvén waves in tokamaks”. PhD thesis. University of Nottingham, 1996 (cit. on p. 26).

[Pin+15] S. D. Pinches et al. “Energetic ions in ITER plasmas”. en. In: *Physics of Plasmas* 22.2 (Feb. 2015), p. 021807. ISSN: 1070-664X, 1089-7674. DOI: `10.1063/1.4908551`. URL: `https://pubs.aip.org/pop/article/22/2/021807/110109/Energetic-ions-in-ITER-plasmas` (visited on 03/06/2024) (cit. on pp. 28, 30).

[Pin+98] SD Pinches et al. “The HAGIS self-consistent nonlinear wave-particle interaction model”. In: *Computer Physics Communications* 111.1-3 (1998), pp. 133–149 (cit. on p. 28).

[Poe+96] S. Poedts et al. *HELENA Installation and User Guide*. RIJNHUIZEN REPORT 98-228, Jan. 1996 (cit. on p. 18).

[Pok+10] G I Pokol et al. “A wavelet based method for detecting transient plasma waves and determining their spatial structure”. en. In: (2010) (cit. on pp. 65, 66).

[PB09] D.G. Pretty et al. “A data mining algorithm for automated characterisation of fluctuations in multichannel timeseries”. en. In: *Computer Physics Communications* 180.10 (Oct. 2009), pp. 1768–1776. ISSN: 00104655. DOI: `10.1016/j.cpc.2009.05.003`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0010465509001507` (visited on 03/05/2024) (cit. on p. 66).

[PM66] Judith M. S. Prewitt et al. “THE ANALYSIS OF CELL IMAGES*”. en. In: *Annals of the New York Academy of Sciences* 128.3 (Jan. 1966), pp. 1035–1053. ISSN: 0077-8923, 1749-6632. DOI: `10.1111/j.1749-6632.1965.tb11715.x`. URL: `https://nyaspubs.onlinelibrary.wiley.com/doi/10.1111/j.1749-6632.1965.tb11715.x` (visited on 08/10/2024) (cit. on pp. xii, 147).

[Pri23] Simon J. D. Prince. *Understanding deep learning*. Cambridge, Massachusetts: The MIT Press, 2023. ISBN: 978-0-262-04864-4 (cit. on pp. 36, 37, 48).

[RPK17] Maziar Raissi et al. *Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations*. arXiv:1711.10561 [cs, math, stat]. Nov. 2017. DOI: `10.48550/arXiv.1711.10561`. URL: `http://arxiv.org/abs/1711.10561` (visited on 09/23/2024) (cit. on p. 36).

[RW05] Carl Edward Rasmussen et al. *Gaussian Processes for Machine Learning*. en. The MIT Press, Nov. 2005. ISBN: 978-0-262-25683-4. DOI: `10.7551/mitpress/3206.001.0001`. URL: `https://direct.mit.edu/books/monograph/2320/Gaussian-Processes-for-Machine-Learning` (visited on 09/23/2024) (cit. on p. 38).

[Rax05] Jean-Marcel Rax. *Physique des plasmas: cours et applications*. fre. Sciences sup. Paris: Dunod, 2005. ISBN: 978-2-10-007250-7 (cit. on p. 15).

[Rax11] Jean-Marcel Rax. *Physique des Tokamaks*. fre. OCLC: 713180855. Palaiseau: Ecole polytechnique, 2011. ISBN: 978-2-7302-1580-0 (cit. on pp. 9, 15–19, 22).

[Rea+18] C Rea et al. "Disruption prediction investigations using Machine Learning tools on DIII-D and Alcator C-Mod". en. In: *Plasma Physics and Controlled Fusion* 60.8 (Aug. 2018), p. 084004. ISSN: 0741-3335, 1361-6587. DOI: `10.1088/1361-6587/aac7fe`. URL: `https://iopscience.iop.org/article/10.1088/1361-6587/aac7fe` (visited on 12/27/2024) (cit. on p. 66).

[Rea23] Cristina Rea. "A Review of Machine Learning Applications to Disruptions". en. In: (2023) (cit. on p. 66).

[RG18] Cristina Rea et al. "Exploratory Machine Learning Studies for Disruption Prediction Using Large Databases on DIII-D". en. In: *Fusion Science and Technology* 74.1-2 (Aug. 2018), pp. 89–100. ISSN: 1536-1055, 1943-7641. DOI: `10.1080/15361055.2017.1407206`. URL: `https://www.tandfonline.com/doi/full/10.1080/15361055.2017.1407206` (visited on 12/27/2024) (cit. on p. 66).

[Red+16] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. arXiv:1506.02640 [cs]. May 2016. DOI: `10.48550/arXiv.1506.02640`. URL: `http://arxiv.org/abs/1506.02640` (visited on 09/18/2024) (cit. on p. 169).

[Rei+24] Dillon Reis et al. *Real-Time Flying Object Detection with YOLOv8*. arXiv:2305.09972 [cs]. May 2024. DOI: `10.48550/arXiv.2305.09972`. URL: `http://arxiv.org/abs/2305.09972` (visited on 09/16/2024) (cit. on p. 169).

[RST22] Nicholas Richardson et al. "SRMD: Sparse Random Mode Decomposition". In: (2022). Publisher: arXiv Version Number: 2. DOI: 10.48550/ARXIV.2204.06108. URL: https://arxiv.org/abs/2204.06108 (visited on 02/20/2024) (cit. on pp. viii, 68, 73, 79, 82, 88).

[Ric08] David S. Richeson. *Euler's gem: the polyhedron formula and the birth of topology*. OCLC: ocn225091369. Princeton, N.J: Princeton University Press, 2008. ISBN: 978-0-691-12677-7 (cit. on p. 11).

[RC78] T. W. Ridler et al. "Picture Thresholding Using an Iterative Selection Method". In: *IEEE Transactions on Systems, Man, and Cybernetics* 8.8 (1978), pp. 630–632. ISSN: 0018-9472. DOI: 10.1109/TSMC.1978.4310039. URL: http://ieeexplore.ieee.org/document/4310039/ (visited on 02/28/2024) (cit. on pp. xii, 147).

[Ros14] Brian C. Ross. "Mutual Information between Discrete and Continuous Data Sets". en. In: *PLOS ONE* 9.2 (Feb. 2014). Publisher: Public Library of Science, e87357. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0087357. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0087357 (visited on 02/27/2024) (cit. on p. 103).

