Refining the Visibility of Diagnostic Information in X-Ray Imaging via Machine Learning

DISSERTATION

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

angenommen durch die Fakultät für Informatik
der Otto-von-Guericke-Universität Magdeburg

von M.Sc. Dominik Eckert

geb. am 06.06.1992 in Bamberg

Gutachterinnen/Gutachter

Prof. Dr. Sebastian Stober
Prof. Dr. Christian Ledig
Dr. Ludwig Ritschl

Magdeburg, den 10.06.2025

Otto von Guericke University Magdeburg

Department of Computer Science
Artificial Intelligence Lab

Ph.D. Thesis

Refining the Visibility of Diagnostic Information in X-Ray Imaging via Machine Learning

Author:
Dominik Eckert

May 23, 2025

Supervisors

Prof. Dr. Sebastian Stober

Otto von Guericke University
Universitätsplatz 2
39106 Magdeburg, Germany

Dr. Ludwig Ritschl
Dr. Julia Wicklein
Dr. Christopher Syben

Siemens Healthineers
Siemensstraße 3
91301 Forchheim, Germany

Eckert, Dominik:
Refining the Visibility of Diagnostic Information in X-Ray Imaging via Machine Learning
Ph.D. Thesis, Otto von Guericke University
Magdeburg, 2024.

Contents

Abstract

Machine learning, particularly deep learning, is a key driver of advancements in various research fields and industrial applications, including medical imaging. Its capability to abstract and manipulate complex data patterns holds significant potential, especially in X-ray image formation. In this process, acquired detector signals must be transformed into human-interpretable images to accurately depict the patient's anatomy and identify potential pathologies. Despite their potential, integrating deep learning models into clinical practice entails distinct challenges. Reliability is essential, as corrupted information may result in incorrect diagnoses. Moreover, traditional image quality metrics often fail to adequately quantify a model's impact on diagnostic accuracy, necessitating more advanced evaluation techniques. Furthermore, the carcinogenic nature of ionizing radiation ethically precludes obtaining X-ray images solely for training purposes. In combination with strict privacy requirements for patient data, this limits the availability of comprehensive datasets. Recognizing both the potential and challenges of integrating deep learning into clinical practice, this work explores how to leverage its capabilities to optimize the conversion of recorded X-ray detector signals into human-readable X-ray images, thereby refining the visibility of diagnostic information. Moreover, by addressing the aforementioned challenges, it aims to facilitate the implementation of the proposed methods in clinical settings. Specifically, this work focuses on the investigation of the following key aspects in X-ray image processing: the removal and supression of artifacts in the X-ray detector signal, namely collimation shadows and noise, and the quantification and adjustment of differences in X-ray image impressions.

Noise in X-ray images is inversely proportional to the radiation dose and can obscure diagnostic information. Consequently, X-ray acquisitions must balance the trade-off between ionizing radiation dose and image quality. To further reduce patient radiation exposure, collimation is employed to exclude non-essential regions. However, shadows from the collimator can decrease the space available for visualizing crucial anatomy and often interfere with subsequent processing steps. Thus, this work proposes deep learning-based methods for denoising and collimator shadow segmentation in X-ray images. To overcome data scarcity and enable supervised training, we propose a physics-based simulation framework to generate matching training pairs. This framework allows for the alteration of the initial acquisition parameters of X-ray images, such as radiation dose, scatter, and collimator shadow patterns. This work demonstrates that models for collimator segmentation and denoising, trained on the simulation framework, generalize well to real-world data. Moreover, to enhance the generalization of the collimation segmentation in clinical practice, this works incorporate the geometric constraint of straight edges into the network architecture, via a differentiable Hough Transform. Additionally, to enable denoising in clinical practice, it must be ensured that tiny details are not mistaken for noise and removed. For this

reason, this work proposes a novel loss function that regularizes the network to prevent overestimation of noise, thus minimizing information loss. Moreover, a differentiated evaluation across different patient types to investigate potential biases in the model's performance is conducted. The combination of a specifically designed loss function and thorough evaluation contributes to the application of the proposed method in clinical settings.

After collimation removal and noise supression, the recored X-ray signal must be compressed into a visible range. Due to the ambiguity caused by overlapping tissues in X-ray images, there is no single optimal solution to this task. Consequently, various X-ray image impressions, also referred to as styles, have emerged to which radiologists have become accustomed due to their experience. Additionally, due to the lack of objective quantification of differences between X-ray image styles from different acquisitions and the absence of reliable, automatically adjustable X-ray image processing algorithms, the styles must be manually adjusted to cater to individual radiologists' needs. Morevoer, the automatic adjustment of X-ray image styles requires special sensitivity to preserving diagnostic information, as converting the recorded X-ray detector signal into a visible image significantly alters the signal. To address these challenges, this work proposes a novel, automatic, reliable, and interpretable algorithm based on the Local Laplacian Filter (LLF) to generate and adjust X-ray image styles. This algorithm converts recorded X-ray detector signals into human-readable X-ray images, allowing for the automatic adjustment of parameters through Stochastic Gradient Descent (SGD) to accommodate the adaptation to different X-ray image styles. Due to the inherent properties of the LLF, the optimized algorithm can be verified to ensure that no diagnostic information is lost. Furthermore, due to the lower number of parameters compared to traditional deep learning models, the algorithm can be optimized on small datasets. To address the absence of objective quantification of differences between X-ray image styles from different acquisitions, we propose a novel deep learning-based metric. This metric uses an encoder trained through Simple Siamese learning to generate X-ray image style representations without requiring labeled style distances. The encoder produces style representations independent of the anatomical structures in the X-ray images. Experiments using t-SNE analysis illustrate that the distances between these style representations correlate with the degree of style difference. Consequently, the encoder, in combination with a distance measurement between the style representations, can quantify style differences between X-ray images from different acquisitions.

The proposed methods aim to refine the visibility of diagnostic information in X-ray images by addressing the challenges of data scarcity, model reliability, and the quantification of differences in X-ray image styles. By leveraging deep learning capabilities, this work aims to facilitate the integration of advanced image processing methods into clinical practice, thereby improving diagnostic accuracy and patient care.

Zusammenfassung

Maschinelles Lernen, insbesondere deep learning (DL), treibt Fortschritte in verschiedenen Forschungsbereichen und industriellen Anwendungen, einschließlich der medizinischen Bildgebung, voran. Die Fähigkeit, komplexe Datenmuster zu abstrahieren und zu manipulieren, birgt erhebliches Potenzial, insbesondere bei der Röntgenbilderstellung. In diesem Prozess müssen erfasste Detektorsignale in für Menschen interpretierbare Bilder umgewandelt werden, um die Anatomie des Patienten genau darzustellen und mögliche Pathologien zu identifizieren. Trotz ihres Potenzials bringt die Integration von DL-Modellen in die klinische Praxis spezifische Herausforderungen mit sich. Zuverlässigkeit ist von entscheidender Bedeutung, da fehlerhafte Informationen zu falschen Diagnosen führen können. Darüber hinaus versagen traditionelle Metriken für die Bildqualität oft darin, den Einfluss eines Modells auf die diagnostische Genauigkeit angemessen zu quantifizieren, was fortschrittlichere Bewertungstechniken erforderlich macht. Zudem schließt die krebserzeugende Natur ionisierender Strahlung ethisch aus, Röntgenbilder ausschließlich zu Trainingszwecken zu erhalten. In Kombination mit strengen Datenschutzanforderungen für Patientendaten schränkt dies die Verfügbarkeit umfassender Datensätze ein. In Anbetracht des Potenzials und der Herausforderungen bei der Integration von DL in die klinische Praxis untersucht diese Arbeit, wie dessen Fähigkeiten genutzt werden können, um die Umwandlung von aufgezeichneten Röntgendetektorsignalen in menschenlesbare Röntgenbilder zu optimieren und dadurch die Sichtbarkeit diagnostischer Informationen zu verfeinern. Darüber hinaus zielt sie, durch die Überwindung der oben genannten Herausforderungen, darauf ab, die Umsetzung der vorgeschlagenen Methoden in klinischen Umgebungen zu ermöglichen. Insbesondere konzentriert sich diese Arbeit auf die Untersuchung der folgenden Schlüsselaspekte bei der Verarbeitung von Röntgenbildern: die Entfernung und Unterdrückung von Artefakten im Röntgendetektorsignal, insbesondere Kollimationsschatten und Rauschen, sowie die Quantifizierung und Anpassung von Unterschieden im Eindruck von Röntgenbildern.

Rauschen in Röntgenbildern ist umgekehrt proportional zur Strahlendosis und kann diagnostische Informationen verdecken. Folglich müssen Röntgenaufnahmen den Kompromiss zwischen ionisierender Strahlendosis und Bildqualität ausbalancieren. Um die Strahlenexposition des Patienten weiter zu reduzieren, wird die Kollimation auf die Interessensregion angewendet. Allerdings können Schatten vom Kollimator den verfügbaren Raum zur Visualisierung entscheidender Anatomien verringern und oft nachfolgende Verarbeitungsschritte stören. Daher schlägt diese Arbeit auf Deep Learning basierende Methoden zur Entrauschung und zur Segmentierung von Kollimatorschatten in Röntgenbildern vor. Um Datenknappheit zu überwinden und überwachte Trainingsmöglichkeiten zu schaffen, schlagen wir ein physikbasiertes Simulationsframework zur Generierung passender Trainingspaare vor. Dieses Framework ermöglicht die Änderung

der ursprünglichen Aufnahmeparameter von Röntgenbildern, wie Strahlendosis, Streustrahlung und Kollimatorschattenmuster. Diese Arbeit zeigt, dass Modelle zur Kollimatorsegmentierung und Entrauschung, die auf dem Simulationsframework trainiert wurden, sich gut auf reale Daten übertragen lassen. Darüber hinaus wird zur Verbesserung der Generalisierung der Kollimationssegmentierung in der klinischen Praxis die geometrische Einschränkung der geraden Kanten über eine differenzierbare Hough-Transformation in die Netzwerkarchitektur integriert. Zusätzlich muss zur Ermöglichung der Entrauschung in der klinischen Praxis sichergestellt werden, dass winzige Details nicht fälschlicherweise als Rauschen erkannt und entfernt werden. Aus diesem Grund wird in dieser Arbeit eine neuartige Verlustfunktion vorgeschlagen, die das Netzwerk reguliert, um eine Überschätzung des Rauschens zu verhindern und somit den Informationsverlust zu minimieren. Zudem wird eine differenzierte Bewertung über verschiedene Patiententypen hinweg durchgeführt, um potenzielle Verzerrungen in der Leistungsfähigkeit des Modells zu untersuchen. Die Kombination aus einer speziell entwickelten Verlustfunktion und gründlicher Bewertung trägt zur Anwendung der vorgeschlagenen Methode in klinischen Umgebungen bei.

Nach der Entfernung von Kollimation und Rauschunterdrückung muss das aufgezeichnete Röntgensignal in einen sichtbaren Bereich komprimiert werden. Aufgrund der Uneindeutigkeit, die durch sich überlappende Gewebestrukturen in Röntgenbildern entsteht, existiert keine einzige optimale Lösung für dieses Problem. Daher haben sich verschiedene Röntgenbilddarstellungen, auch als Stile bezeichnet, entwickelt, an die sich Radiologen aufgrund ihrer Erfahrung gewöhnt haben. Zudem erfordert der Mangel an objektiver Quantifizierung von Unterschieden zwischen Röntgenbildstilen aus verschiedenen Aufnahmen und das Fehlen zuverlässiger, automatisch anpassbarer Röntgenbildverarbeitungsalgorithmen eine manuelle Anpassung der Stile an die individuellen Bedürfnisse der Radiologen. Weiterhin erfordert die automatisierte Anpassung von Röntgenbildstilen besondere Sensibilität zur Erhaltung diagnostischer Informationen, da die Umwandlung des aufgezeichneten Signals des Röntgendetektors in ein sichtbares Bild das Signal erheblich verändert.

Zur Bewältigung dieser Herausforderungen wird in dieser Arbeit ein neuartiger, automatisierter, zuverlässiger und interpretierbarer Algorithmus vorgestellt, der auf dem LLF basiert, um Röntgenbildstile zu erzeugen und anzupassen. Dieser Algorithmus transformiert aufgezeichnete Röntgendetektorsignale in sichtbar darstellbare Röntgenbilder und erlaubt die automatische Anpassung von Parametern mithilfe von SGD, um sich flexibel verschiedenen Röntgenbildstilen anzupassen. Dank der inhärenten Eigenschaften des LLF kann der optimierte Algorithmus überprüft werden, um die vollständige Erhaltung diagnostischer Informationen zu gewährleisten. Des Weiteren ermöglicht die geringere Anzahl von Parametern im Vergleich zu traditionellen Deep-Learning-Modellen eine Optimierung des Algorithmus auf kleinen Datensätzen. Um den Mangel an objektiver Quantifizierung von Unterschieden zwischen Röntgenbildstilen

aus verschiedenen Aufnahmen zu überwinden, wird in dieser Arbeit eine innovative, Deep-Learning-basierte Metrik eingeführt. Dieser Ansatz nutzt einen Encoder, der durch Simple-Siamese-Lernen trainiert wird, um Röntgenbildstil-Repräsentationen ohne die Notwendigkeit gelabelter Stilabstände zu generieren. Der Encoder erstellt Stilrepräsentationen, die unabhängig von den anatomischen Strukturen in den Röntgenbildern sind. Experimente mittels t-SNE-Analyse zeigen, dass die Distanzen zwischen diesen Stilrepräsentationen mit dem Stilunterschied korrelieren. Folglich kann der Encoder in Kombination mit einer Distanzmessung zwischen den Stilrepräsentationen die Unterschiede der Stile zwischen Röntgenbildern aus verschiedenen Aufnahmen quantifizieren.

Die vorgeschlagenen Methoden verbessern die Sichtbarkeit diagnostischer Informationen in Röntgenbildern, indem sie die Herausforderungen der Datenknappheit, Modellzuverlässigkeit und der Quantifizierung von Unterschieden in Röntgenbilddarstellungen adressieren. Durch den Einsatz von Deep-Learning-Techniken zielt diese Arbeit darauf ab, die Integration fortschrittlicher Bildverarbeitungsmethoden in die klinische Praxis zu erleichtern, um sowohl die diagnostische Genauigkeit als auch die Patientenversorgung zu verbessern.

1 Introduction

On December 22, 1895, the first-ever image depicting human anatomy was recorded. The hand of Berta Röntgen was subjected to X-ray radiation for a duration of 20 minutes by the physicist Wilhelm Conrad Röntgen. This exposure revealed not only the bones of her hand but also the ring she was wearing on her finger. This discovery quickly revolutionized the field of medicine, as it became evident that X-rays could be utilized for diagnostic purposes [217, 207]. By directing X-rays towards a patient, the differential attenuation of these rays by various tissues can be leveraged to capture an image of the patient's internal anatomy on a photographic film. This marked the advent of X-ray imaging, a technique that provided unprecedented, non-invasive insight into the internal structure of the human body. However, the euphoria surrounding this discovery was soon tempered by the realization of its potential harm. As early as 1897, merely two years post Röntgen's discovery, instances of skin and tissue damage due to interaction with X-rays began to surface [236, 51]. By 1911, the first case of leukemia was linked to this radiation. This association was further substantiated by a pivotal study in 1944 by March [154], which provided compelling evidence of cancer being a potential consequence of X-ray exposure. Today, it is well-established that X-rays can induce DNA damage, leading to mutations that may result in cancer [22, 32]. Despite significant advancements in technology, a fundamental challenge in radiography persists: balancing the need for high-quality diagnostic images with the imperative to minimize radiation exposure [235]. In the wake of these challenges, the field of radiography has witnessed remarkable technological advancements. A significant leap forward has been the development of digital detectors [126]. These devices have transformed the way X-ray images are captured and processed, replacing traditional photographic film with digital means. Digital detectors capture X-ray images in a digital format, which opens up a plethora of possibilities for image processing. With the aid of sophisticated algorithms and computational power, these digital images can be enhanced, manipulated, and analyzed in ways that were previously unimaginable. This digital revolution has made diagnostic features enhanced visible and discernible [126].

Moreover, digitalization has facilitated the integration of machine learning and especially deep learning algorithms in radiography. These algorithms, which can be trained to recognize patterns in images or manipulate image features [211], have been revolutionizing the field of photographic image processing over the past decade. We posit that the immense potential of machine learning algorithms for enhancing X-ray images remains largely untapped.

1.1 Motivation & Research Objectives

In this work, we explore the potential of machine learning to enhance diagnostic accuracy without necessitating an increase in radiation dosage. From the acquisition of X-ray images to the presentation of processed images to radiologists, various stages are crucial to ensure optimal diagnostic accuracy. For this reason, we have identified four distinct research objectives aimed at improving this process:

1. Investigate machine learning's ability to detect collimation shadows, aiming to enhance the field of view of the X-ray images and establish a basis for further post-processing algorithms.

2. Explore the feasibility of reducing noise levels in X-ray images without compromising diagnostic information, focusing particularly on denoising projections in Digital Breast Tomosynthesis (DBT), where noise is prevalent and radiologist face the especially challenging task of detecting carcinomas (cancer tissue) [119].

3. Optimize an X-ray image processing pipeline using machine learning, with the goal of aligning more closely with the preferred X-ray image appearance of individual radiologists to reduce the effort required to adjust to varying image impressions.

4. Utilize deep learning to quantify differences in X-ray image appearances, with the aim of developing a metric to quantify the challenges radiologists face due to varying image appearances.

1.1.1 Collimator Shadow Detection

Collimation is a technique frequently used in X-ray imaging to minimize unnecessary exposure to non-target areas of the body. However, when present in the final X-ray image, they limit the available space for displaying relevant diagnostic information. Furthermore, despite their lack of diagnostic relevance, shadows from collimation can interfere with processing algorithms. Therefore, the detection and removal of these shadows is vital for facilitating accurate diagnoses and further processing of a X-ray image [142].

The task of detecting collimated areas, however, is significantly challenging due to the phenomenon of scattered radiation. This scattered radiation can reach the detector located behind the collimator, resulting in a brighter collimator shadow [204]. As a result, the collimator shadow can closely mimic the region of interest, posing a challenge for analytical algorithms to differentiate between the two.

Given their proven effectiveness in image processing tasks [128], Artificial Neural Networks (ANNs) hold potential as a valuable solution for collimator shadow detection. Therefore, we investigate the potential of training ANNs to detect and eliminate collimation artifacts from X-ray images. This is achieved by incorporating prior knowledge about the possible geometric manifestations of collimator shadows by utilizing a differentiable Hough Transform (HT) in the network architecture.

Additionally, we explore the possibility of developing a simulation pipeline based on the physical principles underlying collimator shadow formation. This pipeline aims to augment clinical data with collimation shadows, addressing the scarcity of data, particularly labeled data, in medical imaging.

The findings of this study, a collaborative effort between Benjamin El-Zein and the author of this thesis, were showcased in a conference proceeding at Workshop on Data Augmentation Labeling and Imperfections (DALI) as part of the Medical Image Computing and Computer Assisted Intervention (MICCAI) 2023, where they were honored with the best paper award. The publication details are as follows:

> *[252] Benjamin El-Zein & Dominik Eckert et al. "A Realistic Collimated X-Ray Image Simulation Pipeline". In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer. 2023, pp. 137–145.*

1.1.2 Denoising Digital Breast Tomosynthesis Projections

In the intricate field of X-ray imaging, Full Field Digital Mammography (FFDM) stands out as a critical tool in the early detection and diagnosis of breast cancer [7]. However, interpreting these images presents several challenges. First, the similar X-ray attenuation properties between normal and cancerous breast tissues make it difficult to recognize potential malignancies [69]. Second, microcalcifications, which are minute calcium deposits in the breast and key indicators of breast cancer [17], can be easily obscured by noise due to their small size. Third, the Cancer Detection Rate (CDR) is influenced by different breast types; dense breast tissues can mask the presence of microcalcifications and carcinomas, disadvantaging patients with dense breasts [58]. This issue is further compounded by the fact that dense breast tissue is a risk factor for breast cancer [155].

The complexity of breast imaging has spurred the development of DBT [36], a technique that captures multiple images, or projections, of the breast from different angles. This provides a quasi three-dimensional view, addressing challenges inherent to FFDM, such as overlapping tissues and anatomical clutter [94, 58]. Moreover, DBT substantially improves the CDR in dense breast screens compared to low-density screens [135, 50].

Building upon the progress in DBT, a novel development, Synthetic Mammograms (SMs), has been introduced. These are quasi-2D FFDM images processed from the DBT projections. The introduction of SMs marks a significant stride in the evolution of breast imaging, combining the benefits of both approaches. It consolidates information acquired from various DBT projections into a single 2D image. This has the potential to eliminate the need for a separate FFDM acquisition in a routine DBT procedure [220, 249].

While DBT and SM have brought significant advancements in breast imaging, they also introduce a major challenge. To ensure that the total radiation dose remains within the same range as FFDM, each projection in DBT must be acquired with a significantly lower dose. Consequently, the noise level in each projection is higher than in FFDMs, potentially obscuring fine details like microcalcifications, which are critical for accurate diagnosis [15].

This increase in noise motivates the second part of this work, which explores the potential of deep learning in noise removal from DBT projections. The development follows the premise of not compromising diagnostic information and ensuring reliable operation across all patient groups, given the potentially fatal consequences of missing a cancer diagnosis.

Enhancing image quality in DBT projections can facilitate superior reconstruction of the 3D volume, thereby improving the detection of breast cancer [228].

The necessity of training data for denoising is indisputable. However, the nonexistence of pairs of low and high dose images, e.g. due to ethical considerations of not exposing patients to unnecessary radiation merely for data generation, poses a challenge. Similar to the collimation detection, we investigate the possibility of generating physically plausible noise patterns in FFDM images to simulate the noise level in DBT projections. We subsequently investigate deep-learning methodologies, with a particular emphasis on loss functions, to ensure the preservation of diagnostically relevant information.

Lastly, we investigate the performance of the developed deep learning models across various patient groups to ascertain the reliability and enable possible applicability in real world clinical settings.

The findings of this research have been presented at three conferences and published in one academic journal. Notably, the work received the best poster award at the SPIE Medical Imaging 2022 conference. The publications are as follows:

> *[64] Dominik Eckert et al. "Deep learning-based denoising of mammographic images using physics-driven data augmentation". In: Bildverarbeitung für die Medizin 2020: Algorithmen–Systeme–Anwendungen. Proceedings des Workshops vom 15. bis 17. März 2020 in Berlin. Springer. 2020, pp. 94–100.*

> *[62] Dominik Eckert et al. "Deep learning based denoising of mammographic x-ray images: an investigation of loss functions and their detail-preserving properties". In: Medical Imaging 2022: Physics of Medical Imaging. Vol. 12031. SPIE. 2022, pp. 455–462.*

> *[65] Dominik Eckert et al. "Guidance to Noise Simulation in X-ray Imaging". In: Bildverarbeitung für die Medizin 2024: Proceedings, German Conference on Medical Image Computing, Erlangen, March 10-12, 2024. Springer-Verlag. 2024, p. 184.*

> *[63] Dominik Eckert et al. "Deep learning based tomosynthesis denoising: a bias investigation across different breast types". In: Journal of Medical Imaging 10.6 (2023), pp. 064003–064003.*

The following publication, primarily based on the research of Magdalena Herbst, serves as a supplement to the proposed denoising methods. It aims to ensure the reliability of the suggested deep learning method in practical applications:

> *[90] Magdalena Herbst et al. "Noise gate: a physics-driven control method for deep learning denoising in x-ray imaging". In: Medical Imaging 2024: Physics of Medical Imaging. Vol. 12925. SPIE. 2024, pp. 736–739.*

The proposed denoising research contributes to the development of the next-generation DBT system, MAMMOMAT B.brilliant, by Siemens Healthineers. In this system, AI-based denoising is employed to provide a more natural image background for the corresponding Insight 2D image mode.

> *[88] Daan Hellingman et al. "Fast, high-resolution wide-angle digital breast tomosynthesis with MAMMOMAT B.brilliant."*

1.1.3 X-ray Image Impression and Appearance

Besides the removal of artifacts in the image, the presentation of the X-ray image itself is crucial for the diagnostic process. The human eye is unable to simultaneously perceive the entire spectrum of signals acquired by the X-ray detector. Con-

sequently, the signal must be compressed in to a visible range of approximately 500-1000 shades of gray [18]. Inevitably, information is lost in this compression. At the same time crucial diagnostic information is often contained in subtle changes [33]. Hence, the processing of the signal into a visual range is a complex and ambiguous process. Over time, various processing algorithms have emerged, each yielding unique X-ray image impressions. These impressions, often referred to as 'styles', carry their own set of advantages. Besides the actual presentation, the capability of the radiologists to extract the relevant diagnostic information depends on the training, personal preferences and neuro-physilogical processes [33, 117]. Therefore, radiologists have preferred X-ray image styles. Moreover errors, which arise when radiologists fail to recognize the diagnostic information available in the image, account for 60-80% of all diagnostic errors [234]. Consequently, alterations in the style, that is, the presentation of diagnostically relevant information, can influence the diagnostic accuracy of radiologists. Nonetheless, radiologists encounter variations in X-ray machines, equipment modifications, and improvements in image processing pipelines, continuously.

Quantifying these stylistic changes presents a challenge, particularly when comparing images with non-matching content, that is, two X-ray images from different acquisitions. Moreover, these differences significantly influence the process of reading X-ray images. Vendors aim to tailor the image processing pipeline to the radiologist's preferred style, using a set of example images from previous acquisitions. This customization seeks to align with the radiologist's preferred X-ray image style, thereby minimizing the effort required to adjust to varying image impressions. However, this manual process is not only time-consuming and error-prone, but also highly subjective due to the absence of a metric for measuring the style distance between two different X-rays. In this work, we address these issues in two ways. First, we explore the automation of the X-ray image processing algorithm adaptation. Second, we explore the potential of deep learning metrics to abstract high-level features, with the aim of developing a deep-learning-based style metric to quantify differences in the appearance of non-matching X-ray images.

Automatic Adaption of X-ray Image Processing Pipelines

Modern X-ray image processing pipelines primarily depend on the weighting of different frequency bands, a method that has been superseded in photographic image processing [232, 163]. We initially investigate the potential of using the LLF [174], recognized as the state-of-the-art in photographic image processing [54], to enhance the visibility of diagnostic information in X-ray imaging. Following this, we focus on the automatic adaptation of the X-ray image processing pipeline. To do so, we explore the potential of optimizing the parameters of LLF with gradient

descent to automatically match a desired image impression. Furthermore, we investigate its limitations and examine potential improvements, particularly to its remapping function.

The findings of this research have been accepted for presentation at the International Symposium on Biomedical Imaging (ISBI) 2025 conference:

> *[61] Dominik Eckert et al. "An Interpretable X-ray Style Transfer via Trainable Local Laplacian Filter". In: arXiv preprint arXiv:2411.07072 (2024).*

Quantifying X-ray Image Style Differences

Our investigation into quantifying stylistic differences between X-ray images involves multiple stages. Initially, we focus on generating varied image impressions using a transparent linear analysis pipeline. Owing to its linearity, the differences between the generated styles can be traced back to the processing steps. This facilitates the creation of a robust training and test dataset. Moreover, due to its transparency, the pipeline ensures the reproducibility of our research. Secondly, we aim to overcome the non-existence of style distance labels. To this end, we investigate the application of unsupervised training methods to extract high-level X-ray image style features. Thirdly, we explore the feasibility of constructing a style metric based on the trained deep learning model. Lastly, we assess the practical applicability of this metric on clinically relevant image impressions, constructed with a confidential vendor pipeline.

The findings regarding style quantification are in submission to the academic journal Transactions on Medical Imaging (TMI):

> *[66] Dominik Eckert et al. "StyleX: A Trainable Metric for X-ray Style Distances". In: arXiv preprint arXiv:2405.14718 (2024)*

1.2 Thesis Structure

The remainder of this thesis is structured as follows:

Chapter 2 provides an overview of the theoretical background, discussing the essential concepts and methods necessary for understanding the research objectives, as well as the inspiration behind this work. It covers a range of topics from the physical principles underlying X-ray imaging and the peculiarities of mammographic imaging, to key concepts in analytical X-ray image processing, and concludes with a focus on machine learning concepts, specifically deep learning methods.

Chapter 3 discusses the development of a noise simulation pipeline. This pipeline is utilized both as part of the collimation simulation and for the generation of noise patterns in DBT projections.

Chapter 4 presents our research on detecting collimation shadows. It discusses the development of the collimation simulation pipeline as well as the deep learning methodologies investigated for detecting the shadows.

Chapter 5 discusses our research on denoising DBT projections, with a focus on the impact of different loss functions and the development of a novel loss function tailored for mammographic images. It also covers our investigation into the applicability of the developed deep learning models across different patient groups and breast densities, to ensure their reliability in real-world clinical settings.

Chapter 6 details our research on the automatic adaptation of X-ray image processing pipelines to provide a desired image appearance. It is built around the LLF and its gradient-based optimization. The chapter also discusses improvements to its remapping function.

Chapter 7 presents our research on quantifying variations in X-ray image appearances using a deep learning-based style metric. It discusses the creation of diverse image styles, the unsupervised training methods used, and the evaluation of the developed metric on clinically relevant image styles.

Chapter 8 concludes the thesis by summarizing the insights gained by our research and discussing the potential implications of our work on the field of radiography. It also outlines possible future research directions and the limitations of our work.

2
Background

Serving as the foundation for this thesis, this chapter provides the essential concepts and techniques upon which the work is predicated. It is divided into four distinct sections.

The first part offers an overview of the X-ray imaging domain, clarifying the physical principles that underpin it.

Furthermore, given the thesis's specific focus on mammography, the second section investigates the epidemiological and diagnostic aspects of this technique, addressing how these aspects impact the challenges related to image quality.

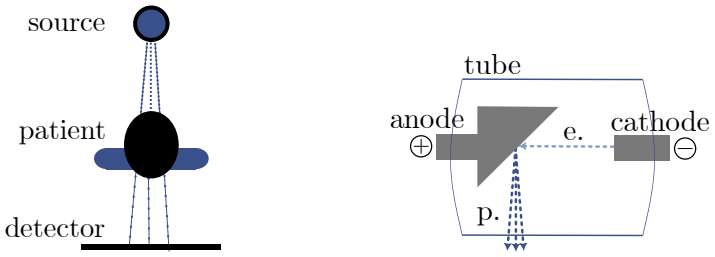
The third section presents the analytical image processing techniques employed in this research, including the Anscombe transformation, the Hough transform, and Image Pyramids.

Considering the investigation of deep learning techniques in X-ray image processing in this work, the final section establishes the foundation for deep learning. It specifically highlights those techniques that have either inspired or been applied in this research.

2.1 Physics of X-ray Imaging

X-rays, a category of electromagnetic radiation, have wavelengths ranging between 10^{-8} m and 10^{-13} m. This range is significantly shorter compared to the wavelengths of visible light, which approximate 10^{-6} m [5]. Their quantum nature allows them to be described both as particles, known as photons, and as waves, a concept encapsulated in the wave-particle duality principle [79, 180]. The energy (E) of the photon is related to its wavelength (ν) as per the equation:

$$E = \frac{hc}{\nu} \tag{2.1}$$



(a) X-ray Imaging (b) X-ray Generation

Figure 2.1: Figure (a) depicts an X-ray imaging system, comprising its three main components: the X-ray source, the patient, and the detector. Figure (b) illustrates the detailed components of the source. The process involves a cathode emitting electrons, which are then accelerated by a high voltage towards a target anode, resulting in their conversion to photons.

where h is Planck's constant (6.626×10^{-34} Js) [79] and c is the speed of light in a vacuum (3×10^{8} m/s) [35].

Compared to visible light, which has a wavelength of approximately 10^{-6} m, X-rays have higher energy and shorter wavelengths. This characteristic allows them to penetrate biological tissues, a property that is utilized in X-ray imaging. In this technique, X-rays are produced and directed towards the body. The degree to which these photons are absorbed or pass through varies depending on the tissue type. The attenuated X-ray beam, representing the differential absorption, is subsequently captured by a detector. This detector then converts the transmitted photons into digital signals, enabling the formation of an image [148]. This process is exemplified in Fig. 2.1a.

2.1.1 X-ray Generation

X-rays for medical imaging are typically generated in a specialized vacuum tube known as an X-ray tube. This tube consists of two electrodes: the cathode and the anode, as depicted in Fig. 2.1b. The cathode, negatively charged, houses a filament that, when heated, emits electrons through thermionic emission. This process, driven by an electric current, enables the filament's outer electrons to overcome their binding energy, creating a free electron cloud or space charge. As the heat corresponds to the tube current, a higher current results in more emitted electrons [148]. Simultaneously, a high voltage applied between the cathode and anode charges the anode positively, driving the free electrons towards it. Therefore, the electrons' kinetic energy is proportional to the tube voltage. The anode, usually composed of high atomic number material such as tungsten, serves as the target for these accelerated electrons [148, 195].

Upon collision of the high-speed electrons with the anode in the X-ray tube, the kinitic energy of the electrons is converted into electromagnetic radiation due to two mechanisms:

Characteristic radiation occurs when a high-speed electron displaces an inner-shell electron in the anode's target atom, leaving a vacancy. An outer shell electron fills this vacancy, and its transition to a lower energy level emits energy in the form of an X-ray photon. The energy of this photon corresponds to the energy difference between the two shells, yielding a characteristic X-ray spectrum unique to the anode material [35].

Bremsstrahlung, is the main X-ray source in an X-ray tube. It occurs when a high-speed electron is deflected by the electric field of the nucleus of a target atom in the anode, leading to energy loss emitted as an X-ray photon. Unlike characteristic radiation, the photon's energy in Bremsstrahlung radiation can vary, resulting in a continuous X-ray spectrum [148].

Due to the physical principle of energy conservation [43], the maximum energy of the generated X-ray photons ($E_{\max}$) is directly related to the tube voltage (U) by the equation:

$$E_{\max} = e \cdot U \tag{2.2}$$

where e is the electron charge (1.602×10^{-19} coulombs) [35].

However, only a minor fraction of about 1% of the electron energy is converted into X-ray photons. The majority of the energy is dissipated as heat, necessitating cooling mechanisms in the X-ray tube to avoid overheating [148].

Thus while the tube current determines the number of X-ray photons produced, the tube voltage influences the amount energy of these photons.

2.1.2 X-Ray Attenuation

As X-ray photons traverse a patient's body, they undergo attenuation due to interactions with body tissues. This attenuation primarily results from the following effects:

Photoelectric Absorption: This phenomenon occurs when an X-ray photon with sufficient energy interacts with an inner shell electron of an atom. The photon transfers all its energy to the electron, causing its ejection, a process known as ionization. The energy of the photon must exceed the electron's binding energy to the atom. Any excess energy is converted into the kinetic energy of the ejected electron. The probability of photoelectric absorption is inversely proportional to the cube of the photon's energy, meaning lower-energy X-ray photons are more likely to be absorbed [250].

Compton Scattering: This occurs when an X-ray photon interacts with a loosely bound outer shell electron, transferring part of its energy and causing the photon to deflect with reduced energy. The scattered photon will have less energy, and therefore a longer wavelength, than the incident photon. The probability of Compton scattering occurring is approximately independent of the energy of the photon for diagnostic X-ray energies, but it decreases slightly as the energy increases [183].

Rayleigh Scattering, the primary form of scattering, happens when an X-ray photon's energy is small compared to an atom's ionization energy. The photon interacts with an atom's electron, exciting the entire atom without causing ionization or electron ejection. This energy is immediately re-emitted as a photon with the same wavelength but possibly a different direction. In diagnostic imaging's energy range, this interaction is unlikely, accounting for less than 5% of X-ray interaction above 70 keV in soft tissues [35, 148].

Pair Production: This effect involves an X-ray photon's energy converting into an electron-positron pair near an atom's nucleus. This process only occurs when the photon's energy is above a certain threshold (1.022 MeV). Therefore, pair production does not play a significant role in diagnostic radiology, which typically uses X-ray photons with energies below this threshold [35].

The combined effects of these interactions result in a spectrum of attenuations within the patient's body, which create contrast in the final image. However, scatter, primarily from Compton scattering, can degrade image quality by creating a fogging effect due to photon deflection from their original path. Additionally, the amount of photoelectric absorption, which depends on the photon energy and the tissue's atomic number, makes the choice of tube voltage critical for determining image contrast. Lower tube voltages generate low-energy photons that are more likely to be absorbed, thus providing better contrast for soft tissue imaging. Conversely, higher tube voltages produce high-energy photons less likely to be absorbed, making them suitable for dense tissue imaging. The tube current, which determines the number of X-ray photons produced, can also enhance image contrast.

2.1.3 Quantum Noise

Inherent uncertainties, known as quantum or Poisson noise, exist in the number of X-ray photons due to attenuation and the randomness of photon generation. Influenced by the Heisenberg uncertainty principle, which posits that the exact position and momentum of an electron cannot be simultaneously known, there is an inherent uncertainty in the timing and energy of each photon emission [79]. Additionally, randomness in the occurrence of attenuation events contributes to

the uncertainty in the number of photons reaching the detector. As a result, the number of X-ray photons reaching the detector during a specific exposure time exhibits statistical variations, following a Poisson distribution [179].

It describes the probability of z photons hitting one detector pixel based on the average photon arrival rate λ. It is defined as:

$$P(z|\lambda) = \frac{\lambda^z e^{-\lambda}}{z!} \tag{2.3}$$

Since λ varies per pixel, the noise also varies accordingly. Hence, Poisson noise is signal-dependent. It is crucial to note that in the Poisson distribution the mean and variance are equal to the mean photon arrival rate [109]:

$$\sigma^2 = \mu = \lambda\,. \tag{2.4}$$

This establishes that the ratio $\frac{\sqrt{\lambda}}{\lambda}$ represents the relationship between the standard deviation and the mean. Consequently, it can be inferred that the relative uncertainty, and by extension, the relative noise within the image, decreases proportionally with an increase in the mean arrival rate, λ [148].

