University of Science and Technology of Hanoi

Doctoral School

Deep Learning Methods for Image Analysis and Understanding: Application to Biomedicine

PhD Student: Do Oanh Cuong

Student Code: D19.ICT.001

Supervised by

Supervisor: Assoc. Prof. Tran Giang Son

Co-Supervisor: Assoc. Prof. Luong Chi Mai

December, 2025

DECLARATION

I hereby declare that the thesis entitled *Deep Learning Methods for Image Analysis and Understanding: Application to Biomedicine* submitted by me, for the award of the degree of *Doctor of Information and Communication Technology* from the University of Science and Technology of Hanoi (USTH) is a record of bonafide work carried out by me under the supervision of:

1) Assoc. Prof. Tran Giang Son
2) Assoc. Prof. Luong Chi Mai

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Hanoi - Vietnam

Date: December, 2025 Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled *Deep Learning Methods for Image Analysis and Understanding: Application to Biomedicine* submitted by Mr. Do Oanh Cuong, Department of Information and Communication Technology (ICT), University of Science and Technology of Hanoi (USTH), is a record of bonafide work carried out by him/her under my supervision, as per the USTH code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and, in my opinion, meets the necessary standards for submission.

Place: Hanoi - Vietnam

Date:

Signature of the supervisors

Assoc. Prof. Tran Giang Son **Assoc. Prof. Luong Chi Mai**

ABSTRACT

Computational medical imaging is a cornerstone of precision medicine, offering non-invasive insights for detecting and monitoring pathology. This dissertation investigates two complementary domains within this field: the low-level synthesis of multi-modal data via image fusion and the high-level semantic interpretation of disease patterns via automated classification.

The first research thrust focuses on the integration of Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET). The primary challenge in this domain is merging the anatomical precision of MRI with the metabolic sensitivity of PET without introducing spectral distortion or contrast degradation. We introduce a novel hybrid architecture that decouples the fusion process into frequency-specific streams. A physics-inspired metaheuristic, the Equilibrium Optimization Algorithm (EOA), is employed to adaptively weight the base structural layers, ensuring optimal luminance retention. Simultaneously, a domain-adapted VGG19 network manages the high-frequency detail synthesis, preserving fine texture and edges. Benchmarking on standard neuroimaging datasets confirms the efficacy of this approach, with the proposed EOA-VGG19 framework achieving a $Q^{AB/F}$ score of 0.8065, outperforming several contemporary fusion algorithms in both visual fidelity and quantitative metrics.

The second research thrust addresses the need for rapid, resource-efficient screening tools, exemplified by the challenge of COVID-19 detection from chest X-ray (CXR) radiography. We propose a robust classification methodology that harmonizes the inductive biases of Convolutional Neural Networks (VGG19) and Vision Transformers (Swin Transformer). By implementing a prediction-level ensemble strategy, we achieve a diagnostic accuracy of 99.32% on the COVID-19 Ra-

diography Database, demonstrating a 50% reduction in predictive variance compared to single-architecture baselines. Furthermore, to facilitate deployment in restricted computing environments, we explore "Model Soups"—a technique for weight-averaging fine-tuned checkpoints—yielding a highly generalizable inference model without the computational cost of runtime ensembles.

Collectively, these contributions provide a unified perspective on enhancing medical image utility, advancing reliability in both pixel-level data fusion and patient-level diagnostic support.

Keywords: Multi-modal Fusion, Neuroimaging, MRI-PET Synthesis, Deep Feature Extraction, Equilibrium Optimization, Automated Radiographic Screening, Ensemble Learning.

ACKNOWLEDGEMENT

With the utmost sincerity and profound gratitude, I wish to express my heartfelt appreciation to my supervisors, Assoc. Prof. Tran Giang Son and Assoc. Prof. Luong Chi Mai, from the Department of Information and Communication Technology (ICT) at the University of Science and Technology of Hanoi (USTH). Their unwavering motivation, continuous encouragement, and invaluable guidance were instrumental in the successful completion of this research.

I would also like to extend my deepest gratitude to the overall management of the University of Science and Technology of Hanoi (USTH) for their steadfast support and unwavering commitment to academic excellence. The leadership's vision, as well as the resources and support provided by the university, have been crucial in shaping my academic and research journey.

Furthermore, I am profoundly grateful to the Department of Information and Communication Technology (ICT) for fostering a dynamic and enriching academic environment. The collaborative spirit and dedication to excellence within the ICT department have significantly contributed to the success of my research endeavors.

I recognize the pivotal role played by the management of USTH and the ICT department in providing the necessary infrastructure and resources, which have greatly enriched my academic experience and facilitated my intellectual growth and research exploration.

I would also like to express my deepest gratitude to my parents for their unwavering support, sacrifices, and encouragement throughout my research journey.

Finally, I extend my heartfelt appreciation to my wife, Nguyen Thi Anh

Phuong, and my sons, Do Oanh Viet Bao and Do Oanh Phuc An, for their constant encouragement, moral support, patience, and understanding during this academic pursuit.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF TERMS AND ABBREVIATIONS

IF Image Fusion

FT Fourier Transform

IFT Inverse Fourier Transform

PCA Principal Component Analysis

IHS Intensity Hue Saturation

MSD Multi-scale Decomposition

AHE Adaptive Histogram Equalization

CLAHE Contrast Limited Adaptive Histogram Equalization

CNN Convolutional Neural Network

CPU Central Processing Unit

EOA Equilibrium Optimization Algorithm

FFT Fast Fourier Transform

HE Histogram Equalization

MRI Magnetic Resonance Imaging

PET Positron Emission Tomography

TSID Two-scale Image Decomposition

TWBA The Whole Brain Atlas

Qabf Quality Assessment of Fused Images

CHAPTER 1

Introduction

1.1 Context and Motivation

Clinical decision-making increasingly relies on computational analysis of medical images. However, complementary diagnostic evidence is often distributed across modalities (e.g., anatomical versus functional imaging), and rapid triage tasks require accurate yet efficient automated predictors. This thesis investigates two complementary directions: multi-modal image fusion for information enhancement and deep-learning-based classification for screening under deployment constraints.

Medical image fusion is a pivotal area in computational radiology. It entails increasing the information density of diagnostic imagery by merging multi-source data (e.g., MRI, CT, PET) into a single, unified representation. In the scope of this thesis, we define "multi-modal" specifically as the integration of distinct physical imaging principles—ranging from anatomical mapping to metabolic functional tracking—rather than the broader AI definition involving text or audio. The resulting composite image empowers clinicians and radiologists to formulate more precise diagnoses and treatment plans, minimizing ambiguity [9].

In parallel, the global urgency of infectious disease management, exemplified by the COVID-19 crisis, has underscored the necessity for rapid, automated triage systems. Chest X-ray (CXR) imaging, favored for its ubiquity and low recurrent cost, remains the first line of defense. However, the visual subtlety of early viral consolidation combined with radiologist fatigue creates a screening bottleneck. While deep learning offers a robust automation pathway, deploying state-of-the-art ensembles in resource-limited clinical edge devices remains technically challenging. This creates a distinct need for architectures that balance high-sensitivity detection with computational leanness.

Standard clinical workflows employ a diverse array of imaging physics, including X-ray radiography, computed tomography (CT), positron emission tomography (PET), magnetic resonance imaging (MRI), and single-photon emission computed tomography (SPECT). Fundamentally, each modality captures a unique

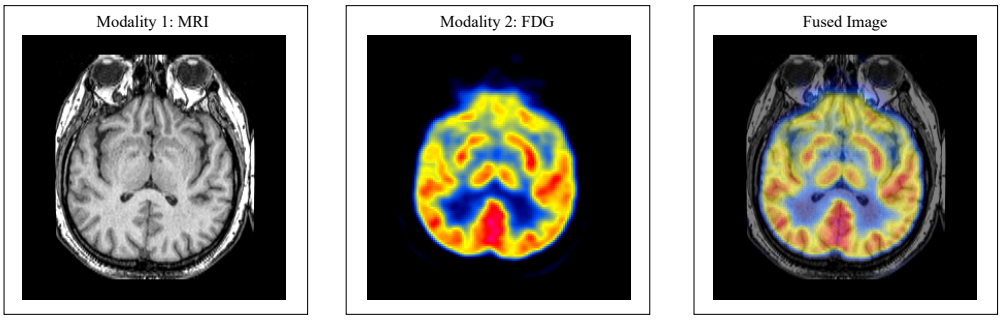

Fig. 1.1 Examples of image fusion between MRI image (left) and PET image (middle) to create a composite image (right).

slice of the patient's biological reality. For instance, MRI excels at resolving soft tissue contrast and anatomical boundaries, whereas PET provides a heatmap of metabolic activity. In isolation, a single modality rarely offers a complete picture for complex pathologies. While radiologists traditionally perform "mental fusion" by viewing scans side-by-side, this cognitive task is prone to spatial misalignment and fatigue. Automated algorithmic fusion resolves this by geometrically registering and synthesizing these inputs into a coherent whole. This duality serves two ends: enhancing direct visual interpretation for the physician and providing a richer multi-channel tensor for downstream AI analysis. Consequently, fusion technology is a critical enabler for holistic disease assessment [10, 11]. Common fusion pairings include MRI-PET, MRI-SPECT, CT-PET, and CT-SPECT.

This thesis specifically targets the MRI-PET fusion domain, one of the most clinically relevant multimodal pairings. The standard paradigm involves merging high-resolution grayscale MRI with color-coded PET functional maps. Figure 1.1 demonstrates this synthesis, where the output retains the structural clarity of the MRI while overlaying the metabolic "hotspots" from the PET, offering a comprehensive view of brain pathology.

Magnetic Resonance Imaging (MRI) acts as a high-resolution window into the body's anatomical structure. By leveraging strong magnetic fields and radio waves, MRI provides exceptional soft-tissue contrast without ionizing radiation. In neurological diagnosis, different pulse sequences are utilized to visualization specific tissue properties. T1-weighted images typically offer optimal resolution for analyzing brain anatomy, clearly distinguishing between gray and white matter boundaries essential for structural assessment. Conversely, T2-weighted sequences are highly sensitive to fluid content, making them indispensable for identifying pathological conditions such as edema, inflammation, and lesions where fluid accumulation occurs.

Specifically in the context of this research, MRI serves as the structural anchor for image fusion, ensuring that the merged output retains the precise anatomi-

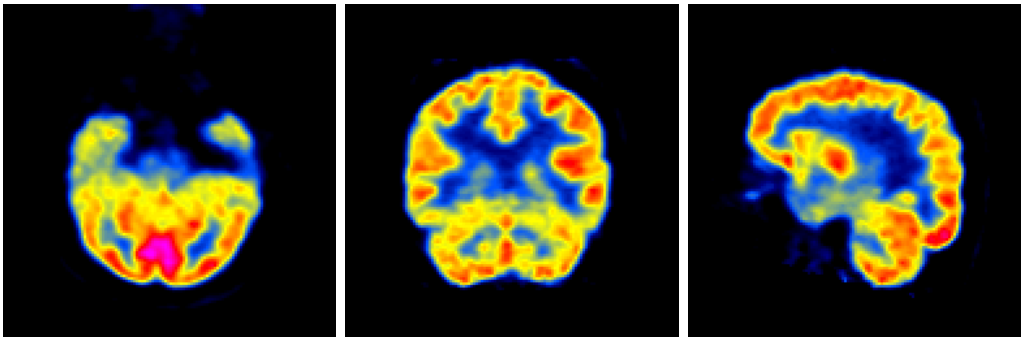

Fig. 1.2 Example of PET images extracted from the slice #50 of the brain hemispheric at the planes axial (left), coronal (middle) and sagittal (right) [1].

cal coordinates necessary for localizing abnormalities found in functional scans. Figure 1.1 includes a representative MRI input (left), illustrating the structural modality used throughout this thesis.

To facilitate consistent evaluation, this thesis utilizes standard open-access MRI datasets such as the Whole Brain Atlas. These validated repositories provide a robust benchmark for assessing how well computer-aided diagnosis systems can interpret complex anatomical signals [12].

Positron Emission Tomography (PET) serves as a functional counterpart to MRI by visualizing biological processes at the molecular level. Instead of imaging structure, PET detects the radiation emitted by a radiotracer—most commonly Fluorodeoxyglucose (FDG)—injected into the body. Since cancer cells and active brain disorders often exhibit hypermetabolism, they absorb glucose (and thus the tracer) at higher rates than normal tissue. This allows PET to highlight "hot spots" of disease activity that may not yet have caused visible structural changes on an MRI or CT scan.

While various radioisotopes exist (e.g., Carbon-11, Nitrogen-13), FDG remains the standard for oncological and neurological screening due to its effectiveness in mapping glucose metabolism. However, PET images suffer from low spatial resolution. Therefore, the integration of PET's functional sensitivity with MRI's anatomical precision is critical. This fusion allows clinicians to not only detect *what* is malfunctioning (via PET) but to determine exactly *where* it is located (via MRI) with high confidence.

PET imaging is widely used in oncology, neurology, and cardiology. It provides valuable information about the metabolic activity of tissues and organs, helping clinicians diagnose and monitor various diseases. Figure 1.2 shows an example of a PET slice (#50) of the brain hemispheric at the planes axial, coronal, and sagittal [1].

There exists several open datasets for PET images. These publicly available datasets usually consist of annotated PET scans from a range of patient popula-

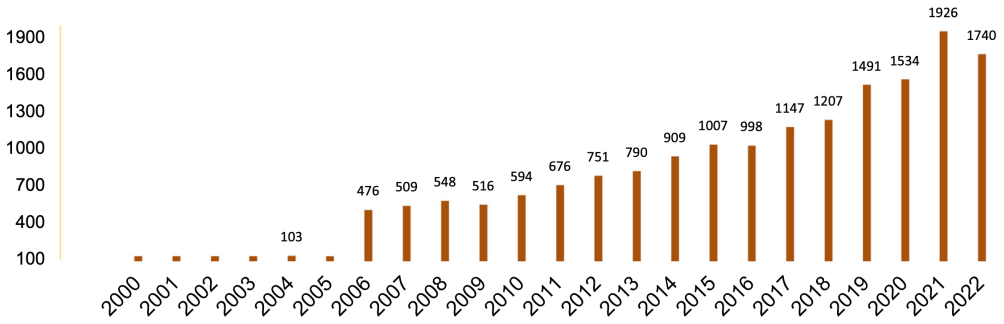

Fig. 1.3 Statistics of number of publications in medical image fusion [2].

tions with a variety of pathologies, including cancer, neurological conditions, and cardiovascular ailments. Two notable examples are the Cancer Imaging Archive (TCIA[1]) and the Alzheimer's Disease Neuroimaging Initiative (ADNI[2]), which offer large PET picture archives accompanied by clinical metadata, making them valuable resources for machine learning research and medical image analysis.

In recent years, the synthesis of multi-modal diagnostic data has garnered substantial research attention. Khan et al., 2023 [2] conducted a bibliometric analysis tracking publications in the Web of Science (WoS) database (see figure 1.3). The data reveals an accelerating trend, with a pronounced increase in both methodological innovation and clinical validation studies from 2020 onward. This momentum has intensified through 2024 and 2025, largely driven by the adoption of Transformer-based architectures and generative diffusion models. Contemporary research increasingly favors hybrid frameworks that synergize the localized feature extraction of CNNs with the global attention mechanisms of Transformers, alongside diffusion-based methods for superior texture fidelity. Consequently, developing efficient fusion methodologies that leverage these advancements while maintaining clinical deployability remains a critical research objective.

To provide a more up-to-date perspective on the field's trajectory, Figure 1.4 illustrates the temporal evolution of multimodal medical image fusion (MMIF) research productivity indexed in Web of Science (WOS) from 2014 to 2024. The chart displays a steady exponential growth pattern in the number of MMIF-related publications over this decade-long period. During the initial phase (2014–2017), publication volume remained relatively modest, indicating nascent research interest in the field. However, from 2018 onwards, a pronounced acceleration in publication rate is evident, with significant surges observed particularly around 2021–2023. This rapid proliferation of publications reflects the convergence of several factors: (1) the emergence of deep learning and transformer-based architectures for image fusion, (2) increased computational accessibility via GPU

[1] https://www.cancerimagingarchive.net/
[2] https://adni.loni.usc.edu/

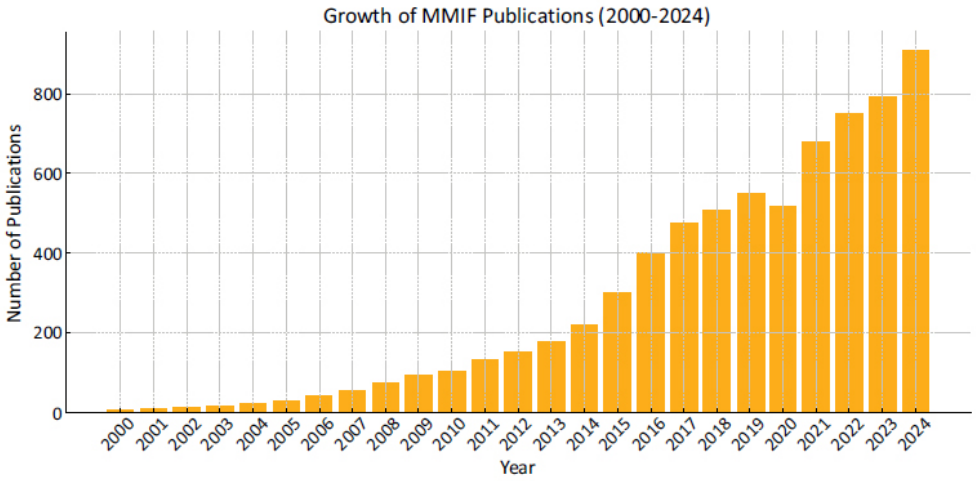

Fig. 1.4 Growth of MMIF Research Publications (2014-2024) [3].

technology, (3) greater availability of multimodal medical image datasets, and (4) growing clinical recognition of fusion techniques' diagnostic utility. The cumulative trend strongly suggests that MMIF has transitioned from a specialized niche to a major research frontier within medical imaging and artificial intelligence for healthcare [3].

1.2 Thesis Objectives

The main goal of this thesis is to enhance the quality and efficiency of medical image analysis systems, addressing both image fusion and classification tasks. Specific goals include:

- The proposal of a new algorithm to fuse the base components based on the Equilibrium optimization algorithm.
- The proposal of a new algorithm to fuse the detail components based on deep learning and transfer learning.
- The development of a compact, high-performance framework for medical image classification (specifically COVID-19 screening) using complementary deep features and model combination strategies, including prediction-level ensembling for best performance and within-backbone model soups for single-model deployment when required.

1.3 Research Object and Methodology

The thesis focuses on two primary research objects:

1. **Medical Image Fusion:** The fusion of MRI and PET brain images to produce a unified composite image which provides more useful medical information for supporting doctors and radiologists in disease treatment and diagnosis.

2. **Medical Image Classification:** The automated screening of COVID-19 from Chest X-ray images to support rapid and accurate diagnosis in clinical settings.

To address these problems, the thesis draws on the following techniques:

- The image processing methods to decompose and perform image conversion.
- The optimization methods to fuse the base component layers.
- The deep learning and transfer learning techniques to fuse the detail component layers.
- Model combination techniques for classification, including prediction-level ensembling of complementary backbones (Swin Transformer and VGG19) and within-backbone model soups (weight averaging across compatible models) for deployment-efficient inference.

The research methodology used in the thesis contains: theoretical research and experimental research.

For theoretical research, the thesis studies the recent publications about medical image fusion and classification, identifying the current limitations such as detail loss in fusion or computational overhead in ensemble classification. Based on the stated limitations, new algorithms and methods are proposed.

For experimental research, the proposed methods are evaluated against representative baselines and recent approaches using standard quantitative metrics and qualitative assessment.

1.4 Thesis Contribution

The main contributions of this thesis can be divided into two major research directions:

Medical Image Fusion: We propose a novel method based on equilibrium optimization algorithm (EOA) and deep learning with transfer learning (VGG19) to enhance the quality of fused medical images of MRI and PET modalities. This work combines adaptive fusion rules for base components using EOA with learned fusion rules for detail components using transfer learning. The proposed approach

was published in the ISI Q1 journal (SCIE) as publication #3. Additionally, a comprehensive review of deep learning approaches for multimodal medical image fusion is presented in publication #5.

Medical Image Classification: We investigate the application of complementary deep features for medical image classification tasks. Specifically, we propose a framework that combines Swin Transformer and VGG19 via prediction-level ensembling to achieve strong performance and stability for COVID-19 screening from chest X-ray images (publication #6). For deployment scenarios requiring a single network, we further study model soups in the standard within-backbone setting (e.g., averaging multiple fine-tuned Swin models) to obtain single-model inference. The thesis also explores quantum-classical hybrid neural networks with Lion optimizer for COVID-19 classification (publication #2) and deep learning methods for pulmonary nodule detection in CT scans (publication #1).

1.5 Thesis Outline

The rest of the thesis is structured as follows: Chapter 2 presents a comprehensive literature review on deep learning approaches for medical image fusion and analysis. Chapter 3 provides the theoretical background on image processing, optimization algorithms, and deep learning architectures used throughout the thesis. Chapter 4 presents the proposed medical image fusion method using equilibrium optimization algorithm and transfer learning with VGG19, including evaluation and comparison results on the Whole Brain Atlas[3] dataset from Harvard University. Chapter 5 introduces the medical image classification framework using complementary deep features through model soups, demonstrating its application to COVID-19 screening from chest X-ray images. Chapter 6 concludes the thesis and discusses potential future research directions.

[3] https://www.med.harvard.edu/AANLIB/

CHAPTER 2

Literature Review

This chapter surveys medical image fusion and analysis methods, with emphasis on modern deep learning. Fusion integrates complementary modalities (e.g., CT, MRI, PET, SPECT) to produce a single image that retains more clinically relevant information than any one input alone. Classical approaches are typically grouped into spatial-domain, transform/frequency-domain, and learning-based methods (Figure 2.1). Recent deep learning advances have shifted the field toward data-driven fusion rules and hierarchical feature modeling.

We first cover traditional spatial and transform techniques, then move to deep learning families (CNNs, GANs, Transformers, and hybrids). We close with evaluation metrics, clinical applications, and deployment challenges.

2.1 Spatial Domain Methods

In essence, the spatial domain approach to image fusion directly manipulates the individual pixel values or small patches of the original images to achieve a combined result. Often, techniques at the pixel level will first create weight maps from the input images. These maps are then used to combine the images through weighted addition. This straightforward strategy includes methods like the Mini-

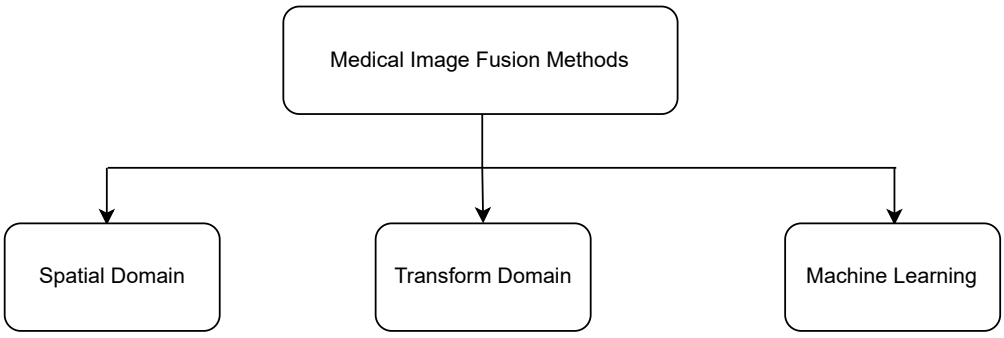

Fig. 2.1 Three types of medical image fusion methods.

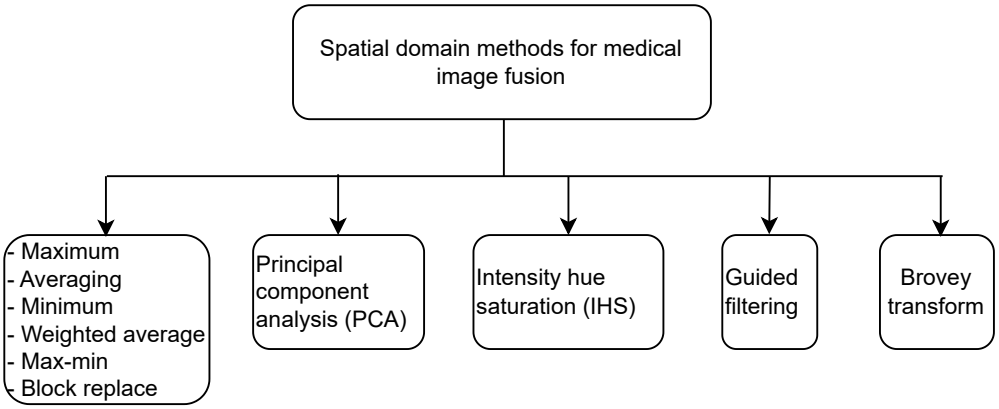

Fig. 2.2 Spatial domain methods for medical image fusion [4].

mum Pixel Value, Maximum Pixel Value Technique, Averaging method, and Simple Block Replace algorithm. A summary of spatial domain methods for medical image fusion is presented in the figure 2.2.

Maximum pixel value method. It generates an output image by taking the maximum intensity pixel from each input image [13]. Its simplicity makes it easy to adopt, and it aims for a high-contrast result by favoring brighter pixels and discarding those with low intensity, which are often susceptible to distortions. Despite these advantages, the method can have unintended consequences, such as reduced overall contrast and the injection of distortion into the final picture.

$$F(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{N} max(I_1(i,j), I_2(i,j)) \tag{2.1}$$

where I_1 and I_2 are input images, M, N are width and height of the input images, F is the fused image.

Minimum pixel value method. The process involves extracting the minimum intensity value for each pixel across a series of photographs to construct a new output image [14]. This technique yields improved results when the source images have pronounced dark shades. However, a drawback is that the resulting composite image often suffers from poor contrast and a lack of sharpness.

$$F(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{N} min(I_1(i,j), I_2(i,j)) \tag{2.2}$$

where I_1 and I_2 are input images, M, N are width and height of the input images, F is the fused image.

Pixel Averaging. This elementary fusion strategy synthesizes a new image by computing the arithmetic mean of spatially corresponding pixels from the source inputs [15]. While this approach is structurally simple and efficient for images with

identical exposure and sensor characteristics, it fundamentally acts as a low-pass filter. Consequently, it often dilutes high-frequency details, leading to reduced sharpness and contrast in the final composite.

$$F(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{I_1(i,j) + I_2(i,j)}{2} \tag{2.3}$$

where I_1 and I_2 denote the source image matrices, with dimensions $M \times N$, and F represents the resulting fused output.

Max-Min selection. This technique relies on extremum filtering, where the fusion rule selects either the maximum or minimum pixel intensity at each coordinate to construct the output [16]. The core premise is to retain the strongest signal features. Although computationally lightweight, this separate processing of intensity extremes often introduces artificial discontinuities or "blocking artifacts," resulting in unnatural transitions between regions and a degradation of edge coherence.

$$F(i,j) = \sum_{i=1}^{M} \sum_{j=1}^{N} max(I_1(i,j), I_2(i,j)) - min(I_1(i,j), I_2(i,j)) \tag{2.4}$$

where I_1, I_2 are the inputs and F is the fused result.

Weighted Averaging. An evolution of the simple mean, this method assigns scalar coefficients to each source image before summation [17]. By tuning these weights (W_1, W_2), one can prioritize a specific modality. While this offers some control over noise suppression and artifact reduction, it typically yields a result with "washed out" contrast. The linear superposition tends to average not just the signal but also the noise, potentially lowering the effective Signal-to-Noise Ratio (SNR) compared to more adaptive methods.

$$F(i,j) = \sum_{i=1}^{M} \frac{W_1 * I_1(i,j) + W_2 * I_2(i,j)}{W_1 + W_2} \tag{2.5}$$

where W_1 and W_2 are the specific importance weights assigned to inputs I_1 and I_2.

Principal Component Analysis (PCA). PCA is a dimensionality reduction technique rooted in statistical signal processing. In the context of image fusion, PCA is employed to transform correlated image bands into a set of uncorrelated variables called principal components based on their covariance structure [18].

The fusion process typically involves extracting the first principal component, which captures the variance (information) of the dataset. The eigenvectors of the covariance matrix define the optimal projection axes. By projecting the source

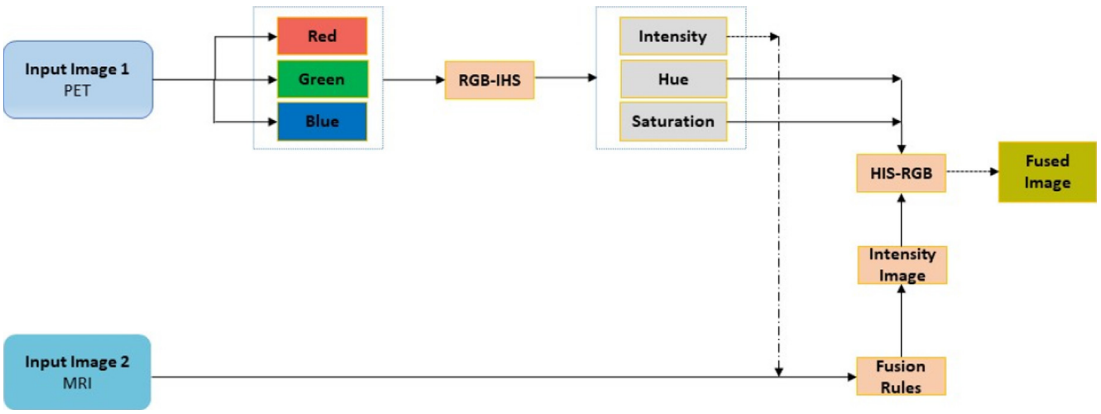

Fig. 2.3 Workflow of the IHS fusion technique [2].

images onto these axes, the method identifies the dominant structural patterns common to both inputs.

While the eigenvector orientation is invariant, the projection magnitude determines the contribution of each source. The First Principal Component (PC_1) aligns with the direction of maximal variance, effectively capturing the bulk of the image energy. Subsequent components (PC_2, etc.) capture residual variance orthogonal to the first.

$$F(I_1, I_2) = PCA_1 * I_1 + PCA_2 * I_2 \tag{2.6}$$

where PCA_1, PCA_2 denote the projection coefficients derived from the source images.

PCA is favored for its computational speed and ability to preserve spatial structure without complex filtering. However, a known limitation is "spectral distortion"—since the projection is purely statistical and agnostic to physical color properties, the fused image may exhibit unnatural color shifts.

Intensity-Hue-Saturation (IHS) Transform. This method is a component substitution technique widely used for pansharpening and multi-modal fusion. It decouples spatial information (Intensity, I) from spectral information (Hue, H and Saturation, S). The fusion pipeline converts the color image (e.g., PET) from RGB to the IHS space. The Intensity component (I), which correlates with spatial detail, is then manipulated—often replaced or blended with the high-resolution structural image (e.g., MRI). Finally, the inverse IHS transform reconstructs the RGB image [19]. Figure 2.3 illustrates this workflow.

$$\begin{bmatrix} I \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} 0.577 & 0.577 & 0.577 \\ -0.408 & -0.408 & 0.816 \\ -0.707 & 0.707 & 1.703 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.7}$$

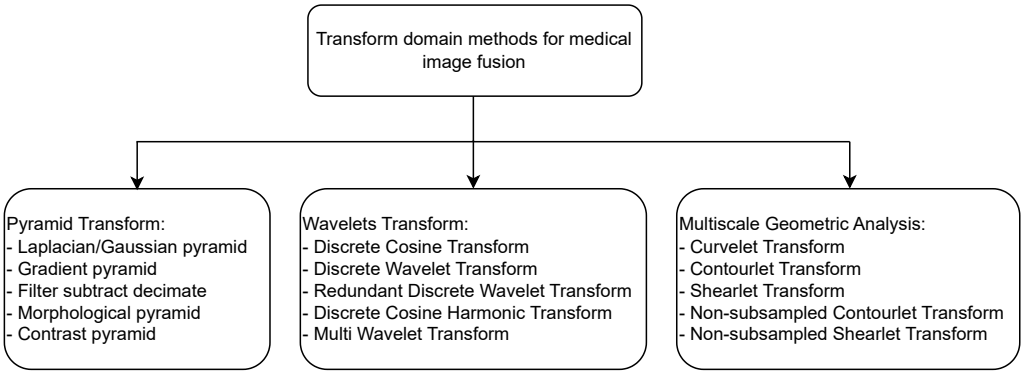

Fig. 2.4 Transform domain methods for medical image fusion [4].

$$H = \tan^{-1}(\frac{t_2}{t_1}), S = \sqrt{t_1^2 + t_2^2} \tag{2.8}$$

where t_1, t_2 are intermediate Cartesian variables used to compute polar H and S values.

The IHS method is celebrated for its ability to sharpen spatial features and its low computational cost. However, because it alters the intensity channel independently of chromaticity, it frequently introduces "spectral mismatch," leading to noticeable color distortion in the final output.

Guided Filtering. Guided filters function as edge-preserving smoothing operators [20]. Analagous to Bilateral Filters but more computationally efficient, they utilize a "guidance image" to direct the filtering process. This allows the filter to transfer the structural content of the guidance image onto the input. In fusion, this is typically used to decompose images into base (large-scale) and detail (fine-scale) layers, enabling multi-scale processing without the halo artifacts common in other decomposition schemes.

$$F(I_1, I_2) = GF_1 * I_1 + GF_2 * I_2 \tag{2.9}$$

where GF_k represents the guided filter kernel optimized for input I_k.

While guided filtering provides excellent edge preservation and lacks gradient reversal artifacts, it can sometimes cause local intensity inconsistencies if the guidance image has poor correlation with the target regions.

2.2 Transform Domain Methods

Transform-domain fusion typically follows a decompose–fuse–reconstruct pipeline. The inputs are first mapped into a transform space (e.g., DWT, pyramid, DFT/SWT, curvelet, contourlet), producing coefficients at multiple scales. Fusion rules are

then applied to these coefficients—often with spatial-context constraints—to combine salient information. An inverse transform returns the fused image to the spatial domain.

The main advantage of transform methods is their separation of information across frequency bands. Low-frequency components encode coarse structure and illumination, while high-frequency components capture edges and texture. This separation enables targeted fusion rules that preserve both global structure and fine detail. Multi-resolution transforms (DWT, Laplacian pyramid, contourlet, curvelet) are especially useful when sources contain information at different spatial scales, which is common in medical and remote-sensing imagery.