[RCM82] David W. Ross et al. "Kinetic description of Alfvén wave heating". en. In: *The Physics of Fluids* 25.4 (Apr. 1982), pp. 652–667. ISSN: 0031-9171. DOI: 10.1063/1.863789. URL: https://pubs.aip.org/pfl/article/25/4/652/825100/Kinetic-description-of-Alfven-wave-heating (visited on 09/12/2024) (cit. on p. 25).

[Rud17] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. arXiv:1609.04747 [cs]. June 2017. DOI: 10.48550/arXiv.1609.04747. URL: http://arxiv.org/abs/1609.04747 (visited on 09/23/2024) (cit. on p. 46).

[Rud+22] Nikita Rudin et al. *Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning*. arXiv:2109.11978 [cs]. Aug. 2022. DOI: 10.48550/arXiv.2109.11978. URL: http://arxiv.org/abs/2109.11978 (visited on 09/23/2024) (cit. on p. 36).

[San04] Bu¨lent Sankur. "Survey over image thresholding techniques and quantitative performance evaluation". en. In: *Journal of Electronic Imaging* 13.1 (Jan. 2004), p. 146. ISSN: 1017-9909. DOI: 10.1117/1.1631315. URL: http://electronicimaging.spiedigitallibrary.org/article.aspx?doi=10.1117/1.1631315 (visited on 02/28/2024) (cit. on pp. xii, 127, 147).

[SSK23] Rachana Sathish et al. *Linear Neural Network as a Fast Solver for Dictionary Learning*. en. preprint. Feb. 2023. DOI: 10.36227/techrxiv.21988181.v1. URL: https://www.techrxiv.org/doi/full/10.36227/techrxiv.21988181.v1 (visited on 02/20/2024) (cit. on p. 85).

[Sat+98] Yoshinobu Sato et al. "Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images". en. In: *Medical Image Analysis* 2.2 (June 1998), pp. 143–168. ISSN: 13618415. DOI: `10.1016/S1361-8415(98)80009-1`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1361841598800091` (visited on 02/29/2024) (cit. on pp. xii, 146, 147).

[SM13] O Sauter et al. "Tokamak coordinate conventions: COCOS". In: *Computer Physics Communications* 184.2 (2013). Publisher: Elsevier, pp. 293–302. DOI: `10.1016/j.cpc.2012.09.010` (cit. on p. 14).

[Sch+17] Erich Schubert et al. "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN". en. In: *ACM Transactions on Database Systems* 42.3 (Sept. 2017), pp. 1–21. ISSN: 0362-5915, 1557-4644. DOI: `10.1145/3068335`. URL: `https://dl.acm.org/doi/10.1145/3068335` (visited on 03/06/2024) (cit. on p. 38).

[Seo+24] Jaemin Seo et al. "Avoiding fusion plasma tearing instability with deep reinforcement learning". en. In: *Nature* 626.8000 (Feb. 2024), pp. 746–751. ISSN: 0028-0836, 1476-4687. DOI: `10.1038/s41586-024-07024-9`. URL: `https://www.nature.com/articles/s41586-024-07024-9` (visited on 02/28/2024) (cit. on p. 66).

[Sha48] C E Shannon. "A Mathematical Theory of Communication". en. In: *The Bell System Technical Journal* 27 (Oct. 1948), pp. 379–423, 623–656 (cit. on p. 100).

[Sha+01] S. E Sharapov et al. "MHD spectroscopy through detecting toroidal Alfvén eigenmodes and Alfvén wave cascades". In: *Physics Letters A* 289.3 (Oct. 2001), pp. 127–134. ISSN: 0375-9601. DOI: `10.1016/S0375-9601(01)00588-6`. URL: `https://www.sciencedirect.com/science/article/pii/S0375960101005886` (visited on 09/23/2024) (cit. on p. 28).

[Sha21] S. E. Sharapov. *Energetic particles in tokamak plasmas*. First edition. Boca Raton: CRC Press, 2021. ISBN: 978-1-138-54554-0 (cit. on pp. 19–21, 26, 30, 31).

[Sha+06] S.E Sharapov et al. "Alfvén cascades in JET discharges with NBI-heating". en. In: *Nuclear Fusion* 46.10 (Oct. 2006), S868–S879. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/46/10/S02`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/46/10/S02` (visited on 04/10/2024) (cit. on p. 125).

[Sha+13] S.E. Sharapov et al. "Energetic particle instabilities in fusion plasmas". en. In: *Nuclear Fusion* 53.10 (Oct. 2013), p. 104022. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/53/10/104022`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/53/10/104022` (visited on 04/10/2024) (cit. on pp. 8, 13, 23, 24, 28, 125).

[SLW16] Bin Shen et al. “Elastic net regularized dictionary learning for image classification”. en. In: *Multimedia Tools and Applications* 75.15 (Aug. 2016), pp. 8861–8874. ISSN: 1380-7501, 1573-7721. DOI: `10.1007/s11042-014-2257-y`. URL: `http://link.springer.com/10.1007/s11042-014-2257-y` (visited on 03/06/2024) (cit. on p. 85).

[She+23] C. Shen et al. “IDP-PGFE: an interpretable disruption predictor based on physics-guided feature extraction”. en. In: *Nuclear Fusion* 63.4 (Apr. 2023), p. 046024. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/acbe0f`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/acbe0f` (visited on 12/27/2024) (cit. on p. 66).

[20] *Single Layer Neural Network : Adaptive Linear Neuron using linear (identity) activation function with batch gradient method - 2020*. 2020. URL: `https://www.bogotobogo.com/python/scikit-learn/Single-Layer-Neural-Network-Adaptive-Linear-Neuron.php` (visited on 09/23/2024) (cit. on p. 45).

[SS12] Deviderjit S. Sivia et al. *Data analysis: a Bayesian tutorial; [for scientists and engineers]*. eng. 2. ed., repr. Oxford science publications. Oxford: Oxford Univ. Press, 2012. ISBN: 978-0-19-856832-2 (cit. on p. 39).

[Škv+20] Vít Škvára et al. “Detection of Alfvén Eigenmodes on COMPASS with Generative Neural Networks”. In: *Fusion Science and Technology* 76.8 (Nov. 2020), pp. 962–971. ISSN: 1536-1055. DOI: `10.1080/15361055.2020.1820805`. URL: `https://doi.org/10.1080/15361055.2020.1820805` (visited on 02/20/2024) (cit. on p. 65).