2.1.4 Photon Recording

The X-ray photon's, attenuated after traversing the patient's body, are captured by a digital X-ray detector and transformed into digital signals for image creation. The process of photon recording in digital flat panel X-ray detectors involves the following four steps [122]:

1. **Photon Conversion:** The sensitive layer of the detector, known as the scintillator, captures and converts X-ray photons into visible light for further processing. However, the scintillator material may distort the spatial information of the emitted light, resulting in image blurring [148]. It is worth noting that in digital mammography, the process differs; instead of converting photons into visible light, electrons are directly converted into electrical charges [148].
2. **Signal Readout:** A Thin Film Transistor (TFT) or Complementary Metal-Oxide-Semiconductor (CMOS) array reads the electrical charges. Each pixel in the array corresponds to a specific location on the detector, and the charge collected at each pixel represents the X-ray intensity at that location [122].
3. **Digital Conversion:** The electrical signals undergo amplification and are subsequently converted into digital signals by an Analog-to-Digital Converter (ADC). This conversion process introduces uncertainties, which manifest as white Gaussian noise, commonly known as electronic noise.

Notably, this noise level is constant and does not depend on the photon count [146, 122].

4. **Image Formation:** The digital signals are processed to form an image, with each pixel value corresponding to the X-ray intensity at a specific location in the patient's body. The image can be further processed to enhance its quality and clarity.

This process enables digital detectors to capture high-resolution images with a wide dynamic range, making them crucial in modern medical imaging.

2.1.5 Influence of Physical Parameters on Image Quality

Radiation has the potential to cause ionization within the body, leading to cell damage and subsequent health risks. Thus, minimizing radiation exposure is of utmost importance. This can be accomplished by meticulously adjusting acquisition parameters such as tube voltage, tube current, and acquisition time to achieve the desired contrast.

The energy of the photons should be calibrated based on the type of tissue under examination, which can be achieved by adjusting the tube voltage, typically given in kilovolts (kV). Lower tube voltages generate low-energy photons that are more likely to be absorbed, making them suitable for soft tissue imaging. This is necessitated by the fact that the level of photoelectric absorption is dependent on both the energy of the photon and the atomic number of the tissue. Furthermore, a balance must be struck between reducing image noise and achieving the desired image quality, a process governed by the tube current and acquisition time, which determine the quantity of X-ray photons emitted and is typically given in milliampere-seconds (mAs) [148].

The quality of X-ray imaging is perpetually tasked with balancing the need for high contrast and low noise, while keeping the radiation dose as low as possible. Therefore, every opportunity to enhance image quality without enhancing the radiation dose should be seized. This underscores the importance of further processing the image algorithmically to enhance the presentation and preparation of the image.

2.2 Mammography

Mammography, a particularly challenging modality in X-ray imaging, is utilized for the screening, diagnosis, and monitoring of breast cancer treatment. The diagnostic process is notably complex due to the close resemblance between cancerous and normal tissue. Using mammography for screening necessitates exposing healthy patients to regular radiation doses. Given these circumstances,

minimizing radiation dose is paramount. Yet, the grave implications of a missed breast cancer diagnosis highlight the crucial balance that must be struck. Therefore, image processing techniques that improve the quality and diagnostic accuracy of mammograms without enhancing the radiation dose are of substantial interest. This work, as a result, focuses specifically on mammography.

2.2.1 Breast Cancer

According to the World Health Organization (WHO), cancer continues to be the second leading cause of death worldwide, responsible for 9.74 million fatalities in 2022 [168]. Among these, breast cancer is the second most prevalent, with 2.296 million cases. Furthermore, it is the most common cancer in women [210, 30]

Cancer, a collection of diseases, is characterized by uncontrolled cell growth resulting from genetic alterations. These alterations can be inherited or induced by environmental factors such as air pollution, alcohol abuse, physical inactivity, or exposure to ionizing radiation [208, 244].

Unlike healthy cells, which differentiate, perform specific functions, and eventually stop dividing, cancer cells divide uncontrollably, disrupting this balance. This abnormal division initially forms benign tumors, which are localized tissue masses. Further genetic alterations can convert these benign tumors into malignant ones that invade adjacent tissues. These malignant cells can metastasize, spreading to other parts of the body and organs through the circulatory system [40].

Breast cancer typically originates in the inner lining of milk ducts, a condition known as Ductal Carcinoma In Situ (DCIS), and constitutes 80% of all cases. Another type, Lobular Carcinoma In Situ (LCIS), originates in the lobules that supply the milk ducts with milk, accounting for 10-15% of cases [203]. In 20-30% of instances, these conditions progress to invasive breast cancer [238], specifically to Invasive Ductal Carcinoma (IDC) and Invasive Lobular Carcinoma (ILC), which invade the surrounding tissues and spread to other parts of the body.

During the process of cellular proliferation in the breast, there is an elevated production of calcium. This excess calcium can precipitate, forming microcalcifications, i.e. small calcium deposits. As such, regions of the breast exhibiting active cellular growth and division frequently harbor these microcalcifications. Therefore, clusters of microcalcifications often serve as early indicators of breast cancer [200]. Notably, 50% of all carcinomas and 90% of DCIS are characterized by microcalcifications and mammography is the sole method capable of detecting microcalcifications [101].

The risk factors for breast cancer are diverse, including genetic components such as BRCA1 and BRCA2 genes, environmental factors, and notably, age and breast

density [124, 218]. Breast density, defined as the proportion of fibroglandular tissue in the breast, is particularly significant. It is noteworthy that dense breast tissue is more prone to cancer development than fatty tissue [242]. Additionally, this dense tissue can obscure the presence of tumors in mammograms, thereby complicating breast cancer detection [242].

To standardize risk assessment, the American College of Radiology proposed the Breast Imaging Reporting and Data System (BI-RADS), which classifies breast density into four categories [245]:

1. Category A (Fatty): The breasts are almost entirely fatty, making it easier to detect abnormalities on a mammogram.
2. Category B (Scattered Areas of Fibroglandular Density): The breasts have some scattered areas of density, slightly increasing the difficulty of detecting abnormalities.
3. Category C (Heterogeneously Dense): The breasts have many areas of fibroglandular density, which can hide abnormalities and make detection harder.
4. Category D (Extremely Dense): The breasts have a high amount of fibroglandular density, significantly increasing the difficulty of detecting abnormalities.

Presently, evidence-based prevention strategies and avoidance of risk factors can prevent between 30% and 50% of cancer cases. Furthermore, early detection, coupled with suitable treatment and patient care, can significantly reduce the impact of cancer. A high cure rate is achievable for many cancers when diagnosed early and treated appropriately [169].

2.2.2 Mammography in Screening

Given the critical importance of early detection in successful cancer treatment, screening programs have been established to detect cancer in its initial stages. As a result, screening typically commences at the age of 40 and includes regular breast examinations one to two times a year. The most common screening methods are FFDM and DBT. However, new technological advancements, such as SM, have the potential to reduce radiation dose and improve diagnostic accuracy [213, 2, 47].

In digital mammography, typically referred to as FFDM, two images of each breast are acquired: the Craniocaudal (CC) view and the Mediolateral Oblique (MLO) view. To CC view captures the breast from top to bottom, while the MLO view captures the breast from the side, including the pectoral muscle and lymph node. To obtain these images, the breast is compressed between two plates to fixate and

spread the tissue apart. This compression is necessary to reduce motion blurring and tissue overlap, thereby enhancing image quality. FFDM has been especially effective in detecting microcalcifications, due to the high resolution and low noise level of the images [245, 84]. However, due to overlapping tissues, direct carcinoma detection can be challenging, particularly in dense breasts [212].

This is where DBT demonstrates its utility. It is an imaging technique that captures multiple projections of the breast from various angles, albeit not in a full 180-degree range, and reconstructs them into a quasi-3D volume [214]. For this reason, DBT exhibits lower susceptibility to tissue overlap, which leads to increased detection rates of invasive cancers and reduced recall rates for additional diagnostic imaging [6]. However, given that between 9 and 25 X-ray projections are acquired, the radiation dose for each projection must be a fraction of the dose for a single FFDM image. This requirement results in an increase in noise within the projections. This can result in microcalcifications being obscured by noise. Consequently, it has been found that the most consistent improvment in breast cancer detection is achieved when DBT is used in conjunction with FFDM [212]. It significantly increases sensitivity in CDR and notably reduces the recall rate [157], thereby decreasing the number of false positives [214]. However, this comes at the cost of a higher radiation exposure for the patient.

To address this issue of double acquisition, SM has been introduced. SM reconstructs a 2D image out of the DBT projections, with the aim to eliminate the need for additional FFDM acquisition. It has been demonstrated that the combination of SM and DBT enhances the CDR compared to using FFDM or DBT independently. However, it is still uncertain whether SM can completely substitute FFDM in the screening setting [2].

In conclusion, both DBT and FFDM have their limitations, specifically in the detection of microcalcifications and carcinomas, respectively. The combination of both yields the lowest CDR but results in a higher radiation dose. SM attempts to address this issue, but it remains uncertain whether SM can fully replace FFDM in the screening setting. We propose that algorithmic improvements, such as denoising, could be particularly beneficial in further enhancing diagnostic accuracy and potentially improving the quality of SMs.

2.2.3 Malmö Breast Tomosynthesis Screening Trial (MBTST)

The Malmö Breast Tomosynthesis Screening Trial (MBTST), a prospective, population-based diagnostic accuracy study conducted at Skåne University Hospital in Malmö, Sweden, has been a notably influential screening trial in the field of DBT and FFDM comparison. The trial's objective was to examine

the accuracy of one-view DBT in population screening, in comparison to the standard two-view FFDM.

Women aged 40 to 74 years were invited to participate in the trial, where they underwent screening with two-view FFDM followed by one-view DBT with reduced compression in the MLO view. The primary outcome measures were the sensitivity and specificity of breast cancer detection.

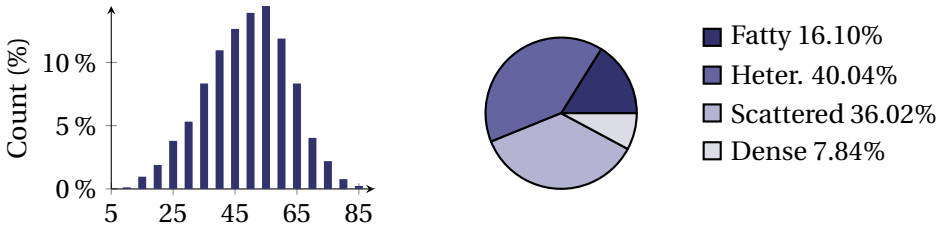
The trial commenced on February 1, 2010, and concluded on September 30, 2019. By March 2015, the recruitment of 14,851 women in Malmö was completed. The study revealed that the sensitivity was higher for DBT than for FFDM (81.1% vs 60.4%), while the specificity was slightly lower for DBT than for FFDM (97.2% vs 98.1%) [249].

This trial is particularly relevant to this work as the data acquired includes both FFDM and DBT images, along with information on breast density and thickness. Crucially, it provides the unprocessed raw data, which is indispensable for the development and evaluation of our proposed deep learning methods. Furthermore, the dataset is publicly available for research purposes [127].

The dataset provided includes both thickness and density information for 7235 patients. Breast thickness, measured in millimeters, is defined as the distance between the compression plates during a mammography examination. On the other hand, breast density is classified according to the BI-RADS categories, as previously outlined. For each patient, the dataset includes images from both the left and right breast, each captured in a CC and MLO view through the FFDM acquisition. This results in four FFDM images and two DBT images per patient. The distribution of breast thickness and density within the dataset is depicted in Fig. 2.2. The thickness values are approximately normally distributed, albeit with a slight skew towards higher measurements, culminating in 55 mm as the most frequently observed thickness. In terms of breast density, scattered and heterogeneously dense breasts are the most prevalent, each constituting approximately 40% of the dataset. Fatty breasts account for 16% of the dataset, while dense breasts represent only 8%. It is important to note that dense breasts present the greatest diagnostic challenge and are most prone to cancer development. Therefore, this data imbalance should be considered when training machine learning algorithms on this dataset.

2.3 X-ray Image Processing

Various analytical X-ray image post-processing algorithms exist to augment the diagnostic information contained within the image. This section will discuss



(a) Breast Thickness (b) Breast Density

Figure 2.2: Breast thickness and density distribution of the MBTST dataset. (a) The histogram shows the distribution of the breast thicknesses in steps of 5 mm, beginning at 5 mm and ending at 85 mm. (b) The pie chart shows the distribution of breast densities in the MBTST dataset.

three concepts that are especially pertinent to this work: the Anscombe transformation, the Hough transform, and image pyramids.

The Anscombe transformation, widely used to transform Poisson noise into signal-independent noise with a constant variance, enhances denoising applications. As a result, it is employed in Chapter 5 to facilitate the denoising of DBT projections.

The Hough transform is a technique used for line detection in images, which is particularly relevant for the detection collimator edges in X-ray images in Chapter 4.

Utilizing image pyramids to decompose an image into various frequency bands serves as a significant method for feature extraction, where each frequency band is treated as a distinct feature. This approach is particularly crucial in the context of automatic X-ray style transfer, as discussed in Chapter 6. Furthermore, this concept forms the foundation of an algorithm designed to generate diverse X-ray image impressions, a key component in the development of an X-ray style loss, as detailed in Chapter 7.

2.3.1 Anscombe Transformation

The primary source of noise in X-ray imaging is Poisson noise, which is signal-dependent, i.e., its variance is proportional to the signal intensity and varies per pixel based on its mean arrival rate, as discussed in Section 2.1.3. This makes Poisson noise particularly challenging for denoising algorithms.

Applying a Variance Stabilizing Transformation (VST) can transform Poisson noise into signal-independent noise with constant variance. This facilitates easier denoising, as the noise is no longer dependent on the signal intensity. A popular

choice for VST is the Anscombe transformation [8], which is an improvement of the square root transformation [19].

The Square Root transformation is defined as:

$$\mathcal{S}(z) = 2\sqrt{z} \tag{2.5}$$

The Anscombe transformation, on the other hand, is defined as:

$$\mathcal{A}(z) = 2\sqrt{z+\frac{3}{8}} \tag{2.6}$$

In both equations, z represents the Poisson-distributed random variable.

Derivation of the Square Root Transformation

To elucidate the variance-stabilizing property of the Anscombe transformation, we first derive this property for the square root transformation, and then explain the rationale behind the improvements made in the Anscombe transformation [151].

To demonstrate the approximated variance-stabilizing property of the square root transformation, we begin by assuming a general VST $f(z)$ that transforms a random variable z. We then calculate the variance of the first-order Taylor approximation of $f(z)$ around the mean of z namely μ. Subsequently, we illustrate that when $f(z) = 2\sqrt{z}$, representing the square root transformation, and z follows a Poisson distribution, the variance of the transformed variable approaches one [151].

Thus the first-order Taylor approximation of $f(z)$ around μ is [173]:

$$f(z) \approx f(\mu) + (z-\mu)\frac{df(\mu)}{d\mu} \tag{2.7}$$

To compute the variance, we subtract $f(z)$, effectively moving it to the other side of the equation, and then square the result, which leads to:

$$(f(z) - f(\mu))^2 = (z-\mu)^2\left(\frac{df(\mu)}{d\mu}\right)^2 \tag{2.8}$$

Upon taking the expectation of the derived expression, the variance can be approximated as follows:

$$\begin{aligned}\sigma^2\big(f(z)\big) &= \mathbb{E}\left\{\big(f(z) - f(\mu)\big)^2\right\} \\ &\approx \mathbb{E}\left\{(z-\mu)^2\left(\frac{df(\mu)}{d\mu}\right)^2\right\} \\ &= \sigma^2(z)\left(\frac{df(\mu)}{d\mu}\right)^2\end{aligned} \tag{2.9}$$

By setting $f(z) = 2\sqrt{z}$ to represent the square root transformation and leveraging the property of the Poisson distribution that $\sigma^2\{z|\mu\} = \mu$, the variance of the transformed Poisson distribution approaches one.

$$\sigma^2\big(f(z)\big) \approx \sigma^2(z)\left(\frac{d2\sqrt{\mu}}{d\mu}\right)^2 = \mu \cdot \left(\frac{2}{2\sqrt{\mu}}\right)^2 = 1 \tag{2.10}$$

Anscombe's extension

Anscombe extended this approach by incorporating the second-order Taylor polynomial. He demonstrated that introducing an additional constant b into the transformation, yielding $f(z) = 2\sqrt{z+b}$, proves beneficial with respect to the second-order Taylor polynomial. Consequently, the Anscombe transformation provides a more accurate approximation of the variance-stabilizing property compared to the square root transformation [8].

The variance of this second order Taylor approximation around μ is defined as follows [151]:

$$\sigma^2\big(f(z)\big) = 1 + \frac{3-8b}{8\mu} + \frac{32b^2-52b+17}{32\mu^2} \tag{2.11}$$

When applying the Anscombe transformation with $b = \frac{3}{8}$, the given expression simplifies as follows:

$$\sigma^2\big(f(z)\big) = 1 + \frac{1}{16\mu^2} \tag{2.12}$$

It is crucial to note that this equation approaches an approximation of one, particularly when μ assumes large values.

Transformation Example and Analysis

In Fig. 2.3, the standard deviations σ of z and their VST-transformed counterparts are plotted against the mean arrival rate, λ. It becomes evident that the Anscombe transformation rapidly approximates a standard deviation of one for $\lambda > 5$. As a result, the variance becomes independent of both the mean arrival rate and the signal intensity. In contrast, the square root transformation not only exhibits a slower convergence but also overshoots the expected variance for low values of λ.

Fig. 2.4 illustrates the impact of the Anscombe transformation on an image with three distinct pixel intensities and Poisson noise. Upon examining the noise map of the original image, the signal-dependent nature of the Poisson noise becomes evident as the noise varies across regions with different intensities. However, following the Anscombe transformation, the noise map appears uniform, indicating a constant variance throughout the image.

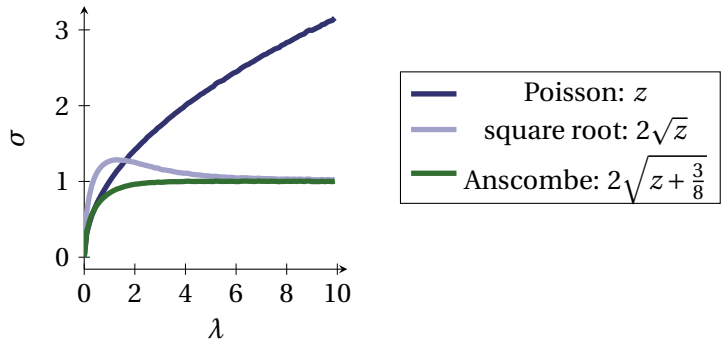
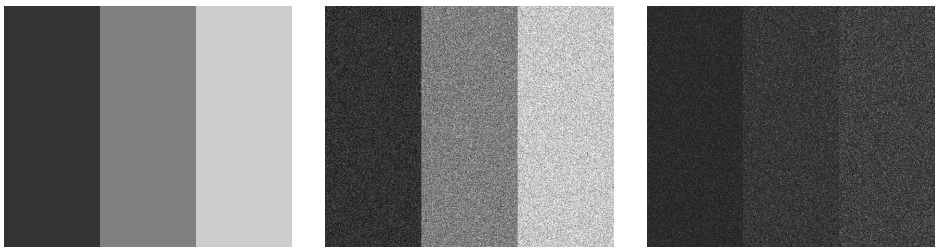
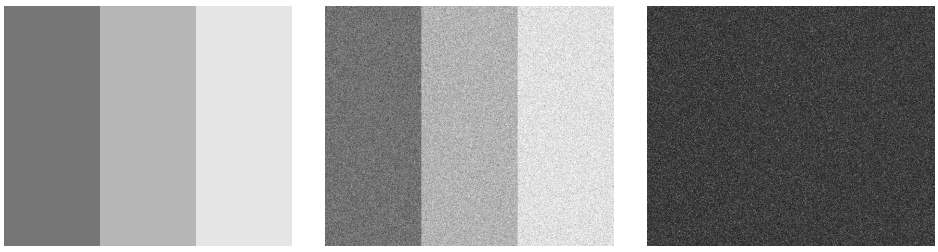


Figure 2.3: The graph depicts the standard deviation σ over the mean arrival rate λ for three types of signals: Poisson distributed signal, square root transformed Poisson distributed signal, and Anscombe transformed Poisson distributed signal.



(a) Without Noise. (b) Noisy Image (c) Poisson Noise



(d) $\mathcal{A}$(Without Noise) (e) $\mathcal{A}$(Noisy Image) (f) $\mathcal{A}$(Poisson Noise)

Figure 2.4: This figure displays an image with three distinct pixel intensities, overlaid with Poisson noise. The top row shows the original image and its corresponding Poisson noise map, illustrating the signal-dependency of the noise. The bottom row presents the same images after the Anscombe transformation, highlighting the elimination of signal-dependency in the transformed noise.

2.3.2 Hessian Normal Form

The Hessian normal form is an alternative representation to the slope-intercept form of a line in Euclidean space. It is utilized by the HT in the subsequent Section 2.3.3, and is therefore discussed in this section.

The slope-intercept form of a line is defined as:

$$y = mx + b \tag{2.13}$$

In this equation, m signifies the slope and b denotes the y-intercept. However, this equation tends towards infinity for vertical lines, rendering it inappropriate for algorithmic implementations.

The Hessian normal form, on the other hand, is defined as:

$$\rho = x\cos(\theta) + y\sin(\theta) \tag{2.14}$$

This equation utilizes ρ and θ instead of m and b, where ρ denotes the line's distance from the origin and θ represents the angle between the line's normal vector and the x-axis, as depicted in Fig. 2.5.

The Hessian normal form can be transformed into the slope-intercept form by dividing Eq. (2.14) by $\sin(\theta)$ and rearranging the terms, yielding:

$$y = -\frac{cos(\theta)}{sin(\theta)}x + \frac{\rho}{sin(\theta)}\,, \tag{2.15}$$

Thus the slope $m = -\frac{cos(\theta)}{sin(\theta)}$ and the y-intercept $b = \frac{\rho}{sin(\theta)}$ can be derived from the Hessian normal form.

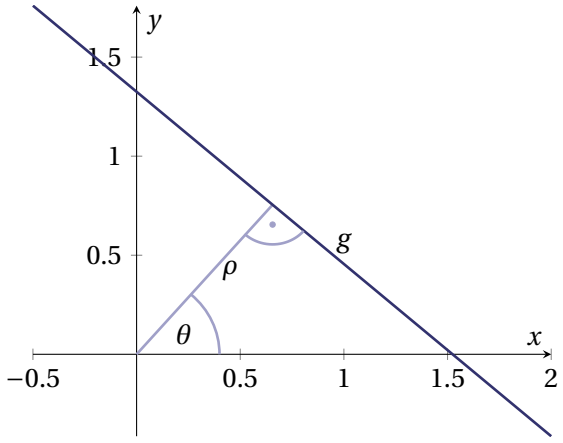
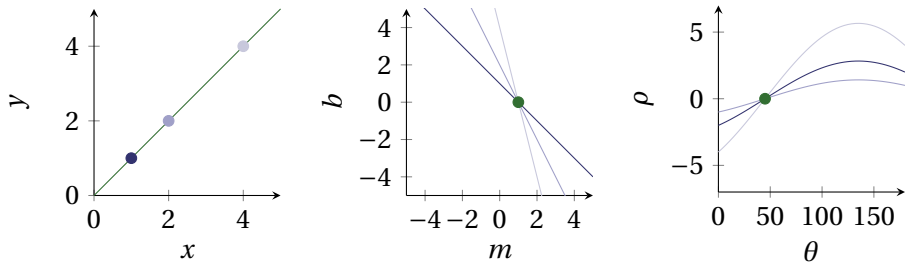


Figure 2.5: In the Hessian normal form, the line g is represented using the angle α and distance ρ.

2.3.3 Hough Transform (HT)

The HT [93] is a technique used in image analysis for feature extraction [83]. Initially developed for line detection in images, its application was later broadened to identify other shapes like circles and ellipses [97]. However, this work focuses primarily on the use of the HT for line detection in images.

The HT transforms an edge map into the Hough Domain (HD), a two-dimensional parameter space where each point signifies a line in the Euclidean space. Consequently, the HD is spanned by the parameters m and b in the slope-intercept form, or ρ and θ in the Hessian normal form. As outlined in Section 2.3.2, the Hessian normal form is preferred because it can represent vertical lines. Fig. 2.6 illustrates the transformation of a line from Fig. 2.6a to the HD. The HD is spanned once by the slope-intercept line parameters as shown in Fig. 2.6b and again by the Hessian normal form parameters in Section 2.3.3.



(a) Euclidean Space (b) Slope-Interecept HD (c) Hessian HD

Figure 2.6: (a) A line in Euclidean space is parametrized with slope $m = 1$ and intercept $b = 0$, or in Hessian normal form with $\theta = 45°$ and $\rho = 0$. Three points on this line are highlighted and transformed into the HD. (b) The HD is spanned by the Slope-Intercept parameters, where the three points manifest as distinct lines that intersect at $m = 1$ and $b = 0$. (c) The HD is spanned by the Hessian normal form parameters, with the three points manifest as sinusoidals intersecting at $\theta = 45°$ and $\rho = 0$.

The HT necessitates a binary input image $\mathbf{I}$, where each pixel is classified as either part of an edge or not. The transformation of $\mathbf{I}$ into the HD is facilitated by the HT algorithm, as detailed in Algorithm 1. This algorithm iteratively processes all pixels that represent edges. For each edge pixel, all potential lines passing through the pixel are computed. This computation corresponds to finding all possible values of ρ and θ that delineate lines passing through the given pixel. Subsequently, the HD value at the position corresponding to the calculated pairs (θ, ρ) is incremented to denote the corresponding lines in HD.

It is crucial to highlight the efficiency of the HT in comparison to the Radon transform [184]. This efficiency becomes apparent considering that an edge

Algorithm 1 Hough Transform

```
for each I(x, y) ≠ 0 do
    for θ := 0 to π do
        ρ := x ∗ cos(θ) + y ∗ sin(θ)
        HD[θ][d] + +
    end for
end for
```

image is typically sparse, and computations are exclusively performed for edge pixels.

In Fig. 2.6, the transformation is exemplified for three marked pixels. Each transformed pixel forms a straight line in the Slope-Intercept HD space. Each point on that line represents the parameters of a line that would pass through the respective pixel in the Euclidean space. All three lines intersect at the same point $(m = 1, b = 0)$. This intersection point is significant as these parameters describe the common straight line that passes through all three pixels in the Euclidean space. A similar pattern is observed in the Poloar Coordinate HD. However, instead of lines, the three pixels are transformed into three sinusoidal curves, intersecting at the same point $(\theta = 45°, \rho = 0)$.

It is important to note that in the HD, lines accumulate as points, but sinusoidal curves can also be observed alongside these points. This presents a particular challenge when retrieving the parameters representing the lines from the HD. Typically, a threshold is applied to the HD to identify the most prominent lines and suppress the sinusoidal curves. However, shorter lines have less prominent points, as fewer edge pixels represent that line, leading to fewer increments at that point. Consequently, short lines might fall below the threshold. Therefore, retrieving the parameters is a non-trivial task.

2.3.4 Image Pyramids

This section will discuss the two primary types of image pyramids: Gaussian and Laplacian. However, before delving into these, it is necessary to understand some prerequisites. Initially, the Fourier Transformation will be covered to provide a brief overview of the frequency domain. This will be followed by an introduction to the Low-Pass Filter, a crucial component of the Gaussian Pyramid. Subsequently, the concepts of downsampling and upsampling with Bilinear Interpolation, will be explained. Finally, the construction of the Gaussian and Laplacian Pyramids will be detailed.

Fourier Transformation

Images are typically represented in the spatial domain, with each pixel corresponding to a specific location. However, they can also be represented as an superposition of sinusoidal functions within the frequency domain. In this domain, each value signifies the amplitude of a specific sinusoidal base function, with a specific frequency. The 2D Fourier Transformation, which enables this transition from the spatial to the frequency domain by determining the amplitudes of the frequencies, is defined as follows [108]:

$$I(u,v) = \mathcal{F}(i(x,y)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} i(x,y)e^{-2\pi j(ux+vy)}dxdy \tag{2.16}$$

The complex sinusoidal function is represented as $e^{j(ux+vy)} = \cos(ux+vy) + j\sin(ux+vy)$, where j is the imaginary unit, and u and v denote the frequencies in the x and y directions, respectively. A 2D image in the spatial domain is denoted as $i(x,y)$, and its corresponding representation in the frequency domain is $I(u,v)$.

The frequency domain can be reverted to the spatial domain using the inverse Fourier Transformation:

$$i(x,y) = \mathcal{F}^{-1}(I(u,v)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(u,v)e^{2\pi j(ux+vy)}dudv \tag{2.17}$$

Low-Pass Filter

Low-pass filters, denoted as $G(u,v)$, are utilized to eliminate high-frequency components of a signal, thereby smoothing image features. This is accomplished by multiplying the signal in the frequency domain with a filter, which exhibits lower values at higher frequencies.

$$L(u,v) = G(u,v)\cdot I(u,v) \tag{2.18}$$

Multiplication in the frequency domain corresponds to convolution in the spatial domain [166]. Therefore, the filtering operation can equivalently be performed in the spatial domain as follows:

$$l(x,y) = (I * g)(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} i(x',y')g(x-x',y-y')dx'dy' \tag{2.19}$$

Note that the filter $g(x,y)$ represents the spatial domain equivalent of $G(u,v)$. Ideally, $G(u,v)$ would be a rectangular filter that is zero for frequencies higher than f_0, described as follows:

$$G(u,v) = \begin{cases} 1 & \text{if } |D(u,v)| \le f_0 \\ 0 & \text{if } |D(u,v)| > f_0 \end{cases} \tag{2.20}$$

Transforming this rectangular filter to the spatial domain results in a Sinc function:

$$g(x,y) = \frac{\sin(\pi x)\sin(\pi y)}{\pi x \pi y} \tag{2.21}$$

One-dimensional representations of both functions are depicted in Figure 2.7a, revealing the inherent flaw of this ideal filter. The Sinc function extends infinitely, necessitating an approximation for practical applications. This characteristic of the Sinc function directly relates to the Uncertainty Principle [167], which sets a limit to the simultaneous precision of certain pairs of physical properties, such as space and frequency. In the frequency domain, an infinite Sinc function would yield an infinitely sharp filter, but it would also result in the loss of all locality information. Conversely, when the Sinc function is limited, as is the case in practical applications, the sharpness of the filter in the frequency domain is reduced. This illustrates the inherent trade-off between space and frequency precision, as dictated by the Uncertainty Principle.

To mitigate this issue, the Gaussian filter is commonly used as a low-pass filter in image processing [141]. Unlike the Sinc function, the Gaussian filter converges to zero more quickly, as illustrated in Figure 2.7b. As a result, an approximated version of the Gaussian filter more closely resembles its infinite size counterpart. The Gaussian filter can be defined as follows:

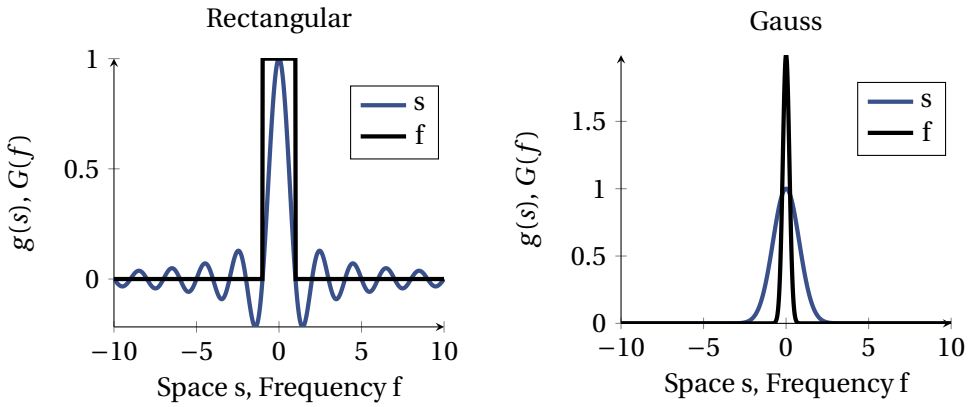
$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.22}$$

and is illustrated in Figure 2.7b. In the frequency domain, the Gaussian filter maintains a bell shape and can be defined as:

$$G(u,v) = e^{-2\pi^2\sigma^2(u^2+v^2)} \tag{2.23}$$

Both representations are Gaussian functions, with the width of the bell shape in each domain being inversely proportional to the other. This inverse proportionality stems from the kernel width being defined in the space domain by $\frac{1}{\sigma^2}$, and in the frequency domain by σ^2. This relationship is again a consequence of the time-frequency uncertainty principle [167], which stipulates that the product of the standard deviation of a function in the space and frequency domains must be at least $\frac{1}{4\pi}$.

While the Gaussian filter serves as an effective low-pass filter, even when the resolution does not extend to infinity, it nonetheless presents certain drawbacks. Specifically, it alters the amplitude of low frequencies due to its lack of a flat top. As an alternative, the Butterworth filter [37] is often employed. This filter offers a better balance between maintaining a flat top and ensuring rapid convergence to zero than the Gaussian filter. Nevertheless, due to its simplicity and proven effectiveness across various applications, the Gaussian filter remains the preferred choice in image processing and is also utilized in this work.



(a) Rectangular function in space and frequency domain.

(b) Gauss function in space and frequency domain.

Figure 2.7: Illustration of the Rectangular and Gauss functions in both the spatial and frequency domains.

Downsampling

Downsampling, also known as downscaling or subsampling, is a process that reduces the resolution of an image or signal by decreasing the number of pixels. For an image $i(x, y)$, the downscaled image $i'(x, y)$ is derived by selecting every K^{th} pixel in both the x and y directions, where K is the downscaling factor. This process can be represented as:

$$i'(x, y) = i(K \cdot x, K \cdot y) \tag{2.24}$$

Typically, a low-pass filter is applied to the image before downsampling to eliminate high frequencies. This step is crucial to prevent aliasing, a phenomenon that introduces distortions or artifacts in the downscaled image if high frequencies are not removed prior to downsampling [181], since during the downsampling process high frequencies are folded into the low frequency band, distorting the signal.

This requirement stems from the Nyquist-Shannon sampling theorem [199], which stipulates that the sampling frequency must be at least twice the highest frequency present in the signal to prevent aliasing. This is represented as:

$$f_{\text{Nyquist}} = \frac{f_{\text{sampling}}}{2} \tag{2.25}$$

Therefore, to avoid aliasing, the cut-off frequency of the low-pass filter must be equal to or lower than f_{Nyquist}.

Upsampling with Bilinear Interpolation

Upsampling, the counter operation to downsampling, improves image resolution by adding new pixels between existing ones. These new pixels' values are usually determined by methods such as Nearest-neighbor, Bilinear, or Bicubic interpolation [75, 113]. Bilinear interpolation, due to its computational efficiency and effectiveness, is often used in image processing and is thus applied in this study.

The upsampling process is mathematically represented as $i'(x,y) = B(i(x,y))$, where $i'(x,y)$ is the intensity of a new image approximated from the original image $i(x,y)$. For each new pixel (x,y), the four closest pixels in the original image are identified, and their intensities are used to compute the new pixel's intensity. The bilinear interpolation function $B(i(x,y))$ performs the following operation:

$$B(i(x,y)) = (1-a)(1-b)\,i(x_1,y_1) + a(1-b)\,i(x_2,y_1) + (1-a)\,v * i(x_1,y_2) + ab * i(x_2,y_2),$$

Here, x_1, y_1, x_2, y_2 are the coordinates of the four nearest pixels, and $a = \frac{x-x_1}{x_2-x_1}$, $b = \frac{y-y_1}{y_2-y_1}$ are the interpolation coefficients, which determine the weight of each pixel's intensity depending on the position of the new pixel. The interpolation operation is applied to each pixel, resulting in a smoothly interpolated image.

Gaussian Pyramid

With the necessary tools in place, the stage is now set to introduce the concept of the Gaussian Pyramid [216].

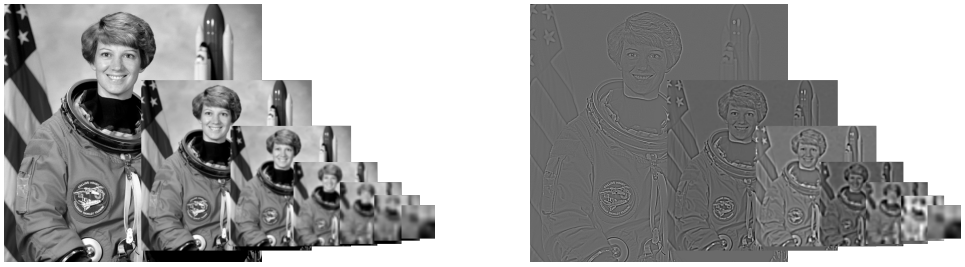
The Gaussian Pyramid is a sequence of N progressively low-pass filtered and downsampled versions of an image, each with a lower resolution than the previous one. The construction process is succinctly described in Algorithm 2. In

Algorithm 2 Gaussian Pyramid

```
P_0(x, y) := i(x, y)                          ▷ Original image
for j in level ≤ N do
    P'_j(x, y) := (P_{j-1} * g)(x, y)         ▷ Apply Gaussian filter
    P_j(x, y) := P'_j(K · x, K · y)           ▷ Downsample the image
end for
```

this algorithm, the original image, denoted as $P_0(x,y)$, forms the first level of the pyramid. Each subsequent level $P_j(x,y)$ is constructed by applying a Gaussian filter to the previous level $P_{j-1}(x,y)$ and then downsampling it. The filtering operation is mathematically represented as $P'_j(x,y) = (P_{j-1} * g)(x,y)$, and the downsampling operation as $P_j(x,y) = P'_j(Kx, Ky)$, where K equals two.

Notably, due to downsampling, the Gaussian kernel doubles in size with each subsequent step. This results in a decrease in high frequencies in each image within the Gaussian Pyramid, as illustrated in Figure 2.8a. Each pixel in a pyramid level approximates the average of four pixels in the level below. Consequently, each pixel represents the average value of its neighboring pixels in the original image. As the pyramid levels increase, each pixel comes to represent an increasingly larger neighborhood.



(a) Gauss Pyramid (b) Laplace Pyramid

Figure 2.8: Application of the Gauss and Laplace Pyramids on the widely recognized photograph of Eileen Collins [131], the first female pilot of a space shuttle.