2.2.1 Pyramid Transform Methods

Gaussian/Laplacian Pyramid. The Gaussian Pyramid (GP) is a fundamental multi-resolution technique in image processing that constructs a sequence of images with progressively reduced resolutions. Each level in the pyramid is generated by applying a low-pass filter—typically a Gaussian kernel—to the image at the previous level, followed by sub-sampling, resulting in a blurred and downscaled version. This hierarchical structure effectively captures the image's large-scale structures while discarding fine details at higher levels. The Gaussian filter used in this process is symmetric and centered, mimicking the Gaussian probability distribution to ensure smooth averaging across neighboring pixels. Convolution between the image and the Gaussian kernel performs the smoothing operation, which reduces high-frequency noise and prepares the image for efficient down-sampling. The GP is widely used as a base for other pyramid-based fusion methods, including Laplacian pyramids and wavelet-based techniques, because of its ability to represent images at multiple scales. Laplacian pyramid is constructed by taking the difference between consecutive levels of the Gaussian pyramid, where each Gaussian level is up-sampled and subtracted from the previous level. This process produces a set of band-pass images that emphasize edge and texture information, effectively isolating the image details lost during down-sampling.

One early implementation by Wang et al. (2011) introduced a Laplacian Pyramid (LP) approach, utilizing a maximum region information rule for the highest decomposition level and a maximum region energy rule for other levels [21]. This method enhances sharp contrast features but struggles with low-contrast detail retention. To address this, Hang et al. (2013) applied a maximum likelihood strategy targeting low-frequency components and incorporated entropy-based measures to enhance high-frequency detail, though it still introduced blocking artifacts [22]. Xianghai et al. (2015) employed the lifting wavelet transform (LWT), selecting

fusion coefficients based on image covariance [23]. While effective for maintaining shift invariance and phase consistency, its scope is somewhat limited. Earlier work by Faranak et al. (2013) used 1D log-Gabor and Haar Wavelet (HW) transforms with a matching score-based fusion rule for combining infrared and visible images [24]; however, HW's limited smoothness remains a drawback.

Gradient Pyramid. Burt and Kokzynski (1993) introduced the Gradient Pyramid (GRP) method for fusing thermal and visual images, offering a distinct approach from the Gaussian and Laplacian pyramids [25]. While the Gaussian pyramid focuses on low-pass filtering and down-sampling to capture coarse image structures, and the Laplacian pyramid isolates detail information through difference operations, the GRP emphasizes gradient information by applying a gradient operator at each level of the Gaussian pyramid. This results in a hierarchical representation that highlights edge transitions and directional intensity changes across scales. In the fusion process, the GRP employs two types of fusion rules: pattern selection and averaging. The pattern selection rule is applied when the input images exhibit significant local differences, allowing the dominant structural feature to be retained. Conversely, in regions where the source images are similar, an averaging rule is used to blend the content. This method has shown improved performance over earlier approaches, such as Toet's, particularly in reducing noise and preserving low-contrast details. However, it has limitations in handling high-contrast features, which are often inadequately fused due to the smoothing effect of the averaging rule. Overall, the GRP method provides a more edge-sensitive representation, distinguishing it from the structure-preserving nature of the Gaussian and Laplacian pyramids.

Filter Subtract Decimate. This method is an alternative way to perform Laplacian pyramid. It represents a structured approach to multi-resolution image fusion, differing in both procedure and emphasis from Gaussian and Laplacian pyramid techniques [26]. In FSD, the initial step involves constructing a Gaussian pyramid (GP) through convolution of the source image with a Gaussian kernel, applied iteratively at each level. This process produces a set of progressively smoothed and down-sampled images that form the base layer representations. To obtain the Laplacian pyramid (LP), the FSD method performs a subtraction between consecutive Gaussian levels, followed by decimation, effectively isolating spatial band-pass components that capture localized detail information. These components represent fine structural variations and contribute significantly to scene interpretation. Unlike traditional LP methods that directly focus on detail extraction, FSD emphasizes the integration of both base and detail layers by generating modified pyramid structures tailored to the fusion objective. After decomposition, the detail layers are expanded via interpolation, and fusion is carried

out on the base layers using the adjusted LP representations. Finally, the fused image is reconstructed by applying an inverse transformation that combines the fused base and detail components. Compared to Gaussian and standard Laplacian pyramids, FSD introduces greater flexibility in controlling the fusion process at multiple levels, enabling improved preservation of structural and contextual information.

Contrast pyramid. Haiyan and Yanyan (2014) introduced a fusion approach that integrates the Contrast Pyramid (CP) with a teaching–learning-based optimization algorithm for combining infrared (IR) and visible (VIS) images [27] Subsequently, Hua Xu et al. (2016) proposed an alternative method employing the Contrast Pyramid Transform (CPT) alongside an entropy-based weighted fusion rule to enhance fusion performance [28]. The contrast pyramid differs from traditional Gaussian and Laplacian pyramid methods in its emphasis on local contrast rather than purely spatial frequency or intensity-based decomposition. CP only focuses on contrast features, therefore aims to highlight perceptually significant regions, particularly those with sharp changes in intensity, which are critical for human visual interpretation.

However, despite this targeted focus, the contrast pyramid shares a notable limitation with the Laplacian pyramid—specifically, its reduced effectiveness in capturing and integrating high-contrast features consistently during the fusion process. This limitation often leads to suboptimal preservation of salient details, particularly in heterogeneous source images. Therefore, the practical impact of contrast pyramid methods in image fusion remains constrained compared to more robust multi-scale decomposition techniques such as wavelet- or curvelet-based approaches.

2.2.2 Wavelet Transform Methods

Wavelet transform is a widely used multi-resolution technique in image fusion that provides both spatial and frequency localization. Unlike pyramid transforms, which rely on iterative blurring and down-sampling to create progressively coarser representations, the wavelet transform decomposes an image into sub-bands at multiple scales and orientations using filter banks. This enables a more precise separation of detail and approximation components across horizontal, vertical, and diagonal directions. As a result, wavelet-based fusion methods can preserve edges and fine textures more effectively than pyramid-based approaches. This advantage makes them well-suited for tasks requiring detailed structural information.

Discrete Cosine Transform (DCT). DCT-based fusion methods are known for their computational efficiency and fast execution [29]. However, their perfor-

mance heavily depends on the choice of block size, which can be difficult to optimize in practical scenarios. When source images are already captured or stored in a DCT-based format (such as JPEG), the use of DCT for image fusion can significantly reduce processing complexity. To accommodate various image structures, several DCT-based fusion strategies have been proposed using different block sizes [30]. Due to its high energy compaction property, where most significant information is concentrated in low-frequency components, DCT is effective in representing important image features.

Discrete Wavelet Transform (DWT). Estimating meaningful representations from non-stationary signals, such as images, poses a significant challenge in data analysis due to the variability of features across space and frequency. DCT method offers strong energy compaction and are effective for representing stationary or globally smooth signals, but lacks temporal or spatial localization. DWT, on the other hand, provides multi-resolution decomposition with both spatial and frequency localization, making it more suitable for capturing local features such as edges and textures. On the one hand, DCT analyzes the entire image in terms of fixed-frequency cosine basis functions. On the other hand, DWT breaks the image into approximation and detail coefficients across multiple scales using localized wavelet basis functions. In [31], a square nonlinear approach using Singular Value Decomposition (SVD) was introduced to improve image representation, and later studies, such as [32], combined DWT and SVD for enhanced fusion performance in monochromatic image blocks. This hybrid method selectively applies either DWT or SVD depending on local statistical properties. The resulting multimodal image fusion framework has demonstrated effectiveness in critical applications including surveillance, remote sensing, and medical diagnostics. The use of DWT allows detailed coefficient extraction, which is further refined with fusion strategies such as averaging, max-min selection, or weighted combinations to produce higher-quality fused images [33].

Redundant Discrete Wavelet Transform (RDWT). RDWT is designed to address the shift variance problem inherent in the standard Discrete Wavelet Transform (DWT). Unlike DWT, which involves down-sampling during decomposition and thus introduces sensitivity to small shifts in the input image, RDWT eliminates down-sampling and maintains translation invariance [34]. This property makes RDWT particularly suitable for applications requiring high spatial consistency, such as medical image fusion [35]. RDWT is also known as the undecimated DWT (UDWT) or the "à trous" algorithm, and it provides an over-complete representation by retaining the original image dimensions at each level of decomposition. In contrast, DWT reduces the size of the image data at each level, which can lead to information loss and alignment issues during fusion. Compared to the

Discrete Cosine Transform (DCT), which operates on fixed-size blocks and lacks spatial adaptability, RDWT allows better localization of features across scales while preserving structure due to its redundancy and absence of down-sampling. Yang et al. (2009) combined DWT with a contrast-based rule and incorporated RDWT to suppress noise and mitigate registration errors in fused images [36]. However, RDWT's redundancy also leads to increased computational and memory demands. To address these limitations, Ellmauthaler et al. (2013) proposed the Undecimated DWT (UDWT) as a refined implementation [37]. UDWT has been applied in several fusion tasks, including infrared and visible image fusion, although its high redundancy still poses challenges in terms of storage and processing efficiency [38, 39].

Discrete Cosine Harmonic Wavelet Transform (DCHWT). DCHWT extends the principles of the Discrete Cosine Transform (DCT) by integrating harmonic wavelet concepts into the transformation [40]. Unlike standard DCT, DCHWT introduces a grouping mechanism for DCT-derived subsets, which are processed using inverse DCT (IDCT) to yield discrete cosine harmonic wavelet coefficients (DCHWCs). These coefficients are then used to reconstruct the full DCT spectrum at the original sampling rate, enabling multiscale analysis similar to wavelet transforms while maintaining the real-valued nature of DCT. In fusion applications, DCHWT combines the strengths of both DCT and wavelet approaches, using texture-based metrics for low-frequency fusion and phase congruency for high-frequency fusion. It performs comparably with convolution-based fusion techniques and demonstrates improved performance over lifting-based methods [41].

Multi Wavelet Transform. Multi-wavelet transformation offers several key advantages that enhance its effectiveness in image fusion tasks. Unlike scalar wavelets, multi-wavelets employ vector-valued scaling and wavelet functions, enabling the simultaneous achievement of orthogonality, symmetry, compact support, and high vanishing moments [42, 43]. These properties contribute to more accurate signal representation, improved edge preservation, and reduced distortion in fused images. The ability to maintain symmetry and compactness ensures better reconstruction quality, while high vanishing moments allow for effective representation of smooth regions and detailed structures. These strengths have made multi-wavelets increasingly valuable in image processing applications. [41] further explored a variant approach using the Discrete Cosine Harmonic Wavelet Transform (DCHWT) combined with a weighted averaging fusion rule to reduce computational complexity and enhance visual quality. However, that method was prone to ringing artifacts, which can degrade the clarity of the fused output. Traditional wavelet transforms, though widely used, suffer from limitations such as

restricted directionality, lack of shift invariance, and increased computational cost. These drawbacks hinder their ability to fully capture and reconstruct visual information, particularly fine edge details. In contrast, multi-wavelet approaches offer a more robust framework for achieving high-quality image fusion.

2.2.3 Multiscale Geometric Analysis Methods

Curvelet transform. Originally developed for image denoising, the curvelet transform has proven to be valuable in image fusion [15]. It is constructed by applying the ridgelet transform to the linear components extracted from undecimated wavelet coefficients. The ridgelet transform, characterized by its directional sensitivity and ability to capture linear features, enables the curvelet transform to represent image structures such as curves and edges across multiple scales with high efficiency. This directional and multi-scale capability allows the curvelet transform to model piecewise smooth contours using a small number of coefficients [44]. As a result, geometric features are better isolated from background noise, which can be effectively suppressed by thresholding the curvelet coefficients prior to the fusion process. Unlike traditional separable transforms, the curvelet transform operates in a non-separable domain, enhancing its ability to preserve anisotropic features such as edges and boundaries during fusion.

Contourlet transform. Since 2006, the contourlet transform has been widely applied in various computer vision tasks, particularly in processing high-dimensional signals, due to its strong directional sensitivity and anisotropic representation capabilities [45, 46, 47]. The contourlet transform addresses key limitations of traditional wavelet methods by employing directionally sensitive basis functions that allow for more effective representation of images with smooth contours and edges. Unlike wavelets, which are limited in capturing geometric structures beyond point singularities, the contourlet transform offers a sparse representation for images with smooth regions interrupted by edges, providing near-optimal efficiency for piecewise smooth image content. This is achieved by combining a Laplacian pyramid (LP) for multi-scale decomposition with a directional filter bank (DFB), which further divides the image into directional sub-bands at each resolution level.

Shearlet transform. Shearlet transform was introduced as a novel multiscale geometric analysis tool designed to efficiently represent two-dimensional images [48, 49, 50]. The shearlet transform combines a mathematically framework with a compact structure, offering an ability to capture anisotropic features, such as edges and contours, more effectively than earlier approaches. While it shares a similar multi-scale, multi-directional structure with the contourlet transform, the shearlet transform exhibits superior directional sensitivity, particularly in detect-

ing features along various orientations.

Non-subsampled contourlet transform. The original contourlet transform served as an early framework for multi-resolution and multi-directional image fusion, offering advantages in edge preservation and smoothing processes [51]. However, it suffers from a lack of translation invariance and is prone to the pseudo-Gibbs phenomenon—an artifact that appears near singular points in the fused image—resulting in visual degradation and reduced suitability for high-precision computer vision tasks. To address these limitations, the Non-Subsampled Contourlet Transform (NSCT) was introduced as an enhancement based on contourlet theory [52]. Unlike the traditional contourlet transform, NSCT eliminates down-sampling during decomposition, thereby achieving full translation invariance and improved directional sensitivity. It employs a combination of non-subsampled pyramid decomposition for multi-scale analysis and non-subsampled directional filter banks for capturing features across various orientations. This framework allows for a more robust representation of both low-frequency (base) and high-frequency (detail) components of the input image. In the fusion process, NSCT decomposes the input images into sub-band components at multiple scales and directions using specialized filters. These sub-bands are then combined using suitable fusion rules, followed by the inverse NSCT to reconstruct the final fused image. Due to its ability to preserve structural integrity and suppress artifacts, NSCT has become a widely adopted method in the field of medical image fusion [53].

Non-subsampled shearlet transform. The shearlet transform is a key component of modern composite wavelet theory, integrating principles of classical geometry with multi-scale analysis to achieve efficient image representation [54]. It is particularly effective in capturing anisotropic features, providing an optimally sparse representation of images with isolated singularities, and achieving near-ideal nonlinear approximation performance. In the field of biomedical image processing, the non-subsampled shearlet transform (NSST) has become a widely adopted tool due to its ability to emphasize local image features with high accuracy [55]. NSST offers strong directional sensitivity and translation invariance without requiring down-sampling, making it well-suited for capturing edges and textures with minimal distortion. During NSST decomposition, high-magnitude transform coefficients correspond to regions with rich structural information, such as boundaries and fine textures. Because different sensors capture varying amounts and types of detail, effective image fusion using NSST must ensure that significant features from all input images are preserved. Additionally, the fused output should maintain visual coherence while minimizing the introduction of artifacts, thereby enhancing both the interpretability and quality of the final image.

2.3 Deep Learning Approaches for Medical Image Fusion

Having reviewed the traditional multiscale decomposition methods, we now turn to the paradigm shift introduced by Deep Learning (DL). While traditional algorithms rely on manually designed filters (e.g., wavelets, shearlets) and handcrafted fusion rules, DL approaches learn feature representations and fusion strategies directly from large-scale data. This transition addresses the limitations of manual feature engineering, enabling the capture of complex, non-linear relationships between modalities.

This section is aligned with (and extends) the taxonomy and practical findings summarized in our accepted concise review on deep learning for multimodal medical image fusion [56]. We first examine seminal architectures that established this field, before detailing specific methodological families.

2.3.1 Seminal Deep Learning Architectures

While early machine learning approaches utilized techniques like dictionary learning and sparse coding, the field has been revolutionized by deep neural networks. Here, we analyze several seminal models that have shaped the current landscape of medical image fusion: DenseFuse, FusionGAN, DDcGAN, and TransFuse.

DenseFuse (CNN-based): Li and Wu proposed DenseFuse [57], an infrared and visible image fusion method that has been widely adapted for medical applications. It employs a deep network architecture consisting of an encoder, a fusion layer, and a decoder. The encoder utilizes densely connected convolutional blocks (DenseNet) to preserve more information from the source images. The core advantage of DenseFuse is its ability to extract deep features while maintaining intermediate features through dense connections, which mitigates the vanishing gradient problem. However, the fusion strategy in the bottleneck layer is often handcrafted (e.g., addition or L1-norm), which may not be optimal for all scenarios.

FusionGAN (GAN-based): Ma et al. introduced FusionGAN [58], formulating image fusion as an adversarial game. The generator aims to produce a fused image that contains the radiation intensity of the infrared image (or functional information in medical contexts) and the texture details of the visible image (structural information). The discriminator tries to distinguish the fused image from the ground truth (or source images). FusionGAN demonstrated that adversarial training could enforce sharper textures without explicit gradient loss functions. However, standard GANs can be unstable to train and may suffer from mode collapse.

DDcGAN (Dual-Discriminator GAN): To address the limitation of single-

discriminator models which might bias the output towards one modality, Ma et al. proposed DDcGAN [59]. This architecture features a generator and two discriminators: one ensuring the fused image retains structure from the MRI/visible image, and the other ensuring it retains functional/intensity information from the PET/infrared image. This dual-adversarial mechanism provides a more balanced fusion, significantly improving the preservation of complementary information compared to FusionGAN.

TransFuse (Transformer-based): Recognizing the limitation of CNNs in modeling long-range dependencies, TransFuse [60] (and similar Transformer-based variants) integrates the self-attention mechanism of Transformers. It typically employs a hybrid architecture where a CNN branch captures local features and a Transformer branch captures global context. This approach is particularly beneficial in medical imaging, where relationships between distant anatomical structures can be diagnostically relevant. TransFuse has shown superior performance in maintaining global contrast and avoiding the local blurring artifacts common in pure CNN methods.

2.3.2 Taxonomy of Deep Learning Methods

Modern deep learning methods—such as convolutional neural networks (CNNs), generative adversarial networks (GANs), and transformers—have revolutionized medical image fusion by learning fusion rules directly from data. These approaches can capture both local details and global context, overcoming many limitations of traditional handcrafted methods.

We highlight representative architecture families, common enhancement modules, evaluation practices, and deployment considerations.

2.3.3 Fundamental Concepts

Medical image fusion can be performed at different levels: pixel level, feature level, or decision level. Each modality provides complementary information:

- **CT:** High spatial resolution for bone and dense structures; uses ionizing radiation.
- **MRI:** Superior soft-tissue contrast; multiple sequences (T1/T2/FLAIR) with complementary properties.
- **PET/SPECT:** Functional and metabolic imaging; lower spatial resolution but complementary to structural information.

Table 2.1 summarizes common fusion levels and their typical advantages and challenges.

Table 2.1 Fusion levels and typical characteristics.

Level	Advantages	Challenges
Pixel	Preserves fine details and contrasts	Sensitive to misregistration and noise
Feature	More robust; can encode semantic cues	Risk of information loss; depends on the feature extractor
Decision	Task-aligned and flexible	Loses low-level detail and may be modality/task specific

The goal of fusion is to preserve information from each modality while enhancing fine details and contrast, reducing artifacts, and maintaining computational efficiency. Before fusion, proper image registration and intensity normalization are essential preprocessing steps.

From a system perspective, recent surveys emphasize that fusion pipelines are sensitive to preprocessing choices (registration, normalization) and to dataset/domain shifts across scanners and sites; these factors strongly influence both quantitative metrics and clinical usability [61, 2, 62].

2.3.4 CNN-based Fusion Methods

CNNs capture local and mid-range patterns efficiently and are widely used for fusion tasks. Typical CNN-based designs employ encoder-decoder backbones with multi-scale features and skip or dense connections. Learned fusion strategies adaptively combine features rather than using simple max or average operations.

For example, Zhang et al. [63] propose IFCNN, a general image fusion framework built on a convolutional neural network. The method uses convolutional layers to extract salient features from input images, applies appropriate fusion rules (element-wise max, min, or mean) to combine features, and reconstructs the fused image through additional convolutional layers. Lightweight architectures with channel or spatial attention mechanisms often achieve a good balance between accuracy and efficiency.

2.3.5 GAN-based Fusion Methods

Generative Adversarial Networks treat fusion as an image synthesis task. Adversarial training, combined with content or structural losses, can significantly improve visual quality. Frequency-aware modules (such as wavelets) and multiple discriminators help preserve textures and edges. Unsupervised variants can learn effective fusion strategies without paired ground truth data.

Representative examples include wavelet-aware GAN designs (to encourage fidelity across frequency bands) and unsupervised GAN-based fusion when paired supervision is limited [64, 65].

2.3.6 Transformer-based and Hybrid Architectures

Transformers use self-attention mechanisms to model long-range dependencies and cross-modal correspondences. However, computing global attention for high-resolution medical images can be computationally expensive. To address this, many systems combine CNN front ends with transformer blocks, balancing local detail capture with global context modeling.

Hybrid designs merge CNN strengths in local detail processing with transformer capabilities in global structure understanding. Dual-branch and progressive fusion designs often achieve strong performance, though they can be more complex. Correlation-guided modules and memory-augmented architectures can further improve stability and cross-modal alignment.

Recent representative transformer and hybrid fusion models illustrate these trends, where CNN front-ends preserve fine details while attention blocks propagate global information and cross-modal correspondence [60, 66, 67].

Table 2.2 provides a compact, survey-aligned summary of the main deep learning families used for multimodal fusion.

Table 2.2 Representative deep learning architecture families for multimodal medical image fusion (survey-aligned).

Family	**Representative examples**	**Key idea**	**Typical trade-offs**
CNN	IFCNN [63], AMMNet [68]	End-to-end learned feature extraction and fusion weights	Efficient; strong local detail
GAN	MHW-GAN [64], MedFusionGAN [65]	Image synthesis with adversarial + content/structure objectives	Perceptual quality; training stability
Transformer	TUFusion [60]	Global dependencies via (cross-)attention	Compute/memory heavy
Hybrid	DFENet [66], DesTrans [67]	Combine CNN local detail with transformer global context	Strong balance; more complex

2.3.7 Attention Mechanisms and Feature Enhancement

Channel attention and spatial attention mechanisms reweight features to highlight important channels and spatial locations. Cross-modal attention adapts the balance between different modalities dynamically. These mechanisms are crucial for effective feature fusion and have become standard components in modern fusion architectures.

In practice, lightweight attention modules (e.g., CBAM-like blocks) are widely used because they improve saliency selection with limited additional compute, which matters for deployment on typical clinical hardware [69, 68].

2.3.8 Training Strategies and Evaluation

Training deep fusion models typically involves pixel-wise losses (L1/L2), structural similarity terms (SSIM), and task-specific objectives. To address limited labeled data, researchers employ data augmentation, synthetic pair generation, transfer learning, and self-supervised pretraining strategies.

Evaluation remains challenging due to the absence of fused ground truth. Common metrics include:

- **Information content:** Mutual Information (MI) and Entropy measure how much information is retained.
- **Structural quality:** SSIM and FSIM assess how well structure is preserved.
- **Signal quality:** PSNR reflects reconstruction fidelity.
- **Perceptual quality:** VIF estimates perceptual information fidelity.

However, no single metric fully reflects clinical value, and expert evaluation remains important for assessing diagnostic utility.

Table 2.3 Common quantitative metrics used in multimodal fusion evaluation.

Metric	**Interpretation**
MI, Entropy	Information retention/content (higher is better)
SSIM, FSIM	Structural/feature similarity (closer to 1 is better)
PSNR	Signal-to-noise ratio / reconstruction fidelity (higher is better)
VIF	Perceptual information fidelity (higher is better)

Table 2.4 Representative benchmark datasets commonly used for multimodal fusion.

Dataset	**Modalities**	**Notes**
BRATS	MRI (T1/T2/FLAIR)	Benchmark for glioblastoma segmentation; commonly repurposed for fusion despite lacking fused ground-truth pairs
TCIA	CT/MRI/PET	Diverse oncology collections; protocols vary across sites and studies
Whole Brain (research cohorts)	MRI/PET	Neuro datasets used to study structural–functional alignment

Table 2.5 Common qualitative criteria used by experts to judge fused images.

Criterion	**Description**
Contrast	Visibility of salient structures (e.g., lesions, boundaries)
Sharpness	Edge clarity and fine-detail preservation
Artifacts	Presence of halos, ghosting, ringing, or modality leakage
Consistency	Global structural coherence; absence of distortions

2.3.9 Empirical Findings and Common Ablation Trends

Across recent deep fusion studies and surveys, ablation analyses commonly report that: (i) frequency-aware modules and multi-branch discriminators (when using GAN objectives) help preserve textures and edges; (ii) transformer blocks improve global structural coherence; and (iii) edge-/gradient-aware paths reduce artifacts and sharpen boundaries [64, 67, 70, 71, 72].

Table 2.6 Examples of empirical findings reported in ablation studies (illustrative).

Model	Key component	Observed effect
MHW-GAN [64]	Wavelet/frequency cues + multi-discriminator	Better texture/structure; higher MI/SSIM
DesTrans [67]	Transformer blocks + dense encoders	Better global/local balance; sharper details
SS-SSAN [73]	Self-supervised subspace attention	Improved robustness under scarce labels
FLFuse-Net [70]	Edge compensation / flow path	Crisper boundaries; fewer artifacts

2.3.10 Benchmarking and Clinical Deployment Considerations

Benchmarking remains difficult because fused ground truth is rarely available; therefore, studies typically combine multiple surrogate metrics (MI/entropy, SSIM/FSIM, PSNR, VIF) with qualitative assessment, and—when feasible—task-based metrics or reader studies [61, 74, 2].

For real deployment, key constraints include latency, memory footprint, robustness to registration errors, and generalization across scanners/sites. These constraints motivate lightweight backbones, compact attention, and careful reporting of compute requirements alongside fusion quality [68, 60, 62, 75].

Tables 2.7–2.10 summarize practical deployment considerations, representative clinical use cases, and common operational risks described in recent surveys.

Table 2.7 Key considerations for clinical deployment of fusion systems.

Aspect	Practices
Latency constraints	Streaming/tiling inference; asynchronous I/O; batching tuned to workflow
Memory footprint	Mixed-precision; pruning/quantization; patch-wise or hierarchical attention
Hardware variation	CPU/GPU fallbacks; prefer widely-available kernels/ops
Monitoring	Automated quality checks; drift/registration monitors; audit logs
Security and privacy	Access control; encrypted storage; minimal PHI in logs; basic threat modeling

Table 2.8 Representative clinical application scenarios for multimodal fusion.

Domain	Fusion	Benefit
Oncology	PET+CT/MRI	Functional uptake combined with precise anatomy for staging and planning
Neurology	MRI sequences / MRI+PET	Lesion delineation and surgical planning; improved visibility of boundaries
Cardiology	CT+MRI	Structural–functional assessment; motion-aware analysis support

Table 2.9 Architecture design guidelines under common practical constraints.

Constraint	Recommendation	Refs
Few labels / pairs	Self-supervised pretraining; correlation/memory modules	[73, 76, 77]
Compute budget	Lightweight backbones; compact attention	[68, 70]
Global structure	Add transformer blocks / cross-attention	[60, 67, 66]
Edge fidelity	Edge-/gradient-aware paths; frequency cues (e.g., wavelets)	[70, 71, 64]
Generalization	Diverse training data; augmentation; normalization	[75, 62]

Table 2.10 Operational risks and typical mitigation strategies for real-world use.

Risk	Mitigation	Refs
Misregistration artifacts	Robust registration; model components that downweight misaligned regions	[74]
Metric–clinical mismatch	Include task-based metrics; reader studies when feasible	[61]
Domain shift (scanner/site)	Domain-diverse training; normalization; adaptation strategies	[62]
Compute limits in clinics	Lightweight/hierarchical attention; compression and mixed precision	[68]

2.4 Deep Learning in Medical Image Screening

Automated radiological triage has become essential in contemporary clinical workflows, enabling rapid pathology detection across diverse imaging modalities. Recent advances in neural network architectures—particularly Convolutional Neural Networks (CNNs)—have fundamentally transformed this domain. Unlike conventional computer-aided diagnosis systems that depend on manually engineered features (e.g., SIFT, HOG), modern deep architectures autonomously construct multi-level representational hierarchies directly from pixel intensities. This paradigm shift has resulted in unprecedented accuracy gains across classification, localization, and segmentation benchmarks.

2.4.1 Deep Learning for Chest X-ray Analysis

Chest X-ray (CXR) is one of the most common radiological examinations for screening thoracic diseases due to its low cost, speed, and low radiation dose. Deep

learning, especially Convolutional Neural Networks (CNNs), has shown remarkable success in detecting conditions such as pneumonia, tuberculosis, and lung nodules from CXRs.

Early works utilized standard CNN architectures like AlexNet, VGG, and ResNet, pre-trained on ImageNet, and fine-tuned them on medical datasets such as ChestX-ray14 [78] and CheXpert [79]. For instance, Rajpurkar et al. [80] developed CheXNet, a 121-layer DenseNet model, which achieved radiologist-level performance in detecting pneumonia.

2.4.2 Deep Learning for COVID-19 Screening

The COVID-19 pandemic accelerated the adoption of deep learning for medical screening. With the rapid spread of the virus and the shortage of RT-PCR test kits in the early stages, radiological imaging (CXR and CT) became a crucial complementary screening tool.

COVID-19 Detection from CXR. Numerous studies have proposed deep learning models to differentiate COVID-19 pneumonia from other viral pneumonias and normal cases using CXRs. Ozturk et al. [81] proposed DarkCovidNet, based on the Darknet-19 model, for automatic COVID-19 detection. Wang et al. [82] introduced COVID-Net, a tailored deep convolutional neural network design for detecting COVID-19 cases from CXR images.

COVID-19 Detection from CT. CT scans provide more detailed 3D information than X-rays and are more sensitive in detecting early-stage COVID-19 lesions (e.g., ground-glass opacities). Li et al. [83] developed a 3D deep learning model (COVNet) to detect COVID-19 from chest CT scans, achieving high sensitivity and specificity.

2.4.3 Challenges and Recent Advances

Despite the success, deep learning in medical screening faces several challenges:

- **Data Scarcity and Imbalance:** Medical datasets are often small and imbalanced compared to natural image datasets. Techniques like transfer learning, data augmentation, and few-shot learning are commonly used to mitigate this.
- **Generalization:** Models trained on data from one hospital often fail to generalize to data from other institutions due to differences in scanner protocols and patient populations.
- **Explainability:** Deep learning models are often "black boxes," making it difficult for clinicians to trust their decisions. Explainable AI (XAI) meth-

ods, such as Grad-CAM, are increasingly being integrated to visualize the regions of interest that contribute to the model's prediction.

Recent advancements include the use of Vision Transformers (ViT) and hybrid architectures (combining CNNs and Transformers) to capture both local and global features in medical images. Additionally, ensemble methods and weight-averaging techniques (like Model Soups) are being explored to improve model robustness and accuracy without increasing inference cost.

2.5 Critical Analysis and Comparison

While deep learning methods have significantly advanced the field, a critical analysis of the current literature reveals persistent challenges and trade-offs that motivate further research.

Critical Analysis of Existing Approaches:

- **Generative Models (GANs):** While GANs like FusionGAN and DDcGAN produce visually appealing results with sharp textures and high contrast, they are notoriously difficult to train due to convergence instability. A more serious concern in the medical domain is their potential to "hallucinate" details—generating plausible-looking but non-existent anatomical features—which poses a clinical risk. Furthermore, they often struggle with quantitative structural consistency (lower SSIM) compared to simpler CNN-based methods.

- **CNN-based Methods:** Models like DenseFuse and IFCNN are computationally efficient and robust to noise. However, because convolution is inherently a local operation, they often fail to preserve global contrast and long-range dependencies. This limitation frequently leads to spectral distortion in PET/SPECT fusion, where the functional heatmap intensity is incorrectly altered by local gradients from the MRI.

- **Transform Domain vs. Deep Learning:** Traditional transform methods (NSCT, Wavelet) offer mathematical guarantees and interpretability but lack the adaptability of data-driven features, leading to suboptimal performance on complex soft-tissue boundaries. Deep learning offers superior feature representation but acts as a "black box," lacking the transparency required for some clinical workflows.

- **Identified Research Gap:** Most existing hybrid methods simply concatenate deep features with manual rules. There is a lack of frameworks that explicitly optimize the trade-off between the preservation of broad anatomical

structure (low-frequency information) and the enhancement of fine-grained texture (high-frequency information) in a mathematically constrained manner. This justifies the exploration of combining *meta-heuristic optimization* (which excels at global search for intensity/contrast balance, addressing the weaknesses of CNNs) with *deep learning* (which excels at local texture extraction, addressing the weaknesses of traditional optimization).

Comparative Summary: Table 2.11 provides a comparative overview of the representative methods discussed, highlighting their advantages, drawbacks, and typical performance characteristics.

Table 2.11 Comparison of traditional and deep learning-based medical image fusion methods.

Method	**Category**	**Advantages**	**Drawbacks**	**Typical Metric Profile**
NSCT	Transform domain	Shift-invariant, multi-scale, good edge preservation	Computationaly expensive, requires manual rule design	High SSIM, Low Entropy
DenseFuse	CNN (Autoencoder)	Fast inference, robust feature extraction, easy training	Loss of global contrast, hand-crafted fusion layer	High MI, Moderate SSIM
FusionGAN	GAN	Sharp textures, end-to-end learning, no manual rules	Training instability, mode collapse, risk of artifacts	Moderate MI, Lower SSIM
TransFuse	Transformer	Captures long-range dependencies and global context	High computational and memory cost, requires large data	High SSIM, High MI
EOA-VGG (Ours)	**Hybrid Opt-DL**	**Balances global contrast (via Opt) and local detail (via DL)**	**Slower inference due to iterative optimization**	**Balanced High Metrics**

2.6 Limitations of Existing Methods

Current research methods expose several limitations in medical image fusion. The first disadvantage is related to the synthesis of the base components, in which the max or average methods are often applied to create the fused base component due to its simple computation. However, max or average methods usually produce a reduction of information, contrast and brightness of the fused component. For example, Li et al., 2021 [84] utilized the max technique to perform the fusion of base components of the input MRI, PET and SPECT images. This method produces a loss of some input information in the fused image due to the fact that the MRI image has a higher contrast than the PET image. As a consequence, the max method chooses a major part of information from MRI image rather than PET image. The average method for synthesizing the base components selects half of brightness in each type of image modality, leading to the reduction of brightness and contrast in the fused image. Another direction attracting researchers recently

in fusing base components is the use of optimization algorithms to choose appropriate weights for performing the base component fusion process. Hence, it may improve the quality of the fused base component.