[Sob+22] Irwin Sobel et al. “Sobel-feldman operator”. In: *Preprint at https://www.researchgate. net/profile/Irwin-Sobel/publication/285159837. Accessed* 20 (2022) (cit. on p. 60).

[SL09] Marina Sokolova et al. “A systematic analysis of performance measures for classification tasks”. In: *Information processing & management* 45.4 (2009), pp. 427–437 (cit. on p. 216).

[Spi58] L. Jr. Spitzer. *The stellarator concept.* Conference INIS-XU–022. INIS-XU–022 INIS Reference Number: 39084571. United Nations (UN), Geneva (Switzerland), 1958, p. 474 (cit. on p. 4).

[Spo+01] D.A Spong et al. “Physics issues of compact drift optimized stellarators”. In: *Nuclear Fusion* 41.6 (June 2001), pp. 711–716. ISSN: 0029-5515. DOI: `10.1088/0029-5515/41/6/305`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/41/6/305` (visited on 09/13/2024) (cit. on p. 18).

[Spo+21] D.A. Spong et al. “Nonlinear dynamics and transport driven by energetic particle instabilities using a gyro-Landau closure model *”. en. In: *Nuclear Fusion* 61.11 (Nov. 2021), p. 116061. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/ac2990`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/ac2990` (visited on 03/05/2024) (cit. on p. 28).

[Sto22] James V. Stone. *Information theory: a tutorial introduction*. eng. Second edition. OCLC: 1395391624. Sheffield, United Kingdom: Sebtel Press, 2022. ISBN: 978-1-73967-270-6 (cit. on pp. 100–103).

[Str82] Jan-Olov Strömberg. “A modified Franklin system and higher-order spline systems on Rn as unconditional bases for Hardy spaces”. In: *Conference on harmonic analysis in honor of Antoni Zygmund*. Vol. 2. 1982, pp. 475–494 (cit. on p. 53).

[Str97] E Strumberger. “Finite- magnetic field line tracing for Helias configurations”. In: *Nuclear Fusion* 37.1 (Jan. 1997), pp. 19–27. ISSN: 0029-5515. DOI: `10.1088/0029-5515/37/1/I03`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/37/1/I03` (visited on 09/13/2024) (cit. on p. 18).

[SG16] E Strumberger et al. “CASTOR3D: linear stability studies for 2D and 3D tokamak equilibria”. In: *Nuclear Fusion* 57.1 (2016), p. 016032 (cit. on p. 28).

[SHL18] David Stutz et al. “Superpixels: An evaluation of the state-of-the-art”. en. In: *Computer Vision and Image Understanding* 166 (Jan. 2018), pp. 1–27. ISSN: 10773142. DOI: `10.1016/j.cviu.2017.03.007`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1077314217300589` (visited on 08/19/2024) (cit. on p. 142).

[Sud+17] Carole H. Sudre et al. “Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations”. In: vol. 10553. arXiv:1707.03237 [cs]. 2017, pp. 240–248. DOI: `10.1007/978-3-319-67558-9_28`. URL: `http://arxiv.org/abs/1707.03237` (visited on 09/20/2024) (cit. on pp. 169, 223).

[SOL15] B.J. Sun et al. “Alfvén eigenmodes including magnetic island effects in the TJ-II stellarator”. In: *Nuclear Fusion* 55.9 (Sept. 2015), p. 093023. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/0029-5515/55/9/093023`. URL: `https://iopscience.iop.org/article/10.1088/0029-5515/55/9/093023` (visited on 03/04/2024) (cit. on pp. 78, 94, 121).

[Sze22] Richard Szeliski. *Computer vision: algorithms and applications*. eng. Second edition. Texts in computer science. Cham: Springer, 2022. ISBN: 978-3-030-34372-9 978-3-030-34371-2 978-3-030-34374-3 (cit. on pp. 50, 58, 59, 145).

[24e] *The ADALINE - Theory and Implementation of the First Neural Network Trained With Gradient Descent.* 2024. URL: https://pabloinsente.github.io/the-adaline (visited on 09/23/2024) (cit. on p. 45).

[Tib96] Robert Tibshirani. "Regression Shrinkage and Selection Via the Lasso". en. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58.1 (Jan. 1996), pp. 267–288. ISSN: 1369-7412, 1467-9868. DOI: 10.1111/j.2517-6161.1996.tb02080.x. URL: https://academic.oup.com/jrsssb/article/58/1/267/7027929 (visited on 09/23/2024) (cit. on p. 48).

[Tib11] Robert Tibshirani. "Regression Shrinkage and Selection via The Lasso: A Retrospective". en. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 73.3 (June 2011), pp. 273–282. ISSN: 1369-7412, 1467-9868. DOI: 10.1111/j.1467-9868.2011.00771.x. URL: https://academic.oup.com/jrsssb/article/73/3/273/7034363 (visited on 02/26/2024) (cit. on pp. 81, 82).

[Til15] Andreas M. Tillmann. "On the Computational Intractability of Exact and Approximate Dictionary Learning". In: *IEEE Signal Processing Letters* 22.1 (Jan. 2015). Conference Name: IEEE Signal Processing Letters, pp. 45–49. ISSN: 1558-2361. DOI: 10.1109/LSP.2014.2345761. URL: https://ieeexplore.ieee.org/abstract/document/6873279 (visited on 02/20/2024) (cit. on pp. 75, 81).

[Tin+22] R.A. Tinguely et al. "Simultaneous measurements of unstable and stable Alfvén eigenmodes in JET". en. In: *Nuclear Fusion* 62.11 (Nov. 2022), p. 112008. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/1741-4326/ac899e. URL: https://iopscience.iop.org/article/10.1088/1741-4326/ac899e (visited on 03/06/2024) (cit. on p. 66).

[TZ21] Una Tuba et al. "Image Denoising by Discrete Wavelet Transform with Edge Preservation". In: *2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI).* Pitesti, Romania: IEEE, July 2021, pp. 1–4. ISBN: 978-1-66542-534-6. DOI: 10.1109/ECAI52376.2021.9515079. URL: https://ieeexplore.ieee.org/document/9515079/ (visited on 02/28/2024) (cit. on p. 127).

[Ult] Ultralytics. *YOLOv8.* en. URL: https://docs.ultralytics.com/models/yolov8 (visited on 09/16/2024) (cit. on pp. xiii, 168).