Laplacian Pyramid

A Laplacian Pyramid [34] is a sequence of images representing the difference between the levels of the Gaussian Pyramid. It is constructed by subtracting each Gaussian Pyramid level from the expanded version of the subsequent level, achieved by upsampling the image of the next level to match the current level's resolution. This process is succinctly described in Algorithm 3. In this algorithm,

Algorithm 3 Laplacian Pyramid

for j in level $< N$ **do**
  $P'_{j+1}(x,y) := B(P'_{j+1})(x,y)$ ▷ Upsample next level
  $L_j(x,y) := P_j(x,y) - P'_{j+1}(x,y)$ ▷ Subtract from current level
end for
$L_n(x,y) := P_n(x,y)$ ▷ Set last level

$B(P'_{j+1})(x,y)$ represents the process of upsampling the image of the next level using bilinear interpolation, and $L_j(x,y) = P_j(x,y) - B(P'_{j+1})(x,y)$ is the equation for constructing each level of the Laplacian Pyramid. The last level in the Laplacian Pyramid is identical to the last level in the Gaussian Pyramid, as there is no subsequent level to subtract from.

Given that the Gaussian Pyramid comprises a sequence of low-pass filtered images, each level includes frequencies not present in the previous level. Conse-

quently, the Laplacian Pyramid can be viewed as a series of band-pass filtered images, as illustrated in Figure 2.8b.

2.4 Optimization in Machine Learning

Unlike their hard-coded counterparts, machine learning algorithms are not governed by predefined rules. Instead, they adjust their functionality based on observations derived from training data. Therefore, a machine learning process consists of two main components: the model that performs the required task during inference, and the process that adjusts the model based on the training data.

Algorithmic advancements, such as backpropagation [194], Convolutional Neural Network (CNN) [70], and self-attention [225], combined with current computational resources and large data volumes, have facilitated the adoption of machine learning, particularly deep learning, across a wide range of fields.

This work explores the significant impact of these algorithms in the field of medical imaging. Before delving into the specifics of deep learning, which employs multi-layered neural networks, the following sections will first lay the groundwork by discussing the core concepts of machine learning. A more detailed exploration of deep learning and its application in medical imaging will be presented in the subsequent Section 2.5.

2.4.1 Optimization Objective

In machine learning, a problem to be solved can be formulated as a function $f^*(\mathbf{x})$ that maps an input $\mathbf{x}$ to a desired output $\mathbf{y}$: $\mathbf{y} = f^*(\mathbf{x})$. For instance, a function $f^*(\mathbf{x})$ can describe the ideal mapping of a noisy image $\mathbf{x}$ to a clean image $\mathbf{y}$.

In general, to solve a problem using machine learning, an optimization process is employed to adjust the parameters $\mathbf{w}$ of a function $f(\mathbf{x};\mathbf{w})$, so that $f(\mathbf{x};\mathbf{w})$ approximates $f^*(\mathbf{x})$ as closely as possible:

$$f^*(\mathbf{x}) \stackrel{!}{=} f(\mathbf{x};\mathbf{w}) \tag{2.26}$$

The parameters $\mathbf{w}$ are estimated by minimizing a loss function L. This function quantifies the difference between $f^*(\mathbf{x}) = \mathbf{y}$ and $f(\mathbf{x};\mathbf{w})$. The process can be expressed as follows:

$$\underset{\mathbf{w}}{\operatorname{argmin}} \quad L(\mathbf{y}, f(\mathbf{x};\mathbf{w})) \tag{2.27}$$

Thus, for a given input $\mathbf{x}$, the parameters $\mathbf{w}$ of $f(\mathbf{x},\mathbf{w})$ are optimized such that the output is an estimate $\hat{\mathbf{y}}$ of $\mathbf{y}$. The function $f(\mathbf{x};\mathbf{w})$ is typically referred to as a model. The process of adjusting the parameters $\mathbf{w}$ to minimize the difference

between $f^*(\mathbf{x})$ and $f(\mathbf{x};\mathbf{w})$ is known as training. Consequently, $\mathbf{x}$ and $\mathbf{y}$ represent the input and target of the training data, respectively.

2.4.2 Gradient Descent

Due to the complexity of many optimization problems that precludes analytical solutions, Gradient Descent (GD) is commonly employed as an effective numerical method. This iterative optimization algorithm aims to find the minimum of Eq. (2.27).

It does so by iteratively calculating the gradient $\frac{\partial L}{\partial \mathbf{w}} = \nabla L(\mathbf{y}, f(\mathbf{x};\mathbf{w}))$ with respect to the parameters $\mathbf{w}$ at each iteration t. Given that the negative gradient indicates the direction of steepest descent and potentially the direction of the minimum, the parameters are updated by taking a step in this direction. It is important to note that this approach yields optimal results for convex functions. However, for non-convex functions, it may only find a local minimum. The update rule for the parameters in each iteration is as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla L(\mathbf{y}, f(\mathbf{x};\mathbf{w}_t))) \tag{2.28}$$

The learning rate η, an important hyperparameter, controls the step size. Small values could lead to slow convergence, while large values might cause the algorithm to overshoot the minimum.

2.4.3 Stochastic Gradient Descent

Calculating the gradient $\nabla L(\mathbf{y}, f(\mathbf{x};\mathbf{w}))$ for the entire optimization function, which encompasses the whole training data, may be too computationally expensive. Furthermore, as previously discussed, GD can potentially become trapped in local minima. Both these issues are addressed by SGD. Instead of computing the gradient for the entire training data, SGD processes the data ($\mathbf{x}$, $\mathbf{y}$) in subsets, commonly referred to as 'mini batches'. Each mini batch, denoted as $(\mathbf{x}_i, \mathbf{y}_i)$, is used to calculate the gradients. The weights $\mathbf{w}$ are then updated for each mini batch according to the equation:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla L(\mathbf{y}_i, f(\mathbf{x}_i;\mathbf{w}_t)) \tag{2.29}$$

The process of iterating over all training subsets can be repeated for a fixed number of times, with one iteration over all subsets referred to as an epoch. Alternatively, the process can continue until the loss value reaches a sufficiently low level. The complete process is outlined in Algorithm 4.

Kleinberg et al. [116] demonstrated that calculating gradients on subsets of data, allows the gradient to escape local minima, provided these minima are not

Algorithm 4 Stochastic Gradient Descent

```
while L(w_t, x, y) > ε do                    ▷ Continue until the loss is low
    for each y_i in y do                     ▷ Iterate over each training example
        w_t − η∇L(y_i, f(x_i, w_t));         ▷ Update the weight vector
    end for
end while
```

present in all batches. Therefore, by employing SGD, the optimization process is less prone to stagnation at these points. Additionally, updating the gradients based on these batches reduces the computational cost of each update, and thus also accelerates the overall optimization process.

2.4.4 Backpropagation

The calculation of the gradient in SGD necessitates the derivative of the loss function with respect to the parameters $\mathbf{w}$. For a large number of parameters, as seen in neural networks, the backpropagation algorithm can be utilized to make this computation feasible [194].

In backpropagation, the to be minimized function $L(\mathbf{y}, f(\mathbf{x};\mathbf{w}))$ is broken down into a sequence of functions:

$$f(\mathbf{x};\mathbf{w}) = f^n\left(f^{n-1}\left(f^{n-2}\left(\dots f^1(\mathbf{x};\mathbf{w}^1);\dots;\mathbf{w}^{n-2}\right);\mathbf{w}^{n-1}\right);\mathbf{w}^n\right) \tag{2.30}$$

Backpropagation then utilizes the chain rule to calculate the gradient of the loss function with respect to the parameters $\mathbf{w}^i$ of each sub-function successively:

$$\frac{\partial L}{\partial \mathbf{w}^i} = \frac{\partial L}{\partial f^n}\frac{\partial f^n}{\partial f^{n-1}}\cdots\frac{\partial f^{i+1}}{\partial f^i}\frac{\partial f^i}{\partial \mathbf{w}^i} \tag{2.31}$$

Thus, it begins by defining a gradient $\boldsymbol{\delta}_n$ for the final function n:

$$\boldsymbol{\delta}^n = \frac{\partial L}{\partial f^n} \tag{2.32}$$

This is done by calculating the gradient with respect to f^n. All intermediate gradients $\boldsymbol{\delta}^i$ can be computed by considering the preceding gradient $\boldsymbol{\delta}^{i+1}$ and the derivative of the function f^i:

$$\boldsymbol{\delta}^i = \boldsymbol{\delta}^{i+1}\frac{\partial f^{i+1}}{\partial f^i} \tag{2.33}$$

Moreover, for calculating the gradients with respect to the parameters $\mathbf{w}^i$ of a function f^i, the gradient $\boldsymbol{\delta}^i$ is utilized:

$$\frac{\partial L}{\partial \mathbf{w}^i} = \boldsymbol{\delta}^i \frac{\partial f^i}{\partial \mathbf{w}^i} \tag{2.34}$$

In summary, backpropagation starts by calculating the derivative of the loss function L with respect to its input f^n. Subsequently, it computes the derivative of the function f^i with respect to its input f^{i-1}, utilizing the previously calculated derivative $\boldsymbol{\delta}^{i+1}$. This process continues until the derivative of the first function f^1 is calculated. Finally, the gradient of the loss function with respect to the parameters $\mathbf{w}^i$ is computed by multiplying the intermediate gradient $\boldsymbol{\delta}^i$ with the derivative of the function f^i with respect to its parameters $\mathbf{w}^i$. Thus, to calculate gradients with respect to $\mathbf{w}^i$, the previous computed gradient $\boldsymbol{\delta}^i$ is utilized. As a result, backpropagation, rather than computing the derivative of the loss function for each parameter individually, leverages previously calculated gradients. This approach facilitates efficient computation of gradients, especially for complex functions with a substantial number of parameters.

In practical implementations, $\boldsymbol{\delta}$ may either vanish or explode after several steps i. This is a phenomenon known as the vanishing or exploding gradient problem. Furthermore, for the implementation of backpropagation, all elements of the function must be differentiable. This differentiability must either be possible analytically or, alternatively, numerically using methods such as subgradients [29].

2.5 Deep Learning

Deep learning, a subfield of machine learning, focuses on a unique design of approximation functions, $f(\mathbf{x};\mathbf{w})$. These functions, known as neural networks, are inspired by the functioning of mammalian brains, or more specifically, their neurons. They have demonstrated the ability to handle a large number of parameters, which can be optimized using SGD and backpropagation. Furthermore, their design facilitates parallelization on GPUs, significantly accelerating the computation of gradients. This section will discuss the fundamentals of neural network designs.

2.5.1 Multi Layer Perceptron (MLP)

The Rosenblatt Perceptron (RP) [191, 192] represents the simplest form of a neural network. The RP is modeled after the operation of a neuron, which integrates several input signals to yield an output signal. It is defined as:

$$y = a(\mathbf{w}^T\mathbf{z} + \mathrm{b}) \tag{2.35}$$

In this context, $\mathbf{w}$ denotes the trainable weights, $\mathbf{z}$ denotes the input to the neuron, and b refers to a trainable bias [76]. In comparison to $\mathbf{x}$, $\mathbf{z}$ can also be the output of a previous neuron. The function a represents the activation function. In the context of the RP, it is a binary step function that introduces a non-linear component into the overall function. The mathematical operations of the RP are graphically illustrated in Fig. 2.9a.

Despite the RP being modeled after the operation of a biological neuron, it lacks the capability to solve non-linearly separable problems, such as the XOR function, as demonstrated by Rosenblatt [191] and Minsky et al. [158].

However, mirroring the structure of mammalian brains where numerous neurons are interconnected, the Multi-Layer Perceptron (MLP) overcomes the limitations of the RP by arranging multiple RPs in layers, as depicted in Fig. 2.9b, with each RP receiving inputs from the outputs of the preceding layer. Thus, a layer f^l in a MLP can be mathematically expressed as:

$$f^l(\mathbf{z}) = a(\mathbf{W}\mathbf{z} + \mathbf{b}) \tag{2.36}$$

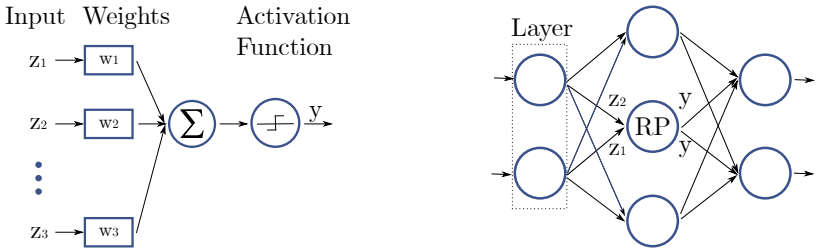
It is important to note that instead of a weight vector $\mathbf{w}$, which signifies the input weights of a single RP, a weight matrix $\mathbf{W}$ is employed. Each row of $\mathbf{W}$ corresponds to the weights $\mathbf{w}^T$ of one RP. Similar, instead of a sclalar bias b, a bias vector $\mathbf{b}$ is used. This layers can now be chained to form the network:

$$f^N(\mathbf{z}, \mathbf{w}^{\text{network}}) = f^j(f^{j-1}(...)) \tag{2.37}$$

This arrangement enables the use of the chain rule to calculate the gradients with backpropagation, as detailed in Section 2.4.4.

Moreover, this chaining of layers does only contribute to the expressive power of the network due to the non-linear activation functions a. The need for a non-linear activation function becomes apparent when examining Eq. (2.37) and Eq. (2.36). Without it, the network's representation would collapse into a single matrix, substantially diminishing its expressive power, as illustrated in the subsequent equation:

$$\begin{aligned} &\mathbf{W}^i(\mathbf{W}^{i-1}\mathbf{z} + \mathbf{b}^{i-1}) + \mathbf{b}^i \\ &= (\mathbf{W}^i\mathbf{W}^{i-1})\mathbf{z} + \mathbf{W}^i\mathbf{b}^{i-1} + \mathbf{b}^i \\ &= \mathbf{W}\mathbf{z} + \mathbf{b} \end{aligned} \tag{2.38}$$



(a) Rosenblatt Perceptron (RP) (b) Multi Layer Perceptron (MLP)

Figure 2.9: The Rosenblatt and Multilayer Perceptron are illustrated. The MLP is a composition of the RP. The outputs y from previous layers serve as the inputs z_1 and z_2 for the RPs of the subsequent layer.

Rosenblatt [191] initially employed the step function as the activation function. Nowadays, however, a variety of activation functions are utilized, such as the sigmoid function, defined as $\text{sigmoid}(x) = \frac{1}{1+e^{-x}}$, or the Rectified Linear Unit (ReLU), defined as $\text{ReLU}(x) = \max(0, x)$ [71]. The key criteria for the activation function include non-linearity and numerical stable gradients during backpropagation. As a result, a diverse range of activation functions exist [59].

Moreover, it has been prooven that a MLP with a non linear activation function, a single hidden layer, and a finite number of neurons can approximate any function [52, 92].

This capability, coupled with the powerful computational resources currently available and vast volumes of data, led to an increase in the complexity and size of neural networks, resulting in increasingly sophisticated and powerful models capable of solving ever more complex tasks [159].

2.5.2 Convolutional Neural Networks (CNNs)

An MLP necessitates a weight for every input value, a requirement that poses a significant challenge for data types such as images or high-resolution time series due to the overwhelming number of parameters involved. This results in substantial computational costs and the potential for overfitting, where the model could inadvertently learn irrelevant patterns in the training data, thereby diminishing its performance on unseen data.

To address this issue, prior knowledge regarding the fundamental characteristics of images and time series can be integrated into the network design. This prior knowledge is embodied in the following three properties: Equivariance, Sparsity, and Shared Weights [76].

Equivariance suggests that any alteration to the input of a neural network should correspondingly change its output. More specifically, a function $f(x)$ is equivariant to a function $g(x)$ if $f(g(x)) = g(f(x))$. In the context of neural networks, $g(\cdot)$ could represent a translation like a shift operation. Consequently, translating the output of the network should yield the same results as translating its input. This property is particularly relevant for image-to-image processing tasks, such as denoising. For instance, shifting a noisy input image should also yield a shifted denoised output image. In the case of MLPs, each input value is assigned a unique weight, thereby not guaranteeing equivariance.

Sparsity refers to the weighted connections between feature values of two consecutive layers. In MLPs, also known as Fully Convolutional Networks (FCNs), each input value is connected to each output value. However, certain image features, such as edges, are localized to specific regions of the image. Therefore, recognizing these features does not necessitate connections between all areas of the image. Consequently, the network can be made more efficient by reducing the number of connections.

Shared Weights implies, that the same weights are applied to multiple input values. This property is particularly relevant for images, as certain features, such as edges, are present in multiple regions of the image. Thus, shared weights can be trained to recognize features irrespective of their location.

Convolution Operation

CNNs, first introduced by Fukushima [70], fundamentally incorporate convolution operations. These convolution operations inherently embody all three required properties.

A fundamental aspect of each CNN is the convolution operation, as illustrated in Fig. 2.10. A convolution kernel $\mathbf{K}$ contains trainable weights, and is depicted with dimensions $a \times b \times c$. This kernel is convolved across the spatial dimensions $h \times w$ of the input feature map $\mathbf{Z}_i$, which has overall dimensions of $h \times w \times c$. The squares in Fig. 2.10 represent the unique values of the feature map and kernel. In the first layer of a CNN, the feature map is the input itself, such as an image, where c denotes the number of RGB channels.

The convolution operation for each kernel k is defined as:

$$\mathbf{Z}_{i+1}(x, y, k) = \sum_{i=1}^{a} \sum_{j=1}^{b} \sum_{c=1}^{C} \mathbf{Z}_i(x + i - a/2, y + j - b/2, c) \cdot \mathbf{K}_k(i, j, c) \tag{2.39}$$

In this equation, each value in the output feature map $\mathbf{Z}_{i+1}$ at position (x, y, k), denoted as $\mathbf{Z}_{i+1}(x, y, k)$, is calculated by summing the product of the input feature

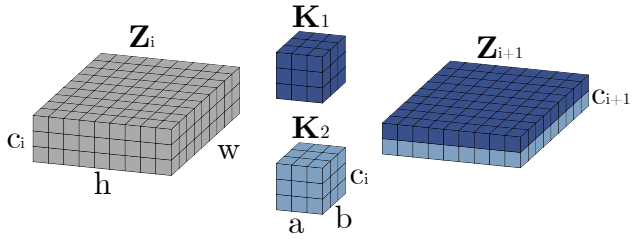


Figure 2.10: The input feature map is represented in grey. Two convolution kernels, $\mathbf{K}_1$ and $\mathbf{K}_2$, are illustrated in dark and light blue respectively. When convolved over the spatial dimension $h \times w$ of the input feature map, each kernel generates a layer of the output feature map, depicted in the color corresponding to each kernel. Each square in the feature maps and kernels represents a distinct numerical value.

map values $\mathbf{Z}_i(x+i-a/2, y+j-b/2, c)$ and the corresponding kernel weights $\mathbf{K}_k(i, j, c)$ of the kth kernel.

The output of the convolution operation is illustrated in Fig. 2.10 using various shades of blue, each representing a specific kernel and its corresponding output channel. Each channel in the output feature map is generated by a unique kernel $\mathbf{K}_k$, as convolution is applied over the $h \times w$ dimensions of the input.

To maintain the same spatial dimensions in the output as in the input, padding can be utilized. This involves adding pixel values around the image, allowing the convolution operation to have as many steps as input values and consequently, an equal number of output values [60]. Additionally, the stride s can be adjusted to control the kernel's steps across the input. As a result, Eq. (2.39) is not applied for all input values x and y, but only for every sth value, leading to a reduction in the output's spatial dimensions.

It can be observed that the convolution operation inherently embodies the properties of equivariance, sparsity, and shared weights. Equivariance is inherent in the convolution operation because the kernel weights, which are independent of the input's location, are applied uniformly across all input locations. Consequently, when the input is translated or shifted, the same convolution operation is applied to the shifted region as was applied to the original region, resulting in a correspondingly shifted output. Applying the same kernel to different regions of the input feature map also ensures that weights are shared across the input. Moreover, the convolution operation exhibits sparsity, as the kernel only covers a specific region of the input and output values in the feature map are dependent solely on a region of the input, specifically the kernel's location. However, regarding the channel dimension, the convolution operation links all input values to

a single output value, thereby representing a fully connected operation in the channel dimension.

Network Architecture

A single convolution operation is insufficient to construct a CNN. Much like a MLP, a CNN requires multiple operations to learn complex features. Moreover, typical CNNs include not only convolution operations but also activation functions, normalization, and pooling operations. The functions and importance of these additional operations will be discussed in this section. Typically, a combination of convolution and some of these operations is referred to as a (convolutional) layer. Sparse connectivity, despite its advantages, imposes a limitation on single convolution operations in terms of their receptive field. This is the region of the input that influences a single output value. However, this limitation can be mitigated by stacking multiple convolution operations, which enhances the receptive field of the subsequent layers. Moreover, akin to a MLP, multiple layers enhance the network's capacity to learn complex patterns.

Mathematically, the receptive field R is defined recursively. For a convolutional operation l with kernel size k_l, stride s_l, and padding p_l, the receptive field R_l can be computed as:

$$R_l = R_{l-1} + (k_l - 1) \cdot \prod_{i=1}^{l-1} s_i \tag{2.40}$$

For the first operation, the receptive field is simply the kernel size: $R_1 = k_1$.

As the network deepens, individual values come to represent larger regions of the network's input. Consequently, single values in deep layers may represent complex features like textures or shapes, derived from a large region of the input image. Depending on the feature to be represented, different channels might be activated, i.e., the output value of a specific channel changes. For this reason, as the network deepens, the importance shifts from the spatial dimension to the representation of a variety of features, necessitating an increase in the number of channels.

Consequently, the number of kernels is typically augmented with the network's depth, increasing the number of channels. Simultaneously, the spatial dimensions are often reduced by increasing the stride s. Therefore, with increasing network depth, feature maps often exhibit a reduced spatial dimension but an increased number of channels.

Constructing a functional network requires more than just stacking convolutional operations. Similar to a MLP, CNNs also need activation functions such as the ReLU or the sigmoid function to introduce non-linearity into the network.

These are the same activation functions employed in MLPs. Furthermore, normalization operations, such as batch normalization, are needed to stabilize the training process [98]. Depending on the batch size and task, other normalization approaches like instance normalization [221] or layer normalization [12] may be more appropriate.

Additionally, pooling operations are often employed. These operations function similarly to convolution operations, using a kernel that traverses the input. However, instead of applying a weighted summation, pooling operations select either the maximum or average value within the kernel [27].

$$\mathbf{Z}_{i+1}(x,y,k) = \max_{i,j} \mathbf{Z}_i(x+i-a/2, y+j-b/2, c, k) \tag{2.41}$$

Pooling, typically applied channel-wise with a stride $s > 1$, reduces the spatial dimensions of the feature map. Notably, Max-Pooling introduces a slight shift-invariance, as the output value is independent of the exact location of the maximum value within the kernel. Consequently, a typical CNN layer consists of a convolution operation, followed by a normalization, an activation function, and a pooling operation, as illustrated in Fig. 2.11.

Figure 2.11: The figure illustrates the typical components of a CNN layer, including batch normalization.

An architecture that encodes an image into an abstract feature map with reduced spatial dimension is commonly known as an encoder. The resulting abstract feature maps, also known as embeddings, are particularly significant for Chapter 7.

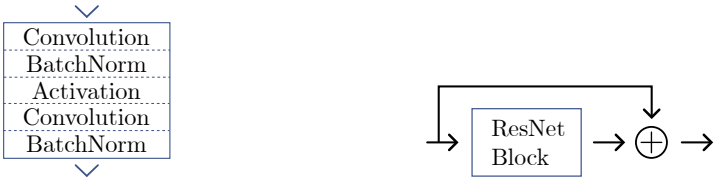
ResNet

Residual Networks (ResNets), as introduced by He et al. [86], are a variant of CNN specifically designed to tackle the problem of vanishing gradients. The architecture of ResNets addresses this issue by introducing an identity connection that bypasses a ResNet block $f(\mathbf{Z})$, as illustrated in Fig. 2.12. In other words, the input $\mathbf{Z}$ is added to the output of the ResNet Block, yielding the final output $f'(\mathbf{Z})$ as follows:

$$f'(\mathbf{Z}) = f(\mathbf{Z}) + \mathbf{Z} \tag{2.42}$$

This identity connection allows $f(\mathbf{Z})$ to learn the residual, that is, the difference between the input and the output. Additionally, identity connections promote a

numerically stable gradient flow through the network by bypassing the ResNet block, thereby enabling the training of deeper networks. Veit et al. [226] also demonstrated that a residual network operates similarly to an ensemble of shallow networks, thereby providing further insight into its success. Various ResNet architectures are available, each denoted by the number of convolutional layers they contain. For instance, ResNet-18 employs 9 ResNet blocks.



(a) ResNet Block Components. (b) ResNet Block with Identity Connection.

Figure 2.12: The left image illustrates the components of a ResNet block, which includes a convolutional layer, batch normalization, and a activation function. The right image depicts the complete ResNet block with a identity connection, which bypasses the convolutional layer and adds the input to the output.

U-Net

U-Net, introduced by Ronneberger et al. [190], is a special CNN architecture, originally designed for biomedical image segmentation. Similar to an Autoencoder [231], the U-net architecture is characterized by an encoder and a decoder path. Initially, the process reduces the spatial dimension while augmenting the channel dimension. Subsequently, in the reverse process of the decoder path, it expands the spatial dimension while diminishing the channel dimension. Additionally, the U-Net architecture introduces skip connections, which transfer information from the encoder to the decoder path on the corresponding matching resolution levels as depicted in Fig. 2.13.

Each encoder block typically comprises two convolution layers, with the first one doubling the channel size and the second one maintaining it. Each convolution layer is succeeded by a batch normalization layer and a ReLU activation function. At the end of each block a max-pooling operation halves the spatial dimensions. Following the encoding blocks, a bottleneck layer is utilized to process the feature maps, maintaining the same spatial dimensions. This bottleneck layer is similar to the encoder blocks, but excludes max-pooling.

It should be noted that Fig. 2.13 depicts only two encoder blocks for clarity, although the original U-Net architecture contains four.

The decoder block employs an upsampling operation, such as bilinear interpolation as detailed in Section 2.3.4 or deconvolution [251], to double the spatial

dimension of the feature map. The corresponding output of the encoder block is then concatenated to this feature map. Subsequently, two convolutions, each with a ReLU activation function, are applied to reduce the channel size.

Consequently, the U-Net architecture, through its combination of high-level features with low-level features via skip connections, has proven to be particularly effective in generating precise segmentation masks.

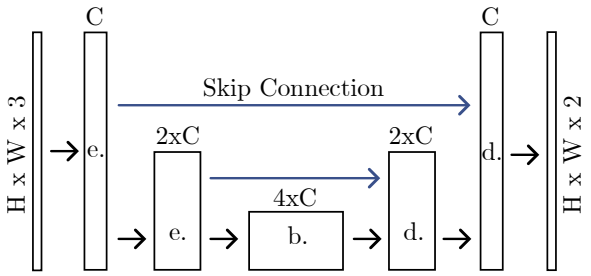


Figure 2.13: The illustration provides a simplified depiction of the U-Net architecture, which includes two encoding blocks (e.), one bottleneck (b.), and two decoding blocks (d.).

2.5.3 Metrics & Loss Functions

To fit the parameters $\mathbf{w}$ of a function $f(\mathbf{x};\mathbf{w})$, i.e to train a neural network or another machine learning model, a loss $L(\mathbf{y}, f(\mathbf{x};\mathbf{w}))$ must be defined, as detailed in Section 2.4.1. This loss function quantifies the difference between the target $\mathbf{y}$ and the prediction $f(\mathbf{x};\mathbf{w})$.

Quantifying this error is crucial as it establishes the objective and consequently, the gradients of the optimization problem. Given that objectives vary across different tasks, a range of loss functions, each with its unique characteristics and use cases, is available.

Moreover, evaluating a model's performance necessitates the use of a metric, which quantifies the difference between the target and the prediction of a test set. Thus, the mathematical concept behind a metric and a loss function is the same, with the difference that the latter is employed for optimization purposes and in case of GD must be differentiable.

Mean Squared Error (MSE)

A widely used and intuitive metric is the Mean Squared Error (MSE), defined as the average of the squared differences between two sets of values, $\mathbf{y}$ and $\hat{\mathbf{y}}$:

$$\text{MSE}(\mathbf{y},\hat{\mathbf{y}}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{2.43}$$

with N being the number of values in the sets. In the context of optimization, $\mathbf{y}$ represents the target and $\hat{\mathbf{y}}$ denotes the prediction $f(\mathbf{x};\mathbf{w})$.

The MSE, which calculates the difference between each value, is easy to interpret. However, not all values are always equally important. For instance, in images, edge pixels contribute more to the overall appearance than background pixels. Therefore, more specific loss functions and metrics are often employed.

Peak Signal-to-Noise Ratio (PSNR)

In addition to the MSE, another important metric in image processing is the Peak Signal-to-Noise Ratio (PSNR). This metric evaluates the ratio of meaningful signal to noise and is typically expressed in dB. It utilizes the MSE between a noise-free signal $\mathbf{y}$ and a noisy signal $\hat{\mathbf{y}}$.

It is defined as:

$$\mathrm{PSNR}(\mathbf{y},\hat{\mathbf{y}}) = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}(\mathbf{y},\hat{\mathbf{y}})}\right), \tag{2.44}$$

with MAX being the maximum possible pixel value of the image.

Structural Similarity Index (SSIM)

The aforementioned limitations of the MSE metric are demonstrated in Fig. 2.14. Despite the perceptible gradual decrease in image quality from Fig. 2.14b to Fig. 2.14d, the MSE between Fig. 2.14a and the other three images of Einstein remains constant. This is because a constant shift in pixel values in Fig. 2.14b contributes substantially to the MSE, despite the fact that this difference is not appreciable to a human observer. In contrast, blurring or artefacts, as depicted in Fig. 2.14c and Fig. 2.14d, are more perceptible.

This issue is addressed by the Structural Similarity Index (SSIM), proposed by Wang et al. [239], which is a metric specifically designed to mimic human perception.

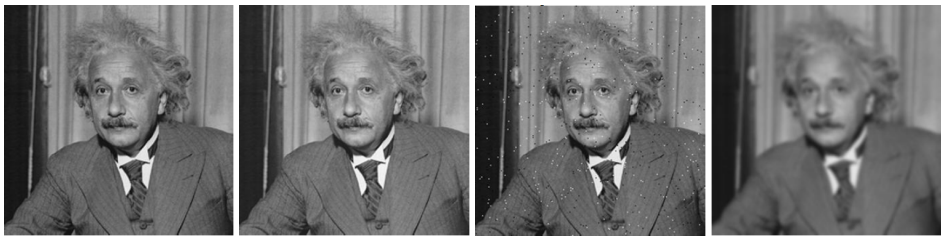
The SSIM is represented as follows:

$$\mathrm{SSIM}(\mathbf{Y},\hat{\mathbf{Y}}) = l(\mathbf{Y},\hat{\mathbf{Y}})^{\alpha} \cdot c(\mathbf{Y},\hat{\mathbf{Y}})^{\beta} \cdot s(\mathbf{Y},\hat{\mathbf{Y}})^{\gamma} \tag{2.45}$$

with $l(\cdot,\cdot)$, $c(\cdot,\cdot)$ and $s(\cdot,\cdot)$ representing the luminance, contrast and structure difference between the two images $\hat{\mathbf{Y}}$ and $\mathbf{Y}$. The constants α, β and γ are used to weight the three components.

The luminance can be stated as:

$$l(\mathbf{X},\mathbf{Y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{2.46}$$



(a)	(b)	(c)	(d)
MSE=0,	MSE=144,	MSE=144,	MSE=144,
SSIM=1,	SSIM=0.988,	SSIM=0.84,	SSIM=0.694,
original	offset added	corrupted pixel	blurred

Figure 2.14: Four images of Einstein [239] are depicted, with the last three being compared against the first using MSE and SSIM. A gradual decrease in image quality is observable, which is not captured by the MSE metric but is reflected by the SSIM metric.

μ_x and μ_y are the two mean values of the to be compared images. C_1 is a constant to avoid instability, if the means get close to zero. Consequently eq. 2.46 is one, if the mean of both images is the same and becomes less otherwise.

The construction of the contrast measurement is akin to the luminance rating, but it utilizes the variances of the images instead of the means. The formula is as follows:

$$c(\mathbf{X},\mathbf{Y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{2.47}$$

The constant C_2 is introduced, similar to C_1, to prevent instability.

A structural comparison can be performed by calculating the correlation between the two entities, as shown in the following equation:

$$s(\mathbf{X},\mathbf{Y}) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{2.48}$$

Here, σ_{xy} represents the covariance between them, which is defined as the inner product of both images:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \tag{2.49}$$

with N being the number of pixels in the image.

Additionally, Wang et al. [239] suggested the application of the SSIM on a local scale rather than a global one. This is achieved by calculating the SSIM for a local square window, which moves pixel by pixel over the image. Moreover, they employed an 11×11 circular symmetric Gaussian weighting function, denoted as

$\mathbf{w} = w_i | i = 1, 2, ..., N$, with $\sigma = 1.5$ and normalized to $\sum_{i=1}^{N} w_i = 1$. Consequently, the contribution of each pixel within the patch is determined by its Gaussian weight, for instance, $\mu_x = \sum_{i=1}^{N} w_i x_i$.

Finally, they integrate the local patches into the Mean Structural Similarity Index (MSSIM) by computing the average of the local SSIM values:

$$\text{MSSIM}(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{M} \sum_{j=1}^{M} \text{SSIM}(\hat{\mathbf{y}}_j, \mathbf{y}_j) \tag{2.50}$$

In this context, $\hat{\mathbf{y}}_j$ and $\mathbf{y}_j$ signify the contents of the image patches under comparison, while M stands for the total count of patches. When utilizing MSSIM or SSIM as a loss function, it must be taken into account that both yield a value of one for identical images and zero for completely dissimilar ones. Consequently, $1 - \text{SSIM}$ is employed as a loss function. The SSIM has proven to be particularly effective in training a denoising neural network, as demonstrated in Chapter 5.

Dice-Sørensen Coeffient

While SSIM is specifically designed for image comparison, the Dice-Sørensen Coefficient [206] or Dice loss is frequently employed to evaluate segmentation tasks. This metric is defined as follows:

$$\text{DSC} = \frac{2|\mathbf{Y} \cap \hat{\mathbf{Y}}|}{|\mathbf{Y}| + |\hat{\mathbf{Y}}|} \tag{2.51}$$

Here, $|\mathbf{Y}|$ and $|\hat{\mathbf{Y}}|$ denote the number of pixels in the ground truth and the prediction, respectively, while $|\mathbf{Y} \cap \hat{\mathbf{Y}}|$ represents the intersection of the two sets. Originally designed for binary data, this coefficient compares the area of overlap between the two sets to the total area of both sets. Given that it only considers the area of segmentation, this coefficient is particularly effective in unbalanced segmentation tasks, where the area of the object to be segmented is small compared to the background. The Dice loss is explicitly employed in Chapter 4.

Perceptual Loss

Gatys et al. [74] proposed the innovative idea of using the comparison of feature maps from a pre-trained neural network as a loss function. Johnson et al. [104] further advanced this concept by training a neural network using this loss function. In this approach, $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ are processed through the pre-trained network $f(\cdot)$. This generates corresponding feature maps, denoted as $\mathbf{Z}_y^l = f_l(\mathbf{Y})$ and $\mathbf{Z}_{\hat{\mathbf{y}}}^l = f_l(\hat{\mathbf{Y}})$. In this context, $f_l(\cdot)$ represents the operation of extracting the feature map at layer l, while $\mathbf{Z}$ denotes the resulting feature map.

The perceptual loss is then computed by comparing $\mathbf{Z}_y^l$ and $\mathbf{Z}_{\hat{\mathbf{y}}}^l$ using the MSE, as follows:

$$\text{PerceptualLoss}(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N} = \| f_l(\mathbf{Y}) - f_l(\hat{\mathbf{Y}}) \|_F^2 \tag{2.52}$$

$$= \frac{1}{N} \sum_{i=1}^{N} (z_{\hat{y}i} - z_{yi})^2 , \tag{2.53}$$

with N being the number of elements in the feature maps, $\| \cdot \|_F$ denotes the Frobenius norm and $z_{\hat{y}i}$ and z_{yi} represent the elements of the feature maps $\mathbf{Z}_{\hat{\mathbf{y}}}^l$ and $\mathbf{Z}_y^l$, respectively.

In a neural network, the depth of layer l is directly correlated with the level of abstraction of the features it represents. Values in shallow layers maintain a close relationship with the input pixel values, whereas elements in deeper layers encapsulate more abstract information such as textures or shapes. As a result, the perceptual loss proves to be especially effective when the task requires the abstract features of two images to be identical. Moreover, the concept of calculating the error between two feature maps serves as a pivotal idea in Chapter 7.

3

Noise Simulation on X-ray Images

Simulating physically accurate noise on X-ray images enables the conversion of high-dose X-rays to those acquired with a lower dose. This is primarily because dose reduction leads to an increase in noise and a decrease in mean signal intensity, ultimately resulting in a reduced Signal-to-Noise Ratio (SNR). Simulating physical noise allows for the augmentation of training data for deep learning models. Specifically, in our work, a sophisticated noise simulation is necessary to train a denoising neural network, as presented in Chapter 5. Moreover, noise simulation is essential for simulating collimator shadows in Chapter 4, as areas of collimator shadows receive a reduced dosage, resulting in higher noise.

For this reason, we propose a comprehensive noise simulation that models different aspects of noise in X-ray images, such as Poisson noise, electronic noise, and scintillator (detailed in Section 2.1.4) blurring.

Furthermore, the noise model's parameters are adjustable, allowing for the simulation of different detectors and dose levels, with the goal of enabling better generalization of the trained models. We evaluate our noise simulation by comparing the simulated images to real X-ray images acquired at different dose levels and assess the impact of various noise components on the simulation's accuracy.

3.1 Related Work

In the field of X-ray imaging, numerous noise simulation methods have been proposed [63, 21, 25], each with its own set of advantages and disadvantages. In this section, we provide an overview of the state-of-the-art noise simulation methods and compare them to our proposed method.

Båth et al. [21] developed a method, which necessitates the understanding of the Noise Power Spectrum (NPS) from two pre-acquired empty (flat field) images at two different dose levels. The known NPS is used to generate noise, which, when added to the original image, simulates the noise characteristics at the lower dose

level. While it is a precise method based on the correct NPS, it requires flatfield images and does not allow for adjustment of the noise characteristics.

Borges et al. [25] introduced a novel approach by adding noise in the Anscombe domain [150]. Similar to [21], this method involves acquiring two flatfield images at different radiation doses and modeling the noise based on the local variance of these images. While this method is more flexible due to its adjustable variance parameter, it does not account for scintillator blurring or electronic noise.