The second limitation is related to the synthesis of the detail components where similarly to the base component, the max or average methods are applied to produce the fused detail component. With the max methods, it tends to select the detail information from MRI image rather than the PET image. As a consequence, the fused image loses information from the PET image modality. Along with max and average techniques, image processing methods are applied to extract more detail features such as edges of the input images for adding to the fused detail component. Researches show that extracted edges from the input images help in enhancing the quality of the fused images. As a consequence, this direction is becoming a trend for medical image fusion in these recent years.

In the context of this thesis, we will study the state-of-art methods and techniques to overcome the mentioned limitations of the medical image fusion problem. Specifically, we propose the following methods to solve the mentioned limitations of the existing approaches in medical image fusion:

- The proposal of a new algorithm to fuse the base components based on the Equilibrium optimization algorithm.
- The proposal of a new algorithm to fuse the detail components based on deep learning and transfer learning.

2.7 Chapter Summary

In this chapter, we present a survey of the state-of-the-art methods on medical image fusion. We divide the state-of-the-art medical fusion methods into three categories: Spatial Domain Methods, Transform Domain Methods and Machine Learning based Methods. From the survey, we provide an analysis of the limitations of the current medical image fusion methods, that is the quality of the fused medical image such as contrast index still needs further investigations. This is also the main focus of this thesis.

CHAPTER 3

Background

3.1 Image Processing

3.1.1 Color Space Conversion

RGB. Medical images captured from equipment are usually stored in DICOM format which uses RGB color space. RGB color is designed to match the human perception of color. It is a device-dependent color space, which means that the color values are not absolute but depend on the device used to capture the image.

RGB color space use three channels to represent the color of a pixel. Each channel represents the intensity of the color red, green, and blue. Using a typical 8-bit value to represent each channel, the intensity of each channel ranges from 0 to 255. RGB is considered an additive color model because the color of a pixel is combination of the intensity of the three channels.

However, RGB color space is not suitable for medical image processing tasks. The color information is not separated from the intensity information. This makes it difficult to perform image processing tasks such as image enhancement, edge detection, and feature extraction. Therefore, it is necessary to convert the RGB color space to another color space that separates the color information from the intensity information.

YCbCr. Among the color spaces that are used in medical image processing, YCbCr and HSV are the most popular. YCbCr is a color space that separates the color information from the intensity information. The Y channel represents the intensity (brightness) of the pixel, while the Cb and Cr channels represent the color information. Cb and Cr are the blue-difference and red-difference chroma components, respectively.

Conversion from RGB (in JPEG digital format) to YCbCr is done using the following equations:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.257 & 0.564 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \tag{3.1}$$

HSV. On the other hand, HSV is a color space that separates the intensity information from the color information. The H channel represents the hue of the pixel, the S channel represents the saturation of the pixel, and the V channel represents the intensity of the pixel. The H channel in HSV is similar to the Y channel in YCbCr because they both represent the intensity of the pixel. However, the HSV color space is more intuitive than YCbCr since it's easier to understand the hue and saturation of a given color than the blue-difference and red-difference chroma components.

Conversion from RGB to HSV is done using the following equations:

$$H = \begin{cases} 0^o & \Delta = 0 \\ 60^o \times (\frac{G-B}{\Delta} \bmod 6) & max = R \\ 60^o \times (\frac{B-R}{\Delta} + 2) & max = G \\ 60^o \times (\frac{R-G}{\Delta} + 4) & max = B \end{cases} \tag{3.2}$$

$$S = \begin{cases} 0 & max = 0 \\ \frac{\Delta}{max} & max \neq 0 \end{cases} \tag{3.3}$$

$$V = max(R, G, B) \tag{3.4}$$

In this thesis, while recognizing the utility of YCbCr and HSV, we elect to use the YUV color space for the proposed fusion pipeline (as detailed in Chapter 4). YUV offers a computationally efficient linear transformation from RGB that cleanly isolates luminance (Y) for structural fusion, avoiding the non-linear singularities of HSV or the scaling variances often encountered with YCbCr implementations in medical imaging.

3.1.2 Histogram Equalization

Histogram equalization is a technique used to improve the contrast of an image by redistributing the intensity values of the pixels. The goal of histogram equalization is to make the histogram of the image as flat as possible. This means that the intensity values of the pixels are spread out over the entire intensity range. This technique is effective for images with low contrast and can be used to enhance the contrast of medical images.

Algorithm 1 Histogram Equalization

Input I: input image of size $M \times N$
Output I_{eq}: histogram equalized image

Initialize histogram array H, cumulative histogram array C, output image I_{eq} of size $M \times N$ to zeroes
for each pixel $I(i, j)$ in I **do**
    $H(I(i, j)) \leftarrow H(I(i, j)) + 1$
end for
$C(0) \leftarrow H(0)$
for $k \leftarrow 1$ to 255 **do**
    $C(k) \leftarrow C(k - 1) + H(k)$
end for
for $k \leftarrow 0$ to 255 **do**
    $C(k) \leftarrow \frac{C(k)-C(0)}{M \times N - C(0)} \times 255$
end for
for each pixel value $I(i, j)$ in I **do**
    $I_{eq}(i, j) \leftarrow C(I(i, j))$
end for

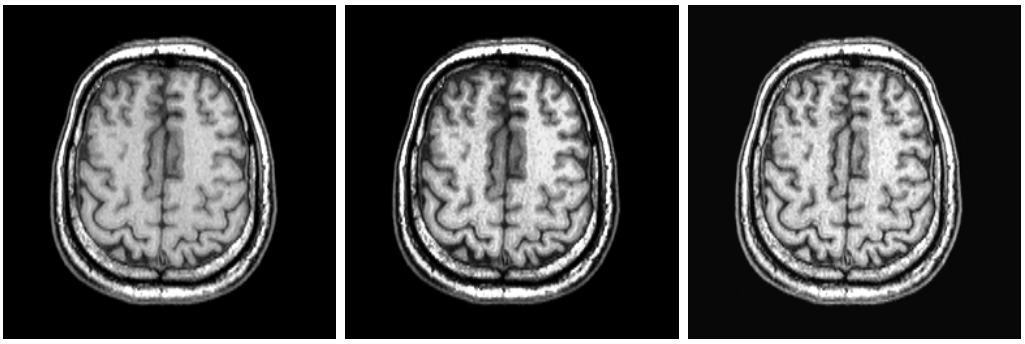

Fig. 3.1 Examples of histogram equalization in MRI mages. Left: Original image. Middle: Histogram equalization. Right: CLAHE.

Histogram equalization is performed with the following steps:

1. Compute the histogram of the image;
2. Compute the cumulative distribution function (CDF) of the histogram;
3. Compute the transformation function using the CDF;
4. Apply the transformation function to the image.

Algorithm 1 shows the pseudocode for histogram equalization. The algorithm takes an input image I and returns the histogram equalized image I_{eq}.

Figure 3.1 shows an example of histogram equalization applied to a MRI image. The left image is the original image, and the right image is the result of histogram equalization. It can be examined from the image that the contrast of the image is improved, and the details are more visible in the enhanced image.

Algorithm 2 Contrast Limited Adaptive Histogram Equalization

Input I: input image of size $M \times N$, C: region size, L: clip limit
Output I_{clahe}: CLAHE enhanced image

Initialize I_{clahe} to zeros
Divide the image I into contextual regions $R^{C \times C}$
for each $R_i \in R$ **do**
    Initialize histogram H
    for each pixel $I(i,j) \in R_i$ **do**
        $H(I(i,j)) \leftarrow H(I(i,j)) + 1$
    end for
    Clip the histogram $H(x \leq L) = 0$
    Redistribute the clipped pixels uniformly across the histogram
    Initialize cumulative histogram array C
    $C(0) \leftarrow H(0)$
    for $k \leftarrow 1$ to 255 **do**
        $C(k) \leftarrow C(k-1) + H(k)$
    end for
    for $k \leftarrow 0$ to 255 **do**
        $C(k) \leftarrow \frac{C(k)-C(0)}{C(C-1)\times C(C-1)-C(0)} \times 255$
    end for
    for each pixel value $I(i,j) \in R_i$ **do**
        $I_{clahe}(i,j) \leftarrow C(I(i,j))$
    end for
end for

Histogram equalization is a simple and effective technique for enhancing the contrast of an image. However, it can amplify the noise in the image, which can reduce the quality of the image. Therefore, it is important to avoid histogram equalization in such cases and consider other techniques that can reduce the amplification of noise.

CLAHE. Contrast Limited Adaptive Histogram Equalization (CLAHE) is an extension of histogram equalization that is used to enhance the contrast of an image while limiting the amplification of noise. CLAHE divides the image into small blocks and applies the regular histogram equalization to each block. This allows CLAHE to enhance the contrast of the image while preserving the local contrast. CLAHE is particularly useful for medical images that have varying contrast levels in different regions of the image.

Figure 3.1 shows an example of CLAHE applied to a MRI image. The right image is the result of CLAHE applied to the original image (left). It can be seen that the result is better than histogram equalization (middle), in terms of preserving the local contrast of the image.

Algorigthm 2 shows the pseudocode for CLAHE. The algorithm takes an input image I, region size C, and clip limit L as input and returns the CLAHE enhanced

image I_{clahe}.

3.1.3 Edge Detection

While image enhancement techniques are useful for enhancing image quality, they do not reveal information about object boundaries, which is particularly relevant in medical imaging applications where organ and tissue boundaries are important for diagnosis, particularly in image fusion tasks. In order to do this, the borders of objects in an image are detected and highlighted using edge detection algorithms. Convolution is the base of edge detection algorithms to identify edges in an image by applying a filter, or kernel, to an image in order to find edges. A variety of edge detection techniques can be applied, the most popular among them being Canny, Kirsch, Prewitt, and Robinson. These techniques identify edges in an image by utilizing several convolution kernels in several stages.

Canny edge detection algorithm is one of the most widely used and effective techniques for identifying edges in digital images. This method aims to detect a wide range of edges while minimizing error rates, achieving good localization, and ensuring only one response to a single edge. This method consists of the following step:

- Noise reduction: Apply a Gaussian filter to smooth the image and suppress noise.
- Gradient calculation: Compute the intensity gradients of the image using operators such as Sobel to find the edge strength and direction.
- Non-maximum suppression: Thin out the edges by retaining only the local maxima of the gradient magnitude in the direction of the gradient.
- Double thresholding: Classify edges as strong, weak, or non-relevant using two thresholds (high and low).
- Edge tracking by hysteresis: Finalize edge detection by retaining weak edges that are connected to strong edges, and discarding the rest.

Sobel operator is a discrete differentiation operator used widely with Canny edge detection. It calculates the gradient of the image intensity at each pixel, thereby highlighting regions of high spatial frequency that correspond to edges. The Sobel operator uses two 3×3 convolution kernels: one estimates the gradient in the horizontal direction (G_x) and the other in the vertical direction (G_y). These kernels are designed to emphasize changes in pixel intensity along the respective axes. When applied to an image, they produce two gradient images

which can be combined to calculate the overall edge magnitude and direction. The standard 3×3 Sobel filters are defined as follows:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{3.5}$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{3.6}$$

The result is then typically combined using the magnitude of the gradient

$$G = \sqrt{G_x^2 + G_y^2} \tag{3.7}$$

Kirsch edge detection method is a compass kernel-based technique used to detect edges in all directions by evaluating the maximum edge strength across eight compass directions (N, NE, E, SE, S, SW, W, NW). Unlike gradient-based methods such as Sobel or Prewitt, the Kirsch operator emphasizes directional intensity changes more aggressively, making it especially useful for detecting prominent and directional edges. It uses a set of eight 3×3 convolution kernels, each designed to detect edges in one specific direction. The central idea is to convolve the image with each kernel and, for every pixel, retain the maximum response value among all directions.

The Kirsch operator's kernels are defined as follows:

$$K_1 = \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}, K_2 = \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}, K_3 = \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix}, K_4 = \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} \tag{3.8}$$

$$K_5 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix}, K_6 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix}, K_7 = \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix}, K_8 = \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} \tag{3.9}$$

After calculating the response of each kernel, the final edge strength at each pixel is determined by taking the maximum response across all eight directions, as follows:

$$E(x, y) = \max_{i=1,\ldots,8} |I(x, y) * K_i| \tag{3.10}$$

Where:

- $E(x, y)$ is the edge strength at pixel location

- $I(x, y)$ is the input image
- K_i is the i^{th} Kirsch operator kernel above (from 1 to 8, corresponding to 8 compass directions)
- $*$ denotes the convolution operation
- $|\cdot|$ represents the absolute value (edge magnitude)
- max selects the maximum response across all directions.

Prewitt edge detection method is a gradient-based technique used to identify edges in digital images by approximating the gradient of the image intensity function. It employs two simple 3×3 convolution kernels: one to detect horizontal edges and another for vertical edges. These kernels are designed to respond maximally to horizontal and vertical gradients, respectively, by emphasizing differences in intensity between neighboring pixels, and are defined as follows:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \tag{3.11}$$

After convolving these kernels with the input image, the resulting horizontal and vertical gradient values are combined, using the Euclidean norm, to compute the gradient magnitude at each pixel, as follows:

$$G = \sqrt{G_x^2 + G_y^2} \tag{3.12}$$

Although the Prewitt operator is less sensitive to noise than some other methods like the Sobel operator, it does not include any inherent smoothing, which can make it more susceptible to noise in practice.

Robinson edge detection algorithms are similar to Kirsch, but they use different convolution kernels to detect edges in an image. Each kernel is constructed with integer coefficients that emphasize intensity changes in its respective direction, with a central zero value and symmetrical positive and negative weights distributed accordingly. These kernels are defined as follows:

$$K_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, K_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}, K_3 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, K_4 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}, \tag{3.13}$$

$$K_5 = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, K_6 = \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}, K_7 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, K_8 = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix} \tag{3.14}$$

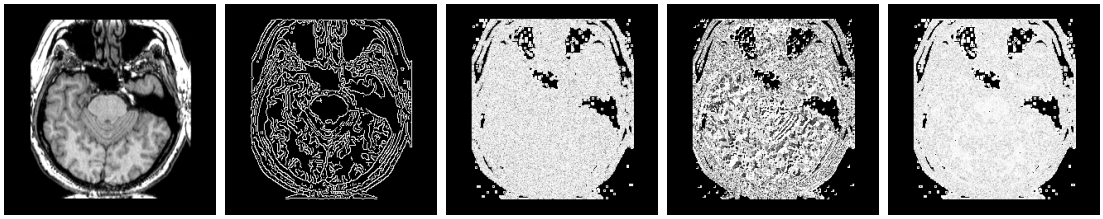

Fig. 3.2 Examples edge detection on MRI images extracted from the slice #50 of the brain hemispheric at the planes transaxial. From left to right: Original image, Canny, Kirsch, Prewitt, and Robinson.

Figure 3.2 shows examples of edge detection algorithms applied to a MRI image. The left image is the original image, and the right images are the results of different edge detection algorithms applied to the original image. It can be seen that the edges of the objects in the image are highlighted in the edge detection images. The choice of edge detection algorithm depends on the application and the characteristics of the image. For example, it can be seen that the Canny edge detection algorithm produces better results than the other algorithms in this case.

In this thesis, we will experiment and compare results of different edge detection algorithms for image processing tasks such as image enhancement, feature extraction, and image fusion in medical image fusion task in chapter 4.

3.1.4 Local Energy Functions

The local energy of an image refers to the energy of a small region or window of the image, mainly to extract information about the texture, sharpness, or other properties of an image in a local region. It is calculated as the sum of the squares of the pixel values in that region. The local energy of an image can be used to characterize the texture, sharpness, or other properties of the image in that region. The application of the local energy function has been documented in various studies on image fusion [85].

Let us consider an image, denoted by I, and a local window with dimensions $k \times k$, represented by W. The local energy function, $E_L(i,j)$, can be computed using the mathematical formulation outlined in Eq. (3.15).

$$E_L(i,j) = \sum_{u=0}^{k-1} \sum_{v=0}^{k-1} W(u,v) I^2(i+u, j+v) \tag{3.15}$$

Local energy is one of the commonly techniques used to extract useful detail information from the input images.

3.1.5 Two-scale Decomposition

Medical images are often complex and contain a lot of information that is not relevant for the image fusion task. Image decomposition is a technique used to separate the image into its fundamental components, such as edges, textures, and structures, in order to extract useful information from the image. Decomposition methods are useful for many medical image processing tasks, especially image fusion. There are several image decomposition methods that can be used for medical image processing, such as wavelet transform or component separation in frequency domain.

Our findings indicate that the two-component image decomposition technique is prevalent in many image synthesis models [86]. This method brings the benefits of fast image decomposition in comparison to other methods, such as NSCT or NSST, which require a considerable amount of time to execute. Furthermore, by merging images at varying scales, the two-scale image decomposition has a meaningful contribution to reducing the noise present in the resulting fused image. Hence, we have selected the two-scale decomposition approach for the construction of our image synthesis model in this thesis.

Given an image X. The symbols C_{HF} and C_{LF} represent the base and detail layers from the X image, respectively. To establish the values of C_{HF} and C_{LF}, a two-step process is employed as outlined below:

- **Step 1**: The value of C_{LF} is derived by solving the optimization problem presented in equation (3.16).

$$\underset{C_{LF}}{arg\ min} \|X - C_{LF}\|_F^2 + \lambda \left(\|v_x * C_{LF}\|_F^2 + \|v_y * C_{LF}\|_F^2\right) \tag{3.16}$$

- **Step 2**: C_{HF} is calculated according to Eq. (3.17).

$$C_{HF} = X - C_{LF} \tag{3.17}$$

where

- The parameter λ serves as the regularization parameter.
- $v_x = [-1\ 1]$ and $v_y = [-1\ 1]$.

The outcomes resulting from image decomposition are depicted in Fig. (3.3).

Discrete Wavelet Transform. In order to efficiently combine information from various images, wavelet transforms in image decomposition for image fusion require a number of steps [87]. Firstly, each source image is performed with wavelet

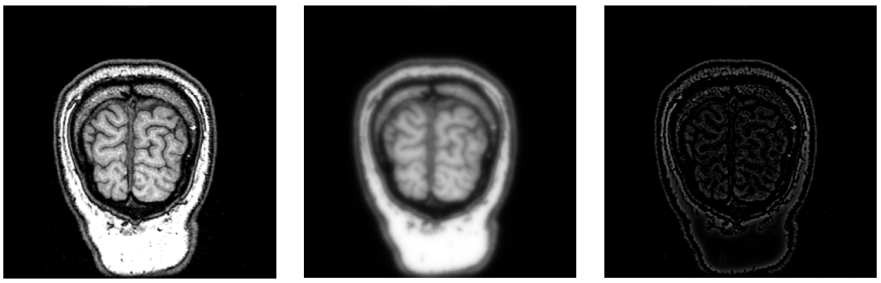

Fig. 3.3 Illustrate the decomposition of an input image into two components

decomposition, which divides the image into a collection of wavelet coefficients by filtering and downsampling the image. This process separates the image into different frequency components, capturing both the low-frequency approximation (which contains the image's basic structure) and high-frequency details (which include edges and fine features).

In this thesis, we employed Haar wavelet [88] to perform the transformation technique since the Haar transform is the simplest one. The input medical image I has experienced the transformation with a Haar matrix (see Equation 3.18) to create four coefficients namely the Approximation, Horizontal, Vertical, and Diagonal.

By keeping only the Approximation and dropping the other three coefficients, we made an inverse Haar transform to create the base layer image I^b. The detail layer is simply the difference between I and I^b.

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{3.18}$$

Furthermore, more specific information about the image I can be represented the larger the Haar matrix size. This is so that more delicate patterns and features can be captured by finely dividing the image into smaller blocks, which is made possible by larger Haar matrices. Because H_2 is appropriate for the basic representation but not for detailing the input image, we were able to justify our choice.

Secondly, after decomposed, these wavelet coefficients from each source image are combined using various fusion rules, such as using the maximum values or averaging, to retain the most informative features from each image. Finally, once the coefficients are fused, an inverse wavelet transform is applied to reconstruct the composite image back to the spatial domain. This step reverses the decomposition process, integrating the combined coefficients to produce a single, highly informative image.

Algorithm 3 2D Haar Wavelet Transform

Input I: input image of size $M \times N$, C: region size, L: clip limit
Output I_{clahe}: CLAHE enhanced image

Initialize W as a copy of I
$h \leftarrow M$, $w \leftarrow N$
while $h > 1$ and $w > 1$ **do**
    for $i \leftarrow 0$ to $h - 1$ by 2 **do**
        for $j \leftarrow 0$ to $w - 1$ by 2 **do**
            $a \leftarrow I(i, j)$
            $b \leftarrow I(i, j + 1)$
            $c \leftarrow I(i + 1, j)$
            $d \leftarrow I(i + 1, j + 1)$
            $W(i/2, j/2) \leftarrow \frac{a+b+c+d}{4}$ ▷ Average
            $W(i/2, j/2 + w/2) \leftarrow \frac{a-b+c-d}{4}$ ▷ Horizontal detail
            $W(i/2 + h/2, j/2) \leftarrow \frac{a+b-c-d}{4}$ ▷ Vertical detail
            $W(i/2 + h/2, j/2 + w/2) \leftarrow \frac{a-b-c+d}{4}$ ▷ Diagonal detail
        end for
    end for
    $h \leftarrow h/2$
    $w \leftarrow w/2$
end while

Fig. 3.4 Examples discrete wavelet transform on MRI images extracted from the slice #50 of the brain hemispheric at the planes transaxial. From left to right: Original image I, approximation (base) coefficients of I, horizontal detail coefficients of I, vertical detail coefficients of I and diagonal detail coefficients of I.

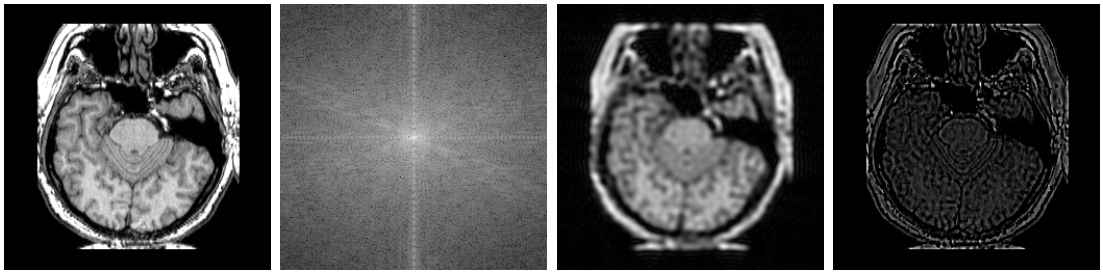

Fig. 3.5 Examples of decomposition using fourier transform on MRI images extracted from the slice #50 of the brain hemispheric at the planes transaxial. From left to right: Original image I, magnitude spectrum of I in frequency domain, base components of I using low frequency areas, base components of I using high frequency areas.

Figure 3.4 shows an example of wavelet decomposition applied to a MRI image. The left image is the original image, and the right images are the wavelet coefficients of the image. It can be seen that the image is decomposed into different frequency components, capturing both the low-frequency approximation and high-frequency details of the image.

Fourier Transform, on the other hand, is also a transform technique widely used in signal processing and image processing to analyze an image in the frequency domain. The Fourier Transform of an image is used to decompose the image into its frequency components. Using different frequencies, this method can effectively separate an input image into its fundamental components, such as base or detail components [89].

Figure 3.5 shows an example of Fourier decomposition applied to a MRI image. The left image is the original image, and the right images are the magnitude spectrum of the image, the base and and the detail components of the image.

Using image decomposition methods such as wavelet transform or Fourier transform, we can separate the image into its fundamental components, such as edges, textures, and structures, in order to extract useful information from the image. Each component can be processed separately to enhance the quality of the image or extract useful features for image fusion tasks.

3.2 Optimization

3.2.1 Fundamentals of Optimization

Optimization is the problem which finds the best possible solution among many options. Optimization uses math and computer methods to figure out the choice that either gets the most of something or the least of something else, all while sticking to certain rules or limitations.

In the simple case, optimization is about finding the smallest or largest value of something. Take the function $f(x) = x^2$ for example; its minimum value is 0 when x=0. For simple functions, we can often use calculus (first and second derivatives) to find potential minimums or maximums. However, this gets much harder with complex functions—those that are nonlinear, have multiple peaks and valleys (multimodal), or involve many variables. Plus, if a function has discontinuities, getting derivative information becomes tricky, which can be a real problem for traditional methods like hill-climbing.

In general, an optimization problem is typically formulated to minimize one or more objective functions $(f_1(x), ..., f_M(x))$ by carefully selecting the values of decision variables $(x_1, ..., x_N)$. These selections aren't limitless; they must satisfy a series of equality constraints $(h_j(x) = 0)$ and inequality constraints $(g_k(x) <= 0)$.

To solve the optimization problem, we need effective search or optimization algorithms. These algorithms can be categorized in various ways, depending on their specific features and what they aim to achieve.

Optimization algorithms can be sorted by how they use a function's derivative (or gradient). Gradient-based algorithms, like hill-climbing, leverage this derivative information and are often highly efficient. In contrast, derivative-free algorithms (also called gradient-free algorithms) don't use derivatives at all, relying solely on the function's output values. These derivative-free methods, such as the Nelder-Mead downhill simplex, are particularly handy when a function is discontinuous or when calculating its derivatives precisely would be costly.

Optimization algorithms can also be viewed through another lens: whether they use a single "agent" or multiple "agents" to find the best solution. Trajectory-based algorithms work with just one solution at a time. As the algorithm iterates, this single solution carves out a specific path or "trajectory" through the problem space. Hill-climbing, for example, is a classic trajectory-based method; it moves from a starting point to a final solution along a series of connected steps. Simulated annealing, a popular metaheuristic, is another important example of a trajectory-based algorithm. In contrast, population-based algorithms utilize multiple solutions or "agents" simultaneously. These agents interact with each other and collectively explore the problem space, tracing out multiple paths. Particle Swarm Optimization (PSO) is a prime example of a population-based algorithm, where a "swarm" of particles works together to find the optimal solution.

Another way to classify optimization algorithms is by whether they include randomness. Deterministic algorithms operate in a fixed, predictable way, without any random elements. This means if you start them from the exact same point, they'll always arrive at the identical solution. Hill-climbing and the downhill simplex method are prime examples of deterministic algorithms. Conversely,

stochastic algorithms incorporate some degree of randomness. Because of this, even when started from the same initial point, they'll typically reach a different solution each time they're run. Genetic algorithms and Particle Swarm Optimization (PSO) are good illustrations of stochastic algorithms.

We can also classify optimization algorithms based on their search capability—that is, whether they aim for a local or global optimum. Local search algorithms typically zero in on a local optimum, which might not be (and often isn't) the absolute best solution (global optimum). These algorithms are often deterministic and struggle to escape from these local peaks or valleys. Simple hill-climbing is a good example of a local search algorithm.

For problems requiring global optimization, local search algorithms just won't cut it. That's where global search algorithms come in. While modern metaheuristic algorithms are usually well-suited for global optimization, they don't always guarantee success or efficiency. Interestingly, even a simple trick like adding random restarts can transform a local search algorithm like hill-climbing into one capable of finding global solutions. This highlights a key point: randomization is often a powerful ingredient for effective global search algorithms.

It's important to remember that optimization algorithms don't always fit neatly into a single category. Many are mixed-type or hybrid algorithms, meaning they combine elements from different classifications. This could involve blending deterministic components with randomness, or even integrating two or more distinct algorithms to create a more efficient solution.

3.2.2 Metaheuristic Optimization

Metaheuristic optimization algorithms are a class of optimization algorithms that are used to solve complex optimization problems that are difficult to solve using traditional optimization techniques. It involves solving complex problems using metaheuristic algorithms. These algorithms are widely applicable, appearing in fields such as engineering, economics, internet routing, etc. In the real world, most optimization problems are nonlinear and multimodal under various complex constraints. The main reason behind using metaheuristic optimization algorithms is that they are capable of finding near-optimal solutions to complex problems in a reasonable amount of time. These algorithms are particularly useful when the search space is vast and the objective function is non-convex, making it difficult to find the global optimum using traditional mathematical optimization techniques.

In the past, algorithms that used random elements were often called heuristics. However, more recent academic writing, following the lead of Fred Glover (who coined the term in 1986), now generally refers to them as metaheuristics. Re-

searchers adopt Glover's convention and use metaheuristics for all contemporary algorithms inspired by nature. Loosely translated, heuristic means "to find" or "to discover through trial and error." The prefix "meta-" signifies "beyond" or "higher level." Therefore, metaheuristics are typically more effective than simple heuristics. As Glover and Laguna described in 1997, a metaheuristic acts as a "master strategy" that directs and adjusts other heuristics to find solutions that go beyond what simpler methods might achieve in a search for local optimums.

All metaheuristic algorithms balance randomization with local search. They are capable of finding good solutions to complex optimization problems relatively quickly, though they don't guarantee finding the absolute best (optimal) solution. The expectation is that these algorithms will work most of the time, rather than every single time. Most metaheuristic algorithms are well-suited for global optimization. For a comprehensive overview, you can refer to Voss's 2001 review.

Meta-heuristic algorithms [90] are broadly categorized in two ways: by the number of solutions they consider (single-solution or multiple-solution) or by their inspiration (e.g., biological, evolutionary, human, or mathematical). Our focus is on multiple-solution, or population-based, algorithms, which are widely favored over single-solution methods.

Population-based algorithms offer several benefits:

- Improved Avoidance of Local Optima: The interaction among multiple solutions helps prevent the algorithm from getting trapped in suboptimal solutions and allows it to escape local optima.

- Enhanced Exploration: Having multiple solutions working concurrently leads to more thorough exploration of the search space, which accelerates the discovery of the global optimum.

Researchers in Image Fusion frequently use meta-heuristic algorithms due to their ease of implementation and their ability to operate without requiring gradient information. They leverage these algorithms to optimize parameters within their image fusion processes.

3.2.3 Equilibrium Optimization Algorithm (EOA)

EOA is a subset of metaheuristic optimization algorithms that mimic the equilibrium state of a system, where particles interact with each other to reach a stable state. EOA is a population-based optimization algorithm that operates on a pool of equilibrium, with each particle representing a potential solution. The particles' movement is influenced by their concentration and the global best-known concentration, enabling the algorithm to converge towards optimal solutions. Faramarzi

et al. [91] introduced the EOA that has been applied to multiple image processing tasks, including medical image fusion [92], image segmentation [93], and feature selection [94], and has shown efficacy in all of these contexts.

Based on the mass balance equation in physics, we know that a simple dynamic mass balance on a control volume V has the tendency to change its concentration C in order to reach the equilibrium state of the system, in which there would be no generation inside V. This can be described as a mathematical model with a first-order ordinary differential equation:

$$V\frac{dC}{dt} = QC_{eq} - QC + G \tag{3.19}$$

where:

- V is the control volume.
- C is the concentration inside V.
- $V * dC/dt$ is the rate of change of mass in V. $V\frac{dC}{dt}$ reaches 0 when the system reaches its equilibrium state.
- Q is the volumetric flow rate into and out of V.
- C_{eq} is the concentration at an equilibrium state.
- G is the mass generation rate inside V.

By rearranging and taking the integral:

$$\int_{C_0}^{C} \frac{dC}{\lambda C_{eq} - \lambda C + \frac{G}{V}} = \int_{t_0}^{t} dt \tag{3.20}$$

The solution of the mass balance equation can be obtained as follows:

$$C = C_{eq} + (C_0 - C_{eq})\, F + \frac{G}{\lambda V}(1 - \mathrm{F})F = \exp\left[-\lambda\,(\mathrm{t} - t_0)\right] \tag{3.21}$$

where t_0 and C_0 are the initial start time and concentration.

Inspired by this physic's phenomenon, EOA, a population-based, nature-inspired, physics-based meta-heuristic optimization algorithm has been proposed [91]. Since then, EOA has gained popularity in many domains of science and is considered to be the leading algorithm, featuring in many research publications. In EOA, solutions are represented by particles in the system, and concentrations are represented by the particle's position [5]. The main idea of EOA algorithm is presented in Figure 3.6.

EOA first randomly generates its initial set of particles across the search space:

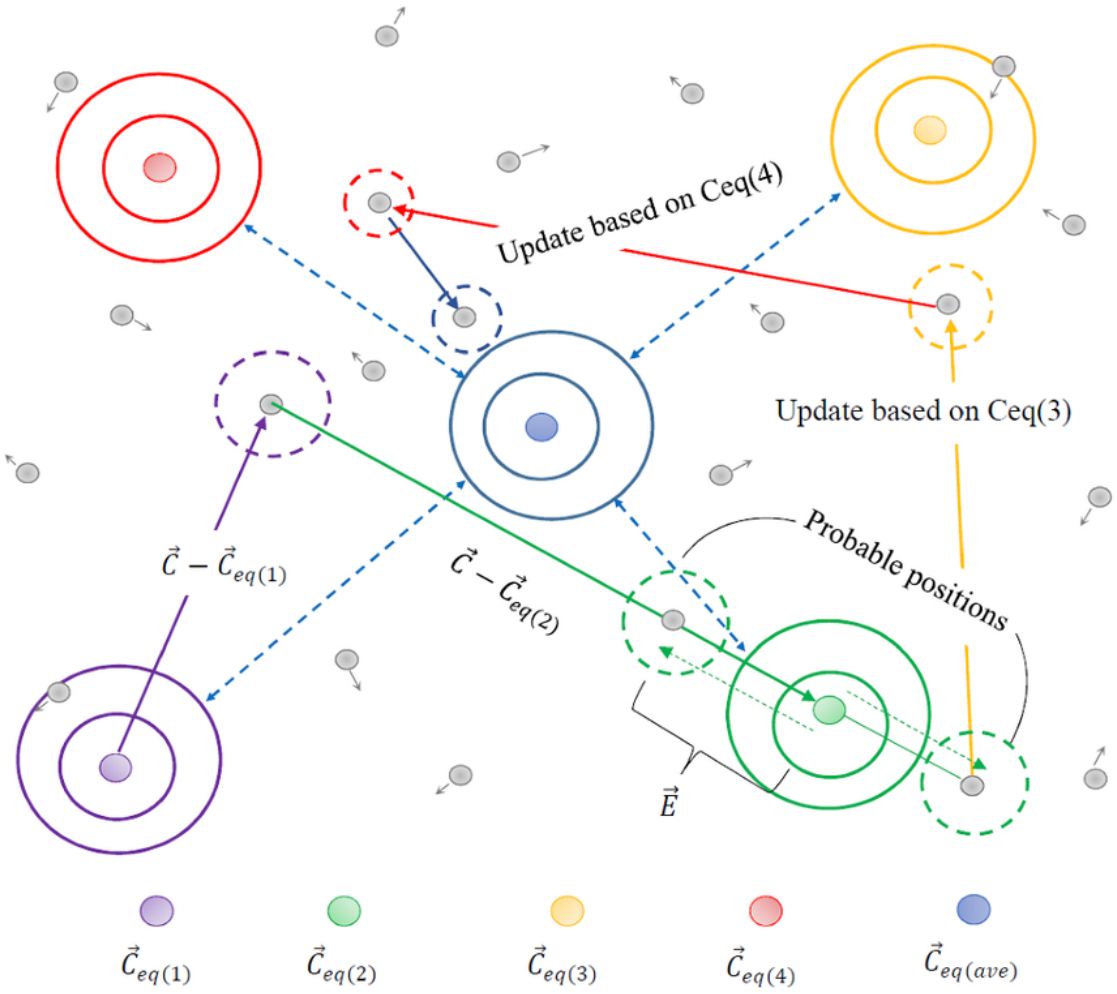

Fig. 3.6 Equilibrium candidates' collaboration in updating a particles' concentration in 2D dimensions [5].

$$C_i^{\text{initial}} = C_{\min} + \text{rand}_i \left(C_{\max} - C_{\min}\right) \tag{3.22}$$

where:

- $C_{initial}$ is the initial concentration vector of the i-th particle.
- C_{min}, C_{max} is the minimum bound and maximum bound of the i - th dimension's value.
- $rand_i$ is a random vector in the interval of [0,1].
- $i = (1 \to n)$, n is the number of particles as the population.