[VF09] Ewout Van Den Berg et al. "Probing the Pareto Frontier for Basis Pursuit Solutions". en. In: *SIAM Journal on Scientific Computing* 31.2 (Jan. 2009), pp. 890–912. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/080714488. URL: http://epubs.siam.org/doi/10.1137/080714488 (visited on 02/20/2024) (cit. on pp. 82, 85).

[VF11] Ewout Van Den Berg et al. “Sparse Optimization with Least-Squares Constraints”. en. In: *SIAM Journal on Optimization* 21.4 (Oct. 2011), pp. 1201–1229. ISSN: 1052-6234, 1095-7189. DOI: `10.1137/100785028`. URL: `http://epubs.siam.org/doi/10.1137/100785028` (visited on 02/20/2024) (cit. on pp. 82, 85).

[Van+14] Stéfan Van Der Walt et al. “scikit-image: image processing in Python”. en. In: *PeerJ* 2 (June 2014), e453. ISSN: 2167-8359. DOI: `10.7717/peerj.453`. URL: `https://peerj.com/articles/453` (visited on 02/28/2024) (cit. on pp. 61, 62, 127, 153).

[Van+06] M. A. Van Zeeland et al. “Radial Structure of Alfvén Eigenmodes in the DIII-D Tokamak through Electron-Cyclotron-Emission Measurements”. en. In: *Physical Review Letters* 97.13 (Sept. 2006), p. 135001. ISSN: 0031-9007, 1079-7114. DOI: `10.1103/PhysRevLett.97.135001`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.97.135001` (visited on 04/10/2024) (cit. on p. 125).

[VSG17] J. Varela et al. “Analysis of Alfven eigenmodes destabilization by energetic particles in TJ-II using a Landau-closure model”. In: *Nuclear Fusion* 57.12 (Dec. 2017), p. 126019. ISSN: 0029-5515, 1741-4326. DOI: `10.1088/1741-4326/aa83c4`. URL: `https://iopscience.iop.org/article/10.1088/1741-4326/aa83c4` (visited on 03/04/2024) (cit. on pp. 78, 94, 121).

[Vas+23] Ashish Vaswani et al. *Attention Is All You Need.* arXiv:1706.03762 [cs]. Aug. 2023. DOI: `10.48550/arXiv.1706.03762`. URL: `http://arxiv.org/abs/1706.03762` (visited on 09/23/2024) (cit. on p. 36).

[Veg+09] J. Vega et al. “Overview of intelligent data retrieval methods for waveforms and images in massive fusion databases”. en. In: *Fusion Engineering and Design* 84.7-11 (June 2009), pp. 1916–1919. ISSN: 09203796. DOI: `10.1016/j.fusengdes.2008.11.097`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S092037960800416X` (visited on 03/05/2024) (cit. on p. 66).

[Veg+15] J. Vega et al. “Disruption precursor detection: Combining the time and frequency domains”. en. In: *2015 IEEE 26th Symposium on Fusion Engineering (SOFE)*. Austin, TX: IEEE, May 2015, pp. 1–8. ISBN: 978-1-4799-8264-6. DOI: `10.1109/SOFE.2015.7482361`. URL: `http://ieeexplore.ieee.org/document/7482361/` (visited on 12/27/2024) (cit. on p. 66).

[Veg+20] J. Vega et al. “Automatic recognition of plasma relevant events: Implications for ITER”. en. In: *Fusion Engineering and Design* 156 (July 2020), p. 111638. ISSN: 09203796. DOI: `10.1016/j.fusengdes.2020.111638`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0920379620301861` (visited on 12/30/2024) (cit. on p. 66).

[Veg+22] J. Vega et al. “Disruption prediction with artificial intelligence techniques in tokamak plasmas”. en. In: *Nature Physics* 18.7 (July 2022), pp. 741–750. ISSN: 1745-2473, 1745-2481. DOI: 10.1038/s41567-022-01602-2. URL: https://www.nature.com/articles/s41567-022-01602-2 (visited on 12/27/2024) (cit. on p. 66).

[Veg+13] Jesús Vega et al. “Results of the JET real-time disruption predictor in the ITER-like wall campaigns”. en. In: *Fusion Engineering and Design* 88.6-8 (Oct. 2013), pp. 1228–1231. ISSN: 09203796. DOI: 10.1016/j.fusengdes.2013.03.003. URL: https://linkinghub.elsevier.com/retrieve/pii/S0920379613002974 (visited on 12/27/2024) (cit. on p. 66).

[Veg+16] Jesús Vega et al. “Advanced Disruption Predictor Based On The Locked Mode Signal: Application To Jet”. en. In: *Proceedings of 1st EPS conference on Plasma Diagnostics — PoS(ECPD2015)*. Frascati, Italy: Sissa Medialab, Oct. 2016, p. 028. DOI: 10.22323/1.240.0028. URL: https://pos.sissa.it/240/028 (visited on 12/27/2024) (cit. on p. 66).

[VV24] Ajantha Vijayakumar et al. “YOLO-based Object Detection Models: A Review and its Applications”. en. In: *Multimedia Tools and Applications* (Mar. 2024). ISSN: 1573-7721. DOI: 10.1007/s11042-024-18872-y. URL: https://doi.org/10.1007/s11042-024-18872-y (visited on 09/23/2024) (cit. on p. 36).

[Wag07] F Wagner. “A quarter-century of H-mode studies”. In: *Plasma Physics and Controlled Fusion* 49.12B (Dec. 2007), B1–B33. ISSN: 0741-3335, 1361-6587. DOI: 10.1088/0741-3335/49/12B/S01. URL: https://iopscience.iop.org/article/10.1088/0741-3335/49/12B/S01 (visited on 09/04/2024) (cit. on pp. 3, 5).

[Wag+82] F. Wagner et al. “Regime of Improved Confinement and High Beta in Neutral-Beam-Heated Divertor Discharges of the ASDEX Tokamak”. en. In: *Physical Review Letters* 49.19 (Nov. 1982), pp. 1408–1412. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.49.1408. URL: https://link.aps.org/doi/10.1103/PhysRevLett.49.1408 (visited on 09/04/2024) (cit. on p. 5).

[Wel24] Welch Labs. *AI can’t cross this line and we don’t know why.* Sept. 2024. URL: https://www.youtube.com/watch?v=5eqRuVp65eY (visited on 09/23/2024) (cit. on p. 44).