Hariharan [82] also approaches the noise simulation by adding noise in the Anscombe domain. The noise characteristics are estimated from the local variance of a high dose and low dose X-ray image. They also account for electronic noise and scintillator blurring.

In contrast to [82] and [25], Cesarelli et al. [41] propose an approach that directly simulates Poisson noise without the need for the Anscombe domain. This is achieved by estimating an increase in this type of noise through the addition of Gaussian noise with zero mean and variance dependent on the expected pixel intensity. However, they do not account for noise already present in the high-dose image, nor do they account for scintillator blurring.

Our approach is based on the premise that deep learning models generalize better when trained with a diverse dataset. Thus, we propose a noise simulation with adjustable parameters directly related to physical properties, such as the variance and mean of the noise. Consequently, this method eliminates the need for the Anscombe domain and does not necessarily require measurements to generate realistic noise. Moreover, it considers the original noise in high-dose images, enabling the precise simulation of low dose characteristics on Ground Truth (GT) images that already contain some noise.

3.2 Methodology

In this chapter, we describe the four steps of the noise simulation pipeline, as shown in Fig. 3.1. First, we examine the fundamental properties of the photon distribution, which form the basis of the noise simulation. Next, we estimate the detector gain to convert pixel values into the photon count domain. The photon count is then adjusted to account for enhanced quantum noise. We then consider the inherent scintillator blurring in the detector, which affects the quantum noise characteristics. Finally, electronic noise is added to the image to complete the noise simulation.

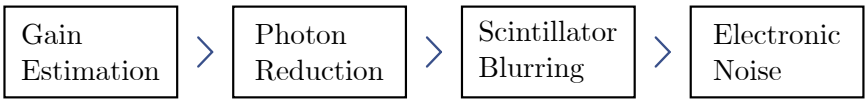


Figure 3.1: The four stages of the noise simulation pipeline.

3.2.1 Poisson Distribution Approximation

The Poisson distribution [178], which describes the probability of z photons hitting a detector pixel based on the average photon arrival rate λ (as discussed in Section 2.1.3), serves as the primary source of noise in X-ray images. Unlike the Gaussian distribution, the Poisson distribution exhibits signal-dependency, indicating that the level of noise varies with pixel intensity.

Two fundamental properties of the Poisson distribution are crucial for the proposed noise simulation. Firstly, as stated in Eq. (2.4), the mean and variance of the Poisson distribution are both equal to the mean photon arrival rate.

Secondly, the Poisson distribution can be approximated by a normal distribution, whose mean and variance equals λ [96]:

$$P(z|\lambda) \approx N_o(\mu_o = \lambda, \sigma_o^2 = \lambda)\,. \tag{3.1}$$

It is important to note that λ varies for each pixel. Consequently, a distinct normal distribution is utilized for the approximation of each pixel. Furthermore, we use the index o as an indicator to denote that the variables are in reference to the photons that originally arrived at one detector pixel.

3.2.2 Detector Gain Estimation

In X-ray imaging, photon counts are converted to visible light using a scintillator. This visible light is then converted into an electrical signal by a photodiode [246]. The electrical signal is subsequently converted by an ADC to pixel intensities i. Therefore, the pixel intensities i do not directly represent the photon counts z. However, the relationship between pixel intensities i and photon counts z can be approximated linearly using the detector gain k:

$$i = k \cdot z\,. \tag{3.2}$$

Consequently, the variance and mean of the photon count domain are also linked to the image pixel domain. The variance of the pixel intensities, σ_i^2, is equivalent

to $k^2\sigma_o^2$. The image mean is linked to the mean of the photon count via $\mu_i = k\mu_o$. As per Eq. (2.4), $\sigma^2 = \mu = \lambda$. Thus, we can obtain k as follows:

$$\frac{\sigma_i^2}{\mu_i} = \frac{k^2\sigma_o^2}{k\mu_o} = \frac{k^2\lambda}{k\lambda} = k \tag{3.3}$$

However, in reality, only i of each pixel is known, not the underlying variance σ_i^2 or mean μ_i of the Poisson distribution. Therefore, these parameters must be estimated. It is important to note that both σ_i^2 and μ_i depend on λ, which varies for each pixel, leading to different values of σ_i^2 and μ_i for each pixel. In order to estimate the parameters σ_i^2 and μ_i, we make the assumption that neighboring pixels exhibit similar noise characteristics, given that the X-ray attenuation should not significantly vary over short distances in most cases. Consequently, by measuring the different intensities i within a small neighborhood, we can compute an estimate for σ_i^2 and μ_i. This assumption holds even more true for areas in the image where there are minimal to no anatomical changes. For this reason, we propose an automatic algorithm that estimates σ_i and μ_i, which are subsequently used to compute the detector gain k. This algorithm is designed to automatically select areas with minimal anatomical variations. This selection process is facilitated by dividing the image into patches and calculating the entropy of each patch. The patches with the lowest entropy, indicating the least amount of anatomical changes, are selected. Consequently, the estimated σ_i^2 and μ_i correspond to the mean and variance of the pixel intensities within these selected patches. Having estimated σ_i^2 and μ_i, the detector gain k can be automatically calculated following Eq. (3.3).

3.2.3 Photon Reduction

A dose reduction by the factor α means a decrease in the number of photons by α and consequently a decrease in the mean photon arrival rate to $\alpha\lambda$. This decrease leads to a higher uncertainty in the photon count, which is reflected in the increased noise. Therefore, to simulate a dose reduction, the noise characteristics must be adjusted, then the number of photons must be scaled. Follwing Eq. (2.4), the new variance and mean of the dose reduced noise is $\sigma_n^2 = \alpha\lambda$ and $\mu_n = \alpha\lambda$. Considering the definition of the SNR as $\text{SNR} = \frac{\mu}{\sigma}$, the SNR of the dose reduced image is

$$\text{SNR}_\alpha = \frac{\mu_n}{\sigma_n} = \frac{\alpha\lambda}{\sqrt{\alpha\lambda}} = \sqrt{\alpha}\frac{\mu_o}{\sigma_o} = \sqrt{\alpha}\text{SNR}_o \tag{3.4}$$

Here, μ_o and σ_o represent the mean and variance of the original photons detected, while SNR_o denotes the original SNR. Consequently, when the dose is reduced by α, the SNR is scaled by $\sqrt{\alpha}$.

The knowledge regarding the degree of SNR reduction in relation to the scaling of the photon count can now be applied to simulate the noise in the dose-reduced image. A pixel-specific Gaussian noise $N_p(0,\sigma_x^2)$ is incorporated into the original image to achieve a new SNR_α. The addition of N_o to the image results in a new variance $\sigma_o^2+\sigma_x^2$. The new SNR is $\frac{\mu_o}{\sqrt{\sigma_o^2+\sigma_x^2}}$. Consequently, the following equation must hold true:

$$\frac{\mu_o}{\sqrt{\sigma_o^2+\sigma_x^2}} \stackrel{!}{=} \sqrt{\alpha}\frac{\mu_o}{\sigma_o}, \tag{3.5}$$

Rearranging the formula results in $\sigma_x^2=(\frac{1}{\alpha}-1)\sigma_o^2$, or equivalently, $(\frac{1}{\alpha}-1)\lambda$. Therefore, the pixel-specific Gaussian noise to be added is solely dependent on the mean photon arrival rate. We estimate λ for each pixel by applying a median filter to the photon-count image. Thus far, only the variance of the image noise has been enhanced. As a result, the mean and variance are not equal, and property Eq. (2.4) does not hold, indicating that the image noise is not Poisson distributed. Hence, a scaling factor s must be found to adjust the image intensities to restore the Poisson distribution. Scaling the image by s results in a new altered variance and mean, which must be equal:

$$s^2\sigma_n^2 \stackrel{!}{=} s\mu_o \tag{3.6}$$

The scaling factor s can be calculated by rearranging Eq. (3.6) as follows:

$$s=\frac{\mu_o}{\sigma_x^2+\sigma_o^2}=\frac{\lambda}{\lambda(\frac{1}{\alpha}-1)+\lambda}=\alpha \tag{3.7}$$

Thus, we can simulate Poisson noise by adding pixel-specific Gaussian noise with variance $\sigma_x^2=(\frac{1}{\alpha}-1)\lambda$ to the image and scaling the image intensities by α.

3.2.4 Scintillator Blurring

In X-ray systems, scintillators convert incoming X-ray radiation into visible light. However, light scattering within the scintillator can cause light to hit neighboring pixels, introducing a correlation between them. Consequently, the Poisson noise characteristics are altered [143, 134]. To account for this spreading effect, the simulated Poisson noise needs to be convolved with a Gaussian kernel defined by σ_s [165]. In our algorithm, we derive σ_s from the NPS of a single high-dose GT image, as suggested by [82], resulting in a value of 0.6 pixels, which is dependent on the scintillator material and thickness. Given that the pixel size of our detector is 0.296 mm, σ_s can also be expressed in millimeters as 0.177 mm. This allows other detectors with different pixel sizes to be simulated by adjusting σ_s accordingly. It is important to note that for deep learning training, this parameter can be varied around our proposed value to generate a diverse dataset. An exact estimation may not be essential, as the value for the required detector is likely to fall within the range of the augmented values.

3.2.5 Electronic Noise

Electronic noise is caused by the detector's electronic components, such as the ADC and the amplifier. These components are independent of the photon count and can be modeled by a Gaussian distribution with a constant variance, σ_e^2 [68]. The electronic noise is added as the final step in the noise simulation to the image with photon count pixel values. The variance of the electronic noise is determined by the detector. For our detector, we once again used the NPS as per [82], resulting in a determination of $\sigma_e^2 = 5$.

3.2.6 Adjustable Parameters

The proposed noise simulation automatically converts the input image to the photon count domain. Subsequently, the image is manipulated based on three adjustable parameters:

1. The dose reduction factor α,
2. The scintillator blurring σ_s.
3. The electronic noise σ_e.

α directly determines the amount of dose reduction or the number of photons in the image. Hence, α should be always set to the desired range of dose reduction, which needs to be simulated.

The scintillator blurring σ_s and the electronic noise σ_e are detector-specific parameters. The proposed simulation is intended to generate training data. Therefore, we recommend varying these parameters around our proposed values to train models for better generalization. However, if the noise simulation needs to be specific to a certain detector, we recommend measuring these parameters directly on that detector.

3.2.7 Noise Power Spectrum (NPS)

For the evaluation of noise simulation, the NPS of the images $\mathbf{I}$ in the photon count domain is computed. The NPS signifies the power of the noise component within a signal or image, denoted here as $\mathbf{I}$, in the frequency domain:

$$\mathrm{NPS}(f) = |\mathcal{F}\{\mathrm{noise}(\mathbf{I})\}|^2 \tag{3.8}$$

The image noise is determined by subtracting a high-dose ground truth image $\mathbf{I}_{gt}$, characterized by barely noticeable noise, from the lower-dose images $\mathbf{I}_n$, which inherently have a higher noise level. In addition to subtraction, the high-dose

	High-Dose	Lvl 1	Lvl 2	Lvl 3
mAs	8.0	2.0	1.0	0.5
kV	80.9	80.9	80.9	80.9
α	1	0.25	0.125	0.0625

Table 3.1: Acquisition parameters of the phantom X-ray images. A constant tube voltage of 80.9 kV was maintained, with variations in the tube current employed to achieve different dose levels and dose reductions α.

image must be scaled down by α to align with the intensity of the simulated low-dose image. This can be represented by the following equation:

$$\text{noise}(\mathbf{I}_n) = \mathbf{I}_n - \frac{1}{\alpha}\mathbf{I}_{gt} \tag{3.9}$$

3.3 Experiments & Results

The capabilities of the proposed noise simulation are examined by acquiring X-ray images at varying dose levels from a phantom. The simulation is applied to the high-dose ground truth image $\mathbf{I}_{gt}$ to generate simulated low-dose images that correspond to the low-dose ground truth phantom X-ray images. The simulated images are subsequently visually compared to the ground truth images. In addition to the visual comparison, a quantitative evaluation is performed by measuring the NPS of the noise maps of both the simulated and ground truth images. Moreover, the MSE between the NPS of these images is also calculated to provide a comprehensive comparison. The impact of the different simulation components, namely scintillator blurring and electronic noise, is analyzed by applying the simulation with and without these components.

3.3.1 Phantom X-ray Image Acquisition

The phantom X-ray images, all taken at the same position, specifically the phantom's chest, are captured at four different dose levels. The acquisition with 8 mAs is considered the high-dose image, and the complete image is illustrated in Fig. 3.2. The remaining three images are classified as low-dose images. These dose levels are achieved by varying the tube current while maintaining a constant tube voltage. Given that the dose level is linearly dependent on the tube current, the dose reduction factor α can be calculated using the ratio of the tube currents. The acquisition parameters for these phantom X-ray images are summarized in Table 3.1.

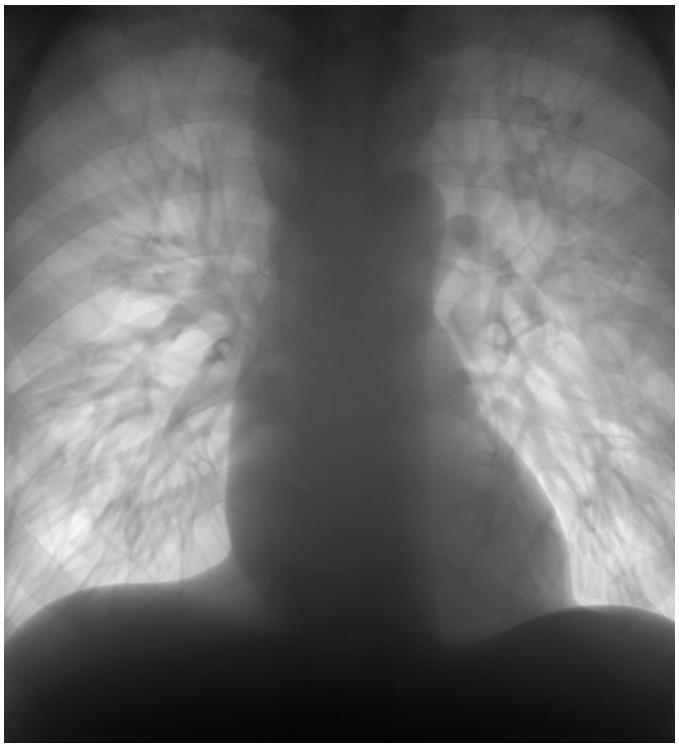


Figure 3.2: Phantom chest X-ray image used for the noise simulation.

3.3.2 Visual Evaluation

The simulations are conducted with corresponding α values to simulate the three dose reductions. Fig. 3.3 depicts a comparison of the simulated images with the ground truth image. The figures display patches to provide a higher resolution, enabling a more precise comparison. Furthermore, the corresponding noise maps of the patches are displayed.

A distinct discrepancy between the ground truth and the simulation without scintillator blurring is evident in both the image patches and the noise maps. The noise appears more fine-grained, a predictable outcome when the pixel correlation due to the scintillator is eliminated. Furthermore, a difference between the ground truth and the simulation without electronic noise is discernible upon closer examination of the noise maps. The electronic noise contributes to the overall noise, adding a fine-grained component visible in the real noise map and the complete simulation. This fine-grained noise results in a sharper appearance of the noise. No differences can be observed between the real and the patch of the complete simulation. Moreover, these effects are observable at all three levels of dose reduction.

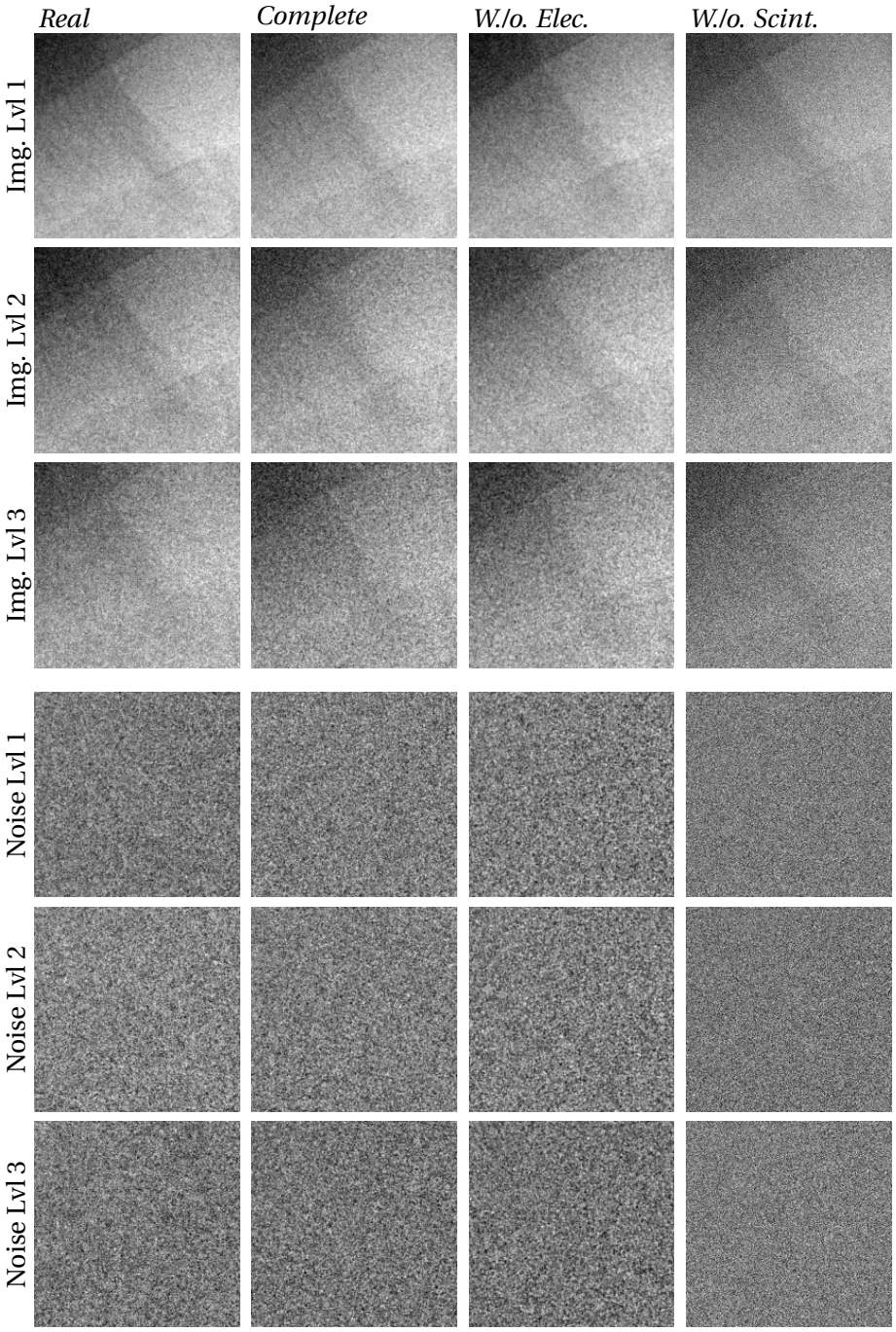


Figure 3.3: Patches represent all three levels of dose reduction. The first three rows display the image patches, while the fourth to sixth rows show the noise maps. The actual physical dose reduction is displayed, along with the complete simulation and the simulation excluding scintillator blurring or electronic noise.

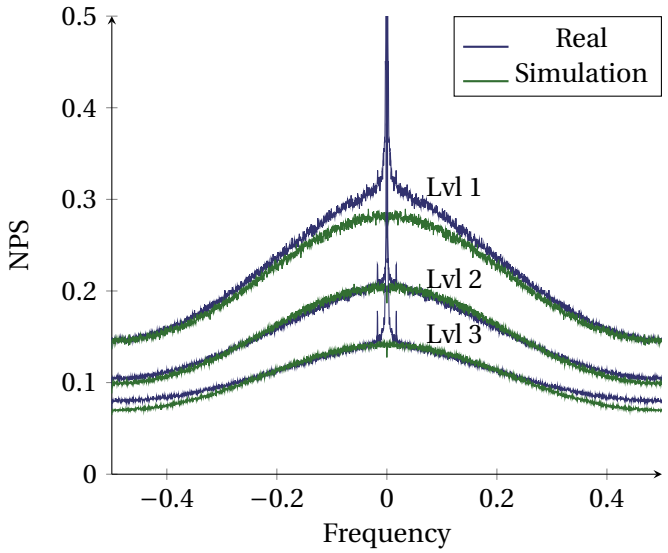


Figure 3.4: NPS of the complete noise simulation at different dose levels, compared against the ground truth NPS.

3.3.3 Noise Power Spectrum Comparison

To quantitatively assess the noise simulation alongside visual comparisons, the NPS of the noise maps is calculated. Fig. 3.4 shows the NPS of the complete simulation at various dose levels, compared to the ground truth. It is worth noting that the NPS of the real noise peaks at zero, suggesting an offset difference in addition to the noise differences. This is due to slight variations in the dosage scaling factor α in a real setup, resulting in a small offset in the noise map when the low-dose image is subtracted from the high-dose image. Excluding the offset absent in the simulated NPS, the NPS of the complete simulation closely mirrors the ground truth noise map, with minor variations contingent upon the noise level. At Lvl 1, the real NPS exhibits a slightly narrower peak than the simulation, while at Lvl 3, it displays a marginally flatter bell curve. These observations suggest that some nonlinear effects in the scintillator blurring are not entirely captured by our noise simulation. However, aside from these minor discrepancies, the NPS of the complete simulation closely resembles the ground truth noise map, indicating that the simulation accurately replicates the noise characteristics of the ground truth images.

Fig. 3.5 compares the NPS of simulations without scintillator blurring and without electronic noise to the ground truth at dose reduction Lvl 1. Comparisons of these simulations for dose reduction Lvl 2 and Lvl 3 are provided in the Appendix (Fig. A.1 and Fig. A.2).

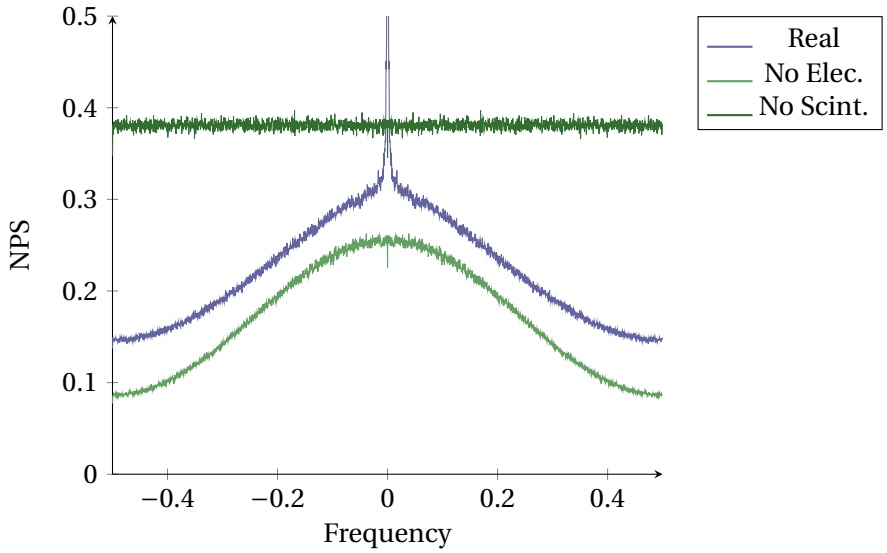


Figure 3.5: NPS comparison of the simulation without electronic noise or scintillator blurring, compared against the NPS of the real acquisition.

Lvl	Complete	W./o. Electron Noise	W./o. Scintillator Blurring
1	0.181	0.289	0.781
2	0.098	0.199	0.898
3	0.269	0.342	0.840

Table 3.2: The MSE between the simulated NPS and the ground truth NPS is calculated for each level of dose reduction across all three noise simulations.

Both NPS show significant differences from the ground truth when compared to the complete simulation. The NPS of the simulation without scintillator blurring resembles white Gaussian noise at a higher level, while the NPS without electronic noise lacks a constant component across all frequencies. This indicates that all components of the simulation are essential for accurately replicating the noise characteristics of the ground truth images.

In Table 3.2, the MSE between the simulated NPS and the ground truth NPS is calculated for each dose reduction level across all three noise simulations. The results align with the visual NPS observations, confirming that the complete simulation most closely resembles the ground truth. The simulation without electronic noise follows, while the simulation without scintillator blurring shows the greatest discrepancy from the ground truth.

3.4 Discussion

The experiments demonstrated that the complete noise simulation closely matches the ground truth noise across three different noise levels. This consistency suggests that our noise model does replicate other noise levels with the same accuracy. Moreover we showed that scintillator blurring and electronic noise are essential components, as the NPS changes when either component is omitted. Furthermore, the adjustable parameters allow for the simulation of different detectors and dose levels, enhancing the generalization of the trained models.

The proposed noise simulation will be employed in the upcoming chapters, Chapter 4 and Chapter 5. In these chapters, the simulation will be used to augment noise in the training data. The models trained on this data will show good generalization on real-world data, reinforcing the measured results of these chapters and underlining the assertion that the noise simulation is sufficiently realistic for training deep learning models.

3.5 Future Work

The noise simulation operates in the photon count domain; hence, an accurate estimate of the conversion factor k is crucial for the simulation's accuracy. The stability of the k estimate has not been investigated and should be addressed in future research. Moreover, at the current state, the detector gain estimation does not take into account scintiallator blurring. Consequently, to further improve the simulation's accuracy, the scintillator blurring can be incorporated in the estimate of the detector gain k.

3.6 Conclusion

We proposed a comprehensive noise simulation for X-ray images aimed at generating training data. The simulation automatically estimates the detector gain and incorporates the existing noise of the high-dose image. It is capable of modeling different dose levels, scintillator blurring, and electronic noise. Most importantly, all parameters of the simulation are adjustable and comprehensible. This allows for the augmentation of different detectors and dose levels in deep learning training, thereby enabling better generalization of the trained models.

4

Collimator Shadow Detection

Collimation minimizes radiation dosage to patients by limiting exposure to the Region of Interest (ROI) [237], as demonstrated in Fig. 4.1. However, the inclusion of collimated areas in the captured image introduces two significant drawbacks. Firstly, it can distort the image processing algorithms due to significant changes in pixel values caused by collimation. Secondly, it reduces the visible area of the image for radiologists, thereby potentially compromising the diagnostic quality of the image.

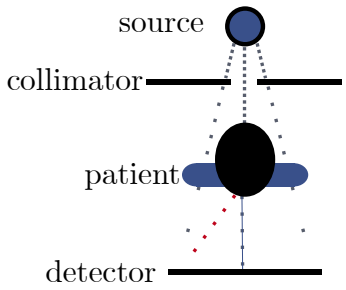
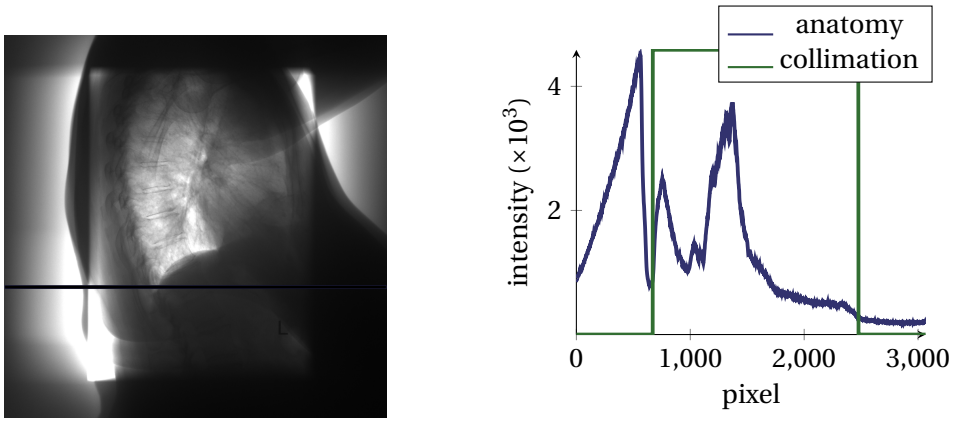


Figure 4.1: The figure illustrates the impact of collimation on dose distribution within a patient. The effect of scatter radiation, depicted in red, complicates the detection of collimation areas.

Therefore, it is essential to remove these collimated areas from the image before further processing. However, as depicted in Fig. 4.1, and further exemplified in Fig. 4.2, scattered radiation can penetrate the detector behind the collimator. This penetration results in a brightening of these areas, which subsequently complicates the detection of shadows.

This chapter presents a deep-learning-based algorithm for detecting the collimation area by estimating the parameters which describe the collimator shadow edges. The algorithm is a collaborative work between Benjamin El-Zein and myself, Dominik Eckert.



(a) X-Ray Acquisition

(b) Profile Plot

Figure 4.2: The figure illustrates the effect on collimation, represented through a profile plot and its corresponding image. Scatter results in higher intensities in the collimated areas compared to the ROI.

4.1 Related Work

Research publications in the field of collimator shadow detection is relatively limited. The most recent study was published in 2015 [170], before the widespread acceptance of deep learning methodologies in medical imaging. To the best of our knowledge, no studies have yet incorporated deep learning into this specific area of research.

Existing works can be broadly categorized into two groups: those that estimate the collimator boundaries or edges in the pixel domain [170, 153], and those that utilize the HT [83] to obtain a parametric description of the collimator boundaries [144, 130, 241, 111].

Ostojić et al. [170] propose a method for detecting the collimator shadow in the pixel domain. This method involves rotating the collimator edges based on the gradient histogram and calculating the Frobenius norm of the Hessian to detect the edges. Similarly, Mao et al. [153] employ a random forest, but instead of using superpixels, they classify each pixel directly to determine its affiliation to the collimator shadow or ROI. After applying a convex hull around the detected pixels to create a coherent area, they identify the corner points of the collimator within this area, as these points provide a complete description of the collimator.

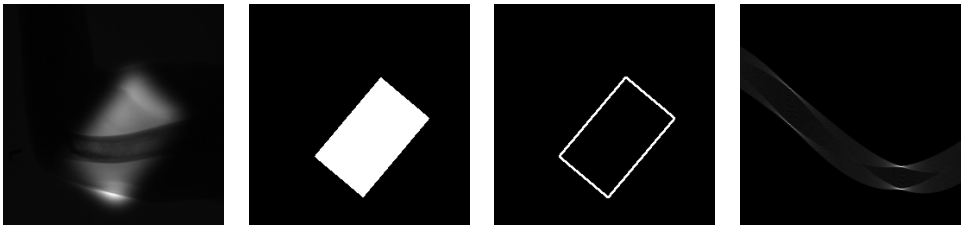
All HT based algorithms employ a similar three-step approach:

1. Estimate the edges of the collimator shadows.
2. Apply the HT to the edge image.
3. Acquire the collimator edge parameters from the HT domain.

The first step typically involves generating a segmentation mask, which is then converted into an edge image. The complete process is exemplified in Fig. 4.3.

However, this approach poses two significant challenges. Scatter can make areas behind the collimator appear brighter than those in the uncovered area, and edge blurring often results in poorly defined edges, as illustrated in Fig. 4.1. Consequently, the first challenge is accurately estimating these edges. Both effects are described in detail in Section 4.3.2.

The second challenge pertains to the determination of the collimator edge parameters from the HT domain, which is described in detail in Section 2.3.3. This process is complicated by several factors. Firstly, the exact number of possible collimator edges is uncertain due to the occasional invisibility of the entire collimator in the image. Secondly, instead of lines being represented by single points in the HT domain, they appear as sinusoidal curves due to resolution constraints, as shown in Fig. 4.3d. These curves' peaks symbolize the collimator's edges, with their amplitudes varying depending on the edge length. For instance, the two less pronounced peaks on the left side in Fig. 4.3d represent the collimator's two short edges, while the more pronounced peak on the right side represents a longer edge. Lastly, the presence of additional points and artifacts in the HT domain, resulting from an imperfect collimator edge image, further complicates the edge parameters' acquisition.



(a) X-ray Image (b) Collimator Mask (c) Edge Mask (d) HT domain

Figure 4.3: Example of collimator edge detection with Hough transform. From left to right: collimated X-ray image, segmentation mask corresponding to the collimator, Sobel edge detection of segmentation mask, Hough domain of edge mask.

The majority of existing literature on collimator detection presents diverse approaches to addressing these two aforementioned challenges. Wiemker et al. [241] and Lehmann et al. [130] estimate an edge image by identifying the edges through the calculation of the image gradients. Conversely, Luo et al. [144] attempts to classify two edges in both x and y directions as collimator edges, employing the method proposed in [197]. This method involves applying a threshold to the background and identifying the transition between the background and

the object. Zhao et al. [258] generates edge images by creating superpixels [1], which are perceptually meaningful atomic regions formed by combining pixels. These superpixels are then classified using random forest [31] to determine if they are located on the edge of the collimator.

The second challenge, acquiring edge parameters from the Hough space, is typically addressed in most studies by incorporating prior knowledge about the collimator edges. Luo et al. [144] posits that collimator edges exhibit a "butterfly" pattern in the HT domain and suppress all peaks that do not display this pattern. In contrast, Lehmann et al. [130] leverage the fact that a line must not be obscured by the collimated area adjacent to another line. Zhao et al. [258] incorporate prior knowledge that intersecting collimator lines form angles of approximately 90 degrees. They also utilize the knowledge gained from superpixel classification to determine if the line reconstructed from the HT domain separates the ROI from the collimator shadow.

4.2 Research Trajectories

Existing methods for collimator detection predominantly rely on analytical image processing techniques. We posit that complex tasks, such as identifying the ROI, generating an edge map with only the collimator edges, and extracting the collimator boundaries from the Hough domain, could be addressed more efficiently using deep learning techniques.

Deep learning typically necessitates a large volume of training data, a requirement often unfulfilled in medical imaging [112]. The collection of sufficient data is often constrained by the essential need for patient privacy. Furthermore, the annotation of medical data is a complex task that demands professional annotators such as radiologists. In particular, the annotation of collimator edges is notably time-consuming, as it involves not only label assignment but also the meticulous task of delineating the collimator edges. Building upon the physics-inspired noise simulation proposed in Chapter 3, we similarly suggest simulating the collimator edges on clinically acquired images. This approach facilitates the augmentation of different shapes and properties during training, thus increasing the number of available training samples. Moreover, by simulating the collimator shadows, the need for label assignment is eliminated, as the collimator edges are known and can serve as the ground truth.

Given sufficient training data, we can pursue two different approaches to train deep neural networks, mirroring existing methods that either rely on the HT or directly estimate the collimator boundaries.

Considering the principle of known operator learning [149], a neural network is likely to approximate the objective function more accurately if all known operators are incorporated into the network architecture. In line with this principle, Zhao et al. [257] proposed a differentiable Hough transform, called Deep Hough Transform (DHT), a known operator that can be seamlessly integrated into the training of a neural network.

Therefore, we propose to incorporate the DHT into a neural network architecture, which, in line with existing literature, consists of three parts:

1. An edge image generator.
2. The DHT.
3. Hough Domain refinement.

The edge image generator's role is to produce an image that exclusively illustrates the collimator edges from the input X-ray image.

The HT domain refinement is designed to remove incorrect edges and artifacts, and it is trained to equalize the amplitude of all points representing the collimator boundaries. This simplifies the extraction of the collimator edge parameters.

4.3 Methodology

Given the proven benefits of known operator learning, we explore the potential for detecting collimator edges with deep learning, by incorporating the HT into neural network architectures. Moreover, we establish a simulation pipeline for collimator shadow augmentation, which serves to generate training data for the proposed networks.

4.3.1 Data

Prior to any image post-processing, the collimation cropping must be performed. This is crucial as the presence of collimator shadows can distort the results of these processes. Consequently, a neural network should be trained to detect the collimator edges on unprocessed images.

Despite the notable scarcity of unprocessed images in the field, we successfully procured a dataset of 1680 raw clinical images from Siemens Healthineers. This dataset comprises images of various body parts, although their distribution is not uniform. The distribution of these images is depicted in Fig. 4.4.

The unprocessed images naturally contain the collimator shadows. Fortunately, a proprietary analytical vendor algorithm generates collimator labels, which are intended to identify the precise location of a collimator edge. As discussed, this

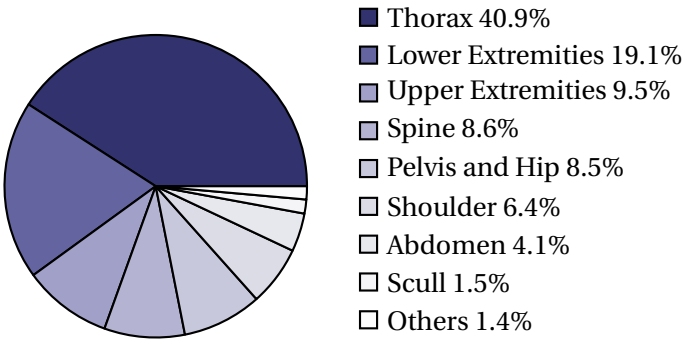


Figure 4.4: The different body parts in the X-Ray dataset and their distribution.

task is complex due to edge blurring and scatter effects, resulting in these labels exhibiting substantial deviations from the actual collimator edges.

The process of generating training data via simulation of collimator shadows requires the cropping out of the actual collimator shadows. We utilized the provided labels for this purpose and added a margin to account for potential errors in the labels. This approach ensures that the cropped image is free of any collimator shadows, even when considering possible inaccuracies in the label boundaries.

To maintain diversity in the test set, we deliberately chose images that represent all body parts found in the entire data set. We aimed to include as many diverse and challenging cases as possible, including those with implants or line artifacts.

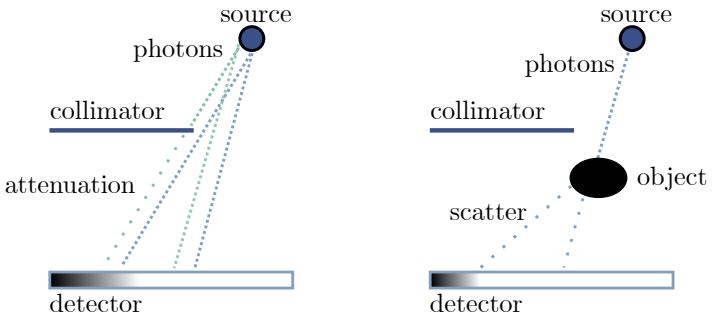
This approach resulted in the creation of three distinct test sets:

1. **General** test set: includes 80 images, ensures representation of all available body parts.
2. **Artifacts** test set: specifically containing 20 images with line artifacts that could be mistaken for collimator edges.
3. **Implants** test set: featuring 30 images with implants that significantly alter the image appearance.