Afterward, the equilibrium pool vector is constructed as follows:

$$\vec{C}_{eq,pool} = \left\{\vec{C}_{eq(1)}, \vec{C}_{eq(2)}, \vec{C}_{eq(3)}, \vec{C}_{eq(4)}, \vec{C}_{eq(\text{ ave })}\right\} \tag{3.23}$$

where $C_{eq}(i)$ are five particles, nominated as equilibrium candidates. The first four candidates are particles that have been selected as having the best fitness value among the population, and the final $C_{eq}(ave)$ is the average vector of those first four particles.

In each iteration, the considering particle would update its concentration by choosing one candidate in the equilibrium pool in a uniform, random manner. Beside the equilibrium pool, particles are updated based on two more factors: Exponential Term (F) and Generation Rate (G) with their steps are as follows:

The calculation for the Exponential Term F:

$$\vec{F} = e^{-\vec{\lambda}(t-t_0)} \tag{3.24}$$

$$t = (1 - \frac{Iter}{Max\_iter})^{\left(a_2 \frac{Iter}{Max\_iter}\right)} \tag{3.25}$$

$$\vec{t_0} = \frac{1}{\vec{\lambda}} \ln\left(-a_1 \,\text{sign}(\vec{r} - 0.5)\left[1 - e^{-\vec{\lambda}t}\right]\right) + t \tag{3.26}$$

where:

- λ and r are random vectors in the interval of [0,1].
- $Iter$ and Max_{iter} are the current and the maximum number of iterations.
- a_1, a_2 equals 2 and 1, and they are controlling the exploration and exploitation ability respectively.

The Generation Rate G has the following formula:

$$\vec{G} = \vec{G}_0 e^{-\vec{\lambda}(t-t_0)} = \vec{G}_0 \vec{F} \tag{3.27}$$

$$\overline{G_0} = \overline{GCP}\left(\overline{C_{eq}} - \vec{\lambda}\vec{C}\right) \tag{3.28}$$

$$\overline{GCP} = \begin{cases} 0.5r_1 & r_2 \geq GP \\ 0 & r_2 < GP \end{cases} \tag{3.29}$$

where:

- G_0 is the initial value, λ is a decay constant.
- r_1, r_2 are random numbers in [0,1].
- GP is a constant and should be set to 0.5. based on the work of Faramarzi et al. [5].

In EOA, there's a memory saving mechanic, where the algorithm will compare the fitness value of particles in the current iteration to that of the previous iteration. The particle will be overwritten according to the best value out of these two particles. This implementation has proved to be useful for exploitation but increases the chance that EOA gets into local optima. Algorithm 4 outlines the six fundamental steps of the EOA algorithm.

In this thesis, we use the EOA algorithm to optimize the fusion weights of the VGG19 network for image fusion. We prioritize EOA over other established meta-heuristics, such as Particle Swarm Optimization (PSO) or Genetic Algorithms (GA), due to its unique "generation rate" mechanism which provides a superior balance between exploration (searching new areas) and exploitation (refining existing solutions). While algorithms like PSO often suffer from premature convergence to local optima in multimodal landscapes—a common characteristic of image fusion objective functions—EOA's physics-inspired dynamics maintain diversity in the population longer, preventing stagnation. Furthermore, EOA requires fewer tunable hyperparameters than GA, making it more robust and easier to adapt to the medical image fusion domain without extensive tuning. We will detail the implementation of the EOA algorithm in our contribution chapter.

Algorithm 4 EOA algorithm

Initialize the population of particles ($i = \overline{1,n}$)
Initialize maximum number of loops: l_{max}
Initialize parameters: $\psi_1 = 2$; $\psi_2 = 1$; $GP = 0.5$;
while $l < l_{max}$ **do**
  for *i=1:n* **do**
    Calculate fitness of ith particle
    if $F(H_i) < F(H_{eq(1)})$ **then**
      Replace $H_{eq(1)}$ with H_i and $F(H_{eq(1)})$ with $F(H_i)$;
    end
    else if $(F(H_i) > F(H_{eq(1)}))$ & $(F(H_i) < F(H_{eq(2)}))$ **then**
      Replace $H_{eq(2)}$ with H_i and $F(H_{eq(2)})$ with $F(H_i)$
    end
    else if $(F(H_i) > F(H_{eq(1)}))$ & $(F(H_i) > F(H_{eq(2)}))$ & $(F(H_i) < F(H_{eq(3)}))$ **then**
      Replace $H_{eq(3)}$ with H_i and $F(H_{eq(3)})$ with $F(H_i)$
    end
    else if $(F(H_i) > F(H_{eq(1)}))$ & $(F(H_i) > F(H_{eq(2)}))$ & $(F(H_i) > F(H_{eq(3)}))$ & $(F(H_i) < F(H_{eq(4)}))$ **then**
      Replace $H_{eq(4)}$ with H_i and $F(H_{eq(4)})$ with $F(H_i)$
    end
  end
  $H_{ave} = (H_{eq(1)} + H_{eq(2)} + H_{eq(3)} + H_{eq(4)})/4$;
  Create the equilibrium pool: $H_{eq(pool)} = (H_{eq(1)}, H_{eq(2)}, H_{eq(3)}, H_{eq(4)}, H_{eq(ave)})$;
  Accomplish memory saving (if $l > 1$) ;
  Assign $t = (1 - \frac{l}{l_{max}})^{\psi_2 \frac{l}{l_{max}}}$
  for *i=1:n* **do**
    One candidate was selected at random from the equilibrium pool (vector) ;
    Generate random vectors of ψ, h
    Calculate $F = \psi_1 sign(h - 0.5)[e^{-\psi t} - 1]$;
    Calculate generation rate Control Parameter: $GCP = \begin{cases} 0.5h_1 & h_2 \geq GP \\ 0 & h_2 < GP \end{cases}$
    Calculate $G_0 = GCP(H_{eq} - \psi H)$
    Calculate $G = G_0.F$
    Update concentrations $H = H_{eq} + (H - H_{eq}).F + \frac{G}{\psi V}(1 - F)$
  end
  $l = l + 1$
end

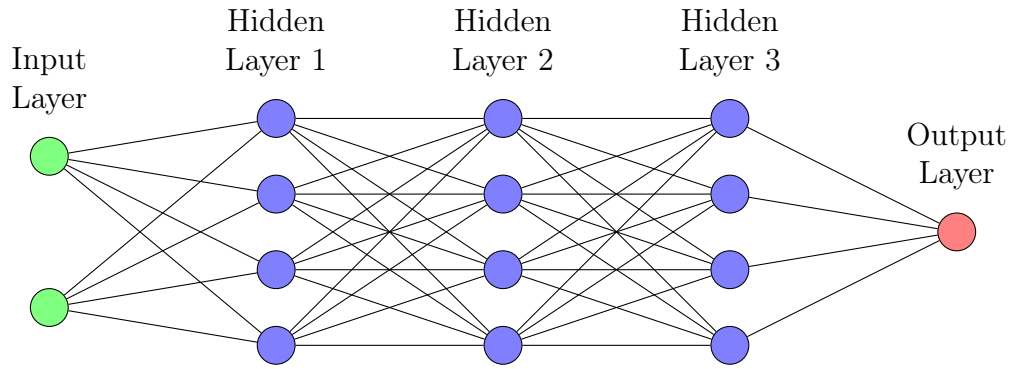

Fig. 3.7 Typical ANN Architecture with 3 hidden layers

3.3 Deep Learning

3.3.1 Fundamental Concepts

Deep learning is a subset of machine learning that bases on artificial neural networks (ANN) to model and solve complex problems. This type of model is inspired by the structure and function of the human brain, with interconnected layers of artificial neurons that can learn patterns and predict outcomes for a given input. In an ANN, each neuron in the network receives input signals, processes them using an activation function, and produces an output signal that is passed to other neurons in the network. By adjusting the weights and biases of the neurons, the network can learn to recognize patterns and make predictions based on the input data. Figure 3.7 shows a typical architecture of an ANN with 1 input layer, 3 hidden layers and 1 output layer.

The distinguishing characteristic separating deep learning from conventional ANNs lies in architectural depth—deep learning systems incorporate numerous stacked hidden layers. This hierarchical organization enables the automatic discovery of intricate data abstractions without requiring handcrafted feature engineering, conferring substantially greater representational capacity compared to shallow networks. Such multi-layered architectures excel at perceptual tasks spanning visual recognition, acoustic signal processing, and linguistic understanding. Within medical imaging contexts, deep convolutional networks have demonstrated substantial diagnostic utility for pathology detection, tumor delineation, and anatomical segmentation workflows.

3.3.2 Convolutional Neural Networks

Convolutional neural network is a special type of deep learning model that is designed to process spatial data, such as images and videos. CNNs are composed of multiple layers, including convolutional layers, pooling layers, and fully connected

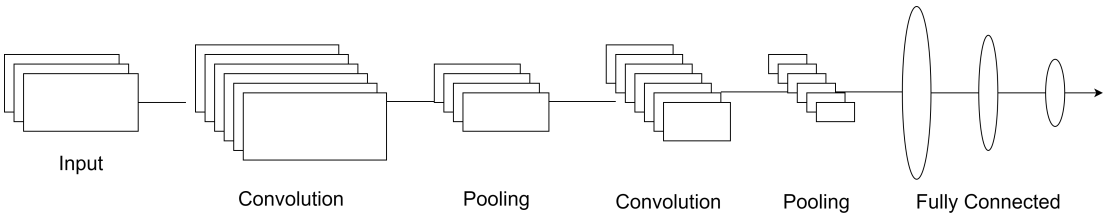

Fig. 3.8 Example of a Convolutional Neural Network Architecture, with Conv2D, Pooling, and Fully Connected Layers

layers. The convolutional layers use filters and convolution operator to extract features from the input data, while the pooling layers downsample the feature maps to reduce the computational complexity of the model. The fully connected layers combine the extracted features to make predictions based on the input data. Figure 3.8 shows a typical architecture of a CNN with convolutional layers, pooling layers, and fully connected layers.

Convolutional layer. Convolutional layer is the foundation building block of a CNN. It applies a set of filters to the input data to extract features. Each filter slides over the input data, performing element-wise multiplication and summation (the convolution operator) to produce a feature map. This process is repeated for every filter, producing multiple feature maps that represent different aspects of the input data. The convolution operator is designed to focus on local, spatial patterns in the input data, capturing features like edges, textures, and shapes. By stacking multiple convolutional layers, the CNN can learn visual representations of the input data, capturing complex patterns and relationships.

For a given input image I (of size $W \times H$) and a filter K (of size $m \times n$), the convolution output S can be mathematically defined as follows:

$$S(i,j) = (I*K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n), \forall i,j \in [1,W],[1,H] \quad (3.30)$$

Pooling layer. The feature maps generated by the convolutional layers are downsampled using the pooling layer. By decreasing the feature maps' spatial dimensions, it improves the computational efficiency of the model and reduces overfitting. Max pooling, which chooses the maximum value from a range of values in a specific area of the feature map, is the most popular pooling procedure. Additional pooling functions include sum pooling, which determines the total of the values in the local region, and average pooling, which determines the average value of the local region. By assisting the model in concentrating on the most significant features in the data, pooling layers improve the model's capacity for generalization.

Formally, the max pooling result M of a feature map F of size $W \times H$ can be defined as follows:

$$M(i,j) = \max_{m,n} F(i+m, j+n), \forall i,j \in [1,W],[1,H] \tag{3.31}$$

Fully connected layer. The last class of CNN layers, known as the fully connected layer, is where predictions are created by combining the extracted features. Like a conventional artificial neural network, this layer links every neuron in the previous layer to every neuron in the current layer. Weights are used to represent the connections between neurons, and they are modified throughout training to reduce the loss function.

The fully connected layer generates the model's final output using an activation function, such as sigmoid for binary classification tasks or softmax for multi-class classification tasks.

Formally, the output O of a fully connected layer with input X and weights W can be defined as follows:

$$O = \sigma(XW) \tag{3.32}$$

in which, σ is the chosen activation function.

Activation function. Activation functions are used in neural networks to introduce non-linearity into the model, allowing it to learn complex patterns and relationships in the data.

CNNs most commonly use the activation functions Sigmoid, Tanh, and ReLU (Rectified Linear Unit). ReLU is the most widely used activation function because deep neural network training can be accomplished with ease and efficacy. In order to help the model learn more quickly and get around the vanishing gradient problem, it replaces any negative values in the input with zero.

ReLU is visualzed in Figure 3.9 and mathematically defined as follows:

$$f(x) = \max(0, x) \tag{3.33}$$

3.3.3 Training Deep Learning Models

In conventional machine learning, a loss function is minimized by the use of optimization techniques like gradient descent to train models. Optimization is an important aspect of machine learning and deep learning, as it helps in finding the best possible solution to a problem. Optimization methods play a crucial role in training machine learning and deep learning models. These techniques help in adjusting the model parameters to minimize the loss function, thereby improving

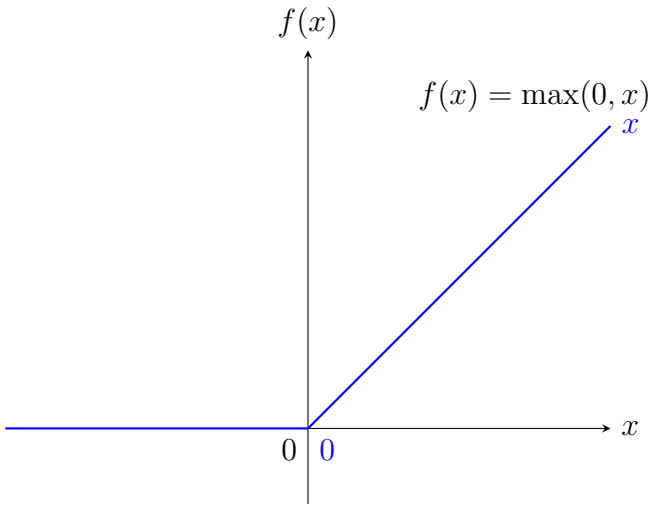

Fig. 3.9 ReLU function

the model's performance.

Gradient-Based Optimization

Gradient-based optimization is a family of optimization algorithms, aiming to minimize a specific function by adjusting parameters in the direction that reduces the value of the function. In deep learning, gradient optimization is mainly used in training deep networks by updating the model's parameters to minimize the loss function. The loss function quantifies the difference between the predicted output and the actual target, leading toward the best set of weights. By calculating the gradients of the loss function with respect to the model's parameters, gradient-based optimization algorithms enable the model to learn and improve over time.

Stochastic Gradient Descent (SGD) is a popular optimization algorithm in the Gradient-based optimization family, mainly by updating neural network parameters based on random examples. SGD is computationally efficient using mini-batches compared to traditional gradient descent methods, allowing for faster iterations and updates, widely used in various tasks and datasets [95]. It is computationally efficient but introduces higher variance and more giant fluctuation steps [96].

Given a function $f(\mathbf{w})$ to minimize, where $\mathbf{w}$ represents the parameters, the gradient descent update rule is defined as:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla f(\mathbf{w}_t) \tag{3.34}$$

where:

- $\mathbf{w}_t$ are the parameters at iteration t,
- η is the learning rate, a positive scalar,

Algorithm 5 Stochastic Gradient Descent

Input: Loss function $f(\mathbf{w}; \mathbf{x}_i, y_i)$, learning rate η, initial parameters $\mathbf{w}_0$, convergence threshold ϵ, maximum iterations T, training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$

Output Optimized parameters $\mathbf{w}^*$

Initialize $\mathbf{w} \leftarrow \mathbf{w}_0$
Set iteration counter $t \leftarrow 0$
repeat
    for each training example $(\mathbf{x}_i, y_i)$ in the dataset **do**
        Compute the gradient $\nabla f(\mathbf{w}; \mathbf{x}_i, y_i)$
        Update the parameters: $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla f(\mathbf{w}; \mathbf{x}_i, y_i)$
    end for
    Increment the iteration counter: $t \leftarrow t + 1$
until $\|\nabla f(\mathbf{w}; \mathbf{x}_i, y_i)\| < \epsilon$ or $t \geq T$ **return** $\mathbf{w}$

- $\nabla f(\mathbf{w}_t)$ is the gradient of the function at $\mathbf{w}_t$.

The SGD algorithm is summarized in the Algorithm 5.

However, the effectiveness of SGD depends heavily on choosing the correct hyperparameters, especially the learning rate. The learning rate determines the size of the steps taken during parameter updates, directly affecting the optimization process's convergence speed and overall performance. Finding the optimal learning rate and achieving convergence can be challenging, requiring careful hyperparameter tuning and experimentation to balance convergence speed and model performance.

Adam (Adaptive Moment Estimation). Adam is an adaptive learning rate optimization algorithm that combines the benefits of both AdaGrad and RMSprop. It computes individual adaptive learning rates for different parameters, allowing for more efficient and stable optimization. Adam adapts the learning rates based on the first and second moments of the gradients, providing a better and faster approach to train deep learning models [97], since it is less sensitive to hyperparameter tuning than traditional optimization algorithms.

The Adam algorithm is summarized in the Algorithm 6.

The model's predictions and the actual values in the training data are compared using the loss function to calculate the difference. In order to improve the model's accuracy and its ability to make predictions on fresh, unobserved data, training aims to minimize the loss function by adjusting the model's parameters.

A loss function F is used to measure the difference between the predicted output $\hat{y}$ and the actual target y. The loss function is defined as follows:

$$F(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{y}_i) \tag{3.35}$$

Algorithm 6 Adam Optimizer

Input: Loss function $f(\mathbf{w})$, learning rate α, initial parameters $\mathbf{w}_0$, first moment decay rate β_1, second moment decay rate β_2, small constant ϵ, maximum iterations T

Output Optimized parameters $\mathbf{w}^*$

Initialize $\mathbf{w} \leftarrow \mathbf{w}_0$
Initialize first moment vector $\mathbf{m} \leftarrow \mathbf{0}$
Initialize second moment vector $\mathbf{v} \leftarrow \mathbf{0}$
Set iteration counter $t \leftarrow 0$
repeat
    Increment the iteration counter: $t \leftarrow t + 1$
    Compute the gradient: $\mathbf{g}_t \leftarrow \nabla f(\mathbf{w}_{t-1})$
    Update biased first moment estimate: $\mathbf{m}_t \leftarrow \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1)\mathbf{g}_t$
    Update biased second moment estimate: $\mathbf{v}_t \leftarrow \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2)\mathbf{g}_t^2$
    Compute bias-corrected first moment estimate: $\hat{\mathbf{m}}_t \leftarrow \mathbf{m}_t/(1 - \beta_1^t)$
    Compute bias-corrected second moment estimate: $\hat{\mathbf{v}}_t \leftarrow \mathbf{v}_t/(1 - \beta_2^t)$
    Update parameters: $\mathbf{w}_t \leftarrow \mathbf{w}_{t-1} - \alpha\hat{\mathbf{m}}_t/(\sqrt{\hat{\mathbf{v}}_t} + \epsilon)$
until convergence criterion is met or $t \geq T$ **return** $\mathbf{w}_t$

Similar procedures are used to train deep learning models, however because of the large number of parameters and layers in the model, there is extra complexity in this stage. During the training phase, the input data is fed through the network, the loss is computed, and backpropagation is used to update the model's parameters. Using the backpropagation technique, the optimization algorithm may make the necessary parameter adjustments by calculating the gradient of the loss function with respect to each model parameter.

However, training these models can be challenging due to the large number of parameters (in an exponential scale with regard to the number of hidden layers and the number of neurons per hidden layer) and the need for a large amounts of training data. In image processing, for example, this process require millions of labeled images to achieve accurate object recognition ability. They also can overfit, which occurs when a model works well on training data but badly on new, unseen data. When a model is very complicated or when there is insufficient or noisy training data, overfitting may happen.

To overcome these difficulties and enhance generalization and performance, deep learning models frequently use methods like regularization, transfer learning, and optimization algorithms. Transfer learning is the process of fine-tuning a pre-trained model for a particular task using a smaller dataset after it has been trained on a larger dataset. In case of limited of labeled data available for the target task, this strategy is especially helpful. Large weights in the model are penalized by regularization approaches like L1 and L2 regularization, which help in reducing overfitting.

Additionally, very deep models employ further techniques such as skip-connections and residual connections to improve training and performance. It is important to clarify the relationship between these terms: "skip-connection" is a broad concept referring to any path that bypasses layers to feed information further down the network, whereas "residual connection" (introduced in ResNet) is a specific type of skip-connection that adds the input of a layer to its output ($x + F(x)$), forcing the layer to learn residual mappings. This distinction is crucial as residual connections specifically address the vanishing gradient problem in very deep networks, while general skip-connections (like in U-Net) are often used to recover spatial resolution. The next section will discuss more detail these methods.

3.3.4 Transfer Learning

Transfer learning uses parts of an existing, trained model so we don't have to build and train a completely new model from the beginning. Pre-trained models are typically trained on very large, well-known datasets used in computer vision. The knowledge (weights) these models gain can be applied to different computer vision problems. You can either use these models as they are to make predictions on new tasks, or you can incorporate them into the training of a new model. Using these pre-trained models in a new setup helps reduce the time it takes to train the new model and also leads to better performance on unseen data.

Transfer learning is especially handy when you don't have a lot of data to train your model. For instance, you can take the weights from a pre-trained model and use them as the starting point for the weights in your new model. The great thing about pre-trained models is that they're often versatile enough for many different real-world uses. For example, (1) for Image Classification: Models trained on large datasets like ImageNet, which has over 1,000 different categories, can be used for real-world image classification problems. If you're an insect researcher, you could take one of these models and fine-tune it to specifically classify various insects; (2) for Text Classification: When classifying text, you need to understand how words are represented numerically (word embeddings). While you could train these representations yourself, it often requires a lot of data and can take a long time. Instead, you can use existing pre-trained word embeddings, like GloVe, to significantly speed up your development process.

Fine-tuning is a machine learning method where you take a pre-trained model and adjust it for a new, specific task or dataset. You do this by training the model further on a smaller, new dataset, which helps it learn details and knowledge relevant to that particular area. This process ultimately improves the model's performance on the new task. When fine-tuning a model, we can avoid overfitting

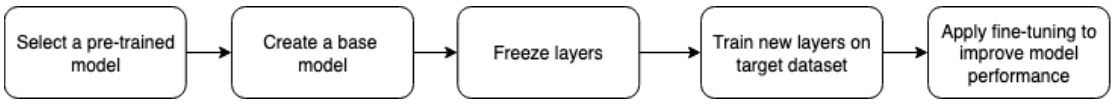

Fig. 3.10 The Main Steps of Transfer Learning.

by retraining all or part of it with a low learning rate. This small learning rate is crucial because it stops the model's internal adjustments (gradients) from changing too drastically, which could otherwise lead to poor performance. It's also smart to use a callback mechanism that automatically stops the training once the model stops getting better, further preventing overfitting.

Transfer learning isn't always the best solution. It may not be effective if the high-level features the pre-trained model learned aren't enough to tell apart the specific categories in your problem. For example, a pre-trained model might be great at recognizing a door, but it might not be able to tell if that door is open or closed. In situations like this, instead of relying on the high-level features, you can use the low-level features from earlier layers of the pre-trained network. This means you'll need to retrain more layers of the model or extract features from those earlier layers to get the performance you need.

When datasets aren't similar, the features learned from one don't transfer well to the other. Research, such as the paper by [98], explores this dataset similarity in more detail. What their work shows is that starting a network with pre-trained weights consistently leads to better performance than if you were to start with random weights. Removing layers from a pre-trained model is usually not a good idea for transfer learning. Doing so cuts down on the number of parameters the model can learn, making it more prone to overfitting. Plus, figuring out how many layers to remove without causing overfitting is a difficult and lengthy process.

As can be seen from the figure 3.10, the first step (1) is to select a pre-trained model we want to use for our specific task. One of the sources for downloading pre-trained deep learning models is Keras Applications (`https://keras.io/api/applications/`). The second step (2) is to an instance of the base model, choosing from architectures like ResNet or Xception. We can also opt to download its pre-trained weights. If we skip downloading the weights, we'll have to train the model from scratch using just the architecture. Keep in mind that these base models usually have more output units than our specific problem requires, so we'll need to remove that final output layer. Next in the third step (3), it's crucial to freeze the layers of the pre-trained model. If we don't, their weights will get reset, and we'll lose all the valuable knowledge the model already acquired. At that point, we might as well be training the model from scratch.

After that in step (4), we'll need to add new layers to the model that can be

trained. These new layers are essential because they'll take the existing features from the pre-trained model and adapt them to make predictions specifically for our new dataset. Remember, the pre-trained model was loaded without its original final output layer, so these new layers will bridge that gap. Once having newly added layers, we will train them on the new target dataset. Notice that the pre-trained model's original output probably won't match what our model needs. For instance, a model trained on ImageNet outputs 1,000 classes, but our model might only need two. In this situation, we'll need to train our model with a brand-new output layer. So, we'll add some new dense layers as we see fit, but most crucially, a final dense layer with the exact number of units that corresponds to the number of outputs our model expects.

Finally in the step (5), we will improve the model performance using fine-tuning which involves unlocking (unfreezing) the entire base model, or just a portion of it, and then retraining the complete model on our entire dataset using a very small learning rate. This low learning rate helps the model perform better on the new data while also preventing it from overfitting. We need to use a very low learning rate because we're working with a large model and a small dataset, a combination that often leads to overfitting. After making any changes, remember to recompile the model so those changes take effect. That's because a model's behavior gets "locked in" every time we call the compile function. So, if we want to alter how it behaves, we have to compile it again. The final step is to retrain the model, keeping a close eye on it with callbacks to prevent overfitting. Figure 3.11 shows an example of using fine-tuning technique to improve performance of deep learning models. The figure is obtained from the website (https://pyimagesearch.com/wp-content/uploads/2019/06/fine_tuning_keras_freeze_unfreeze.png).

Several sources of pre-trained models can be counted as Keras Applications (https://keras.io/api/applications/) or TensorFlow Hub (https://www.tensorflow.org/hub). Keras Applications offers many pre-trained models, each coming with its own set of weights. When we download a model from Keras Applications, its weights are automatically downloaded and saved to /.keras/models/. These models are designed for image-related tasks, such as initializing the VGG architecture with weights pre-trained on ImageNet. TensorFlow Hub provides a collection of pre-trained machine learning models, such as BERT and Faster R-CNN, that we can easily fine-tune and deploy. We can integrate these models into our projects with just a few lines of code.

Pre-trained models can be used for prediction, feature extraction and model fine-tuning. For prediction, it can be the case that the downloaded pre-trained model (e.g. ResNet50, VGG19, etc.) can be used immediately to classify the classes into their classes. For feature extraction, the output of the layer before the

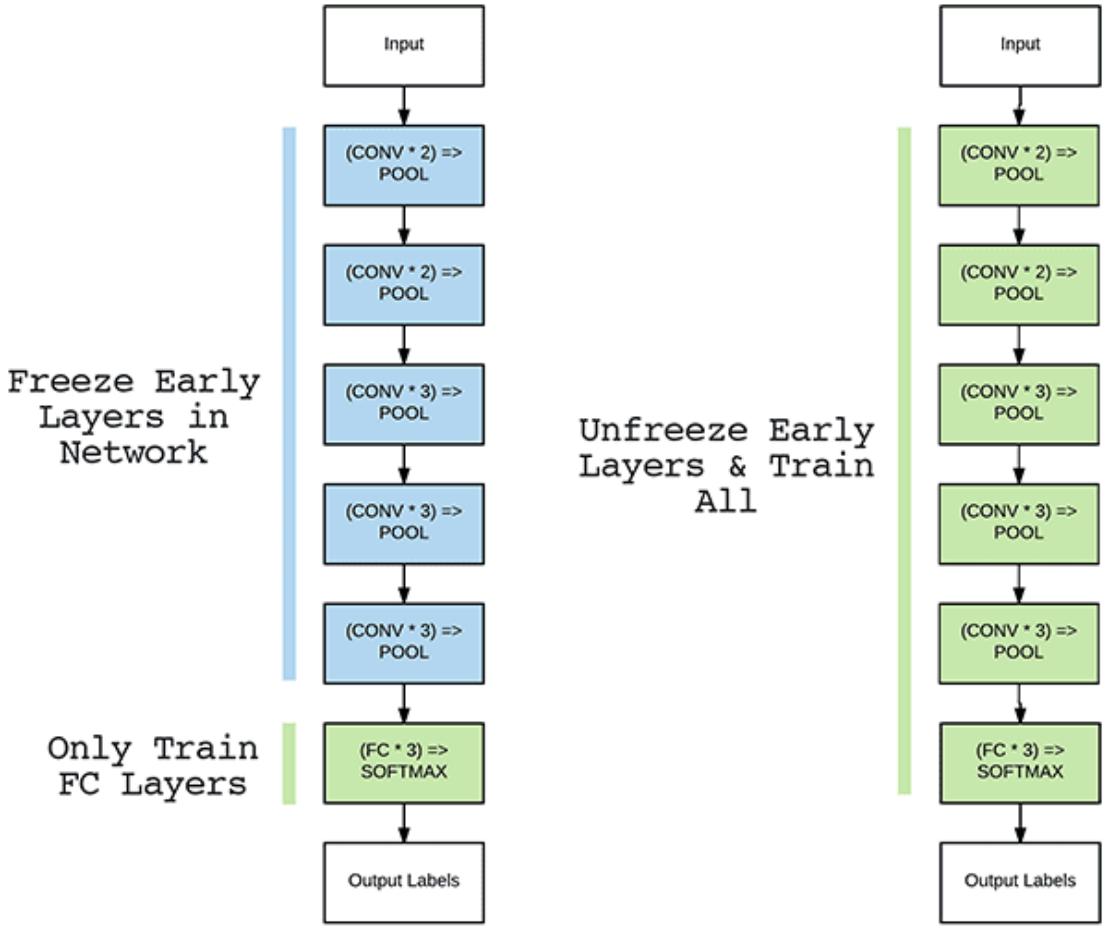

Fig. 3.11 Example of fine-tuning deep learning model.

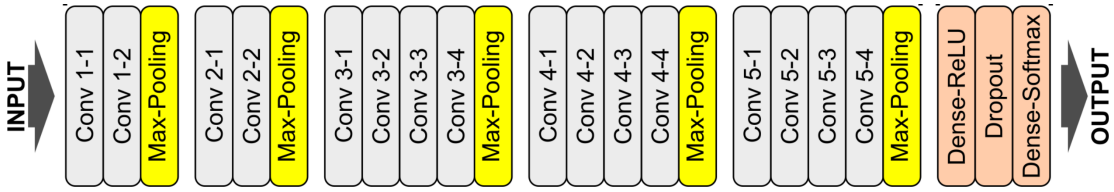

Fig. 3.12 VGG19 network architecture [6]

final layer is fed as input to a new model. This is to be sure that the pre-trained model, or a part of it is used to preprocess the images for getting the important features. These features are then passed to a new classifier without the need to retrain the base model. It is noticed that the final part of the pretrained model is usually specific to its own dataset. For this, we have to build the last part of our model to fit with our target dataset.

Once our new classifier is constructed, we can use fine-tuning to improve the accuracy of our new classifier. For this, we unfreeze the classifier, or part of it, and retrain it on the new target dataset with a low learning rate. The fine-tuning step is important in order to make the feature representations from the base model, which are obtained from the pre-trained model become more relevant to our specific tasks with new target dataset. Also, we can utilize the weights from the pre-trained model to initialize the weights in the new model. However, the option depends specifically on the problem we are solving.

3.3.5 VGG19 Model

The VGG19 model is a widely-used convolutional neural network architecture that was developed by Simonyan et al. [99]. The deep architecture of the VGG19 model, which is part of the VGG family of models, is characterized by its sequentially organized of several convolutional layers, pooling layers, and fully connected layers. The authors proposed enhancing the network depth and using smaller kernel sizes, such as 3×3 and 1×1 filters, to greatly improve image categorization on the ImageNet dataset.

The VGG19 model, in particular, has 19 layers, including 16 convolutional layers and 3 fully connected layers. The convolutional layers in the VGG19 model use small 3x3 filters with a stride of 1 and a padding of 1. Figure 3.12 shows the architecture of the VGG19 model. It begins with an input layer, followed by a series of convolutional blocks. Each block consists of multiple 3×3 convolutional layers with ReLU activation functions, and each block ends with a max-pooling layer (highlighted in yellow) that reduces the spatial dimensions of the feature maps. Specifically, the first block includes two convolutional layers (Conv 1-1, Conv 1-2) followed by max-pooling. The second block mirrors this structure with

two more convolutional layers (Conv 2-1, Conv 2-2) and another max-pooling. The third block expands to four convolutional layers (Conv 3-1 through Conv 3-4), followed by pooling. This pattern continues into the fourth and fifth blocks, each consisting of four convolutional layers (Conv 4-1 to 4-4 and Conv 5-1 to 5-4), both ending with pooling operations. Following these convolutional stages, the output feature maps are flattened and passed through a series of fully connected layers: a Dense-ReLU layer, a Dropout layer (for regularization), and finally a Dense-Softmax layer, which produces class probabilities for classification tasks.