[WC11] John Wesson et al. *Tokamaks.* Vol. 149. Oxford university press, 2011 (cit. on pp. v, 7, 15, 18, 30–32, 164, 165).

[WKO12] Holger Winnemöller et al. “XDoG: An eXtended difference-of-Gaussians compendium including advanced image stylization”. en. In: *Computers & Graphics* 36.6 (Oct. 2012), pp. 740–753. ISSN: 00978493. DOI: 10.1016/j.cag.2012.03.004. URL: https://linkinghub.elsevier.com/

retrieve/pii/S009784931200043X (visited on 03/06/2024) (cit. on pp. 144, 145).

[Wol96] David H. Wolpert. "The Lack of A Priori Distinctions Between Learning Algorithms". en. In: *Neural Computation* 8.7 (Oct. 1996), pp. 1341–1390. ISSN: 0899-7667, 1530-888X. DOI: 10.1162/neco.1996.8.7.1341. URL: https://direct.mit.edu/neco/article/8/7/1341-1390/6016 (visited on 07/27/2024) (cit. on p. 44).

[Woo+20] Benjamin J. Q. Woods et al. "Machine Learning Characterization of Alfvénic and Sub-Alfvénic Chirping and Correlation With Fast-Ion Loss at NSTX". In: *IEEE Transactions on Plasma Science* 48.1 (Jan. 2020). Conference Name: IEEE Transactions on Plasma Science, pp. 71–81. ISSN: 1939-9375. DOI: 10.1109/TPS.2019.2960206. URL: https://ieeexplore.ieee.org/document/8966668 (visited on 02/20/2024) (cit. on p. 65).

[WOS05] Kesheng Wu et al. "Optimizing connected component labeling algorithms". In: *Medical Imaging 2005: Image Processing*. Vol. 5747. SPIE, Apr. 2005, pp. 1965–1976. DOI: 10.1117/12.596105. URL: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5747/0000/Optimizing-connected-component-labeling-algorithms/10.1117/12.596105.full (visited on 04/11/2024) (cit. on pp. xii, 153).

[Wu+19] Yuxin Wu et al. *Detectron2*. https://github.com/facebookresearch/detectron2. 2019 (cit. on pp. xiii, 168).

[WH22] Samuel E. Wurzel et al. "Progress toward fusion energy breakeven and gain as measured against the Lawson criterion". en. In: *Physics of Plasmas* 29.6 (June 2022), p. 062103. ISSN: 1070-664X, 1089-7674. DOI: 10.1063/5.0083990. URL: https://pubs.aip.org/pop/article/29/6/062103/2847827/Progress-toward-fusion-energy-breakeven-and-gain (visited on 09/10/2024) (cit. on p. 21).

[Yam+10] Satoshi Yamamoto et al. "Studies of MHD Stability Using Data Mining Technique in Helical Plasmas". en. In: *Plasma and Fusion Research* 5 (2010), pp. 034–034. ISSN: 1880-6821. DOI: 10.1585/pfr.5.034. URL: http://www.jstage.jst.go.jp/article/pfr/5/0/5_0_034/_article (visited on 03/05/2024) (cit. on p. 66).

[YCC95] Jui-Cheng Yen et al. "A new criterion for automatic multilevel thresholding". In: *IEEE Transactions on Image Processing* 4.3 (Mar. 1995). Conference Name: IEEE Transactions on Image Processing, pp. 370–378. ISSN: 1941-0042. DOI: 10.1109/83.366472. URL: https://ieeexplore.ieee.org/document/366472 (visited on 05/10/2024) (cit. on pp. xii, 147).

[ZRL77] G W Zack et al. "Automatic measurement of sister chromatid exchange frequency." en. In: *Journal of Histochemistry & Cytochemistry* 25.7 (July 1977), pp. 741–753. ISSN: 0022-1554, 1551-5044. DOI: 10.1177/25.7.

70454. URL: http://journals.sagepub.com/doi/10.1177/25.7.70454 (visited on 08/10/2024) (cit. on pp. xii, 147).

[Zap+24a] E D D Zapata-Cornejo et al. "Segmentation of MHD modes using Fourier transform, wavelets and computer vision algorithms". In: *Plasma Physics and Controlled Fusion* 66.9 (Sept. 2024), p. 095016. ISSN: 0741-3335, 1361-6587. DOI: 10.1088/1361-6587/ad6a84. URL: https://iopscience.iop.org/article/10.1088/1361-6587/ad6a84 (visited on 09/16/2024) (cit. on p. 2).

[Zap+24b] E.d.D. Zapata-Cornejo et al. "A novel unsupervised machine learning algorithm for automatic Alfvénic activity detection in the TJ-II stellarator". In: *Nuclear Fusion* 64.12 (Dec. 2024), p. 126057. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/1741-4326/ad85f4. URL: https://iopscience.iop.org/article/10.1088/1741-4326/ad85f4 (visited on 12/30/2024) (cit. on p. 2).

[Zeg+06] Stefan Zegenhagen et al. "Analysis of Alfvén eigenmodes in stellarators using non-evenly spaced probes". In: *Plasma physics and controlled fusion* 48.9 (2006). Publisher: IOP Publishing, p. 1333 (cit. on p. 66).

[Zhe+23] Wei Zheng et al. "Disruption prediction for future tokamaks using parameter-based transfer learning". en. In: *Communications Physics* 6.1 (July 2023), p. 181. ISSN: 2399-3650. DOI: 10.1038/s42005-023-01296-9. URL: https://www.nature.com/articles/s42005-023-01296-9 (visited on 03/06/2024) (cit. on p. 66).

[Zoh15] Hartmut Zohm. *Magnetohydrodynamic stability of tokamaks*. Weinheim, Germany: Wiley-VCH Verlag GmbH & Co. KGaA, 2015. ISBN: 978-3-527-41232-7 (cit. on p. 15).

[Zon+09] F. Zonca et al. "High-frequency fishbones at JET: theoretical interpretation of experimental observations". In: *Nuclear Fusion* 49.8 (Aug. 2009), p. 085009. ISSN: 0029-5515, 1741-4326. DOI: 10.1088/0029-5515/49/8/085009. URL: https://iopscience.iop.org/article/10.1088/0029-5515/49/8/085009 (visited on 07/05/2024) (cit. on p. 167).