Since the provided collimation labels are not accurate, they are manually adjusted for the test sets. This adjustment enables the evaluation of the network's performance on real-world collimator shadows. Furthermore, collimator shadows are simulated on the cropped images of these test sets, creating two versions of each set: one with the real collimator shadow and one with the simulated shadow. This allows for a comparison of the network's performance on real versus simulated collimator shadows.

4.3.2 Collimator Simulation

The simulation of collimator shadows necessitates the consideration of the physical effects that determine the collimator shadow. As illustrated in Fig. 4.5, the collimator shadow significantly impacts the image. Photons emitted by the X-ray source are attenuated by the collimator, leading to reduced intensities on the subsequent detector. However, the non-infinitesimal size of the source results in blurred edges of the collimator shadow. This effect is exemplified in Fig. 4.5a, where one photon beam (depicted in green) emitted from the first position is attenuated, while another beam emitted from the second position at the same angle (depicted in blue) fully penetrates the detector. Furthermore, photons penetrating an object are partially deflected, a phenomenon known as scatter [123, 193]. These deflected photons can penetrate the detector behind the collimator, thereby enhancing the intensity of the collimator shadow in certain areas. This effect is illustrated in Fig. 4.5b. Scatter behind the collimator is the primary reason why collimator edges are challenging to segment, as collimated areas can appear brighter than some parts of the object.



(a) Edge Blurring (b) Scatter

Figure 4.5: Illustration of collimator shadow effects.

As collimation modifies the scatter characteristics, the initial step in the simulation process is to eliminate the existing scatter from the clinical image. Ohnesorge et al. [164] proposed a method to simulate the scatter map on an existing image. Accordingly, this method is applied to the clinical images, and the estimated scatter map is subtracted. The collimator shadow is then simulated on this scatter-free images, taking into account the geometric manifestation, attenuation, and edge blurring. Based on these collimated images, a new scatter map is generated and added. Finally, as the alteration of the scatter and addition of the collimator shadow modify the number of arriving photons, we adjust for the new noise levels by applying the noise simulation of Chapter 3. The four stages of the simulation pipeline are depicted in Fig. 4.6.

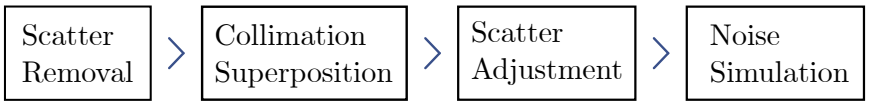


Figure 4.6: The four stages of the collimator simulation pipeline are illustrated.

Scatter Estimation

Ohnesorge et al. [164] suggest that the photon scattering by an object can be approximated using a Gaussian distribution. They employ this assumption to model the scatter by initially defining a scatter potential, $\mathbf{S}_p(\mathbf{I}|\mathbf{I}_0)$, which quantifies the object-induced scatter in the image $\mathbf{I}$. This potential is dependent on the image $\mathbf{I}$ and the primary $\mathbf{I}_0$, representing the image's intensity without any object-induced attenuation. The scatter potential is defined as follows:

$$\mathbf{S}_p(\mathbf{I}|\mathbf{I}_0) = c \cdot \left(\frac{\mathbf{I}}{\mathbf{I}_0}\right)^{\alpha} \cdot \ln\left(\frac{\mathbf{I}_0}{\mathbf{I}}\right)^{\beta} \tag{4.1}$$

The potential for scatter depends on the number of photons penetrating an object. The scatter potential $\mathbf{S}_p$ can be broken down into three components: a scaling factor c, the direct ratio between $\mathbf{I}$ and $\mathbf{I}_0$ exponential weighted by α, and the natural logarithm of the inverse ratio of $\mathbf{I}_0$ and $\mathbf{I}$ exponential weighted by β. This equation was empirically derived by Ohnesorge et al. [164].

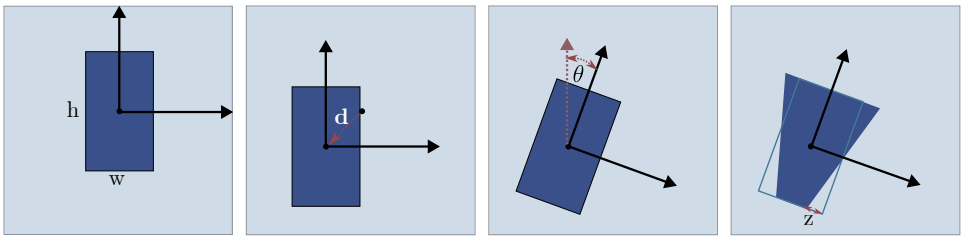
Finally, the estimated scatter $\mathbf{S}_e$ is obtained by convolving the scatter potential with a Gaussian kernel $\mathbf{G}_s$ as demonstrated in this equation:

$$\mathbf{S}_e(\mathbf{I}) = \mathbf{S}_p\left(\mathbf{I}|\mathbf{I}_0\right) * \mathbf{G}_s \tag{4.2}$$

Collimator Shadow

During X-ray image acquisition, situations may arise where the detector is freely positioned behind the patient, a scenario commonly seen in bedside imaging. Such a setup implies that the detector might not be centrally aligned with the X-ray source, could be rotated or tilted, and may have varying distances from the source. These factors, in turn, influence the positioning and geometric appearance of the collimator shadow. We address this by simulating collimator shadows of varying size, positioning, orientation, and skewness, as depicted in Fig. 4.7. The simulation begins by initializing a rectangle with height h and width w. This rectangle is then moved by an vector $\mathbf{d}$, rotated by an angle θ, and skewed by a parameter z. We define the so created binary collimator shadow as $\mathbf{C}_b$.

With the geometric manifestations defined, the attenuation a of the collimator shadow must be considered. Additionally, edge blurring is accounted for by convolution with a Gaussian kernel $\mathbf{G}_e$. Its spread in x and y directions is determined



(a) Initial (b) Position (c) Rotation (d) Skew

Figure 4.7: The simulation of the effects of varying collimator alignments to the source is achieved by transforming a rectangular collimator shadow through changes in position, rotation, and skew.

by the same standard deviation σ. Both σ and a can be adjusted to enhance different levels of edge blurring and augmentation.

The new collimator mask $\mathbf{C}_m$ is then obtained as follows:

$$\mathbf{C}_m = \mathbf{C}_b * \mathbf{G}_e \cdot a \tag{4.3}$$

Comprehensive Simulation Description

The simulation of the collimator on a clinical image, denoted as $\mathbf{I}$, can be expressed through the following steps.

Firstly, a scatter-free image, denoted as $\mathbf{I}_f$, is generated by subtracting the estimated scatter map $\mathbf{S}_e(\mathbf{I}|\mathbf{I}_0)$ from the clinical image $\mathbf{I}$:

$$\mathbf{I}_f = \mathbf{I} - \mathbf{S}_e(\mathbf{I}|\mathbf{I}_0) \tag{4.4}$$

Subsequently, the collimator mask is applied to the scatter-free image:

$$\mathbf{I}_c = \mathbf{I}_f \cdot \mathbf{C}_m \tag{4.5}$$

Lastly, the scatter map is recalculated and added to the image with the collimator shadow:

$$\mathbf{I}_{\text{out}} = \mathbf{I}_c + \mathbf{S}_e(\mathbf{I}_c) \tag{4.6}$$

In the final step, the noise simulation outlined in Chapter 3 is applied to the image with the collimator shadow to adjust for the new noise levels.

Data Generation

The fully set up simulation can be employed to generate training data. It produces modified X-ray images with new collimation, while concurrently preserving the

collimation masks $\mathbf{C}_m$ as target segmentation masks. Furthermore, by providing precise edge parameters, the simulation enables the creation of idealized Hough domain targets, where each edge is represented by a point of equal amplitude.

Phantom Evaluation

Prior to training the network on the simulated collimator shadows, an evaluation of the simulation pipeline is conducted. This process involves acquiring two chest X-ray images from a phantom; one with the collimator in place and the other without it, both captured under otherwise identical conditions. A collimator shadow is then simulated on the open field image, which can be compared against the image containing the actual collimator shadow. The findings of this comparison are elaborated upon in Section 4.4.1.

Feasability of Training with Simulated Data

The feasibility of the proposed collimator simulation is evaluated by training a network on the simulated data to estimate the segmentation mask, a model referred to as Simulation Trained Network (SimNet). Despite the inaccuracy of the real collimator segmentation masks provided in the training set, a second network, Real Data Trained Network (RealNet), is trained on the real collimator shadows. This allows for a comparative analysis between training with real and simulated data. Additionally, the performance of both networks is assessed on all three test sets, each with real and simulated collimators.

Both networks are trained using an identical setup. The chosen architecture is the DeepLabV3 model [44], a well-regarded model for semantic segmentation tasks. The network training employs a Dice loss function [260, 209], detailed in Section 2.5.3, and utilizes the Adam optimizer [115].

4.3.3 Edge Parameter Estimation with Hough Transformation

Upon successful generation of training data, it is necessary to develop and train neural network architectures for estimating the collimation edges. To achieve this, we explore the potential of incorporating a differential implementation of the HT [257] into neural networks.

Loss Functions

Training these networks necessitates two distinct loss functions. One such loss function, which operates on the collimation segmentation masks, is the Dice loss. It is denoted as $\mathcal{L}_{\text{dice}}(\hat{\mathbf{C}}, \mathbf{C})$, where $\hat{\mathbf{C}}$ represents the predicted mask and $\mathbf{C}$ the ground truth mask. The details of the dice loss are provided in Section 2.5.3.

The second loss function is designed to compare the predicted and actual Hough space. This space is predominantly empty, with only sparse points representing the lines. Thus, the loss function must be capable of managing this sparsity. We found the SSIM loss to be highly effective. Originally created to mimic human perception, the SSIM is sensitive to minor variations in the points within the Hough space. These changes are specifically targeted by the cross-correlation in the SSIM. The loss is represented as $\mathcal{L}_{\text{ssim}}(\hat{\mathbf{H}}, \mathbf{H})$, where $\hat{\mathbf{H}}$ denotes the predicted Hough space and $\mathbf{H}$ signifies the actual Hough space. The SSIM loss is detailed in Section 2.5.3.

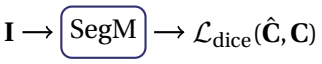
Network Architectures

We investigate three distinct possibilities to integrate the HT in a network architecture, which are compared against a Segmentation Network (SegNet), as illustrated in Fig. 4.8. Inherently, SegNet is limited to predicting segmentation masks and cannot estimate parameters. However, it provides a valuable reference point for evaluating the other networks. Its sole module, Segmentation Module (SegM), is built upon a U-Net [190] architecture. The architecture, as illustrated in Fig. 4.8a, offers greater flexibility for modifications compared to the DeepLabV3 model, making it an ideal choice for the integration of DHT. Consequently, SegM is also utilized as a component in the remaining three networks.

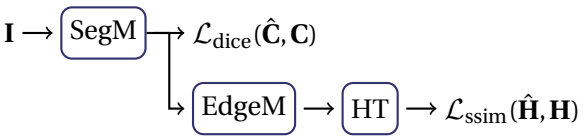
Regularization Network (RegNet) is illustrated in Fig. 4.8b. Like SegNet, it employs SegM, which is again trained with the Dice loss. However, it also aims to incorporate a second loss in the Hough space, which encourages the network to predict straighter lines. To achieve this, the network is enhanced with a second branch that transforms the output into the Hough space. This branch comprises two modules: an Edge Module (EdgeM) and a HT. The EdgeM changes the segmentation mask into a binary edge map. Instead of using an analytical edge detector, such as a Sobel operator, we discovered that the network performs better when the EdgeM is a small convolutional neural network with five convolutional layers, each followed by ReLU and batch normalization. Unlike a Sobel operator, this network can eliminate noise and artifacts that were either introduced or not properly removed by SegM. The output of the HT is then compared to the actual Hough space using the SSIM loss.

Hough Network (H-Net) is employed to investigate the feasibility of training a neural network to detect collimator edges exclusively in the Hough space. The architecture of this process is depicted in Fig. 4.8c.

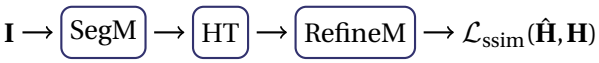
Given that SegM is not optimized for predicting collimator masks, we assume it to be optimized to predict edge maps, thereby negating the need for EdgeM as employed in RegNet. SegM is followed by HT. As elaborated in Section 4.1, the extraction of peaks indicating edges in the HT domain is challenging due to



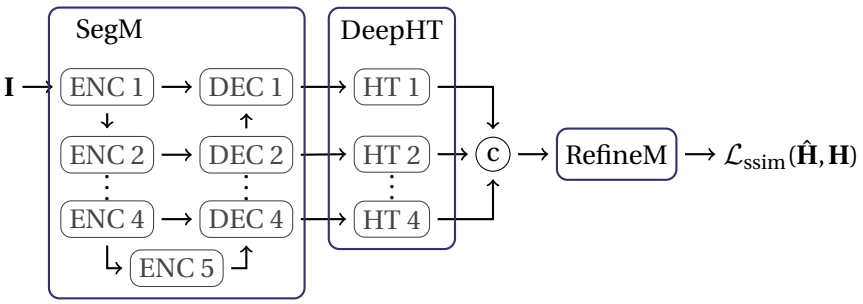
(a) SegNet



(b) RegNet



(c) H-Net



(d) DH-Net

Figure 4.8: The figure illustrates four distinct network architectures. SegNet is trained to predict the segmentation mask without incorporating a Hough Layer. RegNet, on the other hand, integrates a Hough Layer and utilizes its output in a regularization loss. H-Net is designed to predict only the Hough Space. Lastly, DH-Net employs multiple Hough Layers for each Decoding Block.

varying intensities, sinusoidal curves, and artifacts. To address this challenge, we propose the incorporation of a refinement network, Refinement Module (RefineM), following the HT, with the objective of simplifying the HT domain. The training targets in the HT domain are devoid of artifacts and each edge is represented by a single point with the same amplitude. Thus, we anticipate that RefineM is trained to eliminate artifacts and refine the HT domain. We utilized a ResNet-18 [86] as the RefineM, modifying it to match the output dimensions of the Hough space. The output of RefineM is evaluated against the Hough domain targets using the $\mathcal{L}_{\text{ssim}}$ loss.

Deep Hough Network (DH-Net), the final network architecture proposed by Zhao et al. [257], is depicted in Fig. 4.8d. Instead of applying a single HT, the output of each decoding block of SegM is transformed into the Hough Space, with each channel of the outputs processed separately. The outputs of all HT layers

are subsequently concatenated and reduced to one channel through a single 1D convolution, serving as the input to RefineM. All other aspects remain consistent with H-Net.

Training

The networks are trained using the Adam optimizer. The incorporation of HT into the networks necessitates a reduction in the Learning Rate (lr) from 10^{-3}, as used in SegNet, to 10^{-5}. This phenomenon is often observed in known operator learning [149]. The networks are trained for 300 epochs. Given the 1550 distinct X-ray acquisitions used for training, this leads to 465,000 unique collimator shadow manifestations presented to the networks.

Reconstruction & Evaluation

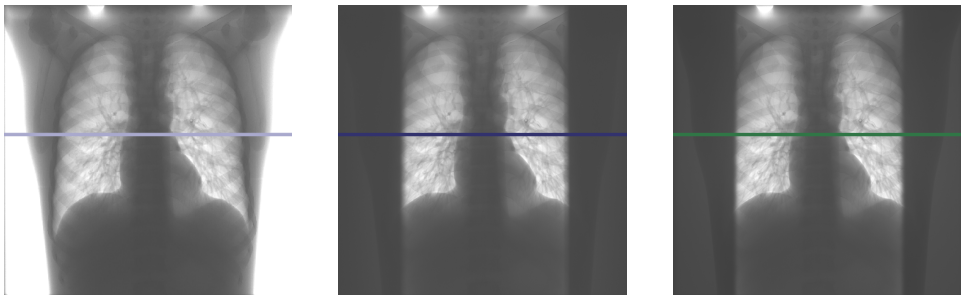
All four networks are evaluated on the three test sets. To enable a comparison between all four, the estimated hough domains of H-Net and DH-Net are reconstructed into a segmentation mask. The edge parameters are obtained by applying the watershed algorithm [120] to the Hough domain and subsequently calculating the center of mass of the resultant regions. The edges are then projected into the image space. In addition to identifying the edges, it is crucial to ascertain which side of each edge corresponds to the collimated and non-collimated areas. To do this, we assume that the non-collimated area has, on average, higher intensities than the collimated area. The non-collimated area is then identified by convolving the original input image with a Gaussian kernel. The spread in the x and y direction is defined by the same $\sigma > 50$, with the maximum value taken as the center of the non-collimated area.

4.4 Experiments & Results

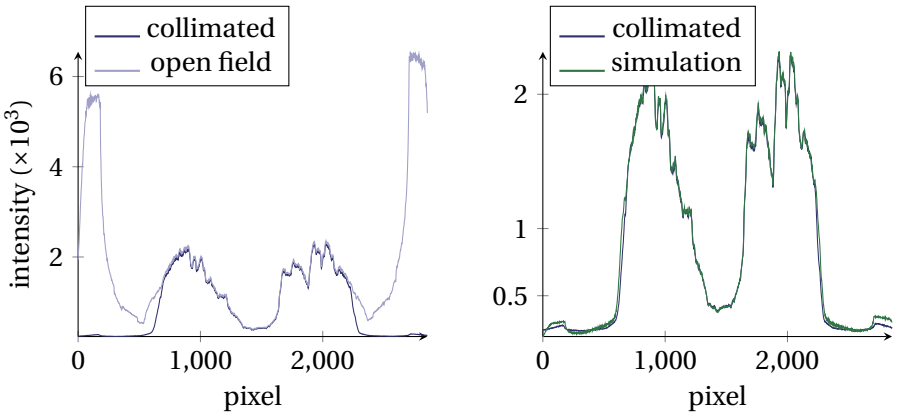
The experiments are divided into two parts. The first part evaluates the simulation pipeline, while the second part assesses the performance of the networks on the test sets.

4.4.1 Simulation Evaluation

Fig. 4.9 presents two acquired phantom images: the open field image (without collimation) and an image taken with the same parameters, but with collimation. Adjacent to these, the open field image with simulated collimation is displayed. To more accurately evaluate the simulation's precision, the pixel values from approximately the middle of all three plots are illustrated in the two line plots Fig. 4.9d and Fig. 4.9e. The positions of these pixels are denoted by colored lines in all three images.



(a) open field (b) real collimated (c) sim. collimated



(d) real collimated vs. open field (e) real vs. sim. collimated

Figure 4.9: Comparison of simulated and real collimation. The top row presents the images of the open field, collimated, and simulated scenarios. The bottom row illustrates the intensity profiles of the open field compared to the collimated scenario, and the collimated scenario compared to the simulated one. The intensity profiles correspond to the positions of the lines indicated in the images.

The difference between the open field and the actual collimation can be observed in Fig. 4.9d. As expected, at the position of the collimations, the intensity of the open field image is markedly higher than that of the image with collimation. These high intensities are transformed by the simulation to closely resemble the actual collimation, as illustrated in Fig. 4.9e. The mean deviation between the simulated and actual collimation is 5.08 %.

Despite the simulation requiring the removal of the initial scatter from the entire open field image, the deviation in intensities within the ROI is only 1.05 %. This implies that the newly added scatter characteristics, which consider the collimation, are realistic.

While confirming the simulation's accuracy, it still leaves the question of whether a network can be successfully trained on this data and if it generalizes to real data. Table 4.1 presents the results of the SimNet and RealNet on the three test sets. As detailed in Section 4.3.3, both networks are trained on the same 1500 X-ray images. However, during the training of SimNet, 465,000 different collimator shadows are simulated. Consequently, even though SimNet is trained exclusively on simulated data, it surpasses RealNet in performance on the general and artifact test sets.

Moreover, particularly for the general test set, the performance of SimNet when tested on real versus simulated collimation, with scores of 0.9749 and 0.9718 respectively, is very similar. This suggests that the network generalizes to real world collimation shadows.

However, there is a significant drop in performance on the implant test set. Hence, the trained network is less accurate on images with implants.

	SimNet		**RealNet**
	Clinical	Simulated	Clinical
General	0.9718 ± 0.027	0.9749 ± 0.041	0.9641 ± 0.048
Artifacts	0.9778 ± 0.025	0.9873 ± 0.014	0.9652 ± 0.038
Implants	0.9494 ± 0.071	0.9820 ± 0.027	0.9780 ± 0.015

Table 4.1: This table compares the performance of SimNet and RealNet on clinical and simulated data. It does so by comparing the dice scores of the estimated collimation against the GT collimation. Higher scores indicate higher similarity between the estimated and GT collimation.

4.4.2 Hough Network Performance

The performance of the four network architectures proposed in Section 4.3.3 is evaluated on the three test sets. The Dice scores between the estimated segmentation masks and the ground truth masks are measured. The results are illustrated in the boxplot in Fig. 4.10. Additionally, the mean scores are presented in the Appendix in Table A.1.

For SegNet and RegNet, the networks directly estimate the segmentation masks, which are then evaluated. Conversely, for H-Net and DH-Net, the masks reconstructed from the Hough domain are used for evaluation. In terms of the Dice score on segmentation masks, SegNet and RegNet yield superior results compared to the reconstructed masks of the two networks trained solely in the Hough domain. RegNet outperforms SegNet on all thre test sets, clearly indicating that

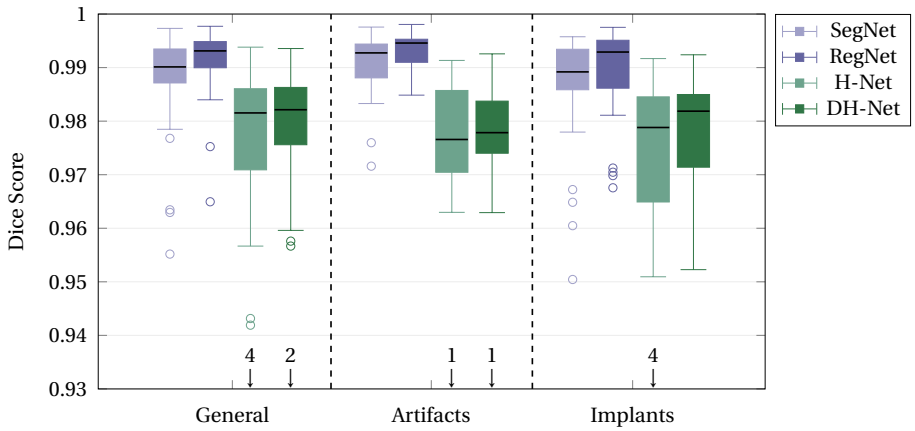


Figure 4.10: This boxplot depicts the results of the four networks on three different test sets.

the HT regularization is beneficial. Furthermore, for both networks, the most extreme outliers still achieve a Dice score over 0.95.

In comparison, the networks trained in the Hough domain, H-Net and DH-Net, perform approximately 0.01 Dice points worse and exhibit a greater variation in their results. However, it is important to note that the first two networks were both optimized on the Dice score. In contrast, the latter two networks were exclusively trained in the Hough domain. Consequently, the former are naturally expected to perform better when tested on the Dice score. This is evident when observing the error of the estimated masks corresponding to a randomly selected image, depicted in Fig. 4.11. The edges of the mask of SegNet appear non-linear and imprecise. As expected, since the DH-Net mask is reconstructed from the Hough domain, its edges are straight lines. However, a small offset in these straight lines can contribute more significantly to a lower Dice score than irregularities in the edge. Moreover, across all three test sets, the results of DH-Net outperform those of H-Net. This superiority is also evident in Fig. 4.11, where the edges of the DH-Net mask are more precise than those of the H-Net mask.

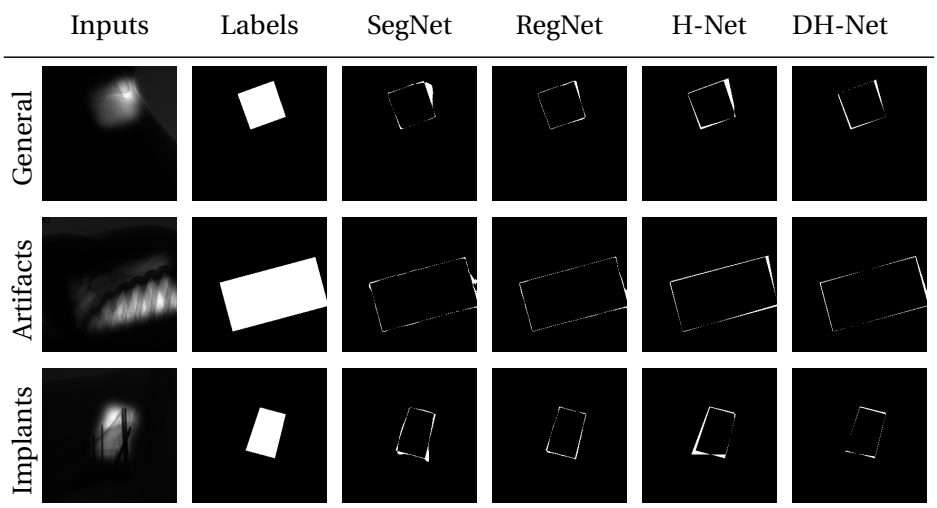


Figure 4.11: This figure depicts the estimated errors of collimations from the four different networks. For each of the three test sets, a representative image is displayed.

Both Hough networks exhibit more severe outliers, as indicated by the arrows in the boxplot. All outliers of DH-Net, along with some from H-Net, yield a Dice score of zero. This can be attributed to the reconstruction's inability to recognize the non-collimated area, as depicted in Fig. 4.12. The edges are accurately reconstructed, indicating that the Hough space is correctly estimated. However, the wrong side of the edges has been assigned to the ROI. Therefore, it's not the DH-Net that needs improvement, but the detection of the ROI.

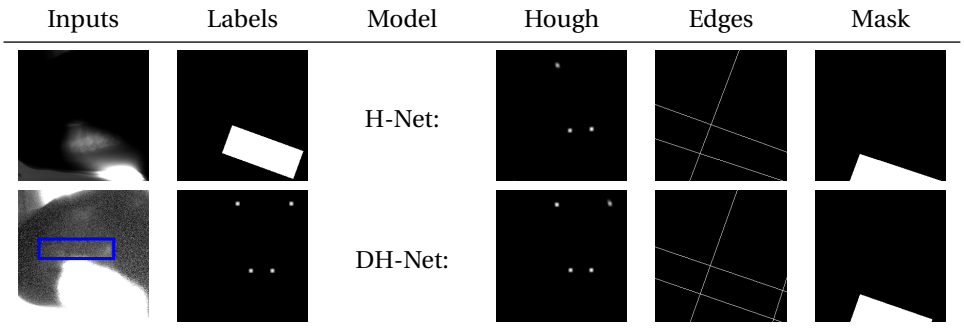
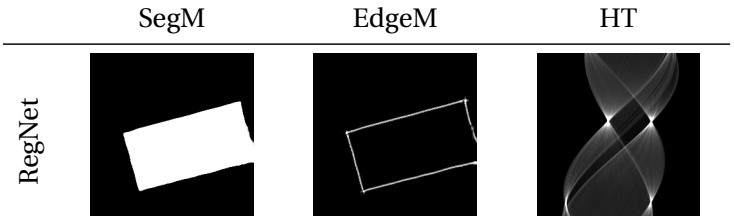


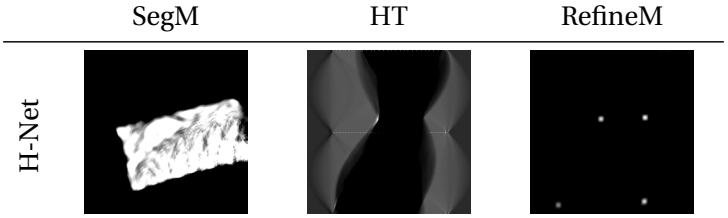
Figure 4.12: Example of an outlier detected by the Hough networks. The edges are correctly reconstructed from the Hough domain, however, the ROI is placed on the wrong side of the edge.

4.4.3 Hough Network Inspection

In Section 4.3.3, different network architectures that incorporate the DHT are proposed and depicted in Fig. 4.8. These networks comprise various modules, each designed with specific tasks in mind. In Fig. 4.13, the outputs of the various modules are examined to determine if they successfully perform their predefined tasks.



(a) Outputs of RegNet Modules



(b) Outputs of H-Net Modules

Figure 4.13: Outputs of the modules of RegNet and H-Net architectures are depicted.

In Fig. 4.13a, the outputs of the three modules SegM, EdgeM, and HT corresponding to a single input image are displayed. All three modules effectively perform their pre-defined tasks. SegM clearly produces a segmentation mask, while EdgeM transforms this mask into an edge map. As expected, when provided with a clean edge map, HT generates the hough points with the corresponding sinusoidal curves.

In Fig. 4.13b, the outputs of SegM, HT, and RefineM are displayed. Since RegNet lacks an EdgeM and is not trained on segmentation masks, we would expect its output to resemble an edge map. However, as observed, the output is more akin to a segmentation mask. As a result, the HT domain is imprecise and riddled with artifacts. However, RefineM can eliminate these artifacts, producing a HT domain where only the edges are represented as peaks. Notably, this process also removes the typical sinusoidal curves. Hence, it can be inferred that the presence of RefineM negates the necessity for SegM to produce clear edge maps during the training process. Beside the aim of RefineM being to eliminate artifacts and

generate a HT domain where all points have equal amplitude, this goal is only partially achieved, as one amplitude remains lower.

4.5 Discussion

The evaluation started with investigating the simulation pipeline in Section 4.4.1. We were able to confirm, on an X-ray image, that the pixel intensities of simulated collimation closely resemble those of actual collimation. Furthermore, we demonstrated the feasibility of training a neural network on simulated collimation, to generalize well on real data. However, a noticeable drop in performance was observed on images with implants. This could indicate that the presence of implants disrupts the simulation's precision, thereby hindering the effective generalization of the networks on these images.

Given that collimator edges invariably form straight lines, we sought to leverage this inherent characteristic. Consequently, we explored the integration of a HT layer, denoted as HT, into the three network architectures, depicted in Fig. 4.8.

In RegNet, HT is incorporated as an additional regularization branch. Our investigations confirm that this integration enhances the performance compared to SegNet, particularly in estimating collimator edges, which appear straighter. This further suggests that the backpropagation of gradients through the HT functions as expected, without any vanishing or exploding gradients.

For this reason, unlike RegNet, both H-Net and DH-Net were trained exclusively in the HT domain. To compare the performance of these networks with SegNet and RegNet, the estimated segmentation masks were reconstructed from the HT domain. Consequently, the obtained segmentation masks inherently possess straight edges. When evaluated using the Dice score, the performance of H-Net and DH-Net is slightly inferior to SegNet and RegNet. This is attributed to the fact that small offsets in straight lines contribute more significantly to a lower Dice score than irregularities in the edge. Because cropping neccesitates straight edges, these irregularities could significantly disrupt cropping algorithms. Conversely, the parameters estimated from the HT domain could be further utilized by these algorithms. Therefore, even though the Dice scores for H-Net and DH-Net are lower compared to SegNet, we argue that the HT networks present clear advantages.

We thus infer that networks, which can estimate their output in the HT domain, are superior to those that produce segmentation masks. This is particularly evident as RegNet outperforms SegNet. However, it is crucial to note that the Dice score is not ideal for performance evaluation, as it is insensitive to minor

offsets in the edges and irregular edges, features that are crucial for collimator edge detection.

4.6 Future Work

The assessment of the collimator simulation in Section 4.4.1 suggests its effective performance. However, SimNet is less accurate on images with implants. Thus, a comprehensive evaluation is necessary to ascertain if the simulation needs adjustments concerning implants, or if additional implant data is required for training.

In the context of HT networks, several opportunities for enhancement can be identified. The components of the two HT networks did not always perform according to their pre-defined tasks. Given the capabilities of RefineM in refining a HT domain abundant with artifacts, SegM was not compelled to generate edge images during the training process. However, this reduced the interpretability and reliability of the HT operation. Moreover, the reconstruction algorithm, however, failed to identify the center of the non-collimated area, resulting in several significant outliers, even though the edges were accurately reconstructed. On the other hand, the HT regularization in RegNet enhanced the quality of the segmentation masks.

We propose that a combination of RegNet and the HT networks is generally beneficial. This would involve a network with a SegM module that outputs segmentation masks. Unlike RegNet, it should also include a RefineM that refines the HT domain, facilitating the easy reconstruction of edge parameters. This approach enables the accurate reconstruction of edge parameters from the HT domain, while simultaneously determining the center of the non-collimated area using the center of mass of the segmentation mask, solving the issue of incorrect ROI determination.

Several possibilities exist to enable a network with a SegM that outputs segmentation masks and a RefineM that does not interfere with the training of SegM:

First, it can be examined, whether a RefineM with fewer parameters does not disrupt the training of SegM and EdgeM.

Secondly, a two-step training approach could be employed. In the first step, SegM can be trained without the HT domain. Subsequently, in the second step, the HT with a RefineM is added and trained, while the weights of SegM are kept constant.

Thirdly, a multi-tasking training approach similar to RegNet could be employed. In this approach, one loss function optimizes the HT domain output, while another loss function enforces SegM to output segmentation masks.

Another potential area of exploration involves integrating the HT domain reconstruction into the training process. Instead of training the HT networks in the HT domain, the networks could be trained on the reconstructed masks. We hypothesize that this approach would further stabilize the reconstruction process, as it would be an integral part of the training process. This approach can be further enhanced by utilizing the segmentation mask as prior knowledge to determine which edges are relevant in the HT domain. However, this approach presents the challenge of extracting the parameters from the HT domain in a differentiable manner. Without this differentiability, integrated reconstruction becomes unfeasible.

Not just the network architectures, but also the loss functions present opportunities for improvement. Rather than applying a Dice loss to the entire segmentation mask, it may be advantageous to confine the Dice loss operation to the edges. This approach would ensure that minor deviations in edges contribute more significantly to the loss.

Furthermore, we have demonstrated that the SSIM loss effectively manages the sparsity of the HT domain. Nevertheless, we hypothesize that this may be primarily due to the cross-correlation calculation in the SSIM loss. As a result, further investigation is necessary to ascertain if cross-correlation as a loss is adequate. Eliminating the elements of SSIM that do not contribute to the loss might result in more accurate gradients.

Additionally, another potential loss function could be a masked MSE loss, which assigns greater weight to the edge-representing points in the HT domain than to the background. We posit that exploring both options is beneficial, as the combination of the important elements of SSIM with a masked MSE loss could prove to be more effective than utilizing SSIM alone.

In addition to a network architecture that incorporates the HT, we propose that it is worth exploring an approach that either directly regresses the edge parameters or outputs a mask that highlights the collimator's edges. The first approach has the advantage of eliminating the need for an analytical reconstruction of parameters, while the second approach simplifies the task for the network, as the output is relatively similar to the input. However, this approach presents the challenge of identifying the edges in the output. Both approaches, similar to the HT, incorporate the prior knowledge that the edges are straight lines. The implementation of these methods facilitates the evaluation of the assumption that the network's performance is enhanced by incorporating a known operator, namely the HT, into its architecture.

4.7 Conclusion

The contributions of this chapter are twofold. Firstly, we have developed a simulation pipeline that generates collimator shadows on clinical images, thereby addressing the scarcity of training data and the labor-intensive task of manual labeling. Secondly, we have explored the application of deep learning for identifying collimator edges, particularly through the incorporation of the HT into the network architecture. The promising approach of incorporating the HT into a regularization branch, alongside a segmentation loss, has demonstrated potential for further utilization and enhancement in future works.

5

Denoising Breast Tomosynthesis Projections

In this chapter, we introduce a novel denoising network specifically designed for mammographic images. Our approach is tailored for practical application in real-world settings, such as clinical routines. Hence, we prioritize reliability and unbiased performance while ensuring that the network preserves as much detail as possible.

5.1 Related Work

Despite the advancements in technology, the practical application of deep learning methods in medical image denoising remains limited. This is largely due to their complex and non-deterministic nature, which raises concerns about their reliability and predictability. There is a prevailing skepticism among practitioners that these models, despite their potential, may behave unpredictably in real-world scenarios, particularly in cases that deviate significantly from the training distribution.

In light of these concerns, analytical algorithms continue to hold their ground in the realm of medical imaging denoising. In fact, there exist relatively recent methods that either build upon these traditional filters or benchmark their techniques against them. Some of the notable analytical methods employed in medical imaging include:

- Block-Matching 3D (BM3D) [4, 16, 81]
- Total Variation (TV) [133, 24]
- Wiener Filter [3, 160, 78]
- Median and Gaussian Filters [125, 105, 57]

Nevertheless, Zhang et al. [253] demonstrated superior performance of a deep-learning based denoising algorithm against analytical algorithms on photo-

graphic images. They achieved this by employing a relatively simple 17-layer CNN, known as Denoising Convolutional Neural Network (DnCNN), which estimates the noise map directly instead of the denoised image. Additionally, DnCNN denoises patches of the image rather than the entire image, allowing for sequential full-resolution denoising.

Despite the stringent requirements for reliability in medical imaging, such advances have spurred extensive research in deep learning-based medical image denoising. Specifically in mammography, research has culminated in the formulation of best practices for denoising with deep learning [102, 138, 202, 42].

A notable contribution to this field was made by Vieira et al. [228], who demonstrated that the most effective results are obtained by denoising the raw projections directly, training the neural network to address physically accurate noise. Additionally, this approach enables the simulation of a physically accurate dose reduction to generate training data [177, 254].

However, physically accurate noise primarily consists of Poisson noise, which is signal-dependent. Several studies have simplified the denoising process by implementing a variance-stabilizing transformation, such as the Anscombe transformation [8], which converts the Poisson noise into signal-independent white Gaussian noise [26, 230, 189].

Besides selecting the appropriate noise and data domain, choosing a suitable loss function is crucial for training denoising networks, a paradigm suggested and thoroughly investigated by Zhao et al. [256].

They discovered that the MSE function falls short in preserving intricate image details. This has led to the exploration of alternative losses for mammographic image denoising, such as SSIM and perceptual loss [104, 198]. Furthering this research, Gao et al. [73] proposed using a discrimantor in the fashion of a Wasserstein Generative Adversarial Network (WGAN) as a potential loss function specifically for denoising mammographic images. Despite these advancements, finding the perfect loss function remains challenging due to the absence of a universally accepted metric that accurately reflects human, especially radiologists', perception of minute structures in mammograms. This inherent subjectivity complicates the task of evaluating and designing the optimal loss function. Hence, we posit that no single best loss function exists so far.