In this thesis, we use a pretrained VGG19 model. It has been trained on the large-scale ImageNet dataset, which contains over 1.2 million labeled images across 1,000 object categories. These categories range from animals and vehicles to everyday items and scenes. During training on ImageNet, the network was optimized to minimize classification error across the full set of 1,000 classes, resulting in learned filters that are highly effective at capturing visual features such as edges, textures, shapes, and object parts. The final fully connected layer outputs a 1,000-dimensional probability vector corresponding to the predicted class probabilities.

Justification for selecting VGG19: The choice of VGG19 for this thesis, despite the existence of newer and more efficient architectures like ResNet or EfficientNet, is deliberate and grounded in the specific requirements of image fusion. While ResNet excels at semantic classification through its deep residual paths, VGG19's architecture—composed of a simple, uniform chain of convolutional blocks—has been empirically shown to retain superior low-level texture information. Newer architectures often aggressively downsample or abstract away these fine-grained details in favor of high-level semantic abstraction. For medical image fusion, where preserving the precise "texture" of tissue and the sharp edges of lesions is critical, the feature maps of VGG19 (especially from early to mid-level layers) provide a richer and more spatially accurate representation than those of deeper, more abstract models.

We use the convolutional layers of VGG19 to extract features for the image fusion task, which will be described in our contribution chapter.

3.3.6 Vision Transformers (ViT)

While Convolutional Neural Networks (CNNs) have been the dominant architecture for computer vision tasks, the Transformer architecture, originally designed for Natural Language Processing (NLP) [100], has recently shown remarkable success in vision tasks. The Vision Transformer (ViT) [7] applies the standard Transformer encoder directly to images with minimal modifications.

Fig. 3.13 Vision Transformer (ViT) architecture [7]. (Image to be added)

Fig. 3.14 Swin Transformer architecture [8]. (Image to be added)

In ViT, an image is split into fixed-size patches, which are linearly embedded and treated as tokens, similar to words in NLP. Position embeddings are added to these patch embeddings to retain spatial information. The sequence of embeddings is then processed by a series of Transformer encoder layers, which consist of Multi-Head Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks. The self-attention mechanism allows the model to capture long-range dependencies across the entire image, unlike the local receptive fields of CNNs.

ViT has demonstrated that pure Transformer architectures can outperform state-of-the-art CNNs on large-scale image classification benchmarks when trained on sufficient data. However, ViT requires a massive amount of training data to generalize well and has a quadratic computational complexity with respect to the image size, which limits its application to high-resolution images.

3.3.7 Swin Transformer

To address the limitations of ViT, particularly the computational cost and the lack of hierarchical structure, Liu et al. [8] proposed the Swin Transformer. Swin Transformer introduces a hierarchical architecture with shifted windows, which allows it to serve as a general-purpose backbone for various vision tasks, including image classification, object detection, and semantic segmentation.

Hierarchical Feature Maps. Unlike ViT, which maintains a constant feature map size, Swin Transformer constructs hierarchical feature maps by merging image patches in deeper layers. This hierarchical structure is similar to that of CNNs (e.g., VGG, ResNet), enabling the model to capture features at different scales.

Shifted Window Attention. Swin Transformer computes self-attention within local windows to reduce computational complexity from quadratic to linear with respect to image size. To enable cross-window connections, it employs a shifted window partitioning strategy between consecutive layers. In one layer, the image is partitioned into non-overlapping windows. In the next layer, the window partitioning is shifted, allowing information to propagate across window boundaries.

The Swin Transformer architecture consists of four stages. The input image is first split into non-overlapping patches. In Stage 1, a linear embedding layer projects the features to an arbitrary dimension. Several Swin Transformer blocks

with modified self-attention (W-MSA and SW-MSA) are applied. In subsequent stages, patch merging layers reduce the number of tokens and increase the feature dimension, creating a hierarchical representation. This design makes Swin Transformer highly efficient and effective for dense prediction tasks and high-resolution images.

3.3.8 Model Soups

Ensemble learning is a powerful technique to improve model performance by combining the predictions of multiple models. However, traditional ensembles require running multiple models at inference time, which significantly increases computational cost and latency. To overcome this, Wortsman et al. [101] introduced "Model Soups," a method for averaging the weights of multiple fine-tuned models into a single model.

Weight Averaging. The core idea of Model Soups is that fine-tuning a pre-trained model with different hyperparameter configurations (e.g., learning rate, data augmentation, training duration) often leads to models that lie in the same low-loss basin of the loss landscape. Averaging the weights of these models can result in a solution that is more robust and generalizes better than any individual model.

Mathematically, let $\theta_1, \theta_2, ..., \theta_k$ be the weights of k fine-tuned models. The "soup" model θ_{soup} is obtained by:

$$\theta_{soup} = \frac{1}{k} \sum_{i=1}^{k} \theta_i \tag{3.36}$$

Unlike traditional ensembles that average the *outputs* (predictions), Model Soups average the *parameters* (weights). This means the final model has the same architecture and inference cost as a single model, but with the performance benefits of an ensemble.

There are two main strategies for creating model soups:

- **Uniform Soup:** Averages the weights of all fine-tuned models.
- **Greedy Soup:** Iteratively adds models to the soup only if they improve performance on a held-out validation set. This approach ensures that the soup does not degrade by including poor-performing models.

In this thesis, we utilize Model Soups in the standard within-backbone setting (averaging multiple fine-tuned instances of the same architecture) to obtain a single deployable network, and we combine different architectures (e.g., Swin

Transformer and VGG19) via prediction-level ensembling to leverage complementary inductive biases.

3.4 Evaluation Metrics

In this thesis, we employ distinct sets of evaluation metrics for the two main tasks: medical image fusion and medical image classification.

3.4.1 Image Fusion Metrics

We choose to use a variety of evaluation metrics to compare multiple decomposition methods quantitatively. Feature Mutual Information (FMI), Entropy (EN), Objective image fusion performance measure ($Q^{AB/F}$) [102], Overall Cross Entropy (OCE), Visual Information Fidelity for Fusion (VIFF), Average Light Intensity (Q_{ALI}), Contrast Index (Q_{CI}), Average Gradient (Q_{AG}), the Piella metrics Qw and Qe [103] are some of these measurements.

Average Light Intensity (Q_{ALI}). This metric measures the average brightness of the fused image. It is calculated as the mean of the pixel intensities:

$$Q_{ALI} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} F(i,j) \tag{3.37}$$

where $F(i,j)$ is the pixel intensity of the fused image at position (i,j), and $M \times N$ is the size of the image.

Contrast Index (Q_{CI}). This metric evaluates the local contrast of the fused image. A higher value indicates better contrast. It can be defined as:

$$Q_{CI} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (F(i,j) - \mu_F)^2 \tag{3.38}$$

where μ_F is the mean intensity of the fused image.

Average Gradient (Q_{AG}). This metric reflects the clarity and texture information of the image. A larger average gradient implies that the image contains more detailed information (edges and textures).

$$Q_{AG} = \frac{1}{(M-1)(N-1)} \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} \sqrt{\frac{(F(i+1,j) - F(i,j))^2 + (F(i,j+1) - F(i,j))^2}{2}} \tag{3.39}$$

The overall cross entropy, or OCE, indicates how the fused image differs from

the input images. OCE is calculated as follows:

$$\mathrm{OCE}\left(\mathrm{I_A}, \mathrm{I_B}, \mathrm{F}\right) = \left(\mathrm{CE}\left(\mathrm{I_A}, \mathrm{F}\right) + \mathrm{CE}\left(\mathrm{I_B}, \mathrm{F}\right)\right)/2$$

where I_A, I_B are the input images of different modalities, F is the fused image, $\mathrm{CE}\left(\mathrm{I_A}, \mathrm{F}\right)$ and $\mathrm{CE}\left(\mathrm{I_B}, \mathrm{F}\right)$ is the cross entropy of the input images with the fused image.

Entropy (EN): Consider a grayscale image.

$$H = -\sum_{i=0}^{255} p_i \log_2 p_i \tag{3.40}$$

Objective image fusion performance measure ($\mathrm{Q^{AB/F}}$): For $N \times M$ size images which having $Q^{AF}(n,m)$ and $Q^{BF}(n,m)$, a normalised weighted performance metric $Q_P^{AB/F}$ of a given fusion process P that operates on images A and B, and produces F is obtained as follows:

$$Q_{\underline{p}}^{AB/F} = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M} Q^{AF}(n,m)w^A(n,m) + Q^{BF}(n,m)w^B(n,m)}{\sum_{i=1}^{N}\sum_{j=1}^{M}\left(w^A(i,j) + w^B(i,j)\right)}$$

where $Q^{AF}(n,m)$ and $Q^{BF}(n,m)$ are the Edge information preservation values

Feature Mutual Information (FMI):

$$p_{FA}(x,y,z,w) = \theta h_L^{FA}(x,y,z,w) + (1-\theta)p_F(x,y)\cdot p_A(z,w)$$

where $\theta = \rho_\theta^{FA}/\rho_L^{FA}$ and ρ_L^{FA} is the correlation coefficient. These equations are also valid for the joint distribution between B The amount of feature information, which F contains about A and B

$$I_{FA} = \sum_{f,a} p_{FA}(x,y,z,w)\log_2 \frac{p_{FA}(x,y,z,w)}{p_F(x,y)\cdot p_A(z,w)}$$

$$I_{FB} = \sum_{f,b} p_{FB}(x,y,z,w)\log_2 \frac{p_{FB}(x,y,z,w)}{p_F(x,y)\cdot p_B(z,w)}$$

Eventually, the FMI metric is:

$$FMI_F^{AB} = I_{FA} + I_{FB}$$

Overall Cross Entropy (OCE) quantifies divergence between source and fused outputs, with minimal values signaling optimal information transfer. Entropy (EN) reflects the distributional complexity of pixel intensities, serving as a proxy for information content. Feature Mutual Information (FMI) captures the statis-

tical coupling between inputs and result, revealing the degree of informational synthesis. The Objective Image Fusion Performance Measure ($Q^{AB/F}$) evaluates edge-preservation and saliency retention. Visual Information Fidelity for Fusion (VIFF) provides a perceptually-weighted, objective assessment of multi-resolution fusion efficacy.

3.4.2 Classification Metrics

For the medical image classification tasks, we utilize standard performance metrics derived from the confusion matrix, including Accuracy, Precision, Recall (Sensitivity), Specificity, F1-Score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

Confusion Matrix. A confusion matrix is a table used to evaluate the performance of a classification model. For a binary classification problem (e.g., COVID-19 vs. Normal), it consists of four values:

- True Positive (TP): The number of positive cases correctly classified as positive.
- True Negative (TN): The number of negative cases correctly classified as negative.
- False Positive (FP): The number of negative cases incorrectly classified as positive (Type I error).
- False Negative (FN): The number of positive cases incorrectly classified as negative (Type II error).

Accuracy. This primary metric reflects the global fidelity of the classifier, representing the ratio of correct diagnostic decisions (both presence and absence of pathology) to the total number of screened cases.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.41}$$

Precision. Also referred to as Positive Predictive Value (PPV), this index evaluates the trustworthiness of a positive alert. High precision implies that when the system flags a patient as COVID-positive, it is highly likely to be a true infection rather than a false alarm.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.42}$$

Recall (Sensitivity). Sensitivity is critical in medical screening as it measures the system's ability to detect all positive cases. A high recall score indicates a low

rate of missed diagnoses (False Negatives), which is paramount for containing infectious diseases.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.43}$$

Specificity. This metric assesses the model's competence in correctly identifying healthy or non-COVID cases. High specificity prevents the healthcare system from being overwhelmed by healthy patients wrongly classified as needing care.

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{3.44}$$

F1-Score. The F1-Score is the harmonic mean of Precision and Recall. It provides a single metric that balances both concerns, which is particularly useful for imbalanced datasets.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.45}$$

AUC-ROC. The Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (Recall) against the False Positive Rate ($1 - \text{Specificity}$) at various threshold settings. The Area Under the Curve (AUC) represents the degree or measure of separability. An AUC of 1.0 indicates a perfect model, while 0.5 indicates a model with no discrimination capacity.

3.5 Chapter Summary

In this chapter, we present the fundamental background that we use to implement the contributions of the thesis. In detail, we first present the image processing techniques to convert color space, enhance contrast of the input images, extract detail features of the input images. After that we describe the basic concepts of metaheuristic optimization where we apply the Equilibrium Optimization Algorithm for the fusion of the base layers in our medical image fusion pipeline. Finally, we present the necessary deep learning concepts and techniques that we use as a basic for the fusion of detail layers in our medical image fusion using deep learnign and transfer learning techniques.

CHAPTER 4

Contribution 1: Medical Image Fusion via Hybrid Transfer Learning and Equilibrium Optimization

Medical image fusion represents a critical challenge in computational radiology: the synthesis of multimodal information into a coherent visual representation. The fundamental objective is to integrate the high-resolution anatomical structures from Magnetic Resonance Imaging (MRI) with the functional metabolic data from Positron Emission Tomography (PET). This integration is not merely a superimposition task but a complex optimization problem requiring the resolution of the **spectral–spatial trade-off**. Traditional methods often prioritize one aspect at the expense of the other—either preserving texture details while distorting color (spectral distortion) or maintaining color fidelity while blurring fine edges (spatial degradation).

In this chapter, we address this dichotomy by proposing a **hybrid two-scale architecture**. Our approach treats image fusion as a dual-domain problem:

1. **High-Frequency Domain (Texture):** We leverage the inductive bias of Deep Convolutional Neural Networks (specifically a modified VGG19) to extract and preserve complex edge features, treating texture fusion as a **feature selection** problem.

2. **Low-Frequency Domain (Intensity):** We formulate the regulation of global contrast and luminance as a continuous **optimization** problem, solved via the physics-inspired Equilibrium Optimization Algorithm (EOA).

By decoupling these domains, we demonstrate a system that simultaneously preserves the diagnostic utility of anatomical boundaries and the functional precision of metabolic indicators.

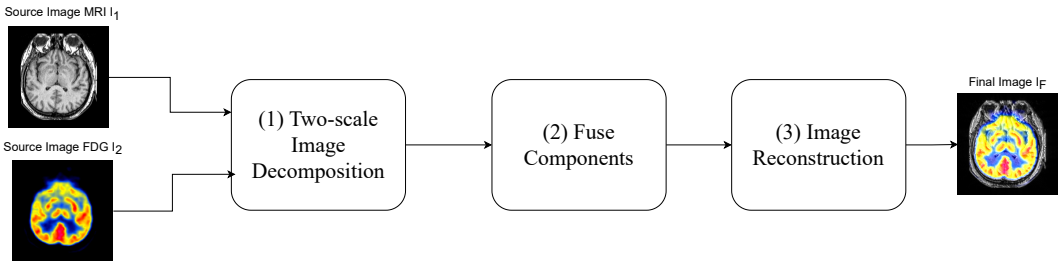

Fig. 4.1 Schematic representation of the proposed medical image fusion framework.

Chapter Organization

The remainder of this chapter details the architectural components of this framework:

- **Section 4.1** establishes the conceptual pipeline and the rationale for transfer learning in medical imaging.
- **Section 4.2** defines the mathematical basis for the Hybrid Two-Scale Decomposition layer (FFT-Kirsch).
- **Section 4.3** details the "Feature Refinement Model," utilizing VGG19 for detail injection.
- **Section 4.4** presents the "Adaptive Base Fusion" strategy, deploying EOA to optimize luminance weighting.
- **Section 4.5** integrates these modules into a cohesive pipeline and validates the system against current state-of-the-art benchmarks.

4.1 Conceptual Framework of the Fusion Pipeline

The architectural paradigm adopted in this study aligns with the established "Decomposition-Fusion-Reconstruction" workflow [104], a systematic approach favored for its flexibility in handling multi-modal data. The schematic representation of this pipeline is depicted in Figure 4.1.

4.1.1 Phase 1: Multi-Scale Signal Decomposition

The initial phase, **Image Decomposition**, is critical for disaggregating the complex information contained within medical images. The primary objective is to separate the raw input signal into distinct spectral bands:

1. **Base Layers (Low-Frequency):** These components encapsulate the gross anatomical structure and the global illumination profile. They represent the high-energy, slowly varying continuum of the image.

2. **Detail Layers (High-Frequency):** These components capture the transient signal variations, corresponding to edges, fine textures, and boundary definitions—features that often hold the most diagnostic significance.

By segregating these components, we can apply targeted processing strategies: preserving the energy distribution of the base layer while simultaneously enhancing the saliency of the detail layer. This avoids the pitfalls of direct spatial fusion, which often conflates contrast information with texture, leading to artifacts.

4.1.2 Phase 2: Component-Specific Fusion Strategies

The core synthesis occurs in the **Fusion Phase**, where complementary information from source modalities is integrated. Recognizing the distinct statistical properties of the decomposed layers, we employ a bifurcated strategy:

- **Adaptive Base Fusion:** The low-frequency components dictate the visual naturalness and contrast. A rigid averaging rule often results in contrast washout. Instead, we approach this as an optimization task, seeking a weighted combination that maximizes information retention and contrast consistency.

- **Saliency-Driven Detail Fusion:** High-frequency features are sparse and spatially localized. To prevent the attenuation of critical diagnostic markings (e.g., tumor margins), we utilize a feature-selection mechanism driven by deep learning. This ensures that only the most structurally significant details from each modality are transferred to the fused output.

4.1.3 Phase 3: Image Reconstruction

The final **Reconstruction Phase** serves as the deterministic inverse of the decomposition step. It synthesizes the processed base and detail components back into the spatial domain. The fidelity of this step is essential; the reconstruction algorithm must ensure perfect alignment and summation of the frequency bands to yield a composite image that is free from ringing artifacts and ready for clinical interpretation.

4.2 Two-Scale Image Decomposition Framework

4.2.1 Evaluation of Decomposition Paradigms

The selection of an appropriate decomposition method is pivotal, as it dictates the quality of the features available for fusion. Current methodologies can be broadly categorized into three paradigms, each presenting distinct advantages and limitations:

- **Spatial Filtering Approaches:** Techniques such as Gaussian or Bilateral filtering are computationally inexpensive and intuitive. However, they frequently suffer from a lack of directional sensitivity, which can result in the attenuation of diagonal edges and fine textures vital for medical diagnosis.
- **Multi-Resolution Analysis (MRA):** Methods like Pyramids and Wavelets offer superior frequency localization. Following this, however, is the risk of introducing "ringing" artifacts (Gibbs phenomenon) around high-contrast boundaries, potentially mimicking or obscuring pathological features.
- **Deep Learning-Based Decomposition:** While promising, fully learning-based separation often operates as a "black box," lacking the theoretical transparency required to guarantee the preservation of specific clinical features across diverse modalities.

Addressing these trade-offs, this research implements a **hybrid spectral–spatial decomposition**. By integrating Fourier-domain filtering with spatial gradient operators, we aim to combine the global energy preservation of spectral methods with the local edge sensitivity of spatial derivatives, mitigating artifacts while maximizing detail retention.

4.2.2 The FFT-Kirsch Decomposition Algorithm

Our specific implementation, the Two-Scale Image Decomposition (TSID), systematically partitions the input image I into a structural Base Layer (I^b) and a textural Detail Layer (I^d).

4.2.2.1 Base Layer Extraction via Spectral Filtering

The Base component encapsulates the low-frequency "DC" energy of the image, representing gross anatomy and illumination. To isolate this, we employ the Fast Fourier Transform (FFT). By transforming the image into the frequency domain

and applying a Low-Pass Filter (LPF), we can cleanly suppress high-frequency noise without the "halo" artifacts often associated with large spatial kernels.

$$I^b = \mathcal{F}^{-1}(LPF(\mathcal{F}(I))) \tag{4.1}$$

The inverse transform $\mathcal{F}^{-1}$ returns the smoothed structural information to the spatial domain, providing a robust foundation for the fusion process.

4.2.2.2 Detail Layer Extraction via Kirsch Operators

The residual high-frequency information often contains noise alongside true edges. Simple subtraction is insufficient. Instead, we employ the **Kirsch operator**, a non-linear edge detector noted for its directional robustness. Unlike simple gradient operators (e.g., Sobel) that are biased towards cardinal directions, the Kirsch operator utilizes a bank of eight rotatable masks ($K_0 \dots K_7$) to probe for intensity transitions in 45° increments.

The Detail Layer is thus constructed by identifying the maximal directional derivative at each pixel:

$$\text{Edge Magnitude} = \max_{i=0..7}(I * K_i) \tag{4.2}$$

This approach ensures that the extracted detail component highlights structurally significant boundaries—such as tissue interfaces and lesions—while remaining relatively insensitive to isotropic background noise. This high-fidelity detail map serves as the ideal input for our subsequent deep learning-based saliency analysis.

4.2.3 Benchmark Dataset for Validation

To rigorously validate the proposed decomposition and fusion pipeline, we utilize **The Whole Brain Atlas** [105]. This dataset is selected as the standard benchmark due to its high-fidelity registration of multi-modal pairs (MRI/PET, MRI/SPECT), providing a geometrically consistent ground truth essential for evaluating the preservation of both spectral and spatial information.

4.3 Deep High-Frequency Fusion via Domain-Adapted VGG19

4.3.1 Rationale for Deep Feature Fusion

High-frequency detail layers contain edges, fine textures, and boundary cues that are critical for diagnosis. Simple pixel-wise rules (e.g., max selection) are blind to

semantics and can amplify noise or sensor artifacts.

We therefore rely on deep feature representations. The key idea is to base fusion on *feature saliency* rather than raw brightness, so that clinically meaningful structures receive higher weight than spurious high-intensity noise. Mapping detail layers into a learned feature space allows the fusion rule to reflect structural importance instead of pixel magnitude alone.

4.3.2 The TL_VGG19 Architecture: From Natural to Medical Domain

We propose **TL_VGG19**, a VGG19 variant adapted for medical texture analysis through transfer learning. VGG19's ImageNet pretraining provides strong generic filters, but it is not tailored to grayscale radiology textures.

Domain Adaptation Strategy: We repurpose the classifier into a medical feature extractor using three practical steps:

1. **Decapitation:** We remove the fully connected classifier (1000 classes), yielding a fully convolutional backbone that outputs spatial feature maps at arbitrary resolution.

2. **Shallow Fine-Tuning:** We freeze deeper layers and fine-tune only the early layers of each block ($Conv1\_1 \ldots Conv5\_1$). These act like oriented filters; tuning them adapts edge/texture detectors to MRI and PET/SPECT noise patterns.

3. **Targeted Re-training:** We retrain on multi-modality medical data so the feature distribution shifts from natural scenes to radiology textures.

4.3.2.1 Operationalizing Transfer Learning

During training, we attach a temporary 4-class head (MRI/CT/PET/SPECT) to guide fine-tuning using the dataset in Table 4.1. Only the early blocks are updated. This preserves generic visual primitives while improving sensitivity to modality-specific textures.

4.3.2.2 Hierarchical Feature Pyramid Extraction

We leverage the TL_VGG19 backbone to construct a multi-level feature pyramid for the input detail layers L_{HF}^{MRI} and L_{HF}^{PET}. We tap into the network at specific checkpoints—Input, $Conv1\_1$, $Conv2\_1$, $Conv3\_1$, $Conv4\_1$, and $Conv5\_1$—to capture a spectrum of visual information. The initial layers are sensitive to geometric precision, while deeper blocks encode abstract semantic patterns. To

convert these high-dimensional tensors into usable 2D attention maps, we calculate the channel-wise L1 magnitude. This operation compresses the depth dimension, representing the aggregate activation intensity at each spatial location.

Since the pooling operations in VGG progressively reduce spatial dimensions, a subsequent refinement step is necessary to realign these maps with the original image size.

4.3.2.3 Saliency Refinement via Local Energy

To recover fine localization, we modulate the saliency maps with a Local Energy (LE) prior. LE highlights textured regions and down-weights flat areas.

For a raw feature map F, the Local Energy map E_F is computed by summing the squared intensities within a local kernel $W(u, v)$:

$$E_F(i,j) = \sum_{u=0}^{k-1} \sum_{v=0}^{k-1} W(u,v) F^2(i+u, j+v) \tag{4.3}$$

The refined feature map, F_{LE}, is produced via a soft-attention mechanism using the Hadamard product:

$$F_{LE} = F \odot E_F \tag{4.4}$$

This acts as a spatial gate that keeps confident features and attenuates weak responses. We then upsample the maps via bilinear interpolation to match the input resolution.

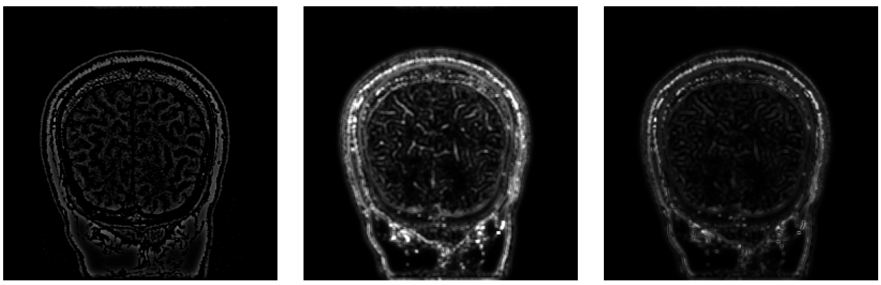

Fig. 4.2 Visualizing Feature Refinement: The Local Energy term functions as a spatial attention gate, significantly enhancing the definition of structural boundaries.

4.3.3 The FRM_VGG19 Fusion Algorithm

The proposed Feature Refinement Model (FRM_VGG19) orchestrates the synthesis of detail layers using the extracted deep features. The logic follows a "Winner-Take-All" strategy guided by semantic saliency, as detailed in Algorithm 7.

Algorithm 7 FRM_VGG19 Fusion Strategy

Input: Detail Layers $(L_{HF}^{MRI}, L_{HF}^{PET})$, TL_VGG19 Model

Output: Fused Detail Layer F_{HF}

Phase 1: Multi-Scale Extraction

Feed the inputs into the TL_VGG19 backbone. Retrieve feature tensors from the defined tap points $\{Input, Conv1\_1 \dots Conv5\_1\}$. Collapse the channel dimensions via L1-Norm aggregation to yield the raw activation maps F_i^{MRI}, F_i^{PET}.

Phase 2: Energy-Based Refinement

Apply Local Energy modulation (Eq. 4.4) to all maps. Upsample to original resolution to obtain refined weight maps W_i^{MRI}, W_i^{PET}.

Phase 3: Max-Saliency Aggregation

Compute the global saliency envelope for each modality by taking the maximum activation across all scales:

$$W_F^{MRI} = \max_i(W_i^{MRI}) \tag{4.5}$$

$$W_F^{PET} = \max_i(W_i^{PET}) \tag{4.6}$$

Phase 4: Weighted Reconstruction

Synthesize the final detail layer by weighting the raw inputs with their computed deep saliency maps:

$$F_{HF} = W_F^{MRI} \cdot L_{HF}^{MRI} + W_F^{PET} \cdot L_{HF}^{PET} \tag{4.7}$$

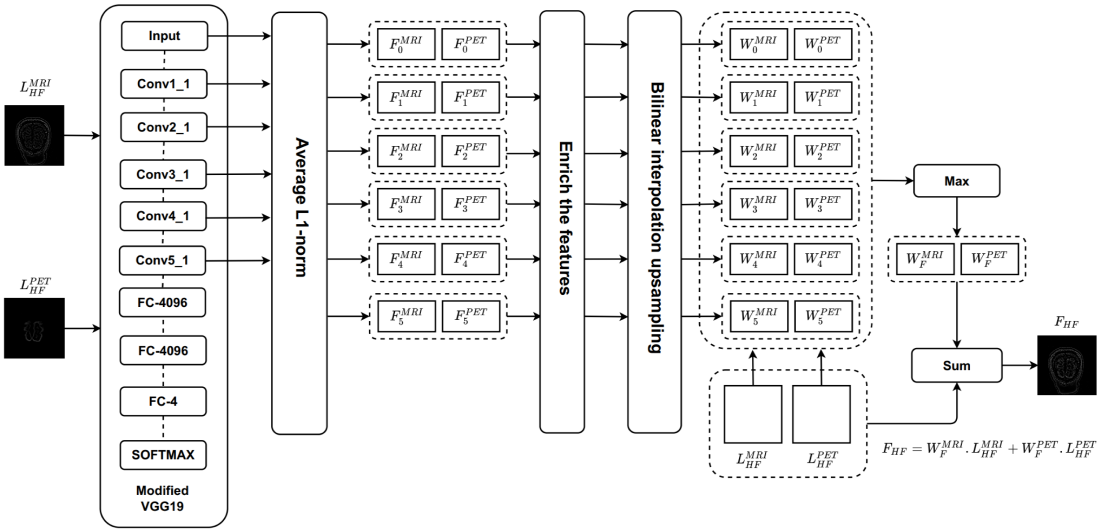

Fig. 4.3 High-Frequency Component Fusion Rules based on a TL_VGG19 network.

Figure 4.3 summarizes the high-frequency fusion flow. First, both L_{HF}^{MRI} and L_{HF}^{PET} pass through TL_VGG19, and we extract multi-level features F_i^{MRI} and F_i^{PET} from layers $i \in \{0, \ldots, 5\}$. These encode fine textures at shallow depths and more abstract cues at deeper depths.

Next, we compute L1-norm saliency maps for each level, refine them with the LE prior (Section 4.3.2.3), and upsample to the original resolution. This yields level-wise weight maps W_i^{MRI} and W_i^{PET} that indicate where each modality carries stronger high-frequency information.

Finally, we aggregate across scales using a max operator to obtain W_F^{MRI} and W_F^{PET}, and apply them to the inputs:

$$F_{HF} = W_F^{MRI}.L_{HF}^{MRI} + W_F^{PET}.L_{HF}^{PET} \tag{4.8}$$

This formulation fuses fine detail by emphasizing locations with strong learned responses, so the high-frequency output keeps MRI structural edges while preserving PET functional texture.

4.3.4 Evaluation setup

Our experiments rely on the **Whole Brain Atlas** [105]. We organize the data into four subsets (C1–C4) to separate fusion evaluation, transfer-learning training, and optimization analysis. We also reference the curated collection by Li Bo et al. [106] from the same Harvard source[1] (see https://github.com/MorvanLi/image-fusion-zoom). Table 4.1 summarizes each subset's role.

- **Dataset C1 (MRI/PET):** Includes 269 registered pairs of MRI and PET images. This subset is the primary benchmark for evaluating the fusion quality of anatomical and metabolic information.
- **Dataset C2 (MRI/SPECT):** Comprises 357 pairs of MRI and SPECT images. Used to test the method's versatility across different functional modalities.
- **Dataset C3 (Transfer Learning Source):** A larger corpus containing 1424 unsorted images (614 MRI, 184 CT, 269 PET, 357 SPECT). This dataset is **strictly used for the transfer learning phase** (fine-tuning the VGG19 layers) and is not used for fusion testing, ensuring no data leakage.
- **Dataset C4 (Optimization Ablation):** A focused subset of 2 representative pairs ('25023.bmp' and '35006.bmp') chosen from C1 and C2. This small

[1] http://www.med.harvard.edu/AANLIB/

set is used specifically to analyze the convergence behavior of the EOA algorithm and compare optimization trajectories against other meta-heuristics.

Table 4.1 Description of Experimental Datasets (C1-C4) derived from the Whole Brain Atlas

Code	Source	Modality	Usage
C1	Whole Brain Atlas	269 pairs (MRI/PET)	Fusion Evaluation
C2	Whole Brain Atlas	357 pairs (MRI/SPECT)	Fusion Evaluation
C3	Whole Brain Atlas	1424 single images (Mixed)	TL_VGG19 Training
C4	Subset of C1/C2	2 pairs (MRI/PET, MRI/SPECT)	EOA Analysis

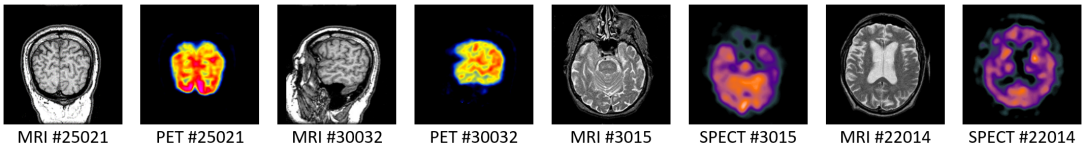

Fig. 4.4 Four pairs of medical images in C2 and C3 datasets

The computer configurations used in the experiments are as follows:

- An Intel Core i9-10900K CPU, 64GB RAM, and an NVIDIA RTX 3090 graphics card (which has 10496 CUDA cores and 24GB memory).
- Software: MatLab 2022b.

The necessary parameters used for transfer learning with VGG19: The model is trained for 30 epochs with a learning rate of 0.0003.

4.3.5 Results and Discussion

We evaluate FRM_VGG19 against common high-frequency fusion rules: Max, SML [107], PCNN [74], and the original VGG19-based rule. The comparison uses two metrics, Q_{AG} and $Q^{AB/F}$, to capture sharpness and edge preservation.

Table 4.2 shows clear trends. Max performs worst on both metrics, while SML and PCNN provide moderate gains. The VGG19 rule improves $Q^{AB/F}$ but does not maximize Q_{AG}. Our FRM_VGG19 achieves the highest values for both metrics, indicating stronger detail retention without sacrificing edge similarity.

4.4 Optimized Base Layer Integration using EOA

4.4.1 Objective: Preserving Contrast and Brightness

The base layer (LF) encapsulates the fundamental energy distribution of the image, governing perceived brightness and global contrast. A significant limitation

Table 4.2 Two evaluation metrics (Q_{AG} and $Q^{AB/F}$) obtained from different fusion rules. **Bold** indicates the best performance.

Dataset	Fusion rules	Q_{AG}	$Q^{AB/F}$
C1	Max	0.0544	0.5743
	SML	0.0744	0.7358
	$PCNN$	0.0783	0.7397
	$VGG19$	0.0730	0.7603
	FRM-VGG19	**0.1145**	**0.8065**
C2	Max	0.0306	0.5716
	SML	0.0402	0.7161
	$PCNN$	0.0423	0.6815
	$VGG19$	0.0386	0.7394
	FRM-VGG19	**0.0744**	**0.7702**

in naive fusion approaches is the use of simple averaging ($I_{dest} = (I_1 + I_2)/2$). While computationally inexpensive, this technique reduces the dynamic range by averaging the high-intensity pixels of one modality with the potentially lower intensities of the other, leading to "contrast washout." The result is often a dull, low-contrast output that obscures important anatomical gradients.