[ZH05] Hui Zou et al. "Regularization and Variable Selection Via the Elastic Net". en. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 67.2 (Apr. 2005), pp. 301–320. ISSN: 1369-7412, 1467-9868. DOI: 10.1111/j.1467-9868.2005.00503.x. URL: https://academic.oup.com/jrsssb/article/67/2/301/7109482 (visited on 02/26/2024) (cit. on pp. ix, 81–83, 90).

[Zyl+22] A. B. Zylstra et al. "Experimental achievement and signatures of ignition at the National Ignition Facility". en. In: *Physical Review E* 106.2 (Aug. 2022), p. 025202. ISSN: 2470-0045, 2470-0053. DOI: 10.1103/PhysRevE.106.025202. URL: https://link.aps.org/doi/10.1103/PhysRevE.106.025202 (visited on 09/02/2024) (cit. on p. 5).

Appendixes

A. Evaluation metrics

Metric	Formula	Evaluates
		Error-based Metrics
Mean Squared Error (MSE)	$\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$	Average squared difference between actual and predicted values
Root Mean Squared Error (RMSE)	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$	Square root of MSE, interpretable in original units
Mean Absolute Error (MAE)	$\frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}_i\rvert$	Average absolute difference between actual and predicted values
		Percentage-based Metrics
Mean Absolute Percentage Error (MAPE)	$\frac{100\%}{N}\sum_{i=1}^{N}\left\lvert\frac{y_i - \hat{y}_i}{y_i}\right\rvert$	Average percentage difference between actual and predicted values
Symmetric Mean Absolute Percentage Error (sMAPE)	$\frac{100\%}{N}\sum_{i=1}^{N}\frac{\lvert y_i - \hat{y}_i\rvert}{(\lvert y_i\rvert + \lvert\hat{y}_i\rvert)/2}$	Average of the absolute percentage errors, symmetrical
		Scale-independent Metrics
R-squared (R^2)	$1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$	Correlation
		Other Metrics
Mean Squared Logarithmic Error (MSLE)	$\frac{1}{N}\sum_{i=1}^{N}\big(\log(1 + y_i) - \log(1 + \hat{y}_i)\big)^2$	Measures the ratio between actual and predicted values, less sensitive to large differences
Median Absolute Error	$\mathrm{median}(\lvert y_i - \hat{y}_i\rvert)$	Median of the absolute differences, robust to outliers

Table 1.: Some Metrics used in regression

Data Class	Classified as 0	Classified as 1
0	true negatives	false positives (Type I Error)
1	false negatives (Type II Error)	true positive

Table 2.: Confusion Matrix

Metric	Formula	Evaluates
		Threshold-based Metrics
Accuracy	$\frac{tp+tn}{tp+tn+fp+fn}$	Overall effectiveness of the classifier
Balanced Accuracy	$\frac{1}{2}\left(\frac{tp}{tp+fn} + \frac{tn}{tn+fp}\right)$	Ability to avoid false classification
		Sensitivity-Specificity Metrics
Specificity	$\frac{tn}{tn+fp}$	Effectiveness in identifying the negative class
Geometric Mean	$\sqrt{\frac{tp}{tp+fn}\cdot\frac{tn}{tn+fp}}$	Effectiveness in identifying both classes
		Precision-Recall Metrics
Precision	$\frac{tp}{tp+fp}$	Correct classifications in the positive class (fewer false positives)
Recall	$\frac{tp}{tp+fn}$	Correct classifications in the positive class (fewer false negatives)
F-score	$\frac{(\beta^2+1)\cdot tp}{(\beta^2+1)\cdot tp+\beta^2\cdot fn+fp}$	Correct classifications in the positive class (fewer false positives and false negatives) $\beta = 1.0$ or 0.5
		Ranking Metrics
AUC ROC [DDC88]	$\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} 1_{p_i>p_j}$	Ability of the classifier to
AUC PR	$\sum_{i=1}^{N} P(k)\cdot\Delta r(k)$ (areas)	avoid false classification
		Probabilistic Metrics
Brier Score	$\frac{1}{N}\sum_{i=1}^{N}\big(p(y_i) - y_i\big)^2$	
Cross-Entropy (i)	$-\frac{1}{N}\sum_{i=1}^{N} y_i\cdot\log(p_i)$	Calibrated probabilities
Log Loss (ii)	$-\frac{1}{N}\sum_{i=1}^{N} -(1 - y_i)\cdot\log\big(1 - p(y_i)\big)$	

Table 3.: Metrics used in binary classification (from [SL09])

Metric	Definition
Accuracy	The ratio of correctly predicted pixels (True Positives + True Negatives) to the total number of pixels: Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
Jaccard Index (IoU)	Intersection over Union (IoU), the ratio of the intersection of the predicted and ground truth masks to their union: Jaccard = $\frac{TP}{TP+FP+FN}$ (only binary classes)
ROC AUC	Area Under the Receiver Operating Characteristic Curve, which measures the ability of the model to distinguish between classes. AUC is the probability that a randomly chosen positive sample ranks higher than a randomly chosen negative sample.
F1 Score	The harmonic mean of precision and recall, providing a balance between the two: F1 = $\frac{2\cdot\text{Precision}\cdot\text{Recall}}{\text{Precision}+\text{Recall}}$
Precision	The proportion of true positives (TP) among all pixels predicted as positive: Precision = $\frac{TP}{TP+FP}$
Recall	The proportion of true positives (TP) among all actual positive pixels: Recall = $\frac{TP}{TP+FN}$
Average Precision (AP)	The average of precision values across all recall thresholds, often computed from the precision-recall curve.
True Negative (TN)	The number of pixels correctly classified as background (negative class).
False Positive (FP)	The number of pixels incorrectly classified as foreground (positive class).
False Negative (FN)	The number of pixels incorrectly classified as background (negative class).
True Positive (TP)	The number of pixels correctly classified as foreground (positive class).