Building on the discussed advancements, this work develops a deep learning-based denoising model for practical application. The proposed noise simulation, detailed in Chapter 3, is applied to simulate a physically accurate dose reduction to generate training data. Additionally, a novel loss function is introduced, specifically tailored to preserve details in medical imaging. This loss function is compared with other state-of-the-art loss functions, emphasizing the preserva-

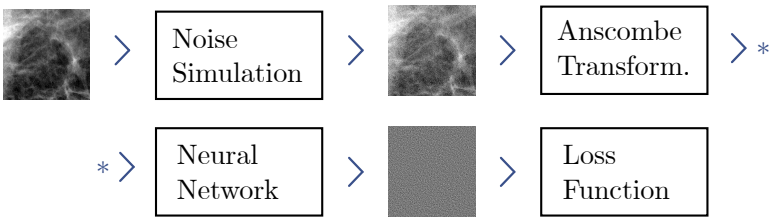


Figure 5.1: Complete setup of denoising training: 1. Dose reduction is simulated on a patch of an input FFDM. 2. Anscombe Transformation is applied for variance stabilization. 3. The neural network estimates the noise map. 4. The estimated noise map is compared against the GT noise map.

tion of microcalcifications. To ensure the model's reliability, extensive evaluations are conducted across various breast groups to prevent bias towards any specific patient group.

5.2 Methodology

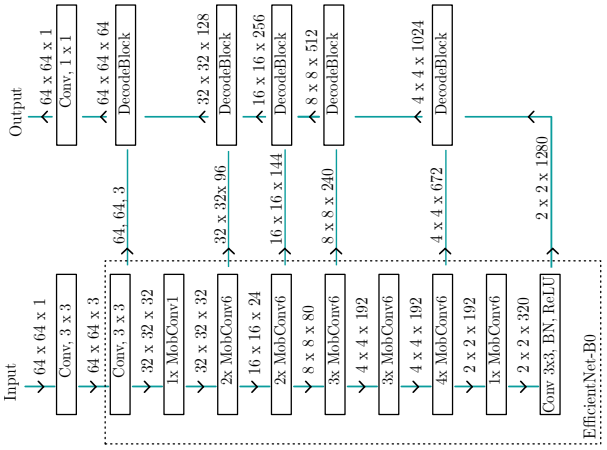
The training of the denoising network follows the best practices proposed in Section 5.1, with the sequential steps illustrated in Fig. 5.1. Digital Mammograms (DMs) are employed as GT targets. To enhance the training data, a dose reduction is simulated on the DMs as suggested in Chapter 3, ensuring the noise corresponds to that of DBT projections. The signal-dependent noise is then transformed to white Gaussian noise employing the Anscombe transformation, as described in Section 2.3.1. Subsequently, a neural network is trained to estimate the noise map, which can later be subtracted from the noisy image to yield the fully denoised image.

Given that DMs have resolutions larger than 2000×2000 pixels, denoising the entire image at once is not feasible due to Graphics Processing Unit (GPU) memory constraints. Consequently, before processing by the neural network, the images are cropped into patches of size 64×64 pixels, with each patch overlapping by 10 pixels. After processing, the patches are stitched back together to form the denoised image.

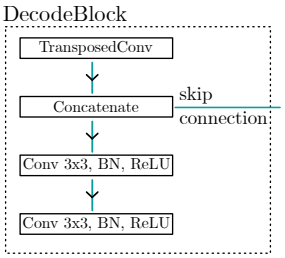
5.2.1 Network Architecture

The denoising network adopts a U-Net-like architecture [190]. To balance image processing time and performance, the U-Net is modified by replacing the encoding path with EfficientNet-B0 [215], as depicted in Fig. 5.2a. EfficientNet-B0 incorporates MobConv blocks from Mobilenetv2 [196], which utilize depthwise separable convolutions [48] and inverted residual blocks. The decoding branch consists of decoding blocks (Fig. 5.2b) connected via skip connections to the

output of the activation functions following the first convolution of the MobConv blocks. Processing an image of size 2082 × 2800 on an NVIDIA RTX 4000 GPU required 8.35 s with a standard U-Net and 3.16 s with the modified U-Net.



(a) Overview of the network architecture.



(b) DecodingBlock used in the decoding branch of the neural network.

Figure 5.2: Network architecture in a U-Net-like fashion, using EfficientNet-B0 [215] as the encoding branch.

5.2.2 Loss Function

The loss function is a critical factor in the denoising process. Although the MSE is a commonly used loss function, it tends to smooth out small structures, which is not ideal for medical imaging [256]. This is particularly significant for mammography, where small microcalcifications play a crucial role in diagnosis and should not be smoothed out under any circumstances.

To address this challenge, we propose a novel loss function, namely the ReLU-Loss ($\mathcal{L}_{\text{ReLU}}$). The central idea of the $\mathcal{L}_{\text{ReLU}}$ is to penalize noise overestimation more than underestimation, unlike most loss functions, such as MSE, which only consider absolute deviations. The underlying principle is that the misinterpretation of image structures as noise equates to an overestimation of noise.

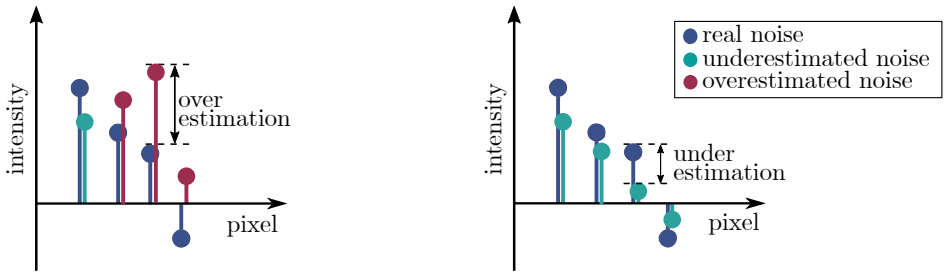
To implement the $\mathcal{L}_{\text{ReLU}}$, a pixel-wise error $e_i = -\text{sign}(\hat{n}_i) \cdot (n_i - \hat{n}_i)$ is first defined. This error is negative for noise underestimation and positive otherwise. Here, $\hat{n}_i$ represents a pixel i from the estimated noise map, and n_i represents the corresponding pixel i from the ground truth noise map. By applying a ReLU function to this error and adding it, weighted by the factor c, to a mean square error, the $\mathcal{L}_{\text{ReLU}}$ is obtained:

$$\mathcal{L}_{\text{ReLU}} = \text{MSE} + \frac{c}{N} \sum_{i=0}^{N} \text{ReLU}[(-\text{sign}(\hat{n}_i) \cdot (n_i - \hat{n}_i)]^2 . \tag{5.1}$$

Therefore, the addition of this term to the MSE introduces a penalty for overestimation. The impact of the $\mathcal{L}_{\text{ReLU}}$ on an exemplary 1D noise map is illustrated in Fig. 5.3.

Additionally, the $\mathcal{L}_{\text{ReLU}}$ is combined with SSIM, as SSIM is a good representation of human perception and represents another important direction for network optimization. Hence, the network loss is defined as follows:

$$\text{Loss} = \mathcal{L}_{\text{ReLU}} + \eta \cdot (1 - \text{SSIM}) . \tag{5.2}$$



(a) Noise estimation with MSE as loss.

(b) Noise estimation with $\mathcal{L}_{\text{ReLU}}$.

Figure 5.3: Illustration of the influence of the $\mathcal{L}_{\text{ReLU}}$ on estimated noise maps in a 1D example. With $\mathcal{L}_{\text{ReLU}}$, the estimated noise intensities do not overshoot the true noise intensity.

All hyperparameters used in the training process, including η and c, are detailed in Table A.2 in the Appendix.

5.2.3 Data

A reliable denoising model must effectively handle a wide variety of patient anatomies and recording settings. Therefore, a training dataset that encompasses a broad range of patients and recording conditions is essential. For this reason, the MBTST dataset, the largest open-source mammographic dataset containing unprocessed FFDM projections, is utilized to train the denoising model. However, the images in the MBTST dataset are all acquired at the same site with the same equipment, introducing potential biases in the training dataset, such as specific dose levels or overrepresentation of certain breast types due to demographic factors. This issue is particularly problematic if the test set also contains only images from the MBTST dataset. To address this issue and identify potential biases in the network, we use the Virtual Imaging Clinical Trial for Regulatory Evaluation (VICTRE) pipeline to generate versatile test data. The VICTRE pipeline allows the use of different breast phantoms with varying densities and thicknesses. Furthermore, it is possible to simulate different dose levels with the Monte Carlo

simulation, which simulates each photon's path. Thus, the VICTRE pipeline enables the generation of a test set completely independent of our noise simulation and training data.

Breast Group Definition

Breast types vary depending on demographic factors; for instance, they differ between Asia and America [39]. They are the most critical factor in mammographic image quality, as breast density and thickness directly influence the CDR [155]. To ensure that no patient with certain breast types is disadvantaged by the denoising algorithm, we evaluate its performance across various breast groups. We have identified 12 distinct breast groups for this purpose. These groups are established by combining four different breast densities (dense, heterogeneous, scattered, and fatty) with three different thicknesses (13-40 mm, 40-60 mm, and 60-83 mm).

Clinical Data

The MBTST dataset is employed to generate the training and one test sets. The objective is to ensure the training set is as unbiased as possible. To achieve this, selection is restricted to data where both breast density and thickness are available, enabling a balanced representation of breast types. This should prevent the network from learning biases towards specific breast types. For each of the defined breast groups, 20 images are sampled, yielding a total of 240 images per set. The test set is generated in the same fashion to allow for an investigation of the network's performance across different breast types.

Synthetic Data

Similar to the MBTST test and training dataset, the VICTRE pipeline is employed to generate two additional test datasets, each containing again the 12 different breast density and thickness groups with 20 images per group.

The first dataset (VICTRE I) is designed to have approximately the same average intensity for each breast type, ensuring that the noise level is consistent and independent of breast types. This approach helps to identify potential biases related to breast types rather than noise levels.

The second dataset (VICTRE II) mirrors the intensity distribution of the MBTST dataset, meaning each breast group has approximately similar noise levels to the MBTST test set. Thus, this dataset facilitates drawing conclusions regarding the differences in denoising behavior between the MBTST test set and the VICTRE simulated data.

Dataset	Mean Intensity Distribution
MBTST	real but unequal distributed
VICTRE I	equal distribution
VICTRE II	distributed like MBTST

Table 5.1: Brief description of the breast density distribution in the different datasets.

All three datasets, including the MBTST test set, are briefly described in Table 5.1.

To generate these test sets, 240 breast phantoms of varying densities are initially created. These phantoms undergo simulated compression, the extent of which depends on the breast density (Table 5.2). The compression process involves moving plates towards each other until the desired thickness is achieved, with displacement maps obtained for each finite element node [49]. After compression, VICTRE-MCGPU [14] is applied to simulate low and high-dose projections. This tool extends the MC-GPU of Badal et al. [13] to replicate a commercial mammography and DBT device, simulating photon noise, scatter, and different radiation doses.

In VICTRE I and VICTRE II, different mean intensity distributions are required, to have same mean intensity for each breast group in VICTRE I and mirror the MBTST mean intensities in VICTRE II. Consequently, the dose, i.e number of emitted photons λ_{req} must be adjusted to achieve the required mean intensity at the detector μ_{req}.

The relation between the detector intensities and the number of photons emitted by the source is influenced by the average attenuation characteristics γ of each individual breast:

$$\mu = \lambda \cdot \gamma + \text{o}\,. \tag{5.3}$$

with o being a known system offset to prevent negative intensities.

Thus, to determine the required number of photons λ_{req}, γ of each breast must be known. This can be achieved by conducting a pre-shot simulation, by setting λ to a known value λ_{preshot} and rearranging Eq. (5.3):

$$\gamma = \frac{\mu_{\text{preshot}} - \text{o}}{\lambda_{\text{preshot}}} \tag{5.4}$$

Thus, to find the required number of photons λ_{req}, Eq. (5.3) can be again exploited:

$$\lambda_{\text{req}} = \frac{\mu_{\text{req}} - \text{o}}{\gamma}\,, \tag{5.5}$$

Type	Compression Size
dense	50 % ± 0.5 %
hetero	40 % ± 0.5 %
scattered	35 % ± 0.5 %
fatty	25 % ± 0.5 %

Table 5.2: Compression sizes for the VICTRE phantom relative to the original size depend on breast density.

and γ can be substitued according to Eq. (5.4), yielding:

$$\lambda_{\text{req}} = \frac{\mu_{\text{req}} - o}{\mu_{\text{preshot}} - o} \lambda_{\text{preshot}} . \tag{5.6}$$

5.2.4 Performance Evaluation Methods

The evaluation of the image quality of mammographic images is a challenging task. Ideally, the evaluation metric reflects the perception of a radiologist and provides information on whether the diagnostic quality of the images has been enhanced. Since there is no straightforward analysis to answer this question, we examine different image qualities using various metrics and methods.

Our evaluation employs three different metrics to assess the denoising performance, namely MSE, SSIM, and PSNR, all detailed in Section 2.5.3. It is important to note that PSNR and MSE are pixel comparison metrics, while SSIM attempts to mirror human perception. Consequently, while the first two metrics are useful for a technical analysis, SSIM is more crucial in assessing the final image quality.

The aforementioned metrics are useful when comparing performance against the same image. However, when comparing performance between different images, two issues arise. First, the background of mammographic images lacks anatomical structure and varies in size depending on the image, which could significantly influence the previous metrics. Moreover, the mean and maximum intensities can vary substantially in X-ray images, making inter-image comparison challenging. Furthermore, according to the Poisson distribution, different mean intensities imply varying noise levels, which can also skew the quantification.

To address these issues, we propose two modified versions of the MSE and MSSIM metrics. Both metrics are applied solely to the foreground pixels, thereby ignoring the background. Furthermore, the images to be compared are normalized by dividing by their mean foreground intensity, yielding the normalized denoised image $\mathbf{D}_{\text{norm}}$, the noisy normalized image $\mathbf{N}_{\text{norm}}$, and the high-dose normalized image $\mathbf{G}_{\text{norm}}$.

Moreover, since different noise levels can influence the denoising performance, we propose a second normalization by normalizing to the MSE between the noisy and high-dose image, yielding:

$$\text{nMSE} = \frac{\text{MSE}(\mathbf{D}_{\text{norm}}, \mathbf{G}_{\text{norm}})}{\text{MSE}(\mathbf{N}_{\text{norm}}, \mathbf{G}_{\text{norm}})}. \tag{5.7}$$

To modify the MSSIM to consider only the anatomical structure, we utilize the local windows of the MSSIM calculation and adjust it to consider only those local windows containing anatomy:

$$\text{mSSIM}(\mathbf{D}_{\text{norm}}, \mathbf{G}_{\text{norm}}) = \frac{1}{N}\sum_{j=1}^{N} m_j \text{SSIM}(\mathbf{d_j}, \mathbf{g_j}) \text{ with } m_j = \begin{cases} 0, & \text{if } \mathbf{g_j} \text{ is background} \\ 1, & \text{otherwise} \end{cases} \tag{5.8}$$

The MSSIM is calculated using the normalized images $\mathbf{D}_{\text{norm}}$ and $\mathbf{G}_{\text{norm}}$, ensuring a fair comparison between images with different mean intensities. However, due to the potential non-linear relationship between noise levels and denoising performance, further normalization of the MSSIM was not performed. This approach allows a direct comparison of denoising performance, enabling the nMSE and mSSIM to effectively complement each other.

5.2.5 Statistical Analysis

To compare the denoising results across different breast groups, a series of statistical tests were conducted. First, Levene's test [132] was used to check if the variances among the groups were equal. This step is crucial because many statistical tests assume equal variances, and verifying this assumption helps ensure the validity of our analysis. Levene's test showed significant differences in variances, indicating that the assumption of equal variances was not met.

Because of this, Welch's ANOVA [240] was performed next. Welch's ANOVA is suitable when variances are unequal, as it adjusts the degrees of freedom to provide a more accurate analysis. This makes it a more reliable choice for comparing group means under these conditions, ensuring that our results are robust despite the variance differences.

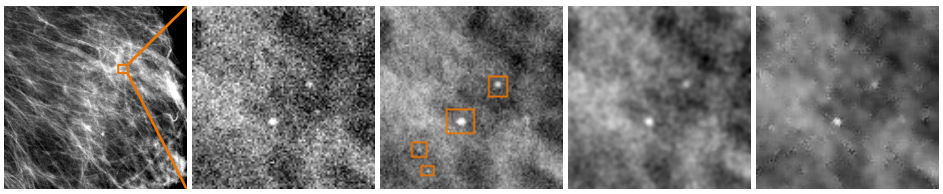
After Welch's ANOVA, the Games-Howell post-hoc test [72] was conducted to identify specific differences between groups. The Games-Howell test is ideal for pairwise comparisons when variances are unequal and sample sizes differ. It helps to pinpoint which breast groups have significantly different denoising performance, providing detailed insights into the data.

5.3 Experiments & Results

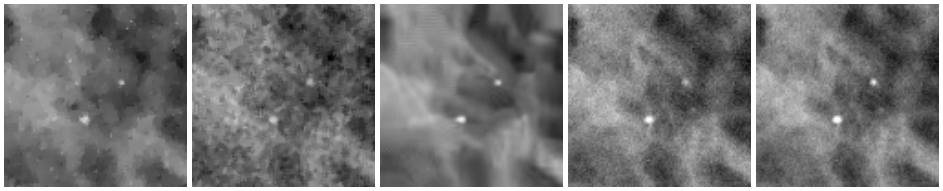
The denoising will be evaluated in several experiments in this section. The first experiment compares the proposed denoising network against the most common analytical denoising algorithms. The second experiment investigates the influence of the loss function on the denoising performance on small structures. Both experiments do not use the proposed three complete test datasets. Instead, they access single images or structures like microcalcifications from the MBTST test set. In the third experiment, we evaluate the denoising performance in terms of reliability and biases towards different breast groups. For this experiment, all three test sets are employed.

5.3.1 Comparison with Base Line Methods

The first experiment compares the proposed denoising network against the most common analytical denoising algorithms, namely the Gaussian Filter, Median Filter, Total Variation, Wiener Filter, and BM3D. Additionally, the DnCNN network, which set the denoising standards in photographic imaging, was retrained on the MBTST train dataset. The algorithms are compared by denoising an image with pleomorphic microcalcifications. A patch of the image, denoised with the different baseline methods, is depicted in Fig. 5.4. The measured results on the image are shown in Table 5.3.



(a) FFDM (b) Noisy (c) GT (d) Gaussian (e) Wiener



(f) Total V. (g) Median (h) BM3D (i) DnCNN (j) Proposed

Figure 5.4: Comparison of denoising methods against the most common baselines. For better clarity, one patch with pleomorphic microcalcifications is shown.

From visual observations, each conventional algorithm has its own flaws. The Total Variation filter and Wiener filter add additional artifacts, whereas the Gaussian and Median filters tend to blur the image. The artifacts of the Total Variation filter can even be confused with tiny microcalcifications. BM3D seems to offer a

Methods	PSNR	SSIM
Noisy	30.136	0.6421
Gaussian	35.348	0.8112
Median	33.649	0.7396
Total Variation	35.622	0.8148
Wiener [107]	35.240	0.8048
BM3D [53]	35.248	0.7966
DnCnn [253]	35.986	0.8384
Proposed	**36.354**	**0.8451**

Table 5.3: Comparison of the denoising model against baseline methods.

good trade-off between visible microcalcifications and fewer artifacts; however, the background appears altered and not close to the GT. The DnCNN, trained with our noise simulation on the MBTST train set, performs reasonably well, but one microcalcification is blurred and the noise is not completely removed. The patch denoised with our proposed method has very little noise, and both microcalcifications are clearly visible. Visually, it outperforms the baseline methods. The results are also reflected in the quantitative measurements in Table 5.3; our proposed method outperforms all other methods in terms of SSIM and PSNR. In the GT image, two more tiny microcalcifications can be spotted. However, these microcalcifications are completely lost in the noisy image and hence cannot be restored by any denoising algorithm. This highlights the limitations of denoising: if information is entirely lost in noise, it cannot be restored.

5.3.2 Loss Evaluation

Zhao et al. [256] demonstrated that using MSE as a loss function tends to smooth out small structures in images, which is undesirable in medical imaging. Therefore, we investigate the impact of various loss functions, including the proposed $\mathcal{L}_{\text{ReLU}}$, on the denoising performance of our network, with a particular focus on microcalcifications and small structures in DMs.

Patches with small structures are extracted from denoised images from the MBTST test set. The images are denoised with networks, trained with different loss functions. The performance is subsequently assessed by quantifying the MSE between the denoised and GT patches. Given that the microcalcifications now cover a substantial portion of the patch area, this measurement method offers a more precise depiction of small structure preservation.

Performance of Loss Functions on Small Structures

The first experiment evaluates the performance of the different loss functions on four small structures illustrated in Fig. 5.5. The corresponding MSE values are shown in Table 5.4. The perceptual loss performs the worst, both visually and quantitatively, introducing noise-like artifacts and proving unsuitable for medical image denoising. Despite the known issues with MSE, it performs comparably well on these structures. Quantitatively, however, the $\mathcal{L}_{\text{ReLU}}$ clearly outperforms the other loss functions.

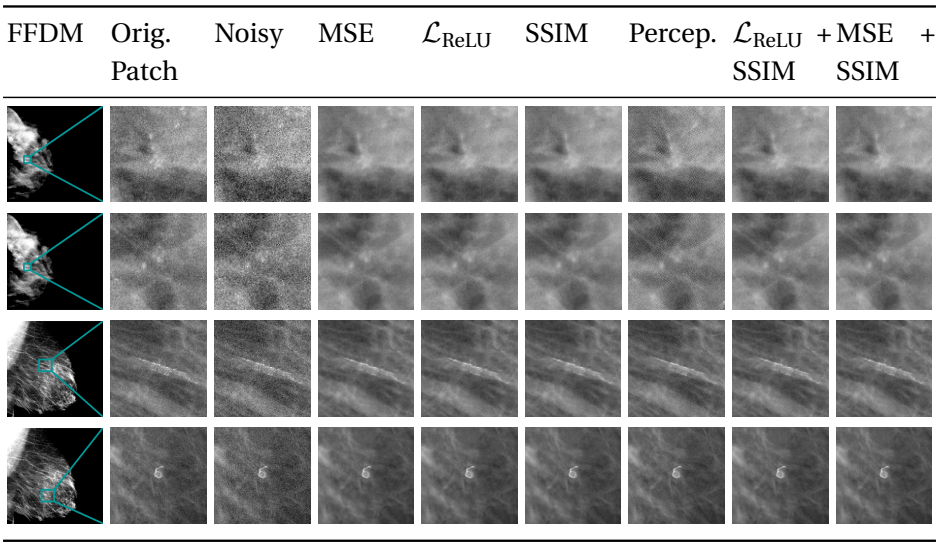


Figure 5.5: Comparison of various loss functions on mammographic structures: Four distinct patches are denoised by six networks, each trained with a different loss function.

Structure	MSE	$\mathcal{L}_{\text{ReLU}}$	SSIM	Perceptual	$\mathcal{L}_{\text{ReLU}}$ + SSIM	MSE + SSIM
1	298.5	**295.6**	299.5	520.1	296.7	299.5
2	304.9	**300.0**	303.2	494.0	301.6	303.9
3	245.8	**242.6**	245.3	427.0	244.0	245.7
4	253.2	**249.9**	252.5	432.0	251.0	253.2

Table 5.4: The MSE between the denoised structures in Fig. 5.5 and the high-dose ground truth patch is shown.

Denoising Performance on Single Microcalcifications

The second experiment evaluates the denoising performance on 15 single microcalcifications, depicted in patches of size 15×15 pixels. Six of these patches are

shown in Table 5.5, along with the average MSE values between the denoised and ground truth patches. The remaining nine patches are shown in the Appendix in Fig. A.3, and Fig. A.4 presents all individual measured values. In these microcalcifications, the smoothing behavior of MSE loss is clearly observed. Hence, SSIM and $\mathcal{L}_{\text{ReLU}}$ offer a clear improvement both visually and quantitatively. The performance of the network can be further improved by combining SSIM and $\mathcal{L}_{\text{ReLU}}$, as demonstrated in Table 5.5.

Calc.	Original	Noisy	MSE	$\mathcal{L}_{\text{ReLU}}$	SSIM	$\mathcal{L}_{\text{ReLU}}$ + SSIM	MSE + SSIM
p1							
p2							
p3							
p4							
p5							
				⋮			
p15							
Average MSE	-	-	900.13	701.06	680.14	**649.52**	730.09

Table 5.5: 6 out of 15 denoised patches with microcalcifications are depicted. The MSE between each patch and the GT is measured, and the average of these measurements is stated.

5.3.3 Bias and Generalization Investigation

Thus far, the denoising performance was evaluated on patches and structures. To investigate the generalization of the denoising network and potential biases towards specific breast types, the performance is now tested on the three complete test datasets described in Section 5.2.3. The investigation covers all 12 breast types, as detailed in Section 5.2.3, to determine if there are biases between these groups. The denoising performance is evaluated using the measurements proposed in Section 5.2.4: nMSE, and mSSIM. Furthermore, the mean intensity of the GT images is evaluated due to its direct correlation with the noise level in the images. This provides additional insights into the complexity of the denoising task, a factor that may exhibit variation across different breast types.

Fig. 5.6 exemplarily shows the denoising results for each breast density in both simulated and clinical data. Fig. 5.7 presents the denoising results for each thickness group across each test set. Both figures are available in the Appendix.

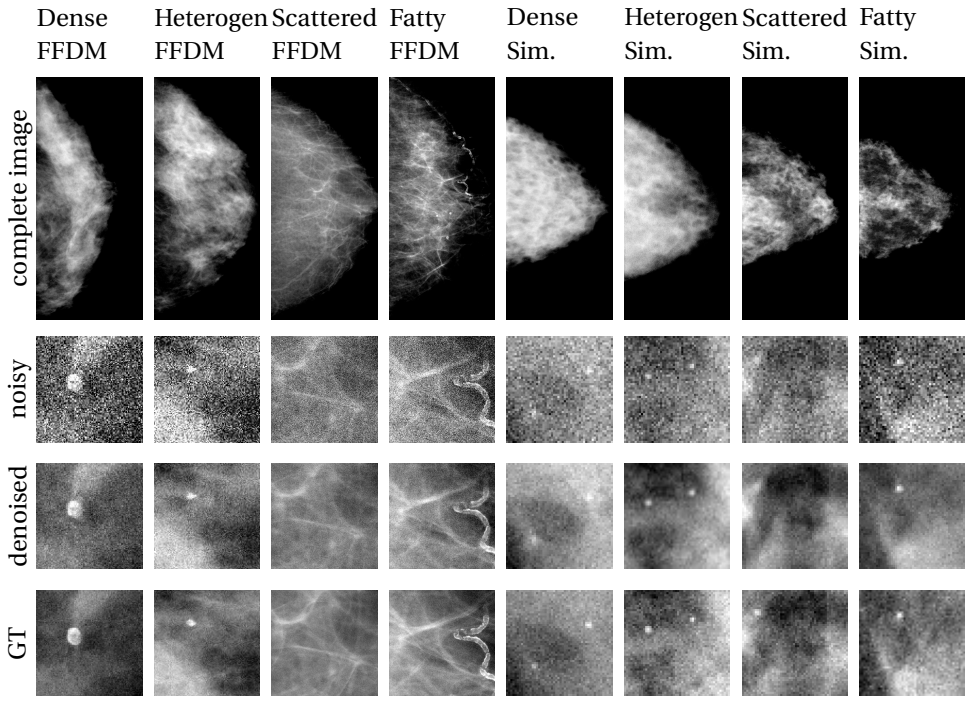


Figure 5.6: Four X-ray images from the MBTST and 4 images simulated with the VICTRE pipeline, each with a different density, are depicted. Patches are cropped out of the images to enable assessment at a higher resolution.

Statistical Difference between Breast Groups and Test Sets

In the initial experiment, we conduct a statistical analysis to identify significant differences in denoising performance and noise levels within each test set, as well as between the three test sets: MBTST, VICTRE I, and VICTRE II. Since Levene's test showes significant differences in the variances of the measurements, Welch's ANOVA is used to evaluate the statistical significance.

In Welch's ANOVA tests, the null hypothesis states that the means of the compared distributions are equal. If the p-value is less than 0.05, the null hypothesis is rejected, indicating that the distributions are significantly different. The results are depicted in Table 5.6. The initial three rows present the statistical differences among the breast groups within each test set, whereas the final row illustrates the statistical differences between the test sets.

The mean intensities between all breast groups and test sets are significantly different, except for VICTRE I, which was initially created to generate a test set with equal mean intensities. Thus, the generation was successful. However, for the MBTST test set and VICTRE II, each breast group has a different noise level distribution.

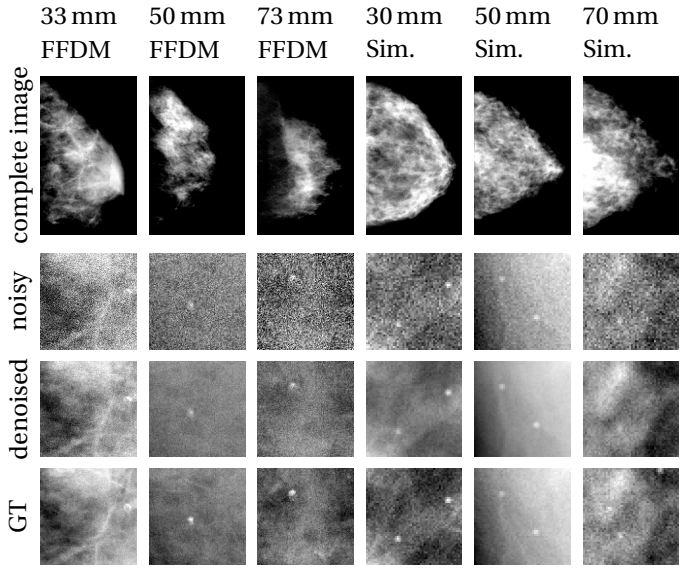


Figure 5.7: Three X-ray images from the MBTST and four images simulated with the VICTRE pipeline, each with a different thickness, are depicted. Patches are cropped out of the images to enable assessment at a higher resolution.

For each breast group, the measured nMSE is significantly different. In contrast, when comparing the mSSIM values, there is no significant difference between the breast groups of the MBTST set. This is an important observation since mSSIM is the most reliable metric and the MBTST set is the most realistic test set. The absence of statistical difference between the denoising behaviors of the groups suggests that the denoising behavior is reliable and potentially unbiased.

However, Welch's Anova revealed a significant difference in the mean intensities, and consequently, the noise levels of the MBTST test set. Therefore, the impact of varying noise levels on the denoising performance remains ambiguous. Additionally, significant differences were identified among the breast groups of the VICTRE datasets and between the denoising performance of the VICTRE datasets and the MBTST test set. This leaves the question of whether the network effectively generalizes to the simulated VICTRE data unresolved. To address these uncertainties, a detailed statistical analysis is conducted in the following section.

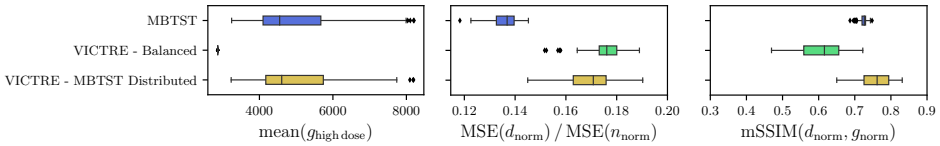
Denoised Breast Groups: Detailed Differences

The detailed statistical analysis is conducted by utilizing the Games-Howell post-hoc test, which allows for pairwise comparisons between groups. The differences in denoising performance and mean intensity between the three datasets are depicted in Fig. 5.8a, which presents the mean intensities, nMSE, and mSSIM for

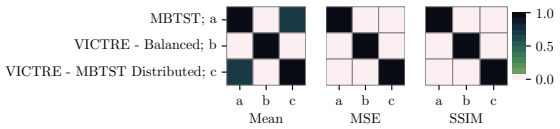
	mean(g_{highdose})	MSE(d_{norm}) / MSE(n_{norm})	mSSIM(d_{norm}, g_{norm})
MBTST	$< 10^{-5}$	$< 10^{-5}$	1.86×10^{-4}
VICTRE I	0.63	$< 10^{-5}$	$< 10^{-5}$
VICTRE II	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$
Between Test Sets	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$

Table 5.6: The statistical significance between the different breast groups is measured using Welch's ANOVA. The table shows the p-value for each measured distribution. This analysis is conducted for the MBTST data set, the simulated data with equal mean intensities (VICTRE I), and the simulated data whose mean intensities follow that of the MBTST data set (VICTRE II). Additionally, the overall distributions of each data set are compared against each other.

each test set. The statistical differences between these groups are illustrated in Fig. 5.8b.



(a)



(b)

Figure 5.8: Measurements on the three (denoised) datasets are depicted in this figure. (a) The distributions of the measurements across the different datasets are shown. (b) The statistical significance of the differences between the datasets is tested using the Games-Howell test. Squares are brightest when the p-value for the difference between two datasets is below 0.05.

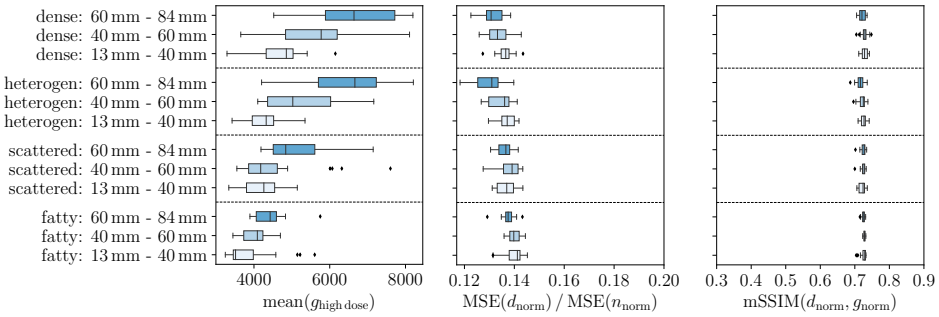
The test reveals no statistically significant difference between the mean intensities of VICTRE II and the MBTST test set. This confirms the successful generation of VICTRE II, as the original goal was to create a simulated dataset with the same intensity distribution as the MBTST dataset.

The denoising performance of VICTRE I is worse than that of the MBTST in terms of nMSE as well as of mSSIM. The significantly lower mean intensities, and thus higher noise levels of VICTRE I, are likely the cause of this inferior performance, as more noise poses greater challenges to denoising.

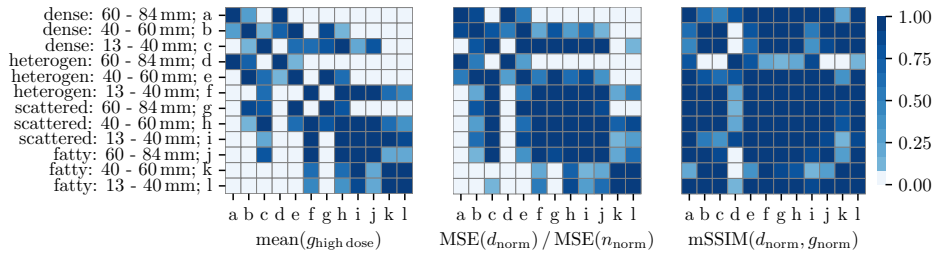
The noise level of VICTRE II is similar to that of the MBTST test set. Consequently, VICTRE II demonstrates superior denoising performance compared to VICTRE I due to its reduced noise levels. Moreover, in terms of mSSIM, the median value for VICTRE II is higher than that for MBTST, although VICTRE II exhibits a wider spread in denoising performance. This indicates, that regarding the mSSIM the network is capable of denoising the VICTRE data similarly well than the MBTST data. However, in terms of the MSE, the denoising performance is poorer on the VICTRE data.

In the next step, the mean intensities, nMSE, and mSSIM for each breast group are depicted, and the statistical differences between these groups are calculated. This analysis is conducted for all three test sets.

MBTST test set results are depicted in Fig. 5.9a. The statistical differences between the groups are shown in Fig. 5.9b.



(a)



(b)

Figure 5.9: Mean intensity, normalized MSE ratio, and SSIM are measured on the denoised X-ray images from the MBTST data set. (a) This figure shows the distribution of the measurements across different groups of the MBTST test set. (b) The differences between the groups are tested for statistical significance using the Games-Howell test. Squares are brightest when the p-value for the difference between two groups is below 0.05.

The mean intensities for the high-dose FFDMs, which serve as ground truth, vary for each thickness and density group. Hence, depending on the breast groups, there are different noise levels.

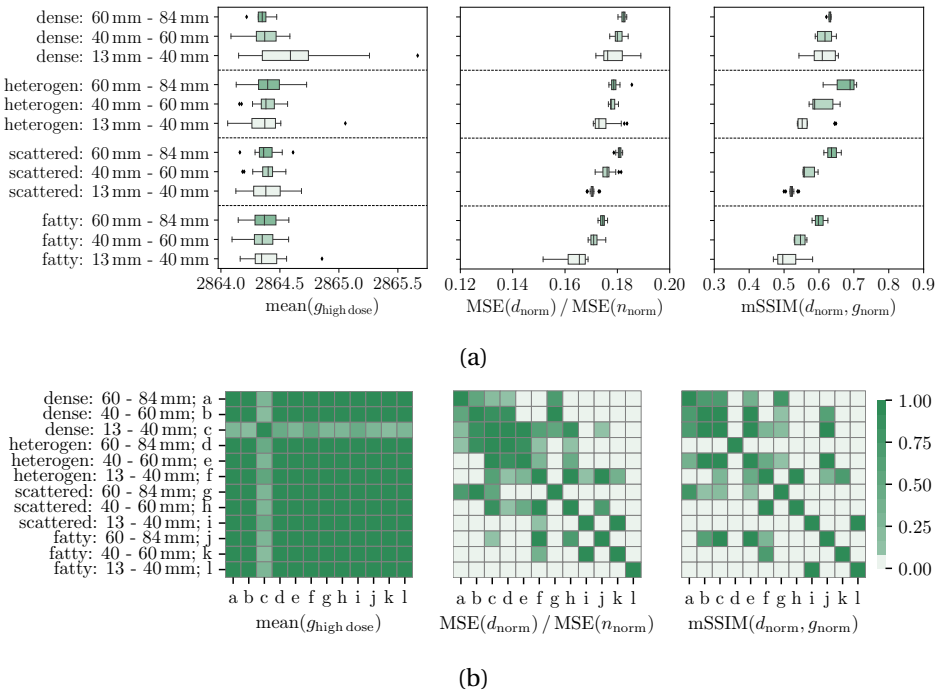
Regarding the nMSE, there are statistically significant differences in the denoising performance across different breast groups, with thicker breasts being denoised more effectively. However, the presence of larger areas with less or smoother background in thicker breasts might contribute to their superior denoising performance in terms of nMSE.