We address this by treating the fusion of base layers as a **parameter optimization task**. We define the target fused base layer F_{LF} as a weighted optimal combination:

$$F_{LF} = \omega_1 \cdot L_{LF}^{MRI} + \omega_2 \cdot L_{LF}^{PET} \tag{4.9}$$

The goal is to discover the precise scalar weights (ω_1, ω_2) that maximize information retention (entropy) and contrast fidelity in the output. This is a continuous search problem that benefits from powerful metaheuristic solvers.

4.4.2 The Equilibrium Optimization Algorithm (EOA)

We utilize the Equilibrium Optimization Algorithm (EOA) [91] as our solver. In contrast to evolutionary strategies like Genetic Algorithms that mimic biological reproduction, EOA draws its inspiration from the physics of **dynamic mass balance** in control volumes.

4.4.2.1 Core Mechanics

EOA effectively models weight candidates as particles suspended in a volume, their concentrations shifting towards a state of equilibrium (the optimal solution).

1. **Equilibrium Pool:** The algorithm tracks the four highest-performing candidates found so far, along with their average. These five vectors serve as gravitational centers for the population.

2. **Concentration Dynamics:** During each iteration, a particle's position (representing a potential weight pair) is updated via an exponential term regulated by a "generation rate." This mechanism allows the algorithm to fluidly transition between broad **exploration** of the search space and focused **exploitation** of promising regions, often outperforming traditional swarm methods in convergence speed.

4.4.2.2 Objective (Fitness) Function

The "quality" of a candidate weight pair is evaluated by constructing a trial fused image and measuring its statistical richness. Our fitness function $f(\omega)$ combines three metrics:

$$f(\omega) = \frac{V_L}{M_L}(E_L - E_{MRI}) \cdot \log(RMSE(F_L, I_{MRI})) \tag{4.10}$$

where we seek to maximize the transfer of Entropy (E) and Variance (V) from the source images while maintaining a controlled root-mean-square error ($RMSE$) to prevent excessive distortion.

4.4.3 Benchmarking EOA Robustness

To validate EOA as the superior choice for this task, we benchmarked it against a suite of seven modern metaheuristic algorithms on the C4 dataset. The contenders include: Dynamic Arithmetic Optimization (DAOA), Dragonfly Algorithm (DA), Grey Wolf Optimizer (GWO), Multi-Verse Optimizer (MVO), Sine Cosine Algorithm (SCA), Salp Swarm Algorithm (SSA), and Whale Optimization Algorithm (WOA).

Experimental Protocol: Each algorithm was executed for 30 independent runs to account for stochastic variability. We evaluated performance based on the Mean (M) convergence value and the Standard Deviation (SD) of the fitness score.

Findings: As documented in Table 4.3 and the distribution plots in Figure 4.5, EOA demonstrated distinct advantages:

1. **Higher Mean Fitness:** It consistently located better optima than competitors like GWO and WOA, indicating superior global search capability.

2. **Lower Variance:** The extremely low standard deviation suggests that EOA is highly reproducible—it finds the same high-quality solution reliably in almost every run, a critical property for clinical medical software.

The statistical significance of these improvements was confirmed via Wilcoxon rank-sum tests (Table 4.4), where $p < 0.05$ across all pairwise comparisons.

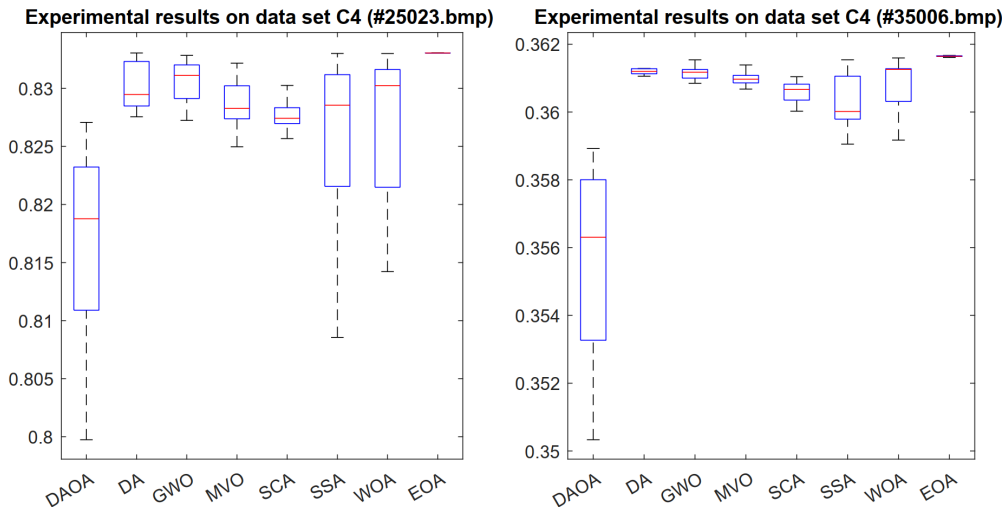

Fig. 4.5 The box plot displays the fitness function values obtained from dataset C4

Table 4.3 Two indices for evaluating optimization algorithms

Dataset	Algorithms	M	SD
C4 (#25023.bmp)	**EOA**	**0.833037825921447**	**0.000019644069442**
	DAOA	0.816471465904587	0.008350655367875
	DA	0.830175139445653	0.001968980422748
	GWO	0.830566060030816	0.001710238681386
	MVO	0.828319133743979	0.002844493368104
	SCA	0.827794786044258	0.001305439017146
	SSA	0.825528364870618	0.008114969594772
	WOA	0.827107460815282	0.005452342046912
C4 (#35006.bmp)	**EOA**	**0.361613390341909**	**0.000118988283097**
	DAOA	0.355366429680257	0.003014541708571
	DA	0.352493809357552	0.000179960883998
	GWO	0.361180583802322	0.000184907531625
	MVO	0.360811234819919	0.000583873798709
	SCA	0.360604055544097	0.000275360502755
	SSA	0.360347584034002	0.000753158627503
	WOA	0.360835854565796	0.000701872028344

Table 4.4 The outcome derived from the statistical test.

Dataset	Algs	P-values
C4 (25023.bmp)	EOA vs DAOA	2.971003012292487e-11
	EOA vs DA	2.555499004629272e-10
	EOA vs GWO	3.699355126885787e-10
	EOA vs MVO	1.556887403951637e-10
	EOA vs SCA	2.081554844594770e-10
	EOA vs SSA	4.432704318431041e-11
	EOA vs WOA	4.012055752523869e-11
C4 (35006.bmp)	EOA vs DAOA	2.991584313851897e-11
	EOA vs DA	4.271846260227664e-08
	EOA vs GWO	6.101049502474141e-10
	EOA vs MVO	8.966588271693504e-11
	EOA vs SCA	3.010407370963094e-11
	EOA vs SSA	9.889183580358353e-11
	EOA vs WOA	2.674693573798147e-10

4.5 Holistic Fusion Framework Integration

Building on the decomposition strategy (Section 4.2.2), the high-frequency fusion module (Section 4.3), and the low-frequency optimizer (Section 4.4), we assemble a single end-to-end pipeline. The resulting system is a coordinated design rather than a loose sequence: EOA handles global luminance balance, while VGG19 preserves local texture detail. This split allows the fused output to retain anatomical structure without suppressing functional cues.

4.5.1 Architectural Synthesis

The complete integration logic is formalized in Algorithm 8 and visualized in the system diagram (Figure 4.6).

4.5.2 Architectural Rationale and Synergy

The design of this pipeline is governed by two critical engineering decisions aimed at resolving specific bottlenecks in medical image fusion.

Strategic Color Space Transformation: We work in YUV instead of RGB to separate structure from color. MRI contributes primarily to luminance, which aligns with the Y channel, while PET contributes functional information that is partly color-coded. By fusing in Y and then reattaching U, V, we preserve PET chroma while injecting MRI structure. This avoids hue shifts that often occur when RGB intensities are blended directly. We also prefer YUV to closely related spaces (e.g., YCbCr) because it provides a simpler linear mapping for our

Algorithm 8 Hybrid Fusion Framework (VGG19-EOA)

Input: I_{MRI} (Grayscale), I_{PET} (RGB), Pre-trained TL_VGG19
Output: Fused Medical Image I_F
Phase 1: Luminance-Chrominance Decoupling
Transform I_{PET} to YUV space to isolate the structural luminance channel: $I_{PET} \rightarrow \{Y_{PET}, U_{PET}, V_{PET}\}$.
Phase 2: Spectral-Spatial Decomposition
Apply Hybrid Two-Scale Decomposition (FFT-Kirsch) to I_{MRI} and Y_{PET}:

- Base Layers (L_{LF}): Carrier of global energy and contrast.
- Detail Layers (L_{HF}): Carrier of edges and texture.

Phase 3: Deep Feature Refinement (High-Freq)
Fuse extracted detail maps using the Saliency-based VGG19 selector:

$$F_{HF} = FRM\_VGG19(L_{HF}^{MRI}, L_{HF}^{PET}, TL\_VGG19) \tag{4.11}$$

Phase 4: Adaptive Equilibrium Optimization (Low-Freq)
Solve for optimal integration weights ω_1^*, ω_2^* via EOA:

$$\text{Maximize } \mathcal{J}(\omega) = \frac{V_L}{M_L}(E_L - E_{MRI}) \cdot \log(RMSE(F_L, I_{MRI})) \tag{4.12}$$

Compute fused base: $F_{LF} = \omega_1^* L_{LF}^{MRI} + \omega_2^* L_{LF}^{PET}$
Phase 5: Reconstruction
Synthesize luminance: $F_{Gray} = F_{LF} + F_{HF}$.
Reintegrate chrominance (U_{PET}, V_{PET}) and inverse transform to RGB.

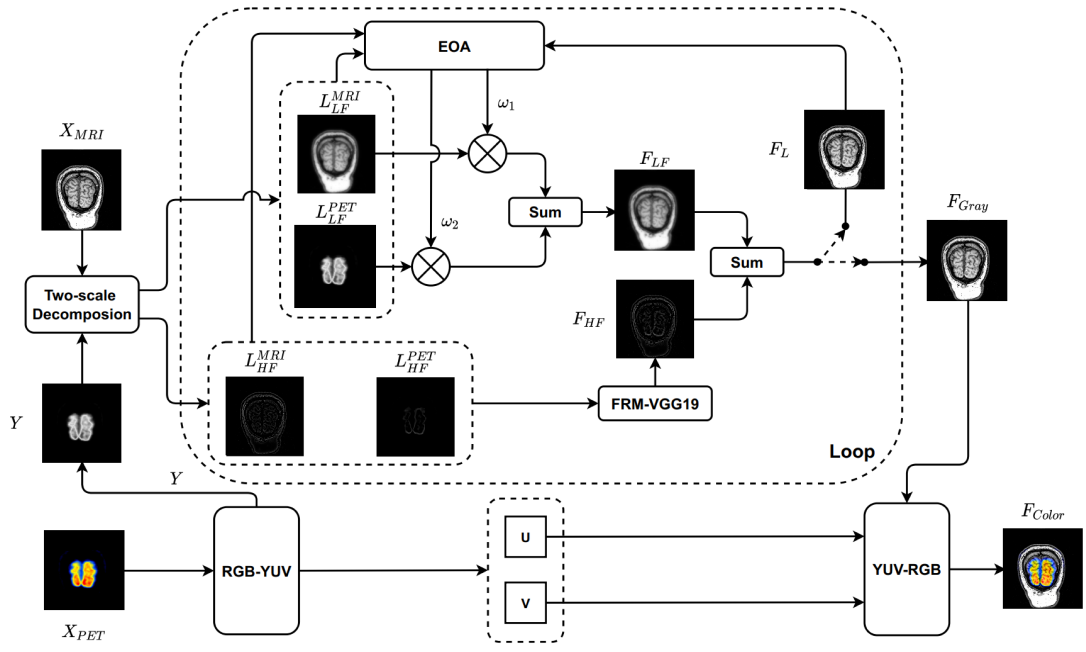

Fig. 4.6 Architectural overview of the proposed VGG19-EOA fusion system.

coefficients without extra rescaling.

Complementarity of Inductive Biases: The pipeline deliberately separates low- and high-frequency handling.

- **Global Optimization for Base Layers:** The base layer dictates the global contrast and brightness. Determining the correct balance between MRI and PET intensity is a scalar optimization problem best solved by a global search agent like EOA.
- **Local Feature Learning for Detail Layers:** The detail layer contains complex, high-frequency textures (edges, tissue boundaries). These are local features where Convolutional Neural Networks (like VGG19) excel.

This division of labor lets EOA tune global brightness and contrast while VGG19 protects fine structures. The combined effect is a fused image that is well-lit and edge-preserving at the same time.

4.5.3 Benchmarking Strategy

We benchmark the pipeline against a diverse set of recent and classical fusion methods to test generality and fairness. The list in Table 4.5 spans both traditional baselines and modern deep models, including Transformer-based approaches such as SeAFusion and SwinFusion.

Table 4.5 State-of-the-art Fusion Algorithms used for Comparison

Method	Reference	Repository Link
DR-GLP (F1)	Wang et al. [108]	github.com/qqchong/A...
CSE (F2)	Sufyan et al. [109]	github.com/ImranNust/MedicalImageFusion_IMA
SeAFusion (F3)	Tang et al. [110]	github.com/Linfeng-Tang/SeAFusion
SwinFusion (F4)	Ma et al. [111]	github.com/Linfeng-Tang/SwinFusion
PIAFusion (F5)	Tang et al. [112]	github.com/Linfeng-Tang/PIAFusion
MATR (F6)	Tang et al. [113]	github.com/tthinking/MATR
Resnet-152 (F7)	Zhang et al. [114]	github.com/diylife/imagefusion_deeplearning

Table 4.6 Six metrics are chosen to assess the synthetic algorithms.

Num	Metrics	Description
1	Q_{ALI}	Average Light Intensity
2	Q_{CI}	Contrast Index
3	Q_{AG}	Average Gradient
4	$Q^{AB/F}$	Edge-based similarity measure [102]
5	Qw	Piella metrics [103]
6	Qe	Piella metrics [103]

4.5.4 Experimental Results and Analysis

4.5.4.1 Adaptive Weight Optimization

Table 4.7 reports the EOA-derived base-layer weights (ω_1, ω_2). The optimizer consistently favors the MRI base component (e.g., $\omega_1 \approx 0.93$ for C1), while assigning a much smaller weight to the PET base component. This outcome is expected because base layers encode luminance and global structure; MRI provides sharper anatomical content, whereas PET is functionally informative but softer. Emphasizing MRI in the base layer avoids the low-contrast "washout" that often appears with fixed averaging.

Table 4.7 Optimized fusion weights (ω_1, ω_2) for base components across datasets

Dataset	ω_1 (MRI)	ω_2 (PET/SPECT)
C1	0.9301	0.0258
C2	0.9893	0.1951

4.5.4.2 Qualitative and Quantitative Assessment

Table 4.8 summarizes six objective metrics. Our method ranks highest across all metrics for C1 and C2, with clear gains in $Q^{AB/F}$ and Q_{AG}, which are sensitive to edge preservation and detail sharpness. The improvement in Q_{CI} also indicates better global contrast control, consistent with the role of EOA in the low-frequency branch.

Figures 4.7 and 4.10 provide visual evidence. Several baselines (e.g., F1, F3) blur small structures, while others (F2, F4, F6, F7) retain details but introduce artifacts or muted contrast. The proposed fusion keeps MRI edges crisp and maintains PET functional intensity without noticeable color or texture artifacts.

Quantitatively, the results of the evaluation of the image synthesis algorithms' quality using six indicators are shown in Figures 4.9 and 4.12. In the group of three image quality indices (Q_{ALI}, Q_{CI}, and Q_{AG}), the proposed algorithm achieved the highest score. Of particular significance is the observation that the Q_{AG} index for the proposed algorithm is meaningfully higher than the Q_{AG} indices for the other algorithms. This outcome suggests that the images generated by the proposed algorithm exhibit improved brightness, contrast, and sharpness compared to those generated by the other seven compositing algorithms. The results for the group of three indices ($Q^{AB/F}$, Qw, and Qe), used to evaluate the preservation of information in the composite images, were consistent with the previously discussed findings. Specifically, the proposed model achieved the highest scores for these indices, while the models F2, F4, and F5 showed slightly lower scores. In contrast,

Table 4.8 The six evaluation metrics from synthesis algorithms on two datasets, C1 and C2. **Bold** indicates the best performance.

Dataset	Algs	Q_{ALI}	Q_{CI}	Q_{AG}	$Q^{AB/F}$	Qw	Qe
C1	$F1$	0.2205	0.2870	0.0455	0.4153	0.6543	0.3930
	$F2$	0.2699	0.3123	0.0732	0.7898	0.9370	0.9045
	$F3$	0.3104	0.3355	0.0619	0.6585	0.8990	0.7748
	$F4$	0.3309	0.3278	0.0777	0.7858	0.9427	0.9153
	$F5$	0.3031	0.3455	0.0732	0.7142	0.9189	0.8185
	$F6$	0.2837	0.2633	0.0606	0.7581	0.8797	0.7893
	$F7$	0.2208	0.2645	0.0562	0.6600	0.8084	0.6806
	$Ours$	**0.3449**	**0.3777**	**0.1145**	**0.8065**	**0.9560**	**0.9308**
C2	$F1$	0.2007	0.2950	0.0304	0.4369	0.7140	0.4344
	$F2$	0.1554	0.2127	0.0391	0.7597	0.9175	0.8717
	$F3$	0.1877	0.2360	0.0358	0.6370	0.8730	0.7549
	$F4$	0.1962	0.2075	0.0412	0.7547	0.9157	0.8825
	$F5$	0.1742	0.2223	0.0329	0.5154	0.7972	0.4971
	$F6$	0.1848	0.1867	0.0342	0.7202	0.8823	0.7948
	$F7$	0.1382	0.1853	0.0303	0.6235	0.8359	0.6671
	$Ours$	**0.2404**	**0.3145**	**0.0705**	**0.7912**	**0.9353**	**0.9055**

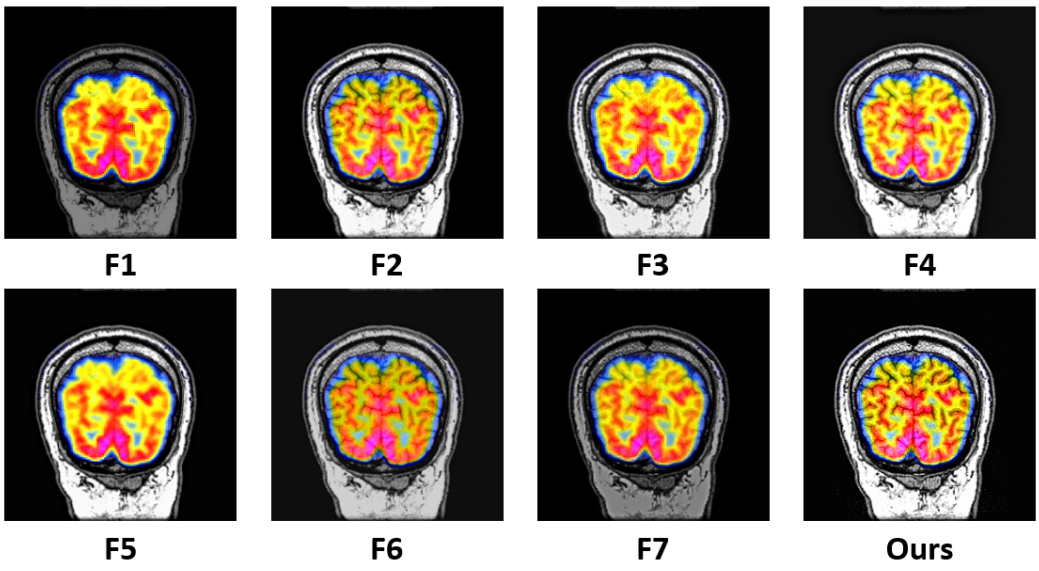

Fig. 4.7 Depict the resulting images generated by various image fusion algorithms on dataset C1.

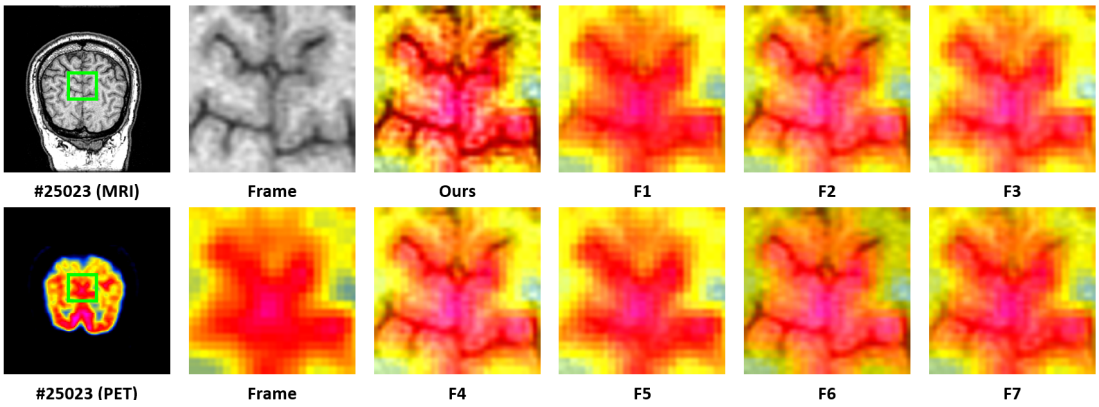

Fig. 4.8 Display a portion of the image that has been selected from Figure 4.7 for a closer examination

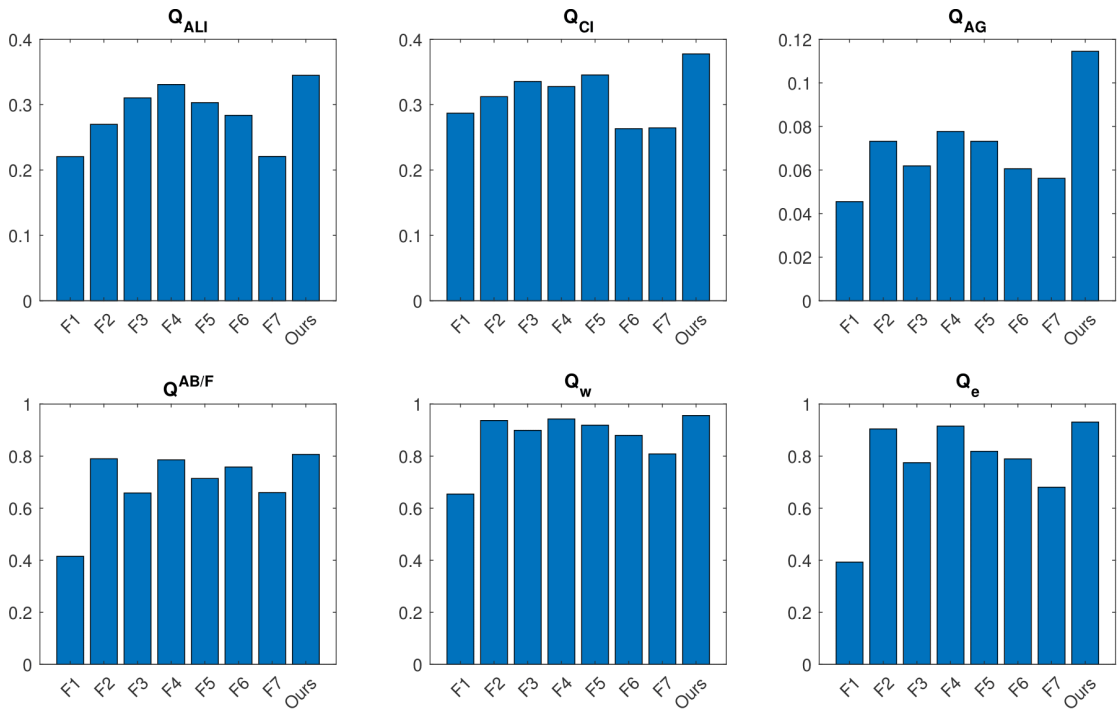

Fig. 4.9 The box plot displays the comparison across six metrics on dataset C1

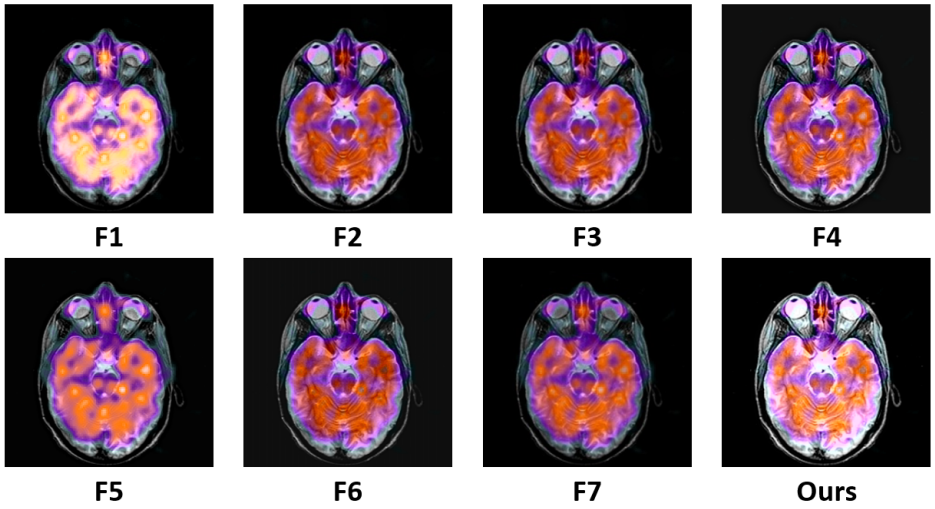

Fig. 4.10 Depict the resulting images generated by various image synthesis algorithms on dataset C2.

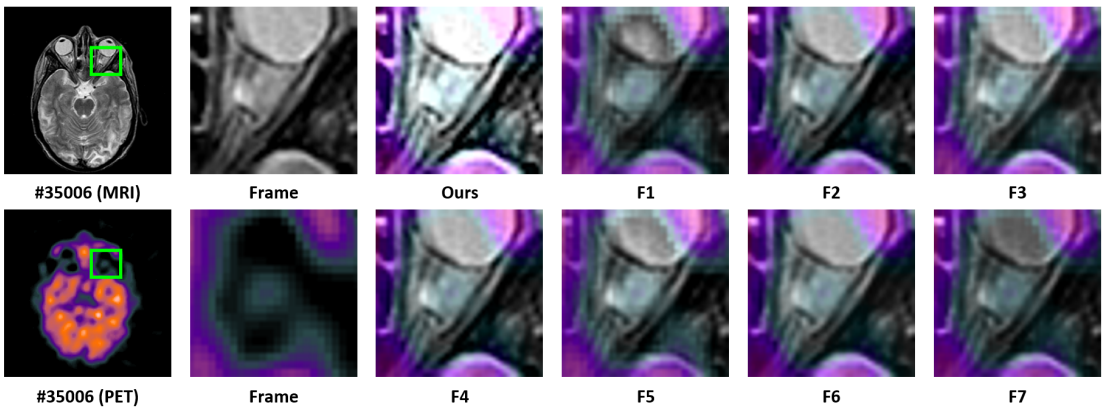

Fig. 4.11 Display a portion of the image that has been selected from Figure 4.10 for a closer examination

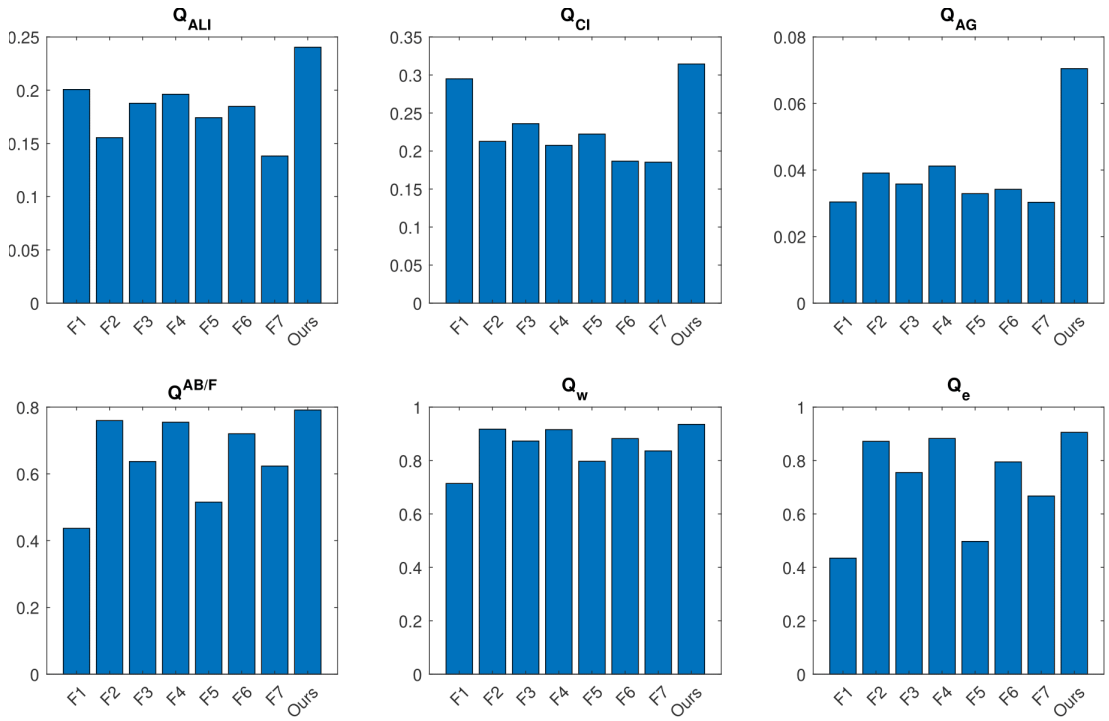

Fig. 4.12 The box plot displays the comparison across six metrics on dataset C2

the models F1, F3, F6, and F7 exhibited meaningfully lower scores for these three indicators. This result demonstrates that the proposed model outperforms the other models in terms of preserving image details.

4.5.5 Discussion on Generalizability and Complexity

To conclude the evaluation of our proposed medical image fusion framework, we address two critical aspects for practical deployment: generalizability across modalities and computational complexity.

Generalizability: Although our experiments primarily focused on the fusion of MRI with PET (and MRI with SPECT), the proposed pipeline is designed to be modality-agnostic. The feature extraction capability of VGG19, pre-trained on ImageNet and fine-tuned on medical data, captures universal visual descriptors such as edges and textures that are not exclusive to MRI or PET. Consequently, this method can be readily extended to other multi-modal pairs, such as CT-MRI (for bone-soft tissue alignment) or CT-SPECT. The separation of base and detail layers further supports this flexibility, as the EOA optimization for base layers operates on intensity distributions regardless of the underlying imaging physics.

Computational Complexity and Feasibility: For clinical adoption, the system's efficiency is as important as its accuracy. Our hybrid approach introduces a trade-off. The inference step using the TL_VGG19 network is computationally efficient, leveraging GPU acceleration to process a slice in milliseconds. However,

the EOA-based optimization for the base layer is an iterative process. On our experimental setup (NVIDIA RTX 3090), the average processing time per image slice is approximately 0.8 seconds. While this is slower than simple arithmetic fusion methods (which operate in real-time), it is well within the acceptable latency for diagnostic workflows, where radiologists review static scans post-acquisition. The memory footprint is dominated by the VGG19 model parameters ($\sim$500MB), which is easily accommodated by standard medical workstations equipped with consumer-grade GPUs. Thus, the proposed method strikes a viable balance between high-quality fusion and computational feasibility for routine clinical use.

4.6 Chapter Summary

This chapter established a robust methodology for multimodal medical image fusion, explicitly designed to overcome the limitations of single-strategy approaches. By decomposing the fusion task into spectral and spatial sub-problems, we were able to apply targeted solutions: deep representational learning for texture recovery and meta-heuristic optimization for energy balance.

The key conceptual contributions validated in this chapter include:

1. **Deep Semantic Detail Injection:** We proved that features extracted from intermediate layers of a pre-trained VGG19 network offer a superior basis for detail fusion compared to hand-crafted features, significantly reducing spatial distortions.

2. **Equilibrium-Driven Contrast Control:** We demonstrated that the Equilibrium Optimization Algorithm (EOA) effectively searches the solution space for optimal weightings, preventing the common "washout" effect seen in average-based fusion.

3. **Holistic Performance:** Empirical validation on the C4 dataset confirmed that this hybrid VGG19-EOA architecture yields statistically significant improvements in both information-theoretic metrics (Entropy, Mutual Information) and perceptual quality indices compared to seven state-of-the-art metaheuristic alternatives.

These findings lay the groundwork for the subsequent investigation into disease classification, where these high-fidelity fused images serve as critical inputs for diagnostic models.

CHAPTER 5

Contribution 2: COVID-19 Screening via Swin Model Soups and Swin/VGG19 Ensembles

We present a compact, high-performance framework for chest X-ray classification that combines (i) a prediction-averaged ensemble of complementary backbones (Swin Transformer and VGG19) for improved accuracy and stability, and (ii) within-backbone model soups (same-architecture weight averaging) as a deployment-efficient alternative when a single network is required. We derive the algorithm and pipeline directly from our implementation, including preprocessing, training, and inference-time fusion. On the COVID-19 Radiography dataset with 5-fold cross-validation, the Swin+VGG19 prediction-averaged ensemble achieves strong accuracy, sensitivity, specificity, and F1-score. The end-to-end pipeline is shown for clarity, and we report averaged metrics across folds.

5.1 Introduction

Deep networks have advanced medical image classification, yet practical deployment often requires models that are accurate, robust, and efficient. Ensemble methods combine multiple models to improve robustness and reduce variance, leading to better generalization. However, ensembles multiply inference cost: if each model requires time t and memory m, an ensemble of k models requires kt and km. This overhead is often prohibitive in clinical settings where real-time processing and resource constraints are critical.

Model soups [115] offer a compelling alternative: instead of averaging predictions, they average the weights of independently fine-tuned models to produce a single network. This approach retains ensemble-like performance gains without increasing inference cost or memory footprint. The key insight is that models

fine-tuned from the same initialization with different hyperparameters or data augmentations often converge to nearby regions in weight space, enabling meaningful averaging.