Table 4.: Definitions of Metrics Used in Binary Mask Segmentation

B. Feature selection and latent representation models

Method Type	Advantages	Disadvantages	Variants
SUPERVISED LEARNING *Filtered*			
	+Model-independent +Very Fast +Generalization	-No interaction with the classifier	Variance Threshold σ^2 Chi-square χ^2 (Categorical) ANOVA (Continuous)
Embedded			
	+Integrated into the model +Less slow than wrappers +Captures dependencies between features	-Depends on model selection	Trees, Elastic-net (Lasso and/or Ridge), LDA, LQA, etc...
Wrapper			
	+Interacts with model +Captures dependencies between features	-Very slow -Risk of overfitting -Model-dependent	Recursive elimination (Forward, backward, or random)
UNSUPERVISED LEARNING *Correlations*			
	+Model-independent	-No target var.	Correlation

Table 5.: Feature Selection Techniques (tabulated data) [p. 17] [BSA15]

Method	Acronym	Data Type
Latent Dirichlet Allocation	LDA	Text data
Autoencoders	AE	Numerical and image data
Variational Autoencoders	VAE	Numerical and image data
Generative Adversarial Networks	GAN	Image and text data
Word2Vec	W2V	Text data
Transformer Models	TM	Text and sequential data

Table 6.: Summary of Latent Representation Methods

C. Image modification in Fourier space

The multiscale character of the Curvelets is demonstrated by examining Figs. 2-5.12 in Fourier space. Note how Fourier Space is modified in Fig.3 to remove TAE antenna signal. All images are in dB scale. The directions filtered can be appreciated.