The pattern observed with mSSIM differs. The disparities between the groups are minimal, and no discernible trend can be identified. This observation is corroborated by the Games-Howell test, which indicates that the differences between most groups are not statistically significant. This outcome aligns with the previous section's Welch's ANOVA, which revealed no significant differences between the breast groups in terms of mSSIM. Unlike the nMSE, the mSSIM is more sensitive to altered image details and structures, akin to a human observer, while smooth areas are less pertinent. Hence, the results suggest that, with regard to human percpetion, the network can denoise all breast groups equally well, without any bias in the MBTST test set.

Compared to the MBTST dataset, the VICTRE datasets exhibit different mean intensities for each breast group. Therefore, to further investigate the impact of varying noise levels on the denoising performance across breast groups, an analysis of the VICTRE datasets is conducted.

VICTRE I results are depicted in Fig. 5.10. Welch's ANOVA test confirms that the mean intensities remain consistent across all groups. However, it also indicates that the denoising performance deteriorates with an increase in breast thickness, a trend that is not evident in the MBTST dataset. This raises the question of whether the unequal denoising performance across different breast groups can be attributed to the varying noise levels or to the differences between the VICTRE data and the MBTST data. This issue is further explored by analysing the VICTRE II results.

VICTRE II results are depicted in Fig. 5.11. VICTRE II follows the mean intensities of the MBTST test set. However, the trend that dense breasts are denoised worse than fatty breasts remains consistent with the results of VICTRE I. Hence, this trend is independent of mean intensities and is only present in the simulated data. Therefore, the amount of varying noise levels between the VICTRE datasets do not alter the relative denoising performance among the breast groups. This suggests that the same could be true for the MBTST test set, implying that the results concerning the MBTST test set are not strongly distorted by diverse noise levels. However, as illustrated in Section 5.3.3, significant fluctuations in noise



(a)

(b)

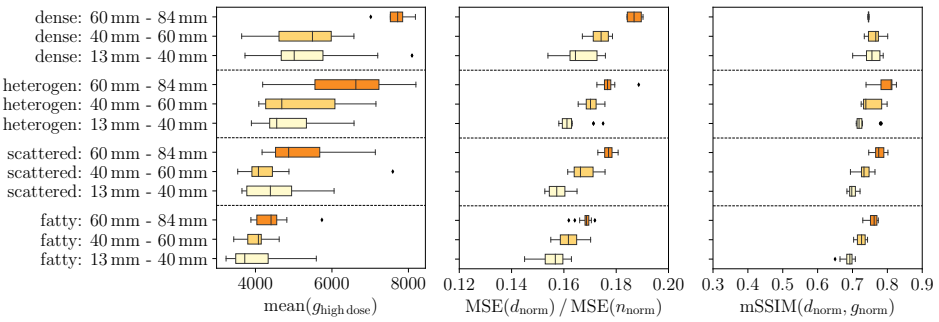
Figure 5.10: Measurements regarding the VICTRE I test set with approximately the same mean intensities are presented in this figure. (a) The distributions of the measurements across different subgroups are depicted. (b) The differences between the groups are tested for statistical significance using the Games-Howell test. Squares are brightest when the p-value for the difference between two groups is below 0.05.

level can affect the denoising performance. Thus, if the mean intensities exceed the difference between the mean intensities of the VICTRE datasets, the uniform denoising performance across all breast groups could potentially be jeopardized.

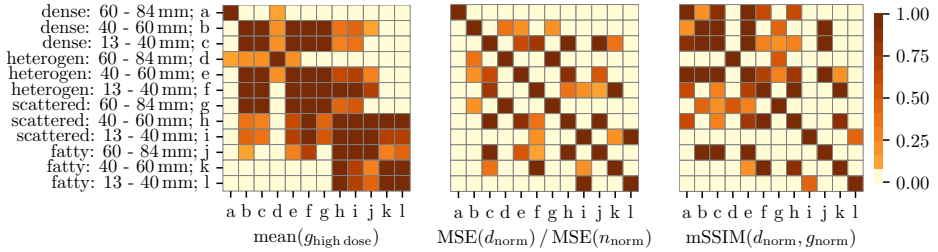
In conclusion, it can be inferred that the network does not exhibit any bias towards any breast group in the MBTST test set. This observation holds true even if different breast groups have a mean intensity distribution that deviates from that of the MBTST test set.

5.3.4 Denoising Example

Lastly, a denoising example is provided to demonstrate the denoising performance in a real-world scenario. The noisy image is displayed in Fig. 5.12a, where microcalcifications are challenging to discern in the actual DBT projection due to the obscuring noise. In contrast, these microcalcifications are clearly visible in the denoised image, as depicted in Fig. 5.12b. Additionally, Fig. 5.13 illustrates



(a)



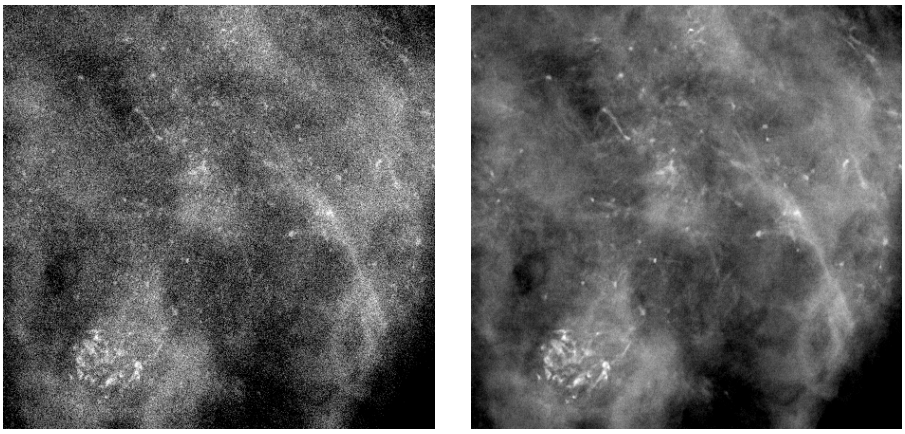
(b)

Figure 5.11: Measurements on the VICTRE dataset, whose group mean distribution follows that of the MBTST set, are depicted in this figure. (a) The distributions of the measurements across different groups are shown. (b) The statistical significance of the differences between the groups is tested. Squares are brightest when the p-value for the difference between two groups is below 0.05.

the application of the denoising network on DBT raw projections. From these projections, an SM is reconstructed and compared against an FFDM acquisition from the same breast. The microcalcifications in the reconstructed SM are even more discernible than in the FFDM. This example underscores the potential of the denoising network to be utilized as a component in DBT reconstruction.

5.4 Discussion

In Section 5.3.1, we showed that the proposed denoising network significantly outperforms analytical algorithms. In these traditional methods, noise was often not adequately reduced, or additional artifacts were introduced. These effects were not observed in the proposed network. These initial results suggested that the deep-learning approach may be a viable and effective method for denoising mammographic images.



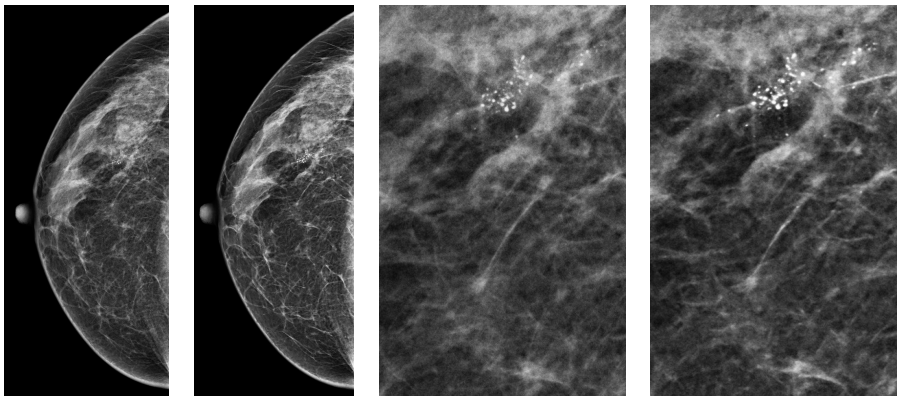
(a) Unprocessed DBT raw projection. (b) Denoised raw projection.

Figure 5.12: Comparison between an unprocessed and a denoised DBT raw projection.

In medical imaging, particularly mammography, preserving small image details is essential for accurate diagnosis. To address this need, we proposed a novel loss function called $\mathcal{L}_{\text{ReLU}}$, specifically designed to preserve small structures. In Section 5.3.2, we compared the performance of $\mathcal{L}_{\text{ReLU}}$ to other loss functions, evaluating its effectiveness in maintaining small structures such as microcalcifications. Our results demonstrated that the choice of loss function significantly influences the preservation of these structures. $\mathcal{L}_{\text{ReLU}}$ consistently outperformed other loss functions in this regard, either as a standalone loss or in combination with SSIM, depending on the experiment.

Besides preserving small structures, a denoising network must operate reliably to prevent fatal errors. It is essential that the network performs well across all breast types to avoid disadvantaging any patient group. Since different ethnicities have varying distributions of breast types, the network must ensure equitable performance for all [39]. For this reason a detailed investigation was conducted in Section 5.3.3.

We demonstrated, using the MBTST test set, that there was no bias towards different breast types in terms of mSSIM. Additionally, VICTRE I & II were employed to explore the impact of varying noise levels on denoising performance and to assess how well the network generalizes to data outside of the training distribution. Since the denoising performance between MBTST and VICTRE II is similar regarding mSSIM, but inferior in terms of nMSE, it can be inferred that the network performs equally well, regarding human perception, but the MBTST data might have more smooth structures, which contribute to the superior denoising performance on the MBTST data regarding the nMSE.



(a) FFDM (b) SM (c) Patch of FFDM (d) Patch of SM

Figure 5.13: Comparison of DM against SM with denoised raw projections.

Moreover, in the detailed investigation of Section 5.3.3 the network denoised the breast groups of VICTRE I & II differently, a trend that was not observed in the MBTST data. Given that VICTRE II maintains the same noise levels for each group as the MBTST dataset, it can be inferred that the variations in denoising performance are attributable to the differences between the VICTRE and MBTST data. This suggests that the VICTRE data possess different structures and features compared to the real-world MBTST data. This observation is further corroborated by visually examining Fig. 5.6 and Fig. 5.7.

Lastly, we demonstrated the denoising performance on a real-world case. We showed that a real DBT projection could be denoised and that an SM reconstructed from denoised projections displayed clearer microcalcifications than an FFDM. Since our previous investigations demonstrated reliability and preservation of small structures, we anticipate that the denoising network can be a valuable tool in clinical practice.

5.5 Future Work

Deep-learning-based denoising in medical imaging has demonstrated excellent results, as evidenced by the image quality achieved in our work. However, traditional metrics such as the SSIM do not provide information on whether these algorithms genuinely improve the diagnostic performance of radiologists, which is the ultimate aim of any denoising algorithm in medical imaging. Furthermore, comparisons between different methods are often challenging due to the lack of freely accessible test data.

Addressing these challenges necessitates the development of model observers, as discussed by [28], representing a crucial direction for future research. These

model observers mathematically simulate radiologists' perception to quantify whether modifications to the image appearance improve a radiologist's diagnostic performance. We believe it is necessary to further drive the development of model observers to create an open-source framework based on clinical data that can be employed to objectively evaluate new methodologies in denoising mammographic images across research groups.

5.6 Conclusion

In this chapter, we proposed a novel deep-learning-based denoising network for mammographic images. We introduced a novel loss function, $\mathcal{L}_{\text{ReLU}}$, designed to preserve small structures in images. Our results showed that $\mathcal{L}_{\text{ReLU}}$ significantly outperformed other loss functions in maintaining small structures, such as microcalcifications. Furthermore, we demonstrated that the network performed reliably across different breast types, showing no bias towards specific groups. The network also generalized well to data outside the training distribution. Lastly, we illustrated the denoising performance on a real-world case, showing that the network could denoise real DBT projections and improve microcalcification visibility in SM reconstructions.

6

Automatic X-Ray Style Adaption

Radiologists often exhibit diverse preferences when it comes to the visual representation of X-ray images. This diversity necessitates the manual tweaking of processing pipelines to cater to these individual preferences. In this chapter, we delve into the potential of automating the adjustment of an X-ray image processing pipeline to better align with the preferences of radiologists. Rather than resorting to deep learning methods for style modification, we maintain an interpretable and adjustable X-ray image processing pipeline even after optimization. To achieve this, we initially propose and investigate the application of the Local Laplacian Filter (LLF) [175] in X-ray image processing. Following this, we optimize the LLF using stochastic gradient descent to attain a specific style. To further enhance the style transfer capabilities of the LLF, we replace the remap function of the LLF with a MLP. This allows for a more nuanced and effective style transfer.

6.1 Related Work

X-ray signals captured by detectors encompass a broad spectrum of pixel values. These values need to be mapped into a visible range, and diagnostic features must be enhanced for better visibility. This task is typically addressed by X-ray image processing algorithms that weight different frequency bands of the image [232, 163, 56]. The fundamental assumption is that large image structures are represented by low frequency bands, while small image structures are represented by high frequency bands. By weighting these frequency bands differently, various structures can be enhanced or suppressed. Low frequency bands are usually attenuated, which scales down the overall range of pixel values and significantly contributes to mapping the desired signal into the visible range [182]. However, this core assumption has a fundamental flaw [136]. The decomposition of a signal into frequency bands breaks the signal down into its sine and cosine components. As a result, edges in an image, which are composed of an infinite

number of frequencies [181], are affected when manipulating frequency bands. This leads to the appearance of so-called halos around edges. Moreover, the human eye is highly sensitive to edges [239]. Beside decomposing an image into frequency bands, the wavelet transform has been proposed as an alternative to the Fourier transform [152]. Instead of decomposing a signal into its sinusoidal components, the wavelet aims to decompose a signal into signals which need fewer components to represent edges [10]. As a result, wavelets have been very successful in compressing image signals [229, 223]. However, when manipulating the wavelet coefficients, artifacts can reappear in the images [174].

Despite these issues, wavelets and frequency weighting remain state-of-the-art in X-ray image processing [139, 243, 162]. In photographic image processing, the bilateral filter was proposed as an alternative for manipulating image signals [67, 175]. It was later replaced by the LLF, developed by Paris et al. [174]. This filter operates on the assumption that the information in the signal lies in the direction of its gradients, and that manipulating the features of an image involves changing the amplitude of the gradients. It showed remarkably results on natural photographic image processing. Despite the rapid development in deep learning, the LLF remains a state-of-the-art method in image processing. Interestingly, Paris et al. [174] assume that "...halos may be tolerable in the context of medical imaging, e.g., [232, 56], [but] they are unacceptable in photography." and hence conclude that their proposed LLF might not be necessary for medical imaging. Radiologists have to distinguish the smallest structures in X-ray images to conduct diagnosis, such as breast cancer detection [99]. Furthermore, a misinterpretation or oversight of a severe disease can have fatal consequences [234]. Contrary to the assumption of [174], we posit that the LLF's unique ability to manipulate image features without halo effects could enhance the quality of X-ray images. In the initial segment of this work, we explore this potential of the LLF in the context of X-ray imagery, examining its efficacy in altering both minor and major structures without inducing halos.

However, as highlighted in Section 1.1.3, radiologists exhibit varying preferences for the appearance of X-ray images. Therefore, in addition to a method that can manipulate X-ray image features, we also require a technique that can automatically adjust the appearance of X-ray images to align with the preferences of radiologists. Most work focusing on the automatic adjustment of X-ray images has utilized deep learning methods such as (Cycle-)GANs [219, 91, 110]. However, these methods primarily aim to optimize the X-ray image impression for Neural Network training, to facilitate better generalization. Several factors hinder the application of deep learning methods for automatic adjustment of X-ray image styles for radiologists. Firstly, these methods necessitate a significant volume of training data. This implies that a new model, backed by a sufficient dataset

reflecting the desired style, must be trained for each radiologist, which is impractical. Secondly, deep learning methods suffer from a lack of interpretability and proving their reliability is a formidable task. This becomes a crucial concern in medical imaging, given the potentially fatal outcomes if the algorithm fails to present all essential diagnostic information. Lastly, even when data is available, the initial image assessment performed by the radiologist may not be optimal, necessitating manual adjustments. However, due to the numerous parameters involved in deep learning models, such manual adjustments are not feasible.

Given these considerations, we propose incorporating prior knowledge about the necessary image manipulations into the optimization process, thereby reducing the number of required parameters. To achieve this, we explore the LLF for the automatic adaptation of image impressions, where image manipulations can be interpreted by investigating its remap function.

We implement the LLF in a differentiable manner, which allows for the automatic optimization of its parameters using backpropagation [194] and stochastic gradient descent [187]. It enables the LLF to be optimized as a component of a more intricate pipeline. Furthermore, we suggest enhancing the LLF with a trainable normalization layer and improving the remap function to express more complex shapes. We demonstrate that it is feasible to optimize the LLF using fewer than 10 training images to achieve a desired image impression. Importantly, the LLF retains its interpretability and allows for manual parameter adjustments post-optimization, should a radiologist wish to alter the image impression. Moreover, we compare our proposed method against Aubry et al. [11]'s style transfer method, which aligns the gradient histograms of two images to transfer the style by manipulating the gradients with the LLF. We demonstrate that our method surpasses the style transfer method in achieving more accurate image impressions.

6.2 Methodology

In this section, we outline the methodology for the automatic alteration of X-ray image impressions using the LLF proposed by Paris et al. [174]. We explore the algorithm, its underlying intuition, and an efficient implementation of the LLF. Furthermore, we discuss the LLF style transfer method proposed by Aubry et al. [11]. Subsequently, we introduce an optimized differentiable implementation of the LLF, enhancing its functionality with a MLP remap function and a normalization layer.

6.2.1 Algorithmic Functionality

The LLF utilizes the Gaussian pyramid, denoted as $\mathcal{G}$, and the Laplacian pyramid, represented as $\mathcal{L}$, as discussed in Section 2.3. The function $\mathcal{G}(\mathrm{p}) \rightarrow \mathbb{R}$ is defined

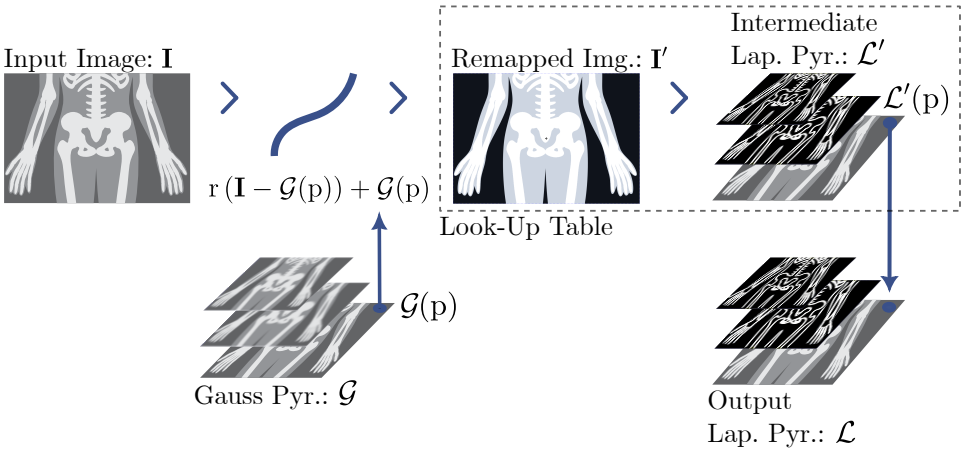


Figure 6.1: Description of the Local Laplacian Filter Algorithm.

such that for any pixel coordinate p within the dimensions of $\mathcal{G}$, $\mathcal{G}(\mathrm{p})$ returns the pixel value at p. Similarly, the function $\mathcal{L} \rightarrow \mathbb{R}$ is defined for any pixel coordinate p within the dimensions of the Laplacian pyramid $\mathcal{L}$, where $\mathcal{L}(\mathrm{p})$ gives the pixel value at p. For simplicity, we omit explicitly stating the level dimension of the pyramids. Hence, when referencing a pixel p, it can be a pixel at any level of the pyramid.

Fig. 6.1 illustrates the structure of the LLF. Initially, the input image $\mathbf{I} \in \mathbb{R}^{M \times N}$ is decomposed into a Gaussian pyramid $\mathcal{G}$. Subsequently, the LLF operates iteratively, cycling through all $\mathcal{G}(\mathrm{p})$ in the Gaussian pyramid $\mathcal{G}$. The first step in each iteration involves computing a remapped image $\mathbf{I}' \in \mathbb{R}^{M \times N}$. To achieve this, the pixel value $\mathcal{G}(\mathrm{p})$ in the Gaussian pyramid is subtracted from the input image $\mathbf{I}$, and the resulting difference $\mathbf{I}_d$ is remapped using a remap function $\mathrm{r}(\cdot) \in \mathbb{R}^{M \times N} \mapsto \mathbb{R}^{M \times N}$: $\mathbf{I}' = \mathrm{r}(\mathbf{I} - \mathcal{G}(\mathrm{p})) + \mathcal{G}(\mathrm{p})$. The remap function will be elaborated upon in the subsequent section 6.2.3. For each transformed image $\mathbf{I}'$, a new intermediate Laplacian pyramid $\mathcal{L}'$ is computed. Only one pixel value, specifically $\mathcal{L}'(\mathrm{p})$, which corresponds to the position of $\mathcal{G}(\mathrm{p})$, is utilized as a new value for the final output Laplacian pyramid $\mathcal{L}$.

Upon completion of the iteration over all $\mathcal{G}(\mathrm{p})$, all p in $\mathcal{L}$ are determined and $\mathcal{L}$ can be collapsed into a single output image $\mathbf{I}_{\text{out}}$.

6.2.2 Intuition behind the Local Laplacian Filter

To intuitively comprehend why the LLF alters edges without generating halos, it is necessary to first understand the underlying principle of the Gaussian pyramid $\mathcal{G}$. Each value $\mathcal{G}(\mathrm{p})$ represents a local average of its neighborhood in the original image. As the pyramid levels increase, the neighborhood represented by each

pixel naturally expands. When $\mathcal{G}(\mathrm{p})$ is subtracted from the input image $\mathbf{I}$, a difference image $\mathbf{I}_d$ is produced, which resembles a gradient image. An edge in the image would be accurately represented by a high value in $\mathbf{I}_d$. However, unlike conventional image gradients, edges with different steepness can still be represented by a single scalar value in $\mathbf{I}_d$. The values of these edges can then be manipulated by $\mathrm{r}(\cdot)$, and $\mathcal{G}(\mathrm{p})$ must be added back to the remapped differences to obtain $\mathbf{I}'$. Since $\mathcal{G}(\mathrm{p})$ represents the local average of the pixel within its neighborhood, remapping differences to this neighborhood is only significant for pixels within it. Consequently, $\mathbf{I}'$ is used to compute a new Laplacian pyramid $\mathcal{L}'$, with only $\mathcal{L}'(\mathrm{p})$ incorporated into the final Laplacian pyramid $\mathcal{L}$. Ultimately, a single value $\mathcal{L}(\mathrm{p})$ is computed for $\mathcal{G}(\mathrm{p})$.

6.2.3 Remap Function

The function $\mathrm{r}(\cdot)$ determines the manipulation of image features. While different remap functions could theoretically be used, the function proposed by [174] has shown remarkable results. It uses only three parameters and aligns with the intuition of the LLF. This function comprises two parts:

$$\mathrm{r}(\mathbf{I}_d) = \begin{cases} \mathrm{r}_d(\mathbf{I}_d) & \text{if } \mathbf{I}_d < \sigma_r \\ \mathrm{r}_e(\mathbf{I}_d) & \text{if } \mathbf{I}_d \geq \sigma_r \end{cases} \tag{6.1}$$

σ_r, the first parameter of r, defines the threshold for processing a pixel of $\mathbf{I}_d$ with either r_d or r_e. Given that $\mathbf{I}_d$ represents differences to the local neighborhood average, σ_r determines whether a difference indicates an image detail or a global image structure. Therefore, r_d remaps local image details, while r_e remaps global image structures.

r_d is defined as:

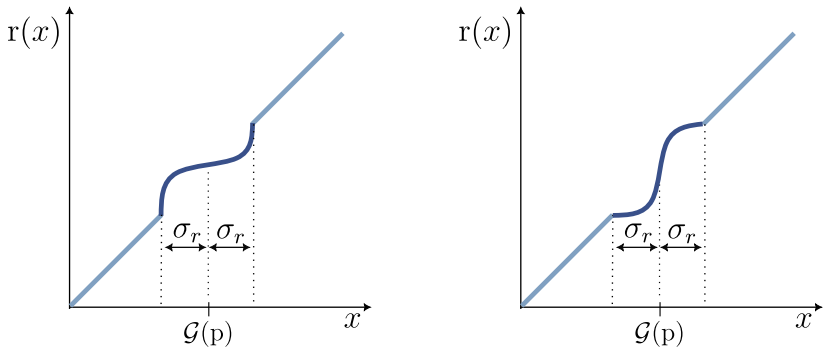
$$\mathrm{r}_d(\mathbf{I}_d) = \mathrm{sign}\big(\mathbf{I}_d(\mathrm{p})\big) \cdot \left(\frac{|\mathbf{I}_d(\mathrm{p})|}{\sigma_r}\right)^{\alpha} \quad \text{for all pixel values p in } \mathbf{I}_d \tag{6.2}$$

r_d operates on the absolute values of $\mathbf{I}_d$, which are normalized to a range between 0 and 1 by dividing them by σ_r. The sign function is employed to maintain the direction of the difference. The parameter α, with $\alpha > 0$, is the second adjustable parameter of r. If $\alpha < 1$, image details are enhanced as smaller differences are amplified. Conversely, if $\alpha > 1$, image details are smoothed as these differences are diminished. The influence of α on the remap function is depicted in Fig. 6.2b and Fig. 6.2a.

r_e is defined as:

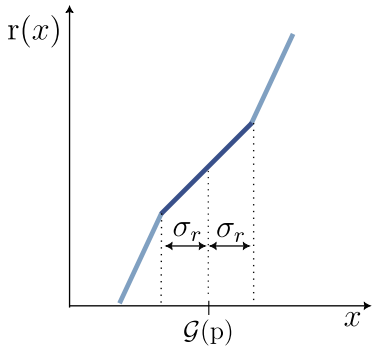
$$\mathrm{r}_e(\mathbf{I}_d) = \mathrm{sign}\big(\mathbf{I}_d(\mathrm{p})\big) \cdot \big(\beta \cdot |\mathbf{I}_d(\mathrm{p}) - \sigma_r| + \sigma_r\big) \tag{6.3}$$

The function r_e operates in a similar manner to r_d, acting on the absolute values of $\mathbf{I}_d$, which are centered around zero for scaling purposes. A sign function is utilized to preserve the direction of the difference. The third definable parameter of r, denoted as β, must be greater than 0. This parameter influences the slope of the remap functions, as illustrated in Fig. 6.2c and Fig. 6.2d. A larger β value results in a more pronounced remapping of the differences. Conversely, a β value less than 1 yields a less steep function, leading to a more subtle remapping of the differences. The function r_e is applied exclusively to differences that exceed the threshold σ_r, thereby serving to remap global changes in the image.

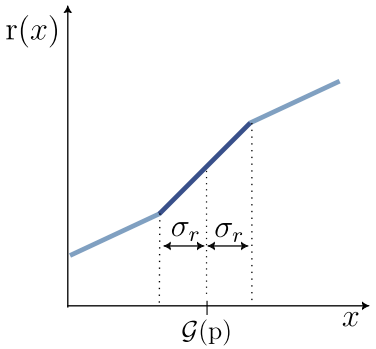


(a) r(x) with $\alpha > 1$

(b) r(x) with $\alpha < 1$



(c) r(x) with $\beta > 1$



(d) r(x) with $\beta < 1$

Figure 6.2: This figure illustrates the influence of the two parameters, α and β, on the remapping function, r. Both parts of r, namely r_d and r_e, are shown in dark and light blue, respectively.

6.2.4 Efficient Implementation

The original LLF, as introduced by [174], mandates the computation of a remapped image $\mathbf{I}'$ for each pixel in the image, along with a new intermediate Laplacian pyramid $\mathcal{L}'$ for each $\mathbf{I}'$. Paris et al. [174] proposed that $\mathcal{L}'$ could be computed on sub-images centered around p, thereby reducing the LLF complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N\log N)$, where N is the total pixel count in $\mathcal{G}$. To

further optimize, [11] suggested using a Look-Up Table (LUT) to precompute the remapped images $\mathbf{I}'$ and the corresponding intermediate Laplacian pyramids $\mathcal{L}'$ (as shown in Fig 6.1). Instead of calculating $\mathbf{I}'$ and $\mathcal{L}'$ for each pixel in $\mathcal{G}$, they are computed for each possible pixel value in $\mathcal{G}$, not exceeding the pixel resolution R, typically less than 256. Thus, $R \ll N$. At the algorithm's onset, the LUT is generated. During the LLF's iterative process, only the precomputed $\mathcal{L}'$ suitable for $\mathcal{G}(\mathrm{p})$ is selected. The algorithm's efficiency can be further boosted by precomputing a LUT with fewer entries than R. This involves selecting the two closest entries in the LUT and linearly interpolating the output coefficient from the precomputed pyramids. This procedure reduces the computational cost to $\mathcal{O}(N)$ [11].

6.2.5 Style Matching with Gradient Histogram Transformation

Aubry et al. [11] propose a style matching method based on the premise that an image's style is primarily characterized by its gradients. They extend the assumption from Paris et al. [174] that the LLF can effectively manipulate image gradients. Their method involves aligning the gradient histograms of two images, a model image $\mathbf{M}$ and a target image $\mathbf{I}$, by applying a remapping function to the LLF. To derive this remapping function the gradient amplitudes $|\nabla\mathbf{I}|$ and $|\nabla\mathbf{M}|$ are computed for both images. They then derive the histogram transfer function, which is used as the remap function for the LLF and is defined as follows:

$$\mathrm{r}(x) = \mathrm{CDF}^{-1}_{|\nabla\mathbf{M}|}\left(\mathrm{CDF}_{|\nabla\mathbf{I}|}(x)\right) \tag{6.4}$$

In the equation above, x denotes the value that requires transformation. Applying the Cumulative Distribution Function (CDF) of values to themselves results in a uniform Probability Density Function (PDF) of the output values [75]. Hence,

$$c = \mathrm{PDF}\left(\mathrm{CDF}_{|\nabla\mathbf{I}|}(|\nabla\mathbf{I}|)\right) \tag{6.5}$$

The application of the inverse CDF to uniformly distributed values yields a distribution that is used to generate the inverse CDF. Consequently, equation 6.4 produces a gradient distribution that closely mirrors the gradient distribution of the model image $\mathbf{M}$. When equation 6.4 is used as a remap function on the LLF, it implies the application of the remap function on $\mathbf{I}_d$. Given that $\mathbf{I}_d$ is closely associated with the image gradient, Aubry et al. [11] predict and demonstrate that the image gradient distribution between $\mathbf{M}$ and $\mathbf{I}$ will converge when the LLF is applied to $\mathbf{I}$.

We implement this method on X-ray images, aiming to transform one X-ray style into another. Furthermore, we strive to convert raw projections into processed X-ray images by identifying the appropriate remap functions based on the gradient distributions.

6.2.6 Differentiability and Prallelization

Contrary to the style transfer method outlined in Section 6.2.5, we suggest an alternative approach for the automatic adjustment of the remap function of the LLF. We implemente the LLF differentiable, with the goal of optimizing the parameters of the remap function using stochastic gradient descent [187] and backpropagation [194].

We employ the PyTorch framework [176] to implement the LLF. PyTorch's functions inherently support automatic differentiation and facilitate parallel execution on GPUs. Hence, no additional effort is required to compute the gradients of the LLF. The pyramids in the LLF possess varying dimensions for each level. Consequently, PyTorch does not allow for parallel computation of $\mathcal{G}(\mathrm{p})$, necessitating the time-consuming LLF iteration over $\mathcal{G}(\mathrm{p})$. To address this, we map both pyramids $\mathcal{G}$ and $\mathcal{L}'$ to 1D tensors:

$$\mathbb{R}^{\mathcal{D}(\mathcal{G})} \rightarrow \mathbb{R}^{|\mathcal{G}|}$$

$$\mathbb{R}^{\mathcal{D}(\mathcal{L}')} \rightarrow \mathbb{R}^{|\mathcal{L}'|}$$

Here, $\mathcal{D}$ refers to the dimensions of the respective pyramids and $|\cdot|$ denotes the set size, i.e., the total number of pixel values in the pyramids. As a result, all operations can be executed in parallel on the 1D tensors, eliminating the need for iteration over the pixel $\mathcal{G}(\mathrm{p})$ values. Thus, utilizing the full potential of the prallel computation capabilities of GPUs.

Compared to the approach proposed by [11], our method enables the application of the LLF as a single element within a more complex pipeline. It enhances the capabilities of the LLF with additional components, while still allowing for automatic optimization of the remap function. If other components also have adjustable parameters, they can be optimized concurrently during the same optimization process.

6.2.7 Normalization Layer

The LLF transforms the relationship between pixel values. As a consequence its capacity to map the total pixel range of an image to a desired range is limited. To address this, we propose the addition of a trainable normalization layer to the LLF. It can be defined as:

$$\mathbf{I}_{\mathrm{norm}} = \mathbf{I} \cdot \gamma + \omega \,, \tag{6.6}$$

where γ and ω represent the trainable parameters of the layer.

6.2.8 Enhanced Remap Function

The remap function proposed by [174] has produced remarkable results. With its three parameters, it is simple to adjust and to interpret. However, these three

parameters also limit the potential shapes of the remap functions. For this reason, we propose an enhanced remap function that is particularly suitable for automatic adjustments. This remap function operates on $\mathbf{I}_d$ and processes each pixel value individually. The revised remap function should exhibit maximum flexibility and be suitable for automatic optimization. To this end, we propose a MLP [147] with a single scalar input and output value. Neural networks, including MLPs, are recognized as universal function approximators [205] and are inherently designed for optimization with stochastic gradient descent and backpropagation. The MLP comprises six linear layers, each followed by a ReLU activation function [161] and batch normalization [98], except for the final linear layer. The number of neurons in the layers are as follows: 3, 12, 24, 24, 12, and 3, resulting in a total of 709 parameters. Despite the MLP having significantly more parameters than the original remap function, it remains interpretable as it still only has one scalar as input and output. Consequently, it is possible to sweep through the entire input space and observe the complete behavior of the remap function. Moreover, verifying the remap function for monotonicity ensures that no image information gets lost in the remapping process [174].

6.2.9 Datasets

To assess the performance of the LLF on X-ray images, we utilize the MBTST dataset as outlined in Section 2.2.3. This dataset includes images from over 7325 patients. However, for the optimization of the LLF, we limit ourselves to a subset of 145 images, with 130 images used for testing and 15 images reserved for optimization. These images are preprocessed using a closed-source vendor pipeline to generate corresponding pairs of unprocessed raw projections and processed mammograms with a clinically relevant image impression.

6.2.10 Optimization

Our goal is to optimize the LLF for transforming raw projections into processed mammograms, emulating the closed-source vendor pipeline. We use matching pairs of processed and raw mammograms to optimize the remap function parameters of the LLF. The raw projections are input images, and their processed counterparts are target images for comparison with the LLF output, as shown in Fig. 6.3. The LLF is optimized with and without the normalizaiton layer attached and with the original remap function as well as the MLP remap function. The LLF is optimized both with and without the attached normalization layer, and with both the original remap function and the MLP remap function. All optimization processes are conducted using the Adam optimizer [115] with a learning rate

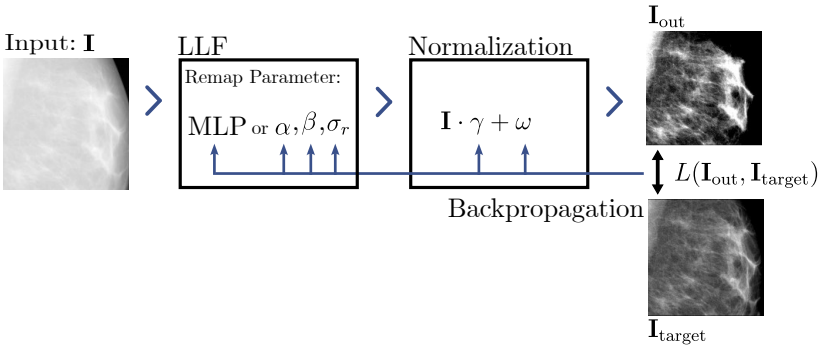


Figure 6.3: This figure depicts the optimization process of the LLF and the Normalization Layer using backpropagation, along with the matching pairs for the loss function.

of 0.0001, and the LLF is trained for 300 epochs. As a loss function, we use a combination of MSE and MSSIM [239]:

$$L(\mathbf{I}_{\text{out}}, \mathbf{I}_{\text{target}}) = \text{MSE}(\mathbf{I}_{\text{out}}, \mathbf{I}_{\text{target}}) + 1 - \text{MSSIM}(\mathbf{I}_{\text{out}}, \mathbf{I}_{\text{target}}) \tag{6.7}$$

We also found that training with the MLP remap function converges faster if the MLP is preinitialized to depict an identity function.

6.3 Experiments & Results

In this section, we evaluate the LLF on X-ray images and explore the optimization of the LLF to achieve a specific image impression. We compare the LLF with the original remap function and the enhanced MLP remap function. Additionally, we compare the LLF with the style transfer method proposed by Aubry et al. [11]. Finally, we assess the processing and training times of the LLF.