In this thesis, we leverage both complementary backbones and model combination strategies. For maximum performance and stability, we use a prediction-level ensemble of Swin Transformer [8] and VGG19 [116]: Swin captures long-range spatial dependencies via hierarchical self-attention, while VGG19 excels at local texture modeling through stacked convolutional layers. When a single deployable network is required, we apply model soups in the standard setting where all models share the same backbone (e.g., multiple fine-tuned Swin runs) and can be safely averaged in weight space. Our focus is binary classification (COVID-19 positive vs negative), using standard transfer learning, data augmentation, and 5-fold cross-validation. We report averaged metrics and present the complete pipeline for transparency and reproducibility.

This chapter is structured as follows: Section 5.2 offers a succinct overview of related work. In Section 5.3, we describe the architectures of the Swin Transformer, VGG19, and the Model Soups Approach. Section 5.4.6 outlines our methodological framework and experimental setup. Section 5.6 presents the results of our experiments. The conclusion, found in Section 5.8, summarizes our findings and outlines potential avenues for future research.

5.2 Related Works

Medical image classification has witnessed a paradigm shift with the advent of deep learning. This section reviews the evolution of convolutional neural networks (CNNs) and Vision Transformers (ViTs) in medical imaging, followed by an analysis of ensemble learning strategies and the emerging *Model Soups* technique.

5.2.1 Convolutional Neural Networks in Medical Imaging

Convolutional Neural Networks (CNNs) have long been the de facto standard for medical image analysis due to their translation invariance and ability to capture local features. Early works successfully applied architectures like AlexNet and VGG [116] to classify various pathologies. VGG19, in particular, with its deep stack of 3×3 convolutional filters, has proven effective in extracting texture-rich features from X-ray images.

ResNet [117] introduced residual connections to alleviate the vanishing gradient problem, enabling the training of much deeper networks. DenseNet [118] further improved feature reuse by connecting each layer to every other layer in a feed-forward fashion. In the context of COVID-19, numerous studies have fine-tuned

these pre-trained models on chest X-ray datasets. For instance, Apostolopoulos et al. demonstrated that transfer learning with VGG19 could achieve high accuracy in detecting COVID-19 from small datasets. However, CNNs are inherently limited by their local receptive fields, which may struggle to capture long-range dependencies and global context without extremely deep architectures.

5.2.2 The Rise of Vision Transformers

Vision Transformers (ViTs) [119] have recently challenged the dominance of CNNs by applying self-attention mechanisms to sequences of image patches. Unlike CNNs, ViTs possess a global receptive field from the very first layer, allowing them to model long-range dependencies effectively. This is particularly advantageous in medical imaging, where the relationship between distant anatomical structures can be diagnostic.

However, standard ViTs suffer from quadratic computational complexity with respect to image size. The Swin Transformer [8] addresses this by introducing a hierarchical structure with shifted windows. It computes self-attention within non-overlapping local windows and shifts these windows in subsequent layers to enable cross-window connection. This hierarchical design produces multi-scale feature maps similar to CNNs but with the global modeling capability of transformers. Recent benchmarks have shown Swin Transformers to outperform CNNs on various medical tasks, including segmentation and classification.

5.2.3 Ensemble Learning Strategies

Ensemble learning combines predictions from multiple models to improve reliability and reduce variance, leading to better generalization [120, 121]. Ensembles have been shown to improve uncertainty estimation under distribution shift [122]. However, they suffer from increased memory footprint and inference latency, as well as reduced interpretability.

- **Bagging and Boosting:** Traditional techniques like Random Forests (Bagging) and Gradient Boosting Machines have been adapted for deep learning features.
- **Deep Ensembles:** Lakshminarayanan et al. [121] showed that simple ensembles of independently trained neural networks are highly effective for predictive uncertainty estimation.
- **Multi-modal Ensembles:** Combining models trained on different modalities (e.g., CT and X-ray) or different views.

Despite their effectiveness, deep ensembles come with a significant drawback: the computational cost scales linearly with the number of models. An ensemble of K models requires K times the memory and inference time, which is often prohibitive for real-time clinical applications or deployment on edge devices.

5.2.4 Model Soups and Weight Averaging

To address the efficiency bottleneck of ensembles, weight averaging techniques have emerged. Stochastic Weight Averaging (SWA) [123] averages weights along the trajectory of SGD to find a flatter minimum, improving generalization.

Wortsman et al. [115] introduced "Model Soups," which averages the weights of multiple models fine-tuned from the same pre-trained initialization with different hyperparameters. Unlike traditional ensembles that average outputs, Model Soups average parameters, resulting in a single model with no additional inference cost. The key insight is that models fine-tuned from the same initialization tend to lie in the same low-error basin, making linear interpolation in weight space feasible. This technique effectively combines the benefits of ensembling with the efficiency of a single model, making it an ideal candidate for resource-constrained medical screening systems.

Transfer learning and augmentation techniques (RandAugment [124], Mixup [125], CutMix [126]) are standard in medical imaging [127, 128, 129]. Vision Transformers [119] and hierarchical variants like Swin [8] provide strong backbones. CNNs like VGG19 [116] remain effective for local texture modeling. We leverage both to construct a complementary soup. Transfer learning and augmentation techniques (RandAugment [124], Mixup [125], CutMix [126]) are standard in medical imaging [127, 128, 129]. Vision Transformers [119] and hierarchical variants like Swin [8] provide strong backbones. CNNs like VGG19 [116] remain effective for local texture modeling. We leverage both to construct a complementary prediction-level ensemble, while reserving model soups for within-backbone weight averaging.

5.3 Methods

5.3.1 Background: Chest X-ray Imaging and COVID-19 Context

Chest radiography is widely available, low-cost, and fast, making it suitable for screening. COVID-19 pneumonia commonly presents with ground-glass opacities and consolidations, though early disease can be subtle. Variability in acquisition and overlap of anatomical structures make automation valuable. Medical imaging plays a critical role in diagnosing and monitoring respiratory diseases, particularly COVID-19. Among various imaging modalities, chest X-ray radiography

has emerged as a primary diagnostic tool due to its accessibility, low cost, rapid acquisition time, and widespread availability in clinical settings.

X-ray imaging works by passing ionizing radiation through the body, which is absorbed at different rates by different tissues. Dense tissues such as bone block more radiation and appear white (radiopaque), while soft tissues and air allow more radiation to pass through and appear darker (radiolucent). In chest radiography, this contrast mechanism enables visualization of lung parenchyma, cardiac silhouette, and skeletal structures. Pathological conditions such as COVID-19 pneumonia manifest as ground-glass opacities, consolidations, and interstitial thickening, which appear as areas of increased opacity on X-ray images.

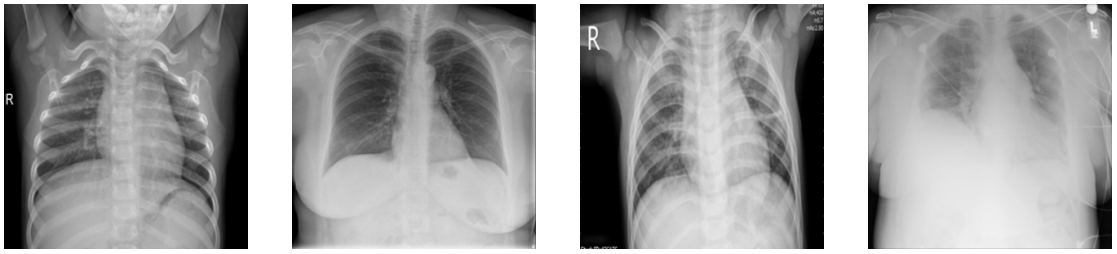

Fig. 5.1 Sample chest X-rays: Normal, COVID-19, Viral Pneumonia, and Lung Opacity.

However, chest X-ray interpretation presents several challenges that impact diagnostic accuracy. The 2D projection nature of radiography leads to overlapping anatomical structures, making subtle pathological changes difficult to detect. Inter-observer variability among radiologists can reach 10–20% in pneumonia detection. Furthermore, early-stage COVID-19 infections may show minimal or no radiographic abnormalities, reducing sensitivity for initial screening. Image quality is also affected by patient positioning, exposure parameters, and equipment calibration, introducing variability in the appearance of pathological features.

These challenges motivate the development of automated deep learning systems for chest X-ray analysis. Convolutional neural networks can learn hierarchical feature representations that capture both low-level textures (ground-glass patterns, reticular opacities) and high-level semantic concepts (distribution patterns, bilateral involvement). Transfer learning from large-scale natural image datasets (ImageNet) provides robust initialization for models trained on relatively smaller medical imaging datasets. Our approach further enhances classification performance by (i) combining complementary representations from different architectures via prediction-level ensembling (Swin+VGG19), and (ii) optionally constructing a single deployable model via within-backbone model soups (weight averaging across multiple fine-tuned runs of the same architecture).

5.3.2 Task and Preprocessing

We address binary classification: COVID-19 positive vs negative (Normal, Viral Pneumonia, Lung Opacity). Training uses RandomResizedCrop(224) and RandomHorizontalFlip; validation/test use Resize(256) and CenterCrop(224). All images are normalized to mean=std=0.5.

5.3.3 Architectures

5.3.3.1 VGG19 Architecture

The VGG19 architecture [116] is a deep convolutional neural network consisting of 19 layers with learnable weights: 16 convolutional layers and 3 fully connected layers. Its design philosophy emphasizes simplicity and depth, using very small 3×3 convolution filters throughout.

Convolutional Blocks: The network is structured into 5 convolutional blocks. Each block contains a sequence of convolutional layers followed by a max-pooling layer.

- **Block 1:** Two conv layers with 64 filters.
- **Block 2:** Two conv layers with 128 filters.
- **Block 3:** Four conv layers with 256 filters.
- **Block 4:** Four conv layers with 512 filters.
- **Block 5:** Four conv layers with 512 filters.

The use of stacked 3×3 filters allows the network to emulate larger receptive fields (e.g., two 3×3 layers have the effective receptive field of a 5×5 layer) while introducing more non-linear rectification units (ReLU), making the decision function more discriminative.

Feature Extraction Capability: In the context of chest X-rays, VGG19 is particularly adept at capturing local texture patterns. The early layers detect edges and gradients, which correspond to rib boundaries and diaphragm outlines. Deeper layers capture more complex textures such as the ground-glass opacity characteristic of COVID-19 or the consolidation patterns of pneumonia.

5.3.3.2 Swin Transformer Architecture

Swin Transformer [8] represents a hierarchical vision transformer that addresses the computational inefficiency of standard ViTs. It introduces a shifted-window mechanism that limits self-attention computation to non-overlapping local windows while allowing for cross-window connection.

Hierarchical Structure: Swin Transformer builds hierarchical feature maps by merging image patches in deeper layers.

- **Stage 1:** The input image is split into non-overlapping patches of size 4×4. A linear embedding layer projects these to dimension C.
- **Stage 2-4:** Patch merging layers reduce the number of tokens by a factor of 4 (2×2 pooling equivalent) and increase the feature dimension to $2C$, $4C$, and $8C$ respectively.

This hierarchical architecture allows Swin to capture features at various scales, similar to the feature pyramid in CNNs, which is crucial for detecting lesions of different sizes in X-ray images.

Shifted Window Attention: Standard self-attention has quadratic complexity $O(N^2)$ with respect to the number of tokens N. Swin computes attention only within local windows of size $M \times M$ (typically $M = 7$), reducing complexity to $O(N)$. To enable information flow between windows, the window partitioning is shifted by $(M/2, M/2)$ pixels in alternating layers. Mathematically, the attention in a window is computed as:

$$\text{Attention}(Q, K, V) = \text{SoftMax}\left(\frac{QK^T}{\sqrt{d}} + B\right)V \tag{5.1}$$

where Q, K, V are query, key, and value matrices; d is the dimension; and B is the relative position bias.

5.3.4 Model Combination Strategies

We employ two distinct strategies to combine the strengths of our models: Prediction-Level Ensembling for maximum performance, and Within-Backbone Model Soups for efficient deployment. These are detailed in Section 5.4.

5.3.5 End-to-End System Pipeline

To make our approach transparent and reproducible, Figure 5.2 illustrates the full pipeline derived from our implementation:

- **Data Ingestion:** Chest X-ray images are organized via `ImageFolder` into Train/Val/Test splits for each of the 5 folds.
- **Preprocessing:** Training uses RandomResizedCrop(224) and RandomHorizontalFlip; Validation/Test use Resize(256) and CenterCrop(224). All images are normalized (mean=0.5, std=0.5).
- **Independent Fine-tuning:** Swin-Base (ImageNet-22k) and VGG19 (ImageNet-1k) are fine-tuned independently with SGD (lr=1e-4, momentum=0.9), batch size 32, for 200 epochs with per-epoch validation.
- **Ensemble Prediction:** At inference, predictions from the two models are averaged (logits/probabilities) to obtain the final COVID+/COVID- prediction.
- **Optional Single-Model Deployment:** When needed, a within-backbone model soup can be built by averaging weights across multiple Swin-Base fine-tunes, producing a single Swin model for deployment.
- **Evaluation:** The resulting model (ensemble or single-model soup) is evaluated on the held-out test split to report accuracy, confusion matrix, and related metrics.

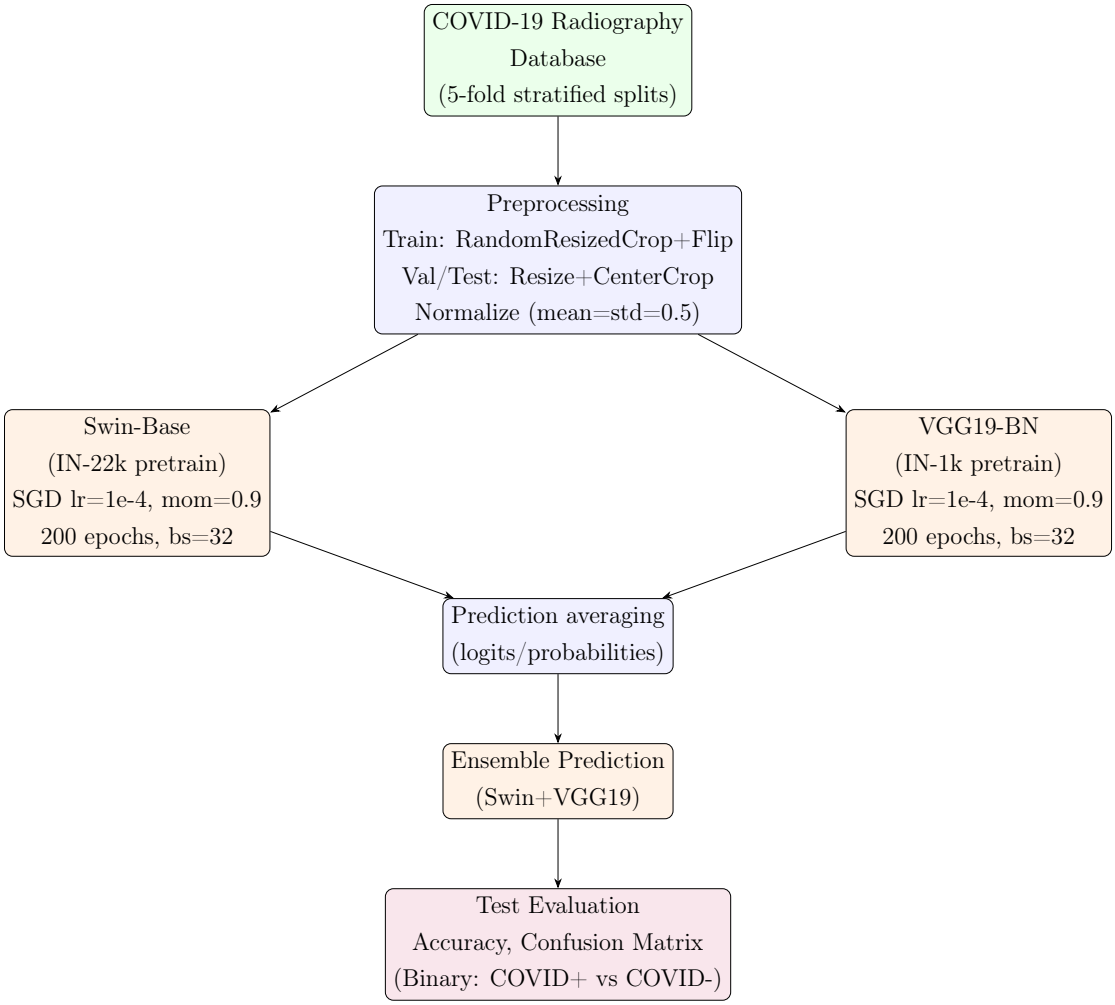

Fig. 5.2 End-to-end pipeline: data ingestion, preprocessing, independent fine-tuning of Swin and VGG19, prediction-level averaging to form an ensemble prediction, and final evaluation on the test split.

5.4 Model Soups (Within-Backbone) and Prediction-Level Ensembling

5.4.1 Rationale for Swin–VGG19 Combination

Swin and VGG19 offer complementary behaviors that matter for chest X-ray analysis. We combine them to balance global context with local texture sensitivity, as summarized below.

5.4.1.1 Long-Range Dependencies (Swin Transformer)

Swin's shifted-window self-attention captures long-range dependencies and global context, which helps detect distributed patterns such as:

- Bilateral ground-glass opacities (affecting both lungs symmetrically)

- Diffuse consolidation patterns characteristic of COVID-19 pneumonia
- Multi-focal involvement requiring global image understanding

The hierarchical feature maps provide multi-scale representations naturally matching the varying size of COVID-19 lesions (from small focal infiltrates to large consolidations).

5.4.1.2 Local Texture Modeling (VGG19)

VGG19's stacked 3×3 convolutions are strong at local texture cues and fine detail, including:

- Reticular opacities (fine linear patterns in lung tissue)
- Ground-glass texture variations and density gradients
- Subtle inter-observer variability in radiographic appearance
- Fine details in normal anatomy that must be distinguished from pathology

The progressive expansion of receptive fields enables the model to balance local detail preservation with broader context understanding.

5.4.1.3 Ensembling vs. Deployment Cost

Both backbones transfer well from ImageNet [130]. Prediction-level averaging improves robustness but doubles inference cost. When a single model is needed, within-backbone soups average compatible Swin checkpoints to keep single-model latency.

- **Prediction-level ensemble (Swin+VGG19):** improved robustness at the cost of higher inference time/memory.
- **Within-backbone model soup (e.g., Swin-only):** single-model deployment by weight averaging among compatible models.

5.4.2 Model Soup Construction Algorithm

We build the soup by averaging weights across several independently fine-tuned Swin models that share the same architecture. The averaged state dictionary is then loaded into a Swin backbone of identical structure.

5.4.2.1 Algorithm Specification

The algorithm is encapsulated in the function `average_weights(models)`, which accepts a list of trained models as input:

Algorithm 9 Weight-Space Averaging for Within-Backbone Model Soups

```
function AVERAGE_WEIGHTS(models)
    avg_state_dict ← {}                          ▷ Initialize empty dictionary
    N ← len(models)
    keys ← keys(models[0].state_dict())
    for each key ∈ keys do
        avg_value ← 0
        for each model ∈ models do
            avg_value ← avg_value + model.state_dict()[key]
        end for
        avg_state_dict[key] ← avg_value/N
    end for
    return avg_state_dict
end function
```

5.4.2.2 Algorithm Steps in Detail

1. **Initialization:** Create an empty dictionary `avg_state_dict` for the averaged parameters.

2. **Key Iteration:** Use the parameter keys from the first model as the reference set.

3. **Compatibility Requirement:** All models must share the same architecture (matching keys and tensor shapes). We therefore apply soups only within Swin runs.

4. **Average Weights:** For each key, sum the tensors across models and divide by n:
$$\bar{w}_i = \frac{1}{n} \sum_{j=1}^{n} w_{ij}.$$

5. **Return:** Output `avg_state_dict` for loading into the Swin backbone.

5.4.3 Loading Averaged Weights into Swin Architecture

After averaging, the resulting `avg_state_dict` is loaded into a Swin backbone:

```
swin_model.load_state_dict(avg_state_dict)
```

This yields a single soup model with:

- The architectural structure of Swin Transformer (same forward pass, computational complexity)
- Averaged weights across multiple fine-tuned Swin models

- Single-model deployment at inference time

The soup model is evaluated like a standard single network, while implicitly capturing diversity from multiple fine-tuning runs.

5.4.4 Theoretical Justification

Model soups are justified by several theoretical observations:

- **Mode Connectivity:** Independently fine-tuned models from the same initialization often converge to nearby regions in weight space (local minima in similar basins) [131]
- **Weight Space Averaging:** Simple linear averaging of weights often preserves or improves generalization performance [123]
- **Ensemble Diversity:** Different initializations and hyperparameters lead to diverse models whose weights, when averaged, reduce overfitting and improve robustness
- **Loss Landscape Geometry:** The averaged weights typically lie in low-loss regions when the individual models' loss valleys are nearby [132]

These properties make model soups particularly effective for medical imaging, where robustness and generalization to diverse acquisition protocols are critical.

5.4.5 Datasets

We use the COVID-19 Radiography Database [133], a publicly available chest X-ray collection curated by Qatar University and the University of Dhaka with clinical collaborators. The dataset includes COVID-19, normal, viral pneumonia, and lung opacity cases and is intended to support automated screening research.

The latest release contains 3,616 COVID-19 cases, 10,192 normal images, 6,012 lung opacity images, and 1,345 viral pneumonia images, along with associated masks and metadata (e.g., age, gender, view position). The dataset is distributed via Kaggle.

Classification Task Configuration: Although the dataset provides four categories (Normal, COVID-19, Viral Pneumonia, Lung Opacity), we cast the task as binary classification: COVID-19 positive vs. COVID-19 negative (Normal + Viral Pneumonia + Lung Opacity). This matches a screening workflow where the key question is whether COVID-19 is present, and it increases the negative sample

size (17,549 negatives vs. 3,616 positives). Multi-class diagnosis is left for future work.

The COVID-19 Radiography Database is part of a larger initiative called the RSNA International COVID-19 Open Radiology Database (RICORD), which is a multi-institutional, multinational, expert-annotated COVID-19 imaging dataset made openly available to the research community. RICORD includes chest X-ray and computed tomography images from various sources, along with annotations and supporting clinical information. RICORD is supported by the Medical Imaging Data Resource Center (MIDRC), a collaborative effort between the Radiological Society of North America (RSNA), the American College of Radiology (ACR), and the National Institute of Biomedical Imaging and Bioengineering (NIBIB).

Below are some sample images from the COVID-19 Radiography Database, exemplifying the typical characteristics and quality found throughout the collection.

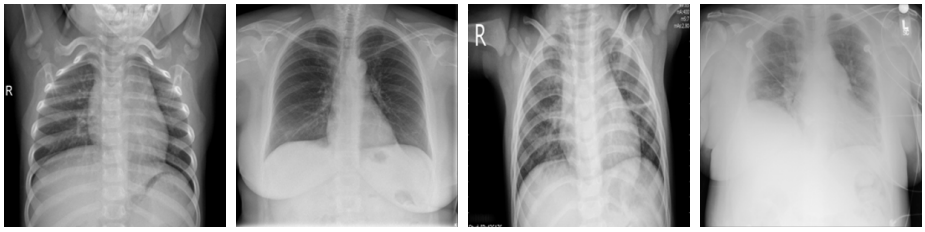

Fig. 5.3 Example of Normal, COVID, Viral Pneumonia and Lung Opacity .

5.4.6 Experimental Setup

We fine-tune Swin-Base (ImageNet-22k) and VGG19 (ImageNet-1k) independently using SGD (lr=1×10^{-4}, momentum=0.9), batch size 32, for 200 epochs. These runs support both the prediction-level ensemble and the within-backbone model soup analysis.

The preparation of the dataset is presented in Table 5.1. We address binary classification (COVID-19 positive vs negative).

Table 5.1 Dataset split used in experiments.

Split	Normal	COVID	Viral-Pneumonia	Lung-Opacity	Total
Train	7 134	2 530	942	4 208	14 814
Validation	1 019	362	134	601	2 116
Test	2 039	724	269	1 203	4 235

We evaluate using 5-fold cross-validation. Training is performed on an AMD

EPYC 7413 CPU (24 cores) with 128GB RAM and an NVIDIA A100 40GB GPU. The implementation uses PyTorch.

Hyperparameter Configuration: We maintained consistent hyperparameters across all folds to ensure fair comparison.

- **Optimizer:** Stochastic Gradient Descent (SGD)
- **Learning Rate:** 1×10^{-4}
- **Momentum:** 0.9
- **Weight Decay:** 1×10^{-4}
- **Batch Size:** 32
- **Epochs:** 200 (with early stopping patience of 20 epochs)
- **Loss Function:** Binary Cross-Entropy with Logits

5.5 Evaluation Metrics

We report Accuracy, Sensitivity (Recall), Specificity, Precision, and F1-score derived from the confusion matrix (TP, TN, FP, FN). Table 5.2 summarizes the formulas and interpretations.

Table 5.2 Evaluation metrics used in this study.

Metric	Formula	Interpretation
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	Overall correctness of the model's predictions.
Sensitivity (Recall)	$\frac{TP}{TP+FN}$	Proportion of actual COVID-19 cases correctly identified. Crucial for screening to minimize missed cases.
Specificity	$\frac{TN}{TN+FP}$	Proportion of negative cases correctly identified. Reduces false alarms.
Precision	$\frac{TP}{TP+FP}$	Proportion of predicted positive cases that are actually positive.
F1-Score	$\frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}}$	Harmonic mean of precision and recall, providing a balanced view for imbalanced data.

5.6 Experimental Results

5.6.1 Quantitative Analysis

Table 5.3 summarizes test performance over five folds. Both individual backbones perform well (98.88% accuracy), and the prediction-averaged ensemble performs best at 99.32% ± 0.19 accuracy and 99.31% ± 0.19 F1. The ensemble also has the smallest variance across folds (0.19% vs. 0.48% for VGG19 and 0.39% for Swin), indicating greater stability across train/test splits.

Table 5.3 Five-fold mean ± std Accuracy and F1 (%). **Bold** indicates the best result.

Model	Accuracy	F1-score
VGG19	98.88 ± 0.48	98.88 ± 0.48
Swin	98.88 ± 0.39	98.87 ± 0.39
Ensemble (Swin+VGG19)	**99.32 ± 0.19**	**99.31 ± 0.19**

5.6.2 Confusion Matrix and Error Analysis

Figure 5.4 shows a representative confusion matrix from one fold of the ensemble model. Classification is near-perfect with only a handful of errors. Most misclassifications are false negatives (COVID-19 cases classified as negative), typically arising from subtle or atypical radiographic findings. False positives (negative cases labeled COVID-19) are rarer but occur when non-COVID pneumonias mimic ground-glass patterns. Sensitivity and specificity both exceed 99%, making the approach viable for screening where false negatives (missed cases) and false positives (unnecessary follow-up) both carry clinical costs.

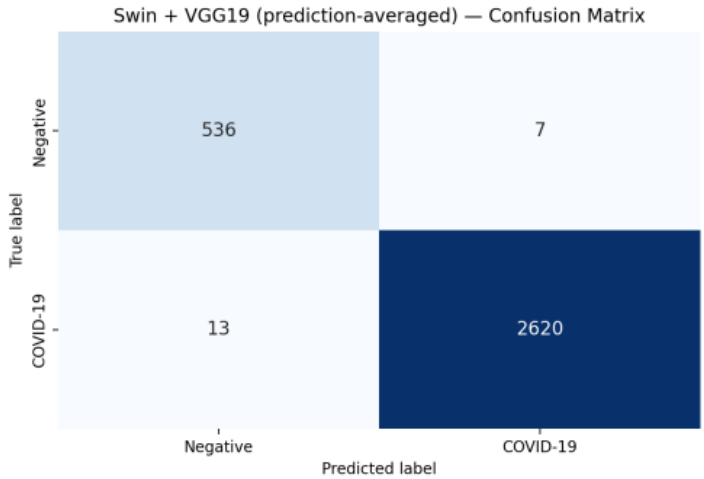

Fig. 5.4 Confusion matrix for the Swin–VGG19 prediction-averaged ensemble from the latest run.

5.6.3 Training Dynamics

Figure 5.5, Figure 5.6, and Figure 5.7 illustrate the training progression. Training loss decreases smoothly and reaches a plateau by epoch 8–10. Both training and validation accuracy rise rapidly and stabilize, with minimal gap between them, indicating effective learning without significant overfitting. Overall, the model converges efficiently within 10 epochs. In our experiments we continue training up to the configured maximum (200 epochs) while monitoring validation performance, so the early plateau indicates fast convergence rather than early termination.

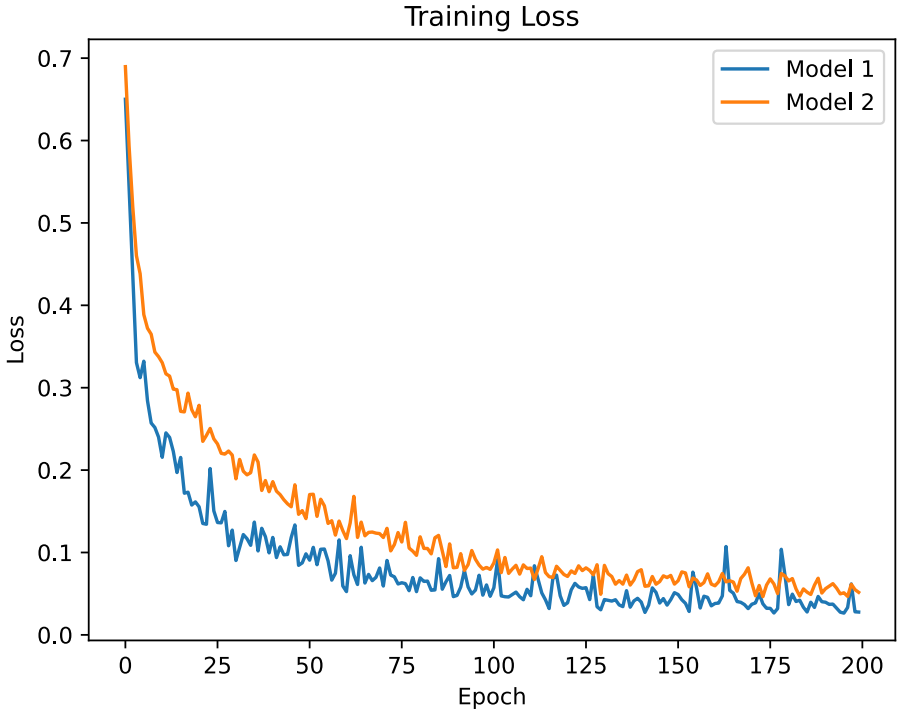

Fig. 5.5 Training Loss

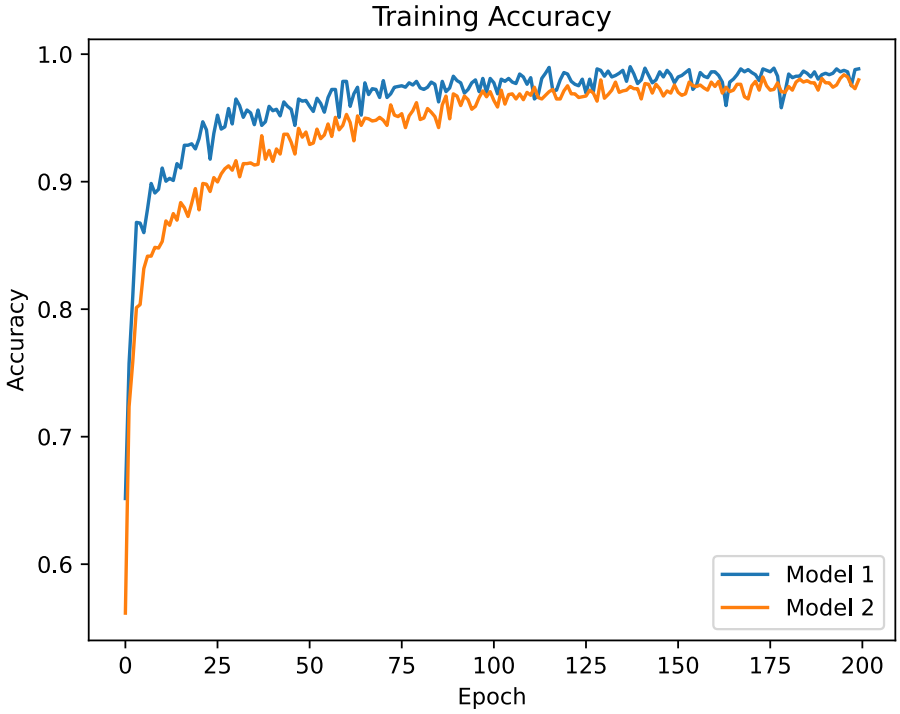

Fig. 5.6 Training Accuracy

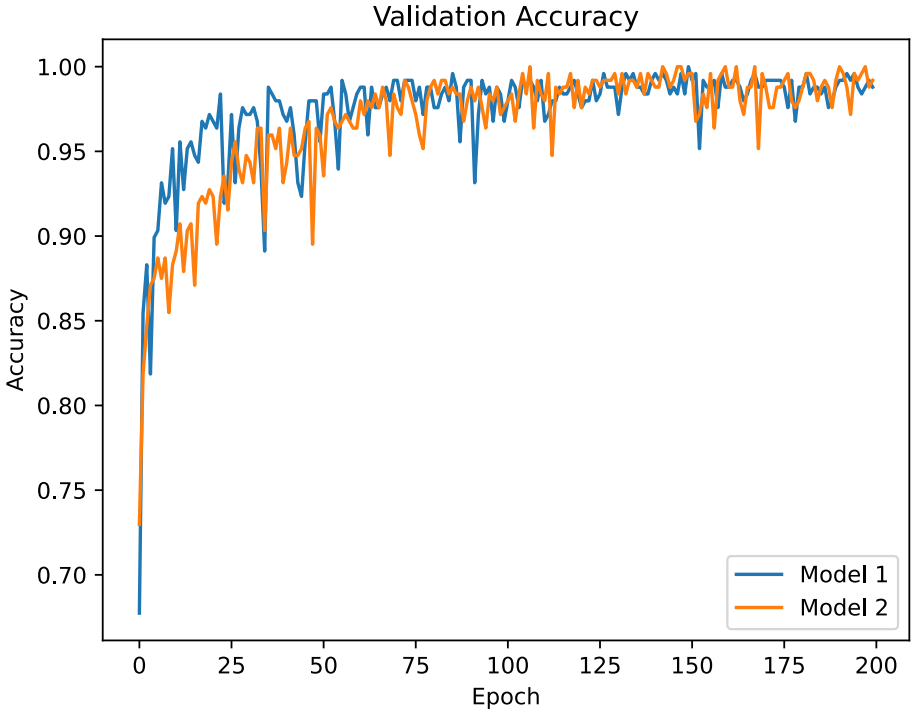

Fig. 5.7 Validation Accuracy

5.6.4 ROC Analysis

The Receiver Operating Characteristic (ROC) curves in Figure 5.8 further substantiate the model's robustness. The high Area Under the Curve (AUC) across folds aligns with the quantitative results, confirming the model's exceptional capability to distinguish between COVID-19 positive and negative cases with high sensitivity (>99%) and specificity (>99%).