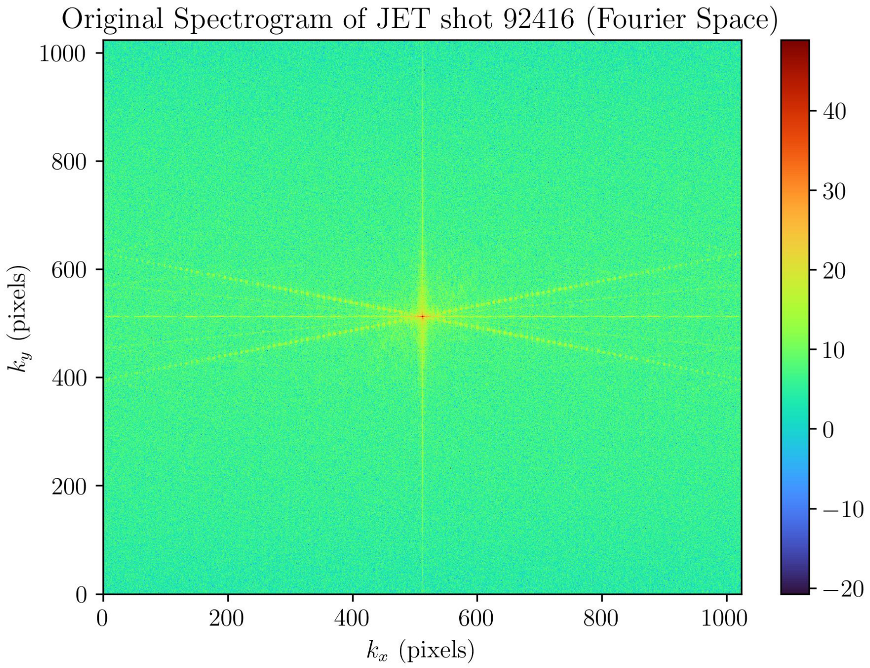


Figure 1.: FFT2D space of original spectrogram

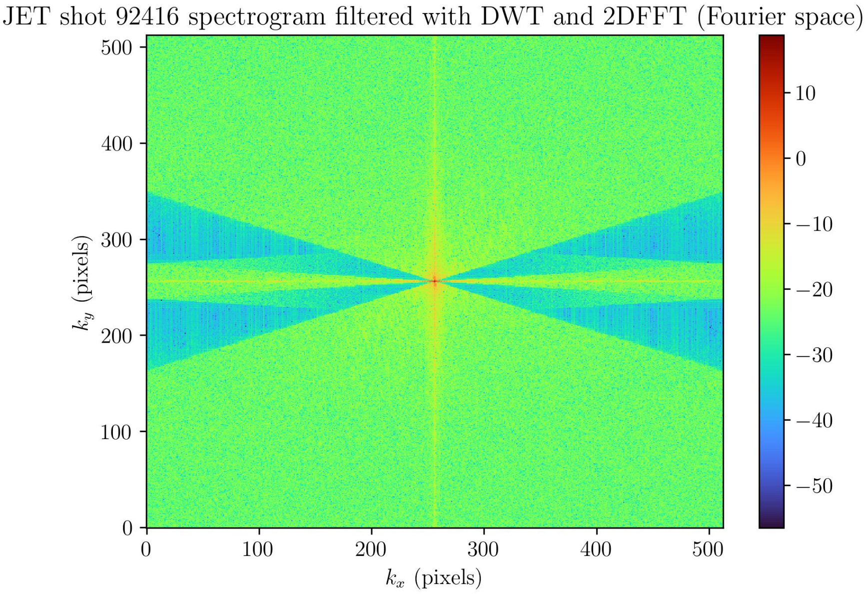


Figure 2.: FFT2D space of processed spectrogram using a radial mask in FFT2D

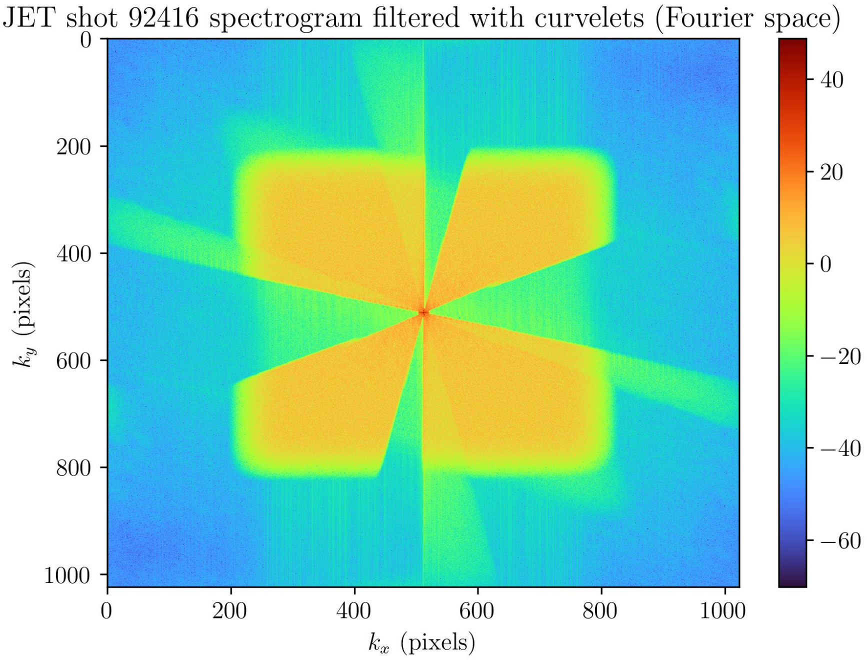


Figure 3.: FFT2D of processed spectrogram using curvelets

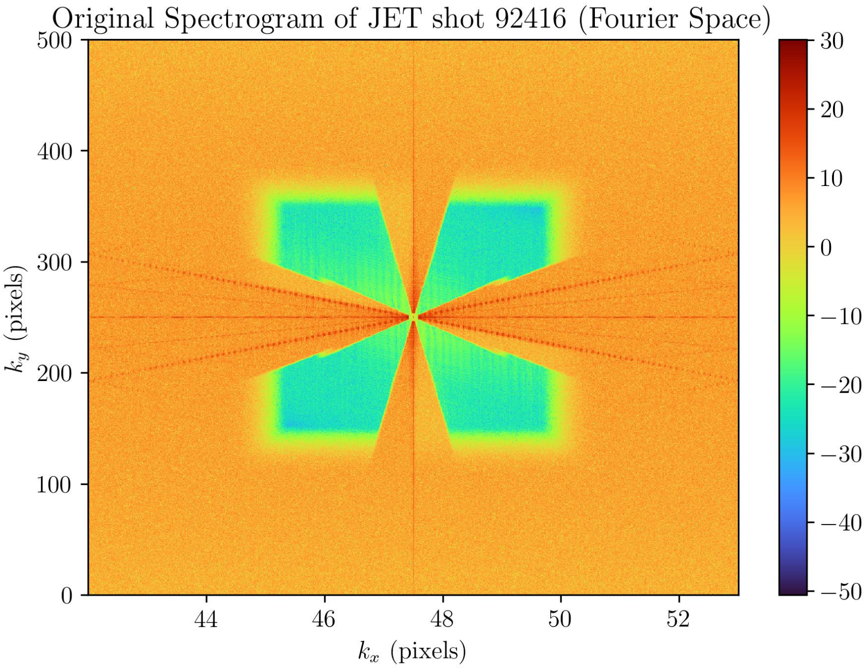


Figure 4.: Noise extracted using curvelets (Fourier space)

D. Evaluation metrics of CV algorithms with error bars

Metric	Bustos et al.	SAM	**Detectron 2**	YOLOv8	Detectron 2 (No DA)	YOLOv8 (No DA)	CV Pipeline
accuracy	0.991	0.988 ± 0.008	0.988 ± 0.007	0.988 ± 0.007	0.988 ± 0.007	0.988 ± 0.008	0.987 ± 0.01
jaccard	0.427	0.486 ± 0.158	**0.498** ± **0.128**	0.478 ± 0.132	0.472 ± 0.146	0.496 ± 0.134	0.238 ± 0.146
roc.auc	0.599	0.857 ± 0.07	0.825 ± 0.092	0.833 ± 0.084	0.833 ± 0.103	0.838 ± 0.078	0.876 ± 0.211
f1	(N/A)	0.637 ± 0.164	**0.654** ± **0.123**	0.635 ± 0.13	0.626 ± 0.150	0.651 ± 0.130	0.362 ± 0.189
precision	(N/A)	0.631 ± 0.23	0.693 ± 0.157	0.652 ± 0.176	0.629 ± 0.157	0.662 ± 0.183	0.262 ± 0.182
recall	(N/A)	0.721 ± 0.141	0.655 ± 0.165	0.673 ± 0.171	0.673 ± 0.207	0.682 ± 0.157	0.811 ± 0.23
tn	0.996	0.993 ± 0.016	0.946 ± 0.120	0.924 ± 0.113	0.992 ± 0.014	0.993 ± 0.014	0.893 ± 0.125
fp	0.004	0.007 ± 0.005	0.054 ± 0.037	0.076 ± 0.065	0.008 ± 0.001	0.006 ± 0.005	0.107 ± 0.088
fn	0.421	0.308 ± 0.006	0.342 ± 0.048	0.330 ± 0.047	0.005 ± 0.004	0.323 ± 0.297	0.213 ± 0.481
tp	0.579	0.692 ± 0.009	0.658 ± 0.069	0.670 ± 0.096	0.995 ± 0.007	0.676 ± 0.456	0.787 ± 0.970

Table 7.: Comparison of SAM, Detectron 2, YOLOv8 on the test set

E. Bayesian optimization of hyperparameters for SAM

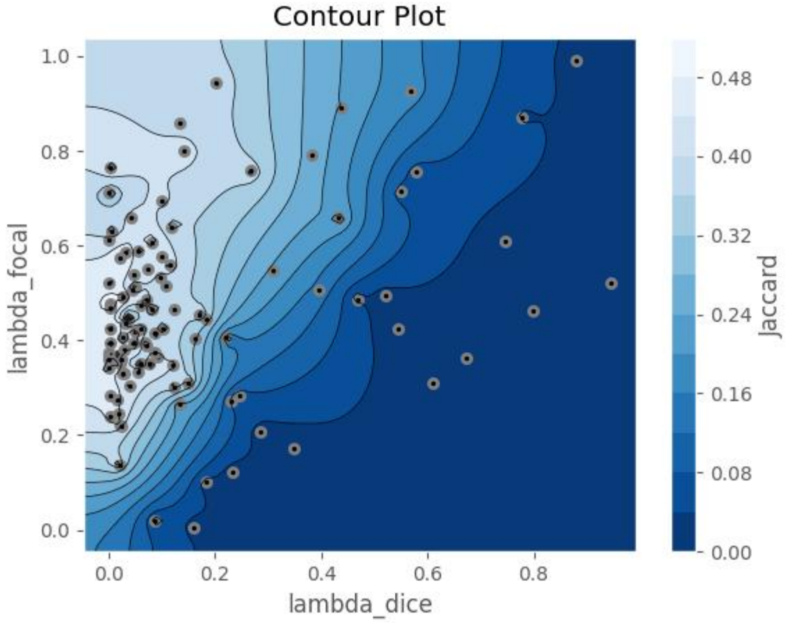


Figure 5.: Contour plot of Jaccard index for SAM as a function of the parameters of loss function, which is a weighted sum of DICE loss [Sud+17] and FOCAL loss [Lin+18]. The results shows that FOCAL loss is more important than DICE loss.

AMDG