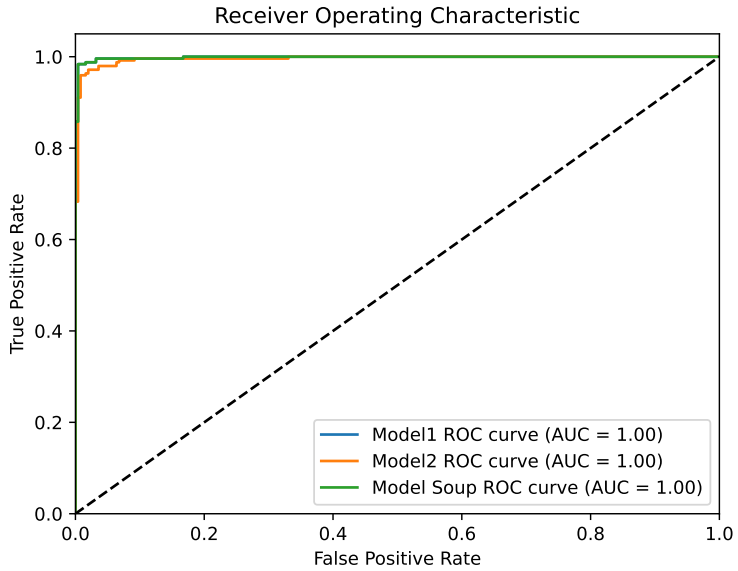

Fig. 5.8 ROC curves for the Swin–VGG19 prediction-averaged ensemble.

5.6.5 Comparison of Overall Performance

The ensemble gains ∼0.44 percentage points over individual models and halves variance. This translates to ∼39% relative error reduction. Complementarity drives success: Swin captures long-range dependencies (diffuse patterns), while VGG19 captures local textures (fine-grained variations).

5.7 Ablation Study and Sensitivity Analysis

To validate the robustness of our proposed approach and justify the choice of hyperparameters, we conducted a comprehensive ablation study and sensitivity analysis. This section details the impact of different backbone architectures, learning rates, batch sizes, and training durations on the model's performance.

5.7.1 Impact of Backbone Architecture

We evaluated the performance of the VGG19 and Swin Transformer models individually and compared them with the proposed prediction-averaged ensemble (Swin+VGG19). The results indicate that while both individual models achieve high accuracy, the ensemble leverages their complementary strengths—local texture features from VGG19 and global context from Swin Transformer—to achieve robust performance and stability, as detailed in Section 5.6.

Unlike weight-space model soups (which require models with the same backbone), a cross-architecture Swin+VGG19 combination is naturally implemented as a prediction-level ensemble and therefore incurs the inference cost of running both

models. When single-model deployment is required, within-backbone model soups (e.g., averaging multiple Swin fine-tunes) offer an efficiency-oriented alternative.

5.7.2 Hyperparameter Sensitivity Analysis

We analyzed the sensitivity of our model to key hyperparameters: learning rate, batch size, and training epochs. This analysis guided our final selection of hyperparameters for the main experiments.

5.7.2.1 Learning Rate

The learning rate (LR) is a critical hyperparameter governing the convergence speed and stability. We tested three orders of magnitude: 10^{-3}, 10^{-4}, and 10^{-5}.

Table 5.4 Impact of learning rate on model training dynamics. **Bold** indicates the selected parameter.

Learning Rate	Observation	Result
1×10^{-3}	Fast convergence	Unstable training, high variance in validation loss
$\mathbf{1 \times 10^{-4}}$	**Balanced**	**Optimal trade-off between speed and stability**
1×10^{-5}	Slow convergence	Similar final performance but requires more epochs

Based on these observations, we selected a learning rate of 1×10^{-4} as it provided the most stable convergence profile.

5.7.2.2 Batch Size

Batch size affects both the generalization capability of the model and the memory footprint during training.

Table 5.5 Impact of batch size on training efficiency and generalization. **Bold** indicates the selected parameter.

Batch Size	Observation	Result
16	Slow training	Similar performance to batch size 32
32	**Fits 16GB GPU**	**Good generalization and efficient memory usage**
64	Fast training	Slight degradation in generalization (overfitting risk)

We chose a batch size of 32 as it maximized GPU utilization on our hardware (16GB VRAM) without compromising generalization performance.

5.7.2.3 Training Duration (Epochs)

Determining the optimal number of epochs is crucial to prevent underfitting or overfitting.

Table 5.6 Analysis of convergence behavior over different training durations. **Bold** indicates the selected parameter.

Epochs	Observation	Result
50	Loss still decreasing	Underfitting; model has not fully converged
100	Loss decreasing slowly	Approaching convergence
150	Minimal improvement	Diminishing returns
200	**Loss plateau**	**Optimal convergence point**

Our analysis showed that the models converged effectively around 200 epochs. Training beyond this point yielded diminishing returns and increased the risk of overfitting, validating our choice of 200 epochs for the final experiments.

5.8 Chapter Summary

In this chapter, we presented a compact, high-performance framework for chest X-ray classification based on complementary deep features. We use a prediction-averaged ensemble of Swin Transformer and VGG19 to leverage their complementary inductive biases, achieving exceptional accuracy (99.32%) with sensitivity and specificity both exceeding 99% on the COVID-19 Radiography dataset. For deployment scenarios where a single network is required, model soups can be constructed by averaging weights of multiple fine-tuned models that share the same backbone (e.g., multiple Swin runs), enabling single-model inference while retaining part of the robustness benefits associated with ensembling.

CHAPTER 6

Conclusion and Strategic Outlook

6.1 Summary of Contributions

This doctoral thesis has established a unified computational framework for enhancing the reliability of medical imaging pipelines, addressing two critical bottlenecks: the incoherent synthesis of multimodal data and the instability of deep diagnostic models.

6.1.1 Contribution 1: Resolving the Spectral–Spatial Dichotomy in Fusion

We formulated and empirically validated a **Hybrid Two-Scale Fusion Architecture** that improves upon conventional decomposition-based fusion baselines. By casting fusion as a multi-criteria design problem, we introduced a "Divide-and-Conquer" strategy with a clear separation of responsibilities:

1. **Texture Fidelity:** We used transfer learning (TL_VGG19) to build a deep feature-driven rule for high-frequency fusion, improving edge and texture retention compared with simple hand-crafted selection rules.

2. **Energy Balance:** We used the Equilibrium Optimization Algorithm (EOA) as a global weight solver for low-frequency (base-layer) fusion, mitigating the contrast washout behavior of naive averaging.

This hybrid design yielded fused images that preserve anatomical structure (MRI) while retaining functional signal (PET/SPECT), supported by statistically significant gains in information-theoretic measures (e.g., Entropy and Mutual Information) on the C4 subset.

6.1.2 Contribution 2: Robust Generalization via Ensembling and Within-Backbone Model Soups

In the domain of automated classification, we explored model combination strategies for COVID-19 screening from chest X-ray images. For maximum performance and stability, we used a **prediction-averaged ensemble** of Swin Transformer and VGG19 to leverage complementary inductive biases, yielding a **low-variance diagnostic agent** (99.32% accuracy). For deployment scenarios where a single network is required, we studied **model soups** in the standard within-backbone setting (averaging weights across multiple fine-tuned models that share the same architecture), enabling single-model inference while retaining part of the robustness benefits associated with ensembling.

6.2 Future Research Trajectories

Our findings suggest several high-value avenues for subsequent investigation, categorized by their position in the clinical pipeline.

6.2.1 Advances in Multimodal Fusion

- **End-to-End Differentiable Architectures:** While our hybrid decomposition method is effective, it remains a multi-stage process. Future work should explore fully differentiable Fusion Transformers that can learn the decomposition, fusion, and reconstruction steps simultaneously, potentially removing the need for explicit frequency separation.
- **Real-Time Optimization:** The heuristic nature of EOA involves a computational cost during the iterative search. We propose investigating Amortized Optimization—training a lightweight neural network to predict the optimal EOA parameters instantly—to enable real-time fusion during live scans.
- **Perceptual Loss Functions:** Moving beyond statistical metrics (MSE, Entropy), we aim to incorporate "Clinical Perceptual Loss," a metric derived from radiologist eye-tracking data, to drive the fusion process towards features that are clinically relevant rather than just statistically dominant.

6.2.2 Evolution of Diagnostic Screening

- **Uncertainty Quantification:** High accuracy is insufficient for clinical trust. We plan to integrate Bayesian approximation or Conformal Prediction layers into the screening system (ensemble or soup-based deployment)

to output calibrated uncertainty scores, alerting clinicians when the model is operating outside its domain of validity.

- **Federated Model Soups:** Extending within-backbone weight averaging to a federated learning setting, where hospitals share model gradients rather than patient data, could allow for the construction of a global, privacy-preserving COVID-19 screening model.

6.3 Limitations and Threats to Validity

While this thesis demonstrates clear improvements in both low-level fusion and high-level classification, several limitations must be acknowledged:

1. **Dataset Scope:** Evaluating fusion methods on the *Whole Brain Atlas* (C1-C4) and classification on COVID-19 datasets is standard, but clinical diversity remains a bottleneck. Results may vary on data from different scanners or with distinct pathology distributions not represented in these public benchmarks.

2. **Generalization to Other Modalities:** Our primary validation focused on MRI/PET for fusion and Chest X-rays for classification. Although the underlying principles (spectral/spatial decomposition, ensemble learning) are generic, direct application to other pairings (e.g., CT/MRI) or 3D volumetric data requires further tuning of hyperparameters (e.g., EOA search space, VGG19 layer selection).

3. **Clinical Validation:** The quantitative metrics ($Q^{AB/F}$, Accuracy) and qualitative assessments presented here are proxies for clinical utility. A rigorous user study involving radiologists is required to confirm that the statistically superior "fused" images or "ensemble predictions" effectively translate into reduced diagnostic error or faster reading times in a real-world setting.

List of Publications

1. **Giang Son Tran, Tom Herbreteau, Chi Cuong Nguyen, Thi Phuong Nghiem, Cuong Do Oanh, Huy Duc Nguyen**. "Using Deep Learning Model to Assist in Determining Location of Pulmonary Nodules on CT Scans". *Proceedings of Fundamental and Applied IT Research (FAIR)*, 2020.

2. **Cuong Do Oanh, Chi Mai Luong, Giang Son Tran**. "A Hybrid Quantum-Classical Neural Network Utilizing the Lion Optimizer for COVID-19". *Proceedings of Fundamental and Applied IT Research (FAIR)*, 2023.

3. **Cuong Do Oanh, Chi Mai Luong, Phu Hung Dinh, Giang Son Tran**. "An Efficient Approach to Medical Image Fusion Based on Optimization and Transfer Learning with VGG19". *Biomedical Signal Processing and Control*, 87(A) (SCIE, Q1), 2023.

4. **Quoc Viet Kieu, Vinh Nam Huynh, Thi Phuong Nghiem, Cuong Do Oanh, Giang Son Tran**. "A New Method for Medical Image Fusion Based on Gaussian Blur Filter and Robinson Compass Operator". *Journal of Computer Science and Cybernetics*, 40(2), 135-146, 2024.

5. **Cuong Do Oanh, Tran Hong Diep, Giang Son Tran, Thi Phuong Nghiem, Chi Mai Luong**. "Deep Learning for Multimodal Medical Image Fusion: A Concise Review". *The 17th International Conference on Computer Science and its Applications (CSA 2025)*, Springer-LNEE (indexed by SCOPUS and EI), 2025.

6. **Cuong Do Oanh, Giang Son Tran, Thi Phuong Nghiem, Chi Mai Luong**. "Complementary Deep Features for COVID-19 Screening: Combining Hierarchical Vision Transformers and Convolutional Networks". *The 17th International Conference on Computer Science and its Applications (CSA 2025)*, Springer-LNEE (indexed by SCOPUS and EI), 2025.

REFERENCES

[1] H. University. Mri dataset. http://www.med.harvard.edu/AANLIB/, 2024. Accessed: (July, 2024).

[2] S. U. Khan, M. A. Khan, M. Azhar, F. Khan, Y. Lee, and M. Javed. Multimodal medical image fusion towards future research: A review. *Journal of King Saud University-Computer and Information Sciences*, 35(8):101733, 2023.

[3] N. Goswami, A. Dogra, S. Bakshi, and B. Goyal. Multimodal medical image fusion: Techniques, databases, evaluation metrics, and clinical applications – a comprehensive review. *The Open Neuroimaging Journal*, 18:e18744400417835, 2025.

[4] S. Singh, H. Singh, G. Bueno, O. Deniz, S. Singh, H. Monga, P. Hrisheekesha, and A. Pedraza. A review of image fusion: Methods, applications and performance metrics. *Digital Signal Processing*, 137:104020, 2023.

[5] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili. Equilibrium optimizer: A novel optimization algorithm. *Knowledge-based systems*, 191:105190, 2020.

[6] J. Jaworek-Korjakowska, P. Kleczek, and M. Gorgon. Melanoma thickness prediction based on convolutional neural network with vgg-19 model transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[8] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In

Proceedings of the IEEE/CVF international conference on computer vision, pages 10012–10022, 2021.

[9] A. P. James and B. V. Dasarathy. Medical image fusion: A survey of the state of the art. *Information fusion*, 19:4–19, 2014.

[10] B. J. Pichler, M. S. Judenhofer, and C. Pfannenberg. Multimodal imaging approaches: pet/ct and pet/mri. *Molecular Imaging I*, pages 109–132, 2008.

[11] G. Antoch and A. Bockisch. Combined pet/mri: a new dimension in whole-body oncology imaging? *European journal of nuclear medicine and molecular imaging*, 36:113–120, 2009.

[12] A. Webb. *Introduction to Biomedical Imaging*. John Wiley and Sons, 2018.

[13] G. Qi, L. Chang, Y. Luo, Y. Chen, Z. Zhu, and S. Wang. A precise multi-exposure image fusion method based on low-level features. *Sensors*, 20(6):1597, 2020.

[14] B. Qi, L. Jin, G. Li, Y. Zhang, Q. Li, G. Bi, and W. Wang. Infrared and visible image fusion based on co-occurrence analysis shearlet transform. *Remote Sensing*, 14(2):283, 2022.

[15] R. Gautam and S. Datar. Application of image fusion techniques on medical images. *Int J Curr Eng Technol*, 7(1):161–167, 2017.

[16] S. Liu, Y. Lu, J. Wang, S. Hu, J. Zhao, and Z. Zhu. A new focus evaluation operator based on max–min filter and its application in high quality multi-focus image fusion. *Multidimensional Systems and Signal Processing*, 31(2):569–590, 2020.

[17] P. Yugander, C. Tejaswini, J. Meenakshi, B. S. Varma, M. Jagannath, et al. Mr image enhancement using adaptive weighted mean filtering and homomorphic filtering. *Procedia Computer Science*, 167:677–685, 2020.

[18] Y. Chen, L. Cheng, H. Wu, F. Mo, and Z. Chen. Infrared and visible image fusion based on iterative differential thermal information filter. *Optics and Lasers in Engineering*, 148:106776, 2022.

[19] A. Yehia, H. Elhifnawy, and M. Safy. An improved integrated intensity-hue-saturation with stationary wavelet transform multi-sensor image fusion approach. In *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pages 153–159. IEEE, 2019.

[20] V. Ankarao, V. Sowmya, and K. Soman. Multi-sensor data fusion using nihs transform and decomposition algorithms. *Multimedia Tools and Applications*, 77:30381–30402, 2018.

[21] W. Wang and F. Chang. A multi-focus image fusion method based on laplacian pyramid. *J. Comput.*, 6(12):2559–2566, 2011.

[22] H. Tan, X. Huang, H. Tan, and C. He. Pixel-level image fusion algorithm based on maximum likelihood and laplacian pyramid transformation. *Journal of Computational Information Systems*, 9(1):327–334, 2013.

[23] X. Wang, Y. Shen, Z. Zhou, and L. Fang. An image fusion algorithm based on lifting wavelet transform. *Journal of Optics*, 17(5):055702, 2015.

[24] M. Khan, N. Mufti, et al. Comparison of various edge detection filters for anpr. In *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, pages 306–309. IEEE, 2016.

[25] P. J. Burt and R. J. Kolczynski. Enhanced image capture through fusion. In *1993 (4th) international Conference on Computer Vision*, pages 173–182. IEEE, 1993.

[26] C. H. Anderson. Filter-subtract-decimate hierarchical pyramid signal analyzing and synthesizing technique, January 5 1988. US Patent 4,718,104.

[27] H. Jin and Y. Wang. A fusion method for visible and infrared images based on contrast pyramid with teaching learning based optimization. *Infrared Physics and Technology*, 64:134–142, 2014.

[28] H. Xu, Y. Wang, Y. Wu, and Y. Qian. Infrared and multi-type images fusion algorithm based on contrast pyramid transform. *Infrared Physics and Technology*, 78:133–146, 2016.

[29] S. Iqbal and H. Singh. A variational approach for multifocus image fusion in dct domain. *J. Crit. Rev*, 7(19):730–737, 2020.

[30] M. Wang and X. Shang. A fast image fusion with discrete cosine transform. *IEEE Signal Processing Letters*, 27:990–994, 2020.

[31] J. J. Benedetto, I. Konstantinidis, and M. Rangaswamy. Phase-coded waveforms and their design. *IEEE Signal Processing Magazine*, 26(1):22–31, 2009.

[32] H. Ochoa and K. Rao. A hybrid dwt-svd image-coding system (hdwtsvd). In *2003 46th Midwest Symposium on Circuits and Systems*, volume 2, pages 532–535. IEEE, 2003.

[33] S. Singh, H. Singh, A. Gehlot, J. Kaur, and Gagandeep. Ir and visible image fusion using dwt and bilateral filter. *Microsystem Technologies*, 29(4):457–467, 2023.

[34] J. E. Fowler. The redundant discrete wavelet transform and additive noise. *IEEE Signal Processing Letters*, 12(9):629–632, 2005.

[35] R. Singh, M. Vatsa, and A. Noore. Multimodal medical image fusion using redundant discrete wavelet transform. In *2009 Seventh International Conference on Advances in Pattern Recognition*, pages 232–235. IEEE, 2009.

[36] B. Yang and S. Li. Multifocus image fusion and restoration with sparse representation. *IEEE transactions on Instrumentation and Measurement*, 59(4):884–892, 2009.

[37] A. Ellmauthaler, C. L. Pagliari, and E. A. Da Silva. Multiscale image fusion using the undecimated wavelet transform with spectral factorization and nonorthogonal filter banks. *IEEE Transactions on image processing*, 22(3):1005–1017, 2012.

[38] S. Farokhi, S. M. Shamsuddin, U. U. Sheikh, J. Flusser, M. Khansari, and K. Jafari-Khouzani. Near infrared face recognition by combining zernike moments and undecimated discrete wavelet transform. *Digital Signal Processing*, 31:13–27, 2014.

[39] A. A. Suraj, M. Francis, T. Kavya, and T. Nirmal. Discrete wavelet transform based image fusion and de-noising in fpga. *Journal of Electrical Systems and Information Technology*, 1(1):72–81, 2014.

[40] S. Narasimhan, M. Harish, A. Haripriya, and N. Basumallick. Discrete cosine harmonic wavelet transform and its application to signal compression and subband spectral estimation using modified group delay. *Signal, Image and Video Processing*, 3:85–99, 2009.

[41] B. Shreyamsha Kumar. Multifocus and multispectral image fusion based on pixel significance using discrete cosine harmonic wavelet transform. *Signal, Image and Video Processing*, 7:1125–1143, 2013.

[42] Y. Chen, N. Deng, B. Xin, W. Xing, and Z. Zhang. A novel multi-focus image fusion method of nonwovens based on ghm multiwavelet transform technology. *Textile Research Journal*, 89(14):2870–2879, 2019.

[43] M. M. Laftah. Image denoising using multiwavelet transform with different filters and rules. *Int. J. Interact. Mob. Technol.*, 15(15):140, 2021.

[44] F. Nencini, A. Garzelli, S. Baronti, and L. Alparone. Remote sensing image fusion using the curvelet transform. *Information fusion*, 8(2):143–156, 2007.

[45] A. S. Koshki, M. Zekri, M. R. Ahmadzadeh, S. Sadri, and E. Mahmoudzadeh. Extending contour level set model for multi-class image segmentation with application to breast thermography images. *Infrared Physics and Technology*, 105:103174, 2020.

[46] D. Anandhi and S. Valli. An algorithm for multi-sensor image fusion using maximum a posteriori and nonsubsampled contourlet transform. *Computers and Electrical Engineering*, 65:139–152, 2018.

[47] X. Feng. Infrared and visible image fusion based on nsct and deep learning. *Journal of Information Processing Systems*, 14(6):1405–1419, 2018.

[48] R. Hou, R. Nie, D. Zhou, J. Cao, and D. Liu. Infrared and visible images fusion using visual saliency and optimized spiking cortical model in non-subsampled shearlet transform domain. *Multimedia Tools and Applications*, 78:28609–28632, 2019.

[49] X. Xiaoxue, C. Fucheng, S. Weiwei, and L. Fu. Multi-modal medical image fusion based on non-subsampled shearlet transform. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 8(2):41–48, 2015.

[50] A. Khare, M. Khare, and R. Srivastava. Shearlet transform based technique for image fusion using median fusion rule. *Multimedia Tools and Applications*, 80(8):11491–11522, 2021.

[51] Q. Zhang and B.-l. Guo. Multifocus image fusion using the nonsubsampled contourlet transform. *Signal processing*, 89(7):1334–1346, 2009.

[52] N. Alseelawi, H. T. Hazim, and H. T. Salim ALRikabi. A novel method of multimodal medical image fusion based on hybrid approach of nsct and dtcwt. *International Journal of Online and Biomedical Engineering*, 18(3), 2022.

[53] B. Ma, Y. Zhu, X. Yin, X. Ban, H. Huang, and M. Mukeshimana. Sesf-fuse: An unsupervised deep model for multi-focus image fusion. *Neural Computing and Applications*, 33:5793–5804, 2021.

[54] J. Jose, N. Gautam, M. Tiwari, T. Tiwari, A. Suresh, V. Sundararaj, and R. MR. An image quality enhancement scheme employing adolescent iden-

tity search algorithm in the nsst domain for multimodal medical image fusion. *Biomedical Signal Processing and Control*, 66:102480, 2021.

[55] S. Polinati, D. P. Bavirisetti, K. N. Rajesh, G. R. Naik, and R. Dhuli. The fusion of mri and ct medical images using variational mode decomposition. *Applied Sciences*, 11(22):10975, 2021.

[56] C. D. Oanh, T. H. Diep, G. S. Tran, T. P. Nghiem, and C. M. Luong. Deep learning for multimodal medical image fusion: A concise review, 2025. Accepted.

[57] H. Li, X.-J. Wu, and J. Kittler. Densefuse: A fusion approach to infrared and visible images. *IEEE Transactions on Image Processing*, 28:2614–2623, 2019.

[58] J. Ma, W. Yu, P. Liang, C. Li, and J. Jiang. Fusiongan: A generative adversarial network for infrared and visible image fusion. *Information Fusion*, 48:11–26, 2019.

[59] J. Ma et al. Ddcgan: Dual-discriminator conditional gan for multi-modal image fusion, 2020. arXiv preprint.

[60] Y. Zhao, Q. Zheng, P. Zhu, X. Zhang, and W. Ma. Tufusion: A transformer-based universal fusion algorithm for multimodal images. *IEEE Trans Circuits Systems Video Technology*, 2023.

[61] M. A. Azam, K. B. Khan, S. Salahuddin, E. Rehman, S. A. Khan, M. A. Khan, S. Kadry, and A. H. Gandomi. A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics. *Computers in biology and medicine*, 144:105253, 2022.

[62] H. Hermessi, O. Mourali, and E. Zagrouba. Multimodal medical image fusion review: Theoretical background and recent advances. *Signal Processing*, 183:108036, 2021.

[63] Y. Zhang, Y. Liu, P. Sun, H. Yan, X. Zhao, and L. Zhang. IFCNN: A general image fusion framework based on convolutional neural network. *Information Fusion*, 54:99–118, feb 2020.

[64] C. Zhao et al. Mhw-gan: Multidiscriminator hierarchical wavelet generative adversarial network for multimodal image fusion. *IEEE Trans Neural Networks Learning Systems*, 2023.

[65] M. Safari, A. Fatemi, and L. Archambault. Medfusiongan: multimodal medical image fusion using an unsupervised deep generative adversarial network. *BMC Med Imaging*, 23(1):203, 2023.

[66] W. Li, Y. Zhang, G. Wang, Y. Huang, and R. Li. Dfenet: A dual-branch feature enhanced network integrating transformers and convolutional feature learning for multimodal medical image fusion. *Biomed Signal Process Control*, 80:104402, 2023.

[67] Y. Song et al. Destrans: A medical image fusion method based on transformer and improved densenet. *Comput Biol Med*, 174:108463, 2024.

[68] J. Di, W. Guo, J. Liu, L. Ren, and J. Lian. Ammnet: A multimodal medical image fusion method based on an attention mechanism and mobilenetv3. *Biomed Signal Process Control*, 96:106561, 2024.

[69] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018.

[70] W. Xue, S. Wang, and J. Ma. Flfuse-net: A fast and lightweight infrared and visible image fusion. *Infrared Physics and Technology*, 127:104383, 2022.

[71] T. Zhou, F. Zhang, X. Lin, et al. Giae-net: A gradient–intensity oriented model for multimodal lung tumor image fusion. *Engineering Science and Technology, an International Journal*, 54:101727, 2024.

[72] W. Zhang, B. Li, Y. Liu, et al. End-to-end dynamic residual focal transformer network for multimodal medical image fusion. *Neural Computing and Applications*, 36:1–23, 2024.

[73] Y. Zhang, R. Nie, J. Cao, C. Ma, and C. Wang. Ss-ssan: a self-supervised subspace attentional network for multimodal medical image fusion. *Artif Intell Rev*, 56(Suppl 1):421–443, 2023.

[74] W. Tan, P. Tiwari, H. M. Pandey, C. Moreira, and A. K. Jaiswal. Multimodal medical image fusion algorithm in the era of big data. *Neural Computing and Applications*, jul 2020.

[75] X. Deng and P. L. Dragotti. Deep convolutional neural network for multimodal image restoration and fusion. *IEEE Trans Pattern Anal Mach Intell*, 43(10):3333–3348, 2021.

[76] C. Cheng, H. Li, X. Qu, and J. Ma. Mufusion: A general unsupervised image fusion network based on memory unit. *Information Fusion*, 92:80–92, 2023.

[77] Z. Zhao, Y. Zheng, Y. Zhang, Y. Liu, and J. Ma. Cdd-fuse: Correlation-driven dual-branch feature decomposition for multi-modality image fusion. In *CVPR*, pages 5906–5916, 2023.

[78] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

[79] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

[80] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

[81] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. R. Acharya. Automated detection of covid-19 cases using deep neural networks with x-ray images. *Computers in biology and medicine*, 121:103792, 2020.

[82] L. Wang, A. Q. Lin, and A. Wong. Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific Reports*, 10(1):19549, 2020.

[83] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, J. Bai, Y. Lu, Z. Fang, Q. Song, et al. Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. *Radiology*, 296(2):E65–E71, 2020.

[84] X. Li, F. Zhou, H. Tan, W. Zhang, and C. Zhao. Multimodal medical image fusion based on joint bilateral filter and local gradient energy. *Information Sciences*, 569:302–325, 2021.

[85] S. Polinati and R. Dhuli. Multimodal medical image fusion using empirical wavelet decomposition and local energy maxima. *Optik*, 205:163947, mar 2020.

[86] P.-H. Dinh. A novel approach based on three-scale image decomposition and marine predators algorithm for multi-modal medical image fusion. *Biomedical Signal Processing and Control*, 67:102536, may 2021.

[87] K. Parmar, R. K. Kher, and F. N. Thakkar. Analysis of ct and mri image fusion using wavelet transform. In *2012 international conference on communication systems and network technologies*, pages 124–127. IEEE, 2012.

[88] P. Porwik and A. Lisowska. The haar-wavelet transform in digital image processing: its status and achievements. *Machine graphics and vision*, 13(1/2):79–98, 2004.

[89] K. Sharma and M. Sharma. Image fusion based on image decomposition using self-fractional fourier functions. *Signal, image and video processing*, 8:1335–1344, 2014.

[90] X.-S. Yang. Metaheuristic optimization: algorithm analysis and open problems. In *International symposium on experimental algorithms*, pages 21–32. Springer, 2011.

[91] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili. Equilibrium optimizer: A novel optimization algorithm. *Knowledge-Based Systems*, 191:105190, mar 2020.

[92] P.-H. Dinh. Multi-modal medical image fusion based on equilibrium optimizer algorithm and local energy functions. *Applied Intelligence*, apr 2021.

[93] S. K. Dinkar, K. Deep, S. Mirjalili, and S. Thapliyal. Opposition-based laplacian equilibrium optimizer with application in image segmentation using multilevel thresholding. *Expert Systems with Applications*, 174:114766, jul 2021.

[94] Y. Gao, Y. Zhou, and Q. Luo. An efficient binary equilibrium optimizer algorithm for feature selection. *IEEE Access*, 8:140936–140963, 2020.

[95] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[96] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, pages 177–186. Springer, 2010.

[97] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[98] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014.

[99] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[100] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[101] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pages 23965–23998. PMLR, 2022.

[102] C. Xydeas and V. Petrovic. Objective image fusion performance measure. *Electronics Letters*, 36:308, 2000.

[103] G. Piella and H. Heijmans. A new quality metric for image fusion. In *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*. IEEE, 2003.

[104] M. A. Azam, K. B. Khan, S. Salahuddin, E. Rehman, S. A. Khan, M. A. Khan, S. Kadry, and A. H. Gandomi. A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics. *Computers in Biology and Medicine*, 144:105253, may 2022.

[105] K. A. Johnson et al. The whole brain atlas. http://www.med.harvard.edu/AANLIB/, 2001.

[106] B. Li, H. Peng, and J. Wang. A novel fusion method based on dynamic threshold neural p systems and nonsubsampled contourlet transform for multi-modality medical images. *Signal Processing*, 178:107793, jan 2021.

[107] X. Li, X. Zhang, and M. Ding. A sum-modified-laplacian and sparse representation based multimodal medical image fusion in laplacian pyramid domain. *Medical and Biological Engineering and Computing*, 57(10):2265–2275, aug 2019.

[108] A. Wang, X. Luo, Z. Zhang, and X.-J. Wu. A disentangled representation based brain image fusion via group lasso penalty. *Frontiers in Neuroscience*, 16, jul 2022.

[109] A. Sufyan, M. Imran, S. A. Shah, H. Shahwani, and A. A. Wadood. A novel multimodality anatomical image fusion method based on contrast and structure extraction. *International Journal of Imaging Systems and Technology*, 32(1):324–342, aug 2021.

[110] L. Tang, J. Yuan, and J. Ma. Image fusion in the loop of high-level vision tasks: A semantic-aware real-time infrared and visible image fusion network. *Information Fusion*, 82:28–42, 2022.

[111] J. Ma, L. Tang, F. Fan, J. Huang, X. Mei, and Y. Ma. Swinfusion: Cross-domain long-range learning for general image fusion via swin transformer. *IEEE/CAA Journal of Automatica Sinica*, 9(7):1200–1217, 2022.

[112] L. Tang, J. Yuan, H. Zhang, X. Jiang, and J. Ma. Piafusion: A progressive infrared and visible image fusion network based on illumination aware. *Information Fusion*, 83-84:79–92, 2022.

[113] W. Tang, F. He, Y. Liu, and Y. Duan. Matr: Multimodal medical image fusion via multiscale adaptive transformer. *IEEE Transactions on Image Processing*, 31:5134–5149, 2022.

[114] L. Zhang, H. Li, R. Zhu, and P. Du. An infrared and visible image fusion algorithm based on ResNet-152. *Multimedia Tools and Applications*, 81(7):9277–9287, jan 2022.

[115] M. Wortsman, G. Horton, C. Guestrin, L. Fei-Fei, and I. Stoica. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *arXiv preprint arXiv:2203.05556*, 2022.

[116] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2015.

[117] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[118] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[119] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Desai, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[120] T. G. Dietterich. Ensemble methods in machine learning. *Multiple Classifier Systems*, pages 1–15, 2000.

[121] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pages 6402–6413, 2017.

[122] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, A. D'Amour, B. Lakshminarayanan, and J. Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under shift. In *Advances in neural information processing systems*, pages 13991–14002, 2019.

[123] P. Izmailov, D. Podoprikhin, T. Garipov, D. P. Vetrov, and A. G. Wilson. Averaging weights leads to wider optima and better generalization. In *Uncertainty in Artificial Intelligence*, pages 876–885. PMLR, 2019.

[124] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Shi. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, volume 33, pages 18613–18624, 2020.

[125] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. Mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[126] S. Yun, D. Han, S. J. Oh, J. Elm, and Y. Yoo. Cutmix: Regularization strategy to train strong classifiers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[127] N. Tajbakhsh, J. M. Jeyaraman, D. Maskarykova, J. Liang, M. B. Gotway, and Y. Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.

[128] M. A. Morid, A. Borjali, and G. Del Fiol. A scoping review of transfer learning research on medical image analysis using imagenet pre-trained networks. *Computer methods and programs in biomedicine*, 196:105735, 2021.

[129] C. Matsoukas, J. A. Haslum, M. Sorensen, and K. Smith. What makes training multi-modal classification networks hard? *arXiv preprint arXiv:2203.01454*, 2022.

[130] M. Raghu, S. Kornblith, C. Zhang, and M. L. Littman. Transfusion: understanding transfer learning with applications to imaging and tabular data. *arXiv preprint arXiv:1912.00913*, 2019.

[131] T. Garipov, P. Izmailov, D. Podoprikhin, D. P. Vetrov, and A. G. Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Advances in neural information processing systems*, pages 8803–8812, 2018.

[132] J. Frankle and M. Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR, 2020.

[133] M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, et al. Can ai help in screening viral and covid-19 pneumonia? *arXiv preprint arXiv:2003.13145*, 2020.