6.3.1 LLF application on X-Ray images

Before investigating the optimization of the LLF, we first evaluate whether it is possible to apply the LLF to manipulate the impressions of X-ray images. In this experiment, we do not investigate the LLF's ability to substitute an entire pipeline but rather its potential use as part of an X-ray image processing chain. Hence, we apply the LLF to a preprocessed mammogram and investigate the influence of different parameter configurations on the image impression. The results are depicted in Fig. 6.4. Furthermore, we compare the LLF manipulated images with plain tone mapping. Tone mapping, akin to the LLF, applies a remap function $r(\cdot)$ to pixel values, but without the use of image pyramids, represented as $r(\mathbf{I})$. Therefore, it offers insights into the advantages of the additional complexity inherent in the LLF. Processing the pixel values with tone mapping to reduce the

amount of large (i.e., bright) pixel values reduces the bright areas. However, as a side effect, small details in these bright areas are also reduced, since they get mapped into a smaller range. In contrast, this is an effect that the LLF avoids. It is possible to manipulate large bright structures in the mammogram to enhance or suppress them, while simultaneously ensuring that small details in these areas are not affected. This manipulation can be achieved by changing the parameter β, as demonstrated in Fig. 6.4c and Fig. 6.4d. Adjusting the parameter α allows for the enhancement or suppression of small details, as shown in Fig. 6.4e and Fig. 6.4f. This effect can be observed in the appearance change of the small microvessels in the image. Most importantly, manipulating the mammogram with the LLF does not produce side effects such as halos or the removal of details from the image.

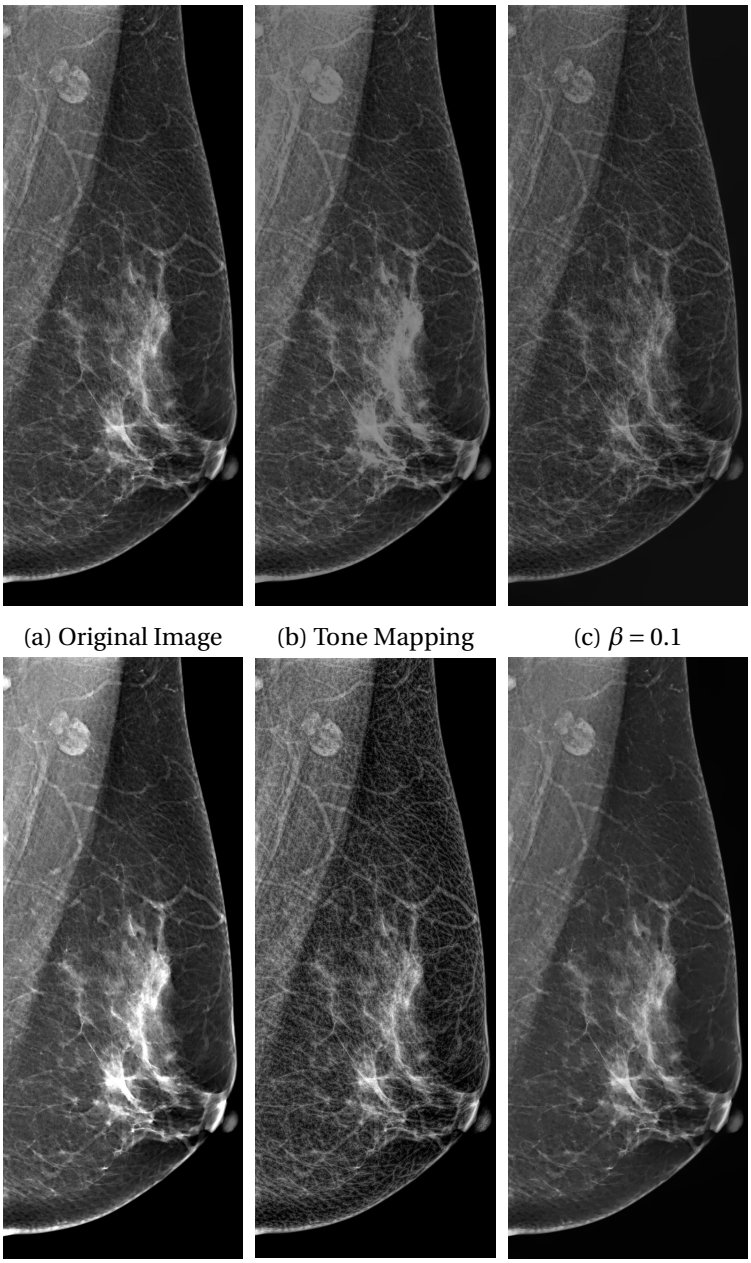
6.3.2 Automatic Optimization to Match Image Impressions

In this experiment, we examine the LLF's capability to convert a raw projection into a mammogram that provides a clinically relevant image impression. This is accomplished by automatically optimizing the remap function, as suggested in Section 6.2.6. We optimize both the original remap of [174] and the enhanced MLP remap function. Each of these approaches are optimized twice: once with the normalization layer described in Section 6.2.7 and once without it. Our optimization method is compared with the style transfer technique proposed by Aubry et al. [11] and detailed in Section 6.2.5. Altogether, we present five different implementations:

1. Original Remapping Function optimized w/o Norm. (Orig-RM)
2. Original Remapping Function optimized with Norm. (OrigNorm-RM)
3. MLP Remapping Function optimized w/o Norm. (MLP-RM)
4. MLP Remapping Function optimized with Norm. (MLPNorm-RM)
5. Remap Function generated with Gradient Matching (∇-Matching)

Empirical Visual Evaluation

Fig. 6.5 presents an image from the test set, processed using the LLF and its various remap functions. The raw input image differs substantially from the target image, with structures being challenging to discern due to the concentration of crucial information within a narrow pixel value range. The image processed with ∇-Matching, as shown in Fig. 6.5e, exhibits a marked improvement over the input image. However, the dense tissue appears brighter than in the target image. In this particular example, MLP-RM yields the best results. While the breast edges are less visible than in the target image and the non-dense tissue areas appear slightly darker, the overall image impression closely resembles the target image. The MLPNorm-RM performs slightly less effectively compared



(a) Original Image (b) Tone Mapping (c) $\beta = 0.1$

(d) $\beta = 2.0$ (e) $\alpha = 0.1$ (f) $\alpha = 2.0$

Figure 6.4: Example of enhancing or suppressing features in a mammogram using LLF, compared to the method of simple tone mapping.

to MLP-RM, as does OrigNorm-RM. In both corresponding output images, the contrast between non-dense and dense tissues is less pronounced than in the

target image. This effect is particularly noticeable in the output image generated with OrigNorm-RM.

Quantitative Evaluation

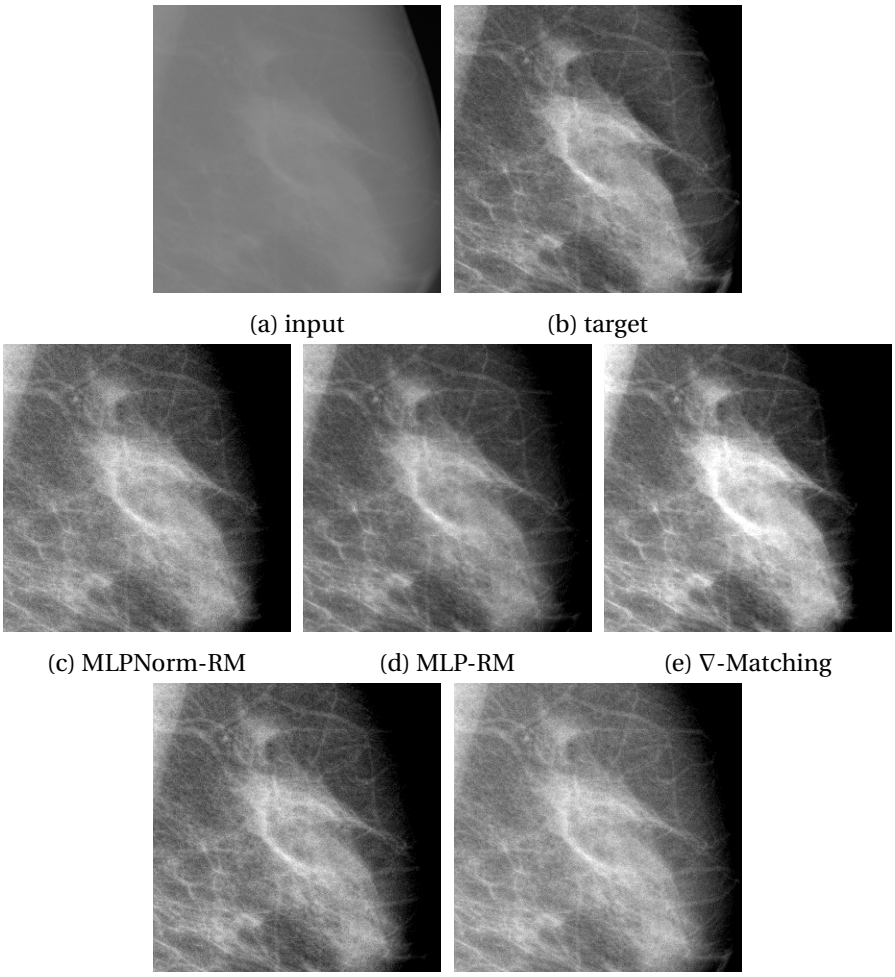
The findings obtained from this individual image are largely supported by the measured results across the entire test set, as detailed in Table 6.1. In contrast to the example image, MLPNorm-RM demonstrates the best performance in terms of the SSIM measurement when applied to the complete test set. The other trends observed align with those from the example image. Additionally, the evaluation using the MSE reveals that the output images processed with ∇-Matching deviate more from their targets than the input image does from the target. This divergence can be ascribed to the overly bright dense tissues, which considerably influence the MSE calculation. Nevertheless, it still succeeds in producing images that are more akin to the target images than the input images, as per the SSIM metric.

	input	MLP-RM	MLPNorm-RM	Orig-RM	OrigNorm-RM	∇-Matching
SSIM	0.587	0.9426	**0.9441**	0.9190	0.9107	0.8174
MSE	0.0270	**0.0064**	0.0066	0.0264	0.0105	0.0738

Table 6.1: Different LLF optimizations are evaluated based on SSIM and MSE measurements between the target mammogram and the LLF output. The comparison also includes the input image against the target mammogram.

Evaluation of Remap Functions

Fig. 6.6 depicts the optimized remap functions. The two MLP remap functions are displayed in Fig. 6.6a, while ∇-Matching, Orig-RM, and OrigNorm-RM are depicted in Fig. 6.6b. All four remap functions demonstrate similar behavior when the input is close to zero. Orig-RM and OrigNorm-RM are identical, indicating that the normalization layer did not affect the parameters of the remap functions. The overall shape of these remap functions is akin to the MLP remap functions. However, the MLP remap functions display a more complex behavior and possess a steeper slope. Both the steeper slope and the more complex behavior cannot be realized with the original remap function. The shape of MLP-RM and MLPNorm-RM is similar; however, without the normalization layer, the remap functions have a steeper slope, most likely compensating for the missing normalization. The ∇-Matching is particularly distinctive as a remap function, forming an 'S' shape. Since it is applied on absolute values, it maintains point symmetry around zero. Similar to MLP-RM, it extends to approximately -1.5 and 1.5. The different shape of ∇-Matching suggests that generating matching gradients is not the same

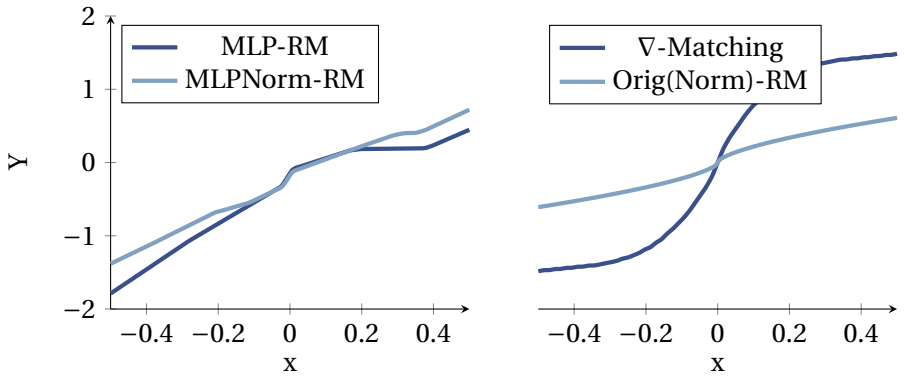


(a) input (b) target

(c) MLPNorm-RM (d) MLP-RM (e) ∇-Matching

(f) OrigNorm-RM (g) Orig-RM

Figure 6.5: Images mapped using optimized remap functions through LLF. The display includes gradient matching, MLP with and without normalization, and the original remap functions, both with and without normalization layer.



(a) MLP Remap Functions

(b) Original and Gradient Matching Remap Functions

Figure 6.6: Optimized remap functions of LLF for transforming raw projections into target mammograms are depicted. These include the MLP remap functions of MLP-RM and MLPNorm-RM, the original remap function of Orig-RM and OrigNorm-RM, and the remap function of ∇-Matching.

objective as optimizing the LLF using our proposed loss function and stochastic gradient descent, and may miss some information about the image impression.

Required Number of Training Images

The LLF optimization process requires only a limited number of training data. In this experiment, we investigate the influence of the number of training images on the performance of the LLF. Again, we measure the MSSIM and MSE between the target mammograms and the outputs of the LLF on the test data. Fig. 6.7 illustrates the results. We optimize with a number of training images ranging from one to 15. The LLF is optimized with the corresponding number of training images only where a marker is visible in Fig 6.7. Optimizing the LLF with only one training image yields the worst results, indicating that the LLF overfits on that single image. However, already with two training images, the performance of the LLF improves. The MSSIM fluctuates for image numbers between two and 15. We assume that this is due to statistical fluctuations of the optimization process and it is difficult to draw a conclusion from the MSSIM values about which number of training images, ranging from two to 15, is best. However, the MSE exhibits a more steady trend and is best for image numbers between two and five. Hence, we conclude that two to five training images are sufficient to optimize the LLF for a specific image impression.

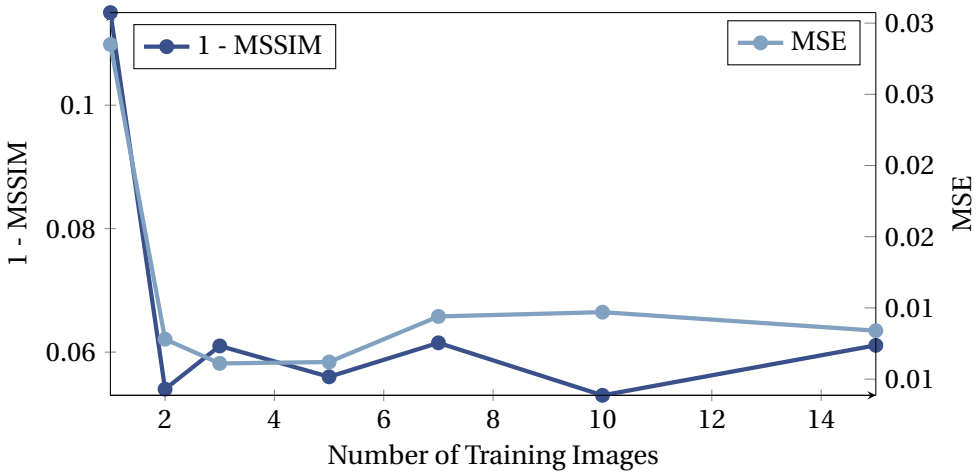


Figure 6.7: LLF algorithm performance as the number of training data points varies. The metrics MSSIM and MSE are depicted for each data point.

Processing Time

Applying the LLF to process X-ray images necessitates a reasonable execution time, ideally allowing for real-time processing. Moreover, adapting an image processing pipeline to meet a radiologist's needs should also be accomplished within a reasonable timeframe. Consequently, we implemented the LLF as proposed by [11] and detailed in Section 6.2.4 which suggests a computational complexity of $\mathcal{O}(N)$. Additionally, we facilitated parallel computation of the LLF on GPUs as outlined in Section 6.2.6.

To assess the effectiveness of this implementation, we examine the execution and training time of the LLF. We explore the impact of various parameters, including image size, the number of pyramid levels, and the pixel resolution of the LUT, on the execution time. The results are presented in Fig. 6.8. In the plot, we denote the image resolution as the length of one side of a square image. Consequently, the number of pixels N is the square of the resolution. However, the execution time of the LLF is almost linear to the image resolution. This is due to the parallel implementation of the LLF on the GPU. An image with a resolution of 512×512 pixels takes 61ms to execute, allowing for real-time processing at this image size. Decreasing the number of pyramid levels does not significantly reduce the execution time. This is because the initial levels in the pyramids are the most computationally intensive, and eliminating the lower levels does not significantly affect the execution time. Reducing the pixel resolution of the LUT correlates linearly with the execution time. Therefore, decreasing the resolution from 256 to 120 reduces the time from 61ms to 29ms. This reduction can be leveraged to accelerate the execution time of the LLF.

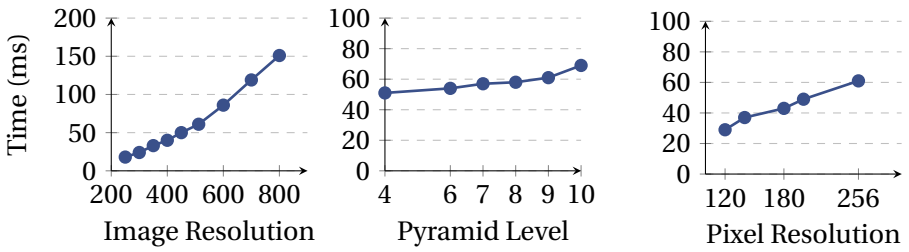


Figure 6.8: Execution times of the LLF under varying parameters: image resolution, pyramid level number, and pixel value resolution.

6.4 Discussion

The LLF can be employed to manipulate features in X-ray images, akin to its application on photographic images, as illustrated in Section 6.3.1. This establishes it as a feasible component within an X-ray pipeline. In Section 6.3.2, we further demonstrated that the LLF can be automatically optimized to align with the image impression of a vendor pipeline through the processing of raw projections. Moreover, substituting the remap function with the MLP improved the adapted image impression. Optimizing the LLF through backpropagation facilitates its refinement as a segment of a larger pipeline. Consequently, we incorporated a trainable normalization layer, which notably improved the LLF's performance when used with the original remap function, leading to satisfactory outcomes. This approach retains the advantage of using the original remap function, which, due to its low number of trainable parameters, can be manually adjusted if required. When the LLF was further enhanced with the MLP remap function, its performance improved even more, normalizing intrinsically and eliminating the need for an additional normalization layer.

The proposed method demonstrates superior performance over ∇-Matching, likely due to the fact that style is not solely encoded in gradients. For medical imaging, especially when considering soft tissue and lesion contrast, the edges are often diffuse and blend into the background texture rather than being sharply defined. Therefore, purely gradient histogram matching cannot capture the full complexity of a medical image.

Our optimization approach fully utilizes the capabilities of the LLF, optimizing it to manipulate image impression across all features that the LLF can modify. This includes edges, fine-grained structures, and large image structures. This superiority is further reflected in the remap functions, where all functions optimized with our method display similar shapes, in contrast to the different remap function of ∇-Matching.

The LLF can be optimized using a limited number of training images, making it a practical solution for real-world applications where medical image data is scarce. This is particularly useful when the image data must align with the specific image impression of an individual radiologist.

Furthermore, the LLF can operate in real-time. Any increase in image resolution or pyramid depth contributes at most linearly to the execution time. Leveraging parallel computing on GPUs, an image can be processed in under 100ms.

6.5 Future Work

In our parallel computation of the LLF, the number of pixel resolution values determines the number of precomputed Gaussian Pyramids that must be stored simultaneously in the GPU memory. This limits the maximum resolution of the LUT or the maximum image size. To overcome this limitation, a more efficient implementation of the LLF could be developed, which precalculates the LUT only for a subset of pixel values. The LLF can then be computed iteratively for different pixel ranges, enabling a trade-off between computation time and memory usage. Additionally, instead of using PyTorch for implementation, the LLF could be implemented in C++ and CUDA to further optimize the execution time. In our work, we examined the original remap function and a MLP, representing two extremes of the spectrum of remap functions in terms of parameter number. Future work could explore remap functions with a moderate number of parameters, such as a spline function, to determine the optimal balance between complexity and the allowance for manual adjustments. Moreover, we demonstrated the feasibility of optimizing the LLF as a standalone algorithm and as part of a larger pipeline by adding a normalization layer. However, this concept can be extended to far more components, enabling the creation of a fully sophisticated pipeline that can be optimized using backpropagation. We optimized the LLF using matching pairs. However, matching pairs are often unavailable in medical imaging, especially when a set of images must match the image impression required by an individual radiologist. Most of the time, only already processed images from devices the radiologist is familiar with are available. Therefore, a metric that can measure the similarity of two images from different acquisitions, i.e., with different content would allow for the optimization of the LLF without matching pairs. The next chapter will introduce such a metric.

6.6 Conclusion

In this chapter, we demonstrated the feasibility of applying the LLF to X-ray images. We optimized the LLF to match the image impression of a vendor pipeline by processing raw projections. We showed that the LLF can be optimized with

backpropagation to match a certain image impression and that it can be integrated into a larger pipeline. Simultaneously, the LLF remains interpretable and can be manually adjusted if necessary. Moreover, the LLF can be executed in real-time with parallel computation on GPUs.

7

A trainable metric to quantify style differences.

We previously discussed the need for a metric to quantify style differences on non-matching pairs in Chapter 6, as the lack of such a metric limited our ability to optimize the LLF only on matching pairs. Besides this optimization, such a metric is also beneficial to experts who adjust X-ray image styles for radiologists, as it allows quantification of the differences between styles, thereby objectifying the typically very subjective process of style selection. In this section, we delve into the development of the Style Metric for X-ray Images (StyleX), that quantifies style differences between X-ray images of non-matching pairs.

7.1 Related Work

To our knowledge, no existing research in the field of medical imaging has specifically focused on developing a style metric for non-matching pairs. However, numerous studies have examined the generalization of neural networks to inter-modality and intra-modality appearance differences, implicitly or explicitly addressing style differences [261, 140]. Therefore, their work has some resemblance to ours.

In addressing the generalization of neural networks, two primary strategies are typically employed. The first, domain translation, aims to convert images from one domain to another [38, 248], such as from CT to MRI [172] or CT to X-Ray images [222, 77], with the goal of transferring the image to the domain on which a neural network has been trained. The second strategy focuses on training networks on domain-invariant features. This can be achieved either by harmonizing the domain of the training data [80, 20, 263] or by augmenting various domain appearances during training. This augmentation compels the network to disregard domain-specific features [137, 247].

Three methodologies are predominantly employed to address style differences: Generative Adversarial Networks (GANs) [118, 219, 110], Diffusion models [114, 171, 188], and Autoencoder disentanglement approaches [121, 129, 95]. Each of these methodologies may yield a style loss or style representations as a byproduct of their training process. Given our work's focus on the development of a style metric based on style representations, we will explore these methodologies in relation to their style representations and losses.

During the training of GANs [262, 106, 100], the generator is trained to produce images that resemble the target domain. In the context of style transfer, the target domain represents the desired style. Concurrently, the discriminator is trained to differentiate between real and generated images. Thus, the discriminator's loss can be construed as a style loss, given that it is fundamentally trained to recognize the style of the target domain. However, it is worth noting that the discriminator also learns to recognize unrealistic artefacts generated at pixel level, as it is trained to discern between real and generated images. Furthermore, the discriminator's capability is limited to distinguishing between a discrete set of styles, typically confined to just the input and target domain. Consequently, its application as a standalone style loss is limited.

In an effort to enhance the training of GANs on medical image style transfer, Armanious et al. [9] and Hémon et al. [89] have integrated the well-known style loss from Gatys et al. [74] and Johnson et al. [104] into their training process. The style loss is constructed by extracting features from a pre-trained VGG network [201]. These features are then permuted via a Gram matrix [85], resulting in the loss of their locality. This approach allows the features to effectively represent artistic elements such as brush strokes, textures, and colors. Implementing it as a fixed loss that remains unoptimized during the GAN training process might enhance the overall training. However, when used as a standalone style loss without the discriminator, merely eliminating the locality of features in X-ray images is insufficient to capture the style differences between these images.

Zhang et al. [255] enhanced their GAN training by introducing a handcrafted style loss, premised on the assumption that style information resides in the high-frequency domain, to generate different x-ray image styles. While high-frequency information does contain some style information - for instance, contrast enhancement results in more dominant high frequencies - it is not solely confined to these frequencies. Style information also encompasses the distribution of pixel intensities, texture, and the weighting of low frequency background structures. As such, a style loss based solely on high-frequency information is insufficient to capture the full scope of style information. Additionally, it is worth noting that high frequencies also encapsulate content information, which is not a desired attribute in a style loss.

Zhao et al. [259] developed a specific loss for their diffusion model by training a network to differentiate between two distinct domains. This led to the construction of a deep-learning-based domain loss. However, it should be noted that their focus was on discrete non-medical image domains, specifically differentiating between categories like 'cats and dogs' or 'female and male'. Yet, the idea of pretraining a network on different domains and then using this pretrained network to classify between the domains is a promising approach. However, this method falls short in providing a quantifiable measure of style distance, which is essential for a complete style loss.

The approach of incorporating two autoencoders into the GAN training process, as applied to medical images by [233, 247, 80], was initially proposed by [121, 129, 95]. In this method, one autoencoder's latent space encapsulates content information, while the other represents style. The decoder combines the embeddings from the latent spaces of both autoencoders, aiming to reconstruct an image that reflects the style of the style embedding and the content of the content embedding. A discriminator is subsequently trained to assess the accuracy of the content and style in the reconstructed image. This approach necessitates an additional decoder and a discriminator, which evaluates the accuracy of the reconstructed images on a pixel-level basis and it does not directly optimize the accuracy of style representations in the latent space. Despite this, the concept of generating style representations in the latent space, independent of the image's content, will be incorporated into StyleX.

Unlike the methods proposed in previous studies, our research uniquely develops a style metric capable of quantifying style differences of non-matching pairs. Our method does not rely on a decoder for embedding reconstruction, a pixel-wise loss, a discriminator, or handcrafted style features. Furthermore, we specifically focus on exploring and refining the capacity of our proposed style loss to accurately distinguish between all styles.

7.2 Methodology

Our objective is to devise a style metric that enables quantifiable comparison between the styles of images of non-matching pairs. The construction of this metric is twofold. Firstly, we obtain a multi-dimensional vector $\mathbf{r}$ representing the style of an image $\mathbf{I} \in \mathbb{R}^{M \times N}$. This is achieved using an encoder network $\mathbf{e}_\Theta(\cdot) \in \mathbb{R}^{M \times N} \mapsto \mathbb{R}^D$, with M, N being the image dimensions, D the embedding dimension, and Θ the weights of the encoder. Secondly, we apply a distance metric $d(\cdot)$, capable of quantifying the distance between two vectors. This mea-

surement reflects the stylistic differences between the images. Consequently, the style metric, StyleX, can be formulated as follows:

$$\text{StyleX}(\mathbf{I_1}, \mathbf{I_2}) = d(\mathbf{r}_1, \mathbf{r}_2) = d\left(\mathbf{e}_\Theta\left(\mathbf{I_1}\right), \mathbf{e}_\Theta\left(\mathbf{I_2}\right)\right) \tag{7.1}$$

7.2.1 Essential Prerequisites for Training a Style Encoder

To develop a meaningful style metric, it is necessary to train an encoder $\mathbf{e}_\Theta$. This encoder should generate style representations r from the input image, which are disentangled from the images' content.

Training the encoder, however, cannot rely on parameters from imaging pipelines that compute stylized X-ray images for information about style differences. This is due to the fact that these parameters are either non-comparable when styles from different pipelines are used, or they are undisclosed vendor secrets, effectively rendering them as black boxes. Additionally, non-matching content complicates pixel distance computation, such as Euclidean distance. Thus, while the training data provides the style class c, it does not offer information about the distance between styles, a critical element for deriving a meaningful distance metric.

Consequently, supervised methods cannot be used to train $\mathbf{e}_\Theta$. An unsupervised method is required, capable of generating style representations whose distances reflect the stylistic differences. This excludes unsupervised methods that rely on maximizing distances between negative pairs, such as Siamese learning [156, 23], or contrastive learning [45, 87]. These methods artificially enlarge distances, even when some styles might be close to each other.

7.2.2 Limitations of Siamese Training

In Siamese training, $\mathbf{e}_\Theta$ is optimized by presenting image pairs, which are either matching or non-matching. The encoded embeddings, $\mathbf{r}_1$ and $\mathbf{r}_2$, are then compared using a loss function. If the images are matching (i.e., they belong to the same class, denoted by the label $c = 0$), the embeddings $\mathbf{r}$ should be similar, and the distance between them, $d(\mathbf{r}_1, \mathbf{r}_2)$, should be minimized. If the images are non-matching (i.e., they belong to different classes, $c = 1$), $d(\mathbf{r}_1, \mathbf{r}_2)$ should be maximized. However, to stabilize the training process, a margin m is introduced, which defines the maximum distance between the embeddings of non-matching pairs. This results in the following total loss function, which must be minimized in the training process [156]:

$$L_{\text{siam}} = (1 - c) \cdot d(\mathbf{r}_1, \mathbf{r}_2) + c \cdot \max(0, m - d(\mathbf{r}_1, \mathbf{r}_2)) \tag{7.2}$$

The function $\max(\cdot, \cdot)$ returns the maximum of the two arguments. The second term in the loss function prevents the use of Siamese training for the training of

$\mathbf{e}_\Theta$. The distance maximization between non-matching pairs is problematic for two reasons. Firstly, a fixed margin m must be defined beforehand; it sets the maximum distance between two non-matching styles. This margin is arbitrary and challenging to define accurately, as the maximum distance between different styles is not known beforehand. Secondly, the loss function consistently attempts to maximize the distance between two styles, irrespective of their actual proximity. Some styles are very similar, and enforcing a large distance would not effectively reflect their differences.

7.2.3 Employing SimSiam as a Style Encoder Training Method

To surmount the discussed barriers, we propose employing the Simple Siamese (SimSiam) approach from Chen et al. [46], a method originally developed to pre-train neural networks in an unsupervised manner to learn important image features for downstream tasks like segmentation or classification. In contrast to classical Siamse learning [156, 23], SimSiam relies solely on positive pairs, i.e., images sharing the same style.

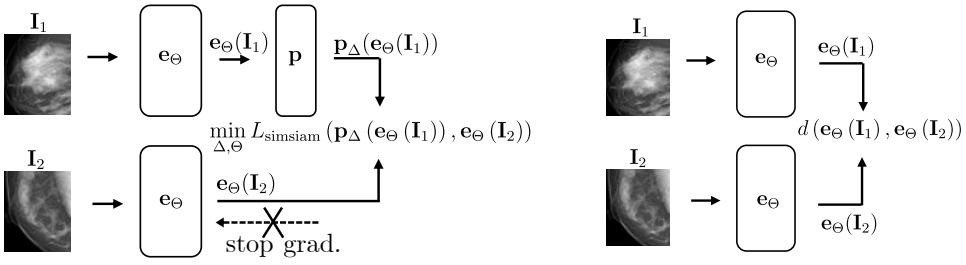
Chen et al. [46] showed that SimSiam implicitly learns to embed distinct features, which allows to differentiate negative pairs, despite being trained only on positive pairs. Hence, we propose to train the encoder in the SimSiam fashion, by presenting only matching images with the same style to the encoder. Since no artificial distance margin m must be enforced, this approach allows the style representations to be freely positioned in the embedding space. We anticipate that the distances between these representations in the embedding space will reflect the stylistic differences between the images.

Training an encoder with only positive pairs in a Siamese approach bears the risk of mode collapse, i.e., the encoder learns to map all images to the same point in the embedding space. To prevent this, we adopt the approach of Chen et al. [46] and utilize two asymmetries during training: 1) a neural network $\mathbf{p}_\Delta(\cdot) \in \mathbb{R}^{\mathbf{D}} \mapsto \mathbb{R}^{\mathbf{D}}$ with weights Δ, and 2) a one-sided gradient flow, as depicted in Fig. 7.1a. This results in the following loss function:

$$L_{\text{simsiam}} = d\left(\mathbf{e}_\Theta(\mathbf{I}_1),\, \mathbf{p}_\Delta(\mathbf{e}_\Theta(\mathbf{I}_2))\right), \tag{7.3}$$

which does not necessitate the maximization of distances.

For inference, the network $\mathbf{p}_\Delta(\cdot)$ is discarded, and the style representations are directly obtained from the encoder $\mathbf{e}_\Theta(\cdot)$. The distance between the output style representations is then computed. This completes the setup of StyleX, as depicted in Fig. 7.1b.



(a) Training (b) Inference

Figure 7.1: This figure illustrates the training process (left) where an encoder learns style representations, and the inference process of StyleX (right) used to compute the distance between two images.

7.2.4 The Distance Measure for StyleX

Up to now, no measure $d(\mathbf{r}_1, \mathbf{r}_2)$ to quantify the distance between the style representations has been defined. We employ the same distance metric in inference as in training. This is because, during inference, StyleX should accurately quantify the distances of the style representations, as the encoder has been optimized to do so. Having different distance metrics in training and inference might introduce additional compelxity and potential errors, if the two distance measurements do not align.

Chen et al. [46] applied the negative cosine similarity cs as the distance metric $d(\mathbf{r}_1, \mathbf{r}_2) = -cs(\mathbf{r}_1, \mathbf{r}_2)$. cs is defined as the dot product between the vectors divided by each vector's Euclidean norm: follows:

$$cs(\mathbf{r}_1, \mathbf{r}_2) = \frac{\mathbf{r}_1 \cdot \mathbf{r}_2}{\|\mathbf{r}_1\| \, \|\mathbf{r}_2\|} \tag{7.4}$$

As a consequence, cs measures the angle between two multidimensional representations, as it is normalized to the magnitude of the representations. Accordingly, the range of cs is $-1 \leq cs \leq 1$, where 1 represents identical representations and -1 represents representations pointing in opposite directions.

Both characteristics are beneficial for the encoder training, as the fixed output range and the disregard of the magnitude does stabilize the gradients of the optimization process. To employ cs as the inference distance measurement $d_{\text{inf}}(\cdot, \cdot)$ of StyleX, we convert it's range to a more intuitive range of $[0, 1]$, with 0 representing the minimum distance. Therefore, we normalize cs for inference as follows:

$$d_{\text{inf}}(\mathbf{r}_1, \mathbf{r}_2) = cs_n(\mathbf{r}_1, \mathbf{r}_2) = \frac{1 + cs(\mathbf{r}_1, \mathbf{r}_2)}{2} \tag{7.5}$$

It should be noted that this normalization does not alter our initial proposition that the training and inference loss should be identical, as we are merely mapping the output range from $[-1,1]$ to $[0,1]$.

7.2.5 Image Processing

The operation of the encoder $\mathbf{e}_\Theta$ and the evaluation of StyleX require processed X-ray images. Consequently, we employ two distinct pipelines to convert raw X-ray image projections into stylized versions, as they would be typically presented to radiologists:

1. The Linear Analysis Pipeline (LAP), designed for reproducible research and straightforward analysis.
2. The Proprietary Advanced Style System (PASS), an advanced closed-source prototype pipeline for processing clinically relevant image impressions.

Linear Analysis Pipeline (LAP)

LAP serves as a transparent pipeline, wherein the parameters influencing the style are comprehensible. The core functionality of LAP is twofold. Firstly, it weighs the frequency bands of the images, created with a Laplacian pyramid [34, 227, 232], as proposed in Section 2.3.4. Consequently, it enables the manipulation of image structures, defined by the frequencies, which represent them. Fine details are represented by high frequencies, medium-sized structures by low to mid frequencies, and background characteristics by low frequencies. Secondly, LAP maps a parameterized portion of the full pixel range to the final image. Depending on the size of this mapping window, the image's contrast is adjusted accordingly [103]. This results in three adjustable parameters of the pipeline:

1. w (window): This defines the image's contrast by setting a the mapping window with.
2. l (low- to mid frequencies): This determines the extent to which medium-sized structures are highlighted or suppressed by weighting low- to mid frequencies.
3. h (high frequencies): This emphasizes fine image details by weighting high frequencies.

A detailed flowchart and description of the pipeline are provided in Appendix A.

Proprietary Advanced Style System (PASS)

PASS, a prototype pipeline developed by Siemens Healthineers, aids radiologists by generating clinically relevant image impressions. Unlike LAP, PASS offers

advanced capabilities for complex image feature manipulation. We applied it to generate 32 distinct styles.

The Benefits of PASS and LAP

The availability of LAP's parameter settings serves as a valuable tool for analyzing style representations. Due to the linear nature of the pipeline, parameter settings that are closer together are expected to generate more similar styles. Thus, the labels not only indicate identical styles but also signify the distance between styles. This additional distance information is crucial for evaluating our method. However, LAP is a simplified pipeline, and our proposed method for creating a style metric should also be applicable to clinically relevant styles. For this reason, we also train the encoder $\mathbf{e}_\Theta$ and evaluate StyleX using data processed with PASS. On the other hand, unlike LAP, PASS operates as a black box with undisclosed parameters, making the relationship between the produced styles unclear.

7.2.6 Datasets

We generate two training sets and various test datasets using the LAP and PASS pipelines, by processing the raw data from MBTST, as detailed in Sec. 2.2.3. This choice of data source provides two significant advantages. Firstly, the public availability of the MBTST data ensures the reproducibility of our research. Secondly, the raw form of the data, as captured by the detector, allows us complete control over the processing and generation of styles. For our training and evaluation, we utilized only the data where information on breast density and thickness was available, resulting in a total of 7325 patients. Out of these 7325 DMs, we allocated 70 % (5064 images) for the creation of training and validation datasets, and the remaining 30 % (2171 images) for testing.

LAP Parameter Selection:

To describe the datasets generated with the LAP pipeline more efficiently, we define functions for selecting values from all three LAP parameters. These functions generate the corresponding parameter value within the range of its minimum and maximum values and are denoted as follows: $W(i) = i \cdot w_{\max} + w_{\min}$, $L(j) = j \cdot l_{\max} + l_{\min}$, and $H(k) = k \cdot h_{\max} + h_{\min}$, with $i, j, k \in [0,1]$. As a result, a style is defined by the combination of all three parameters values, represented by the tuple $\big(W(i), L(j), H(k)\big)$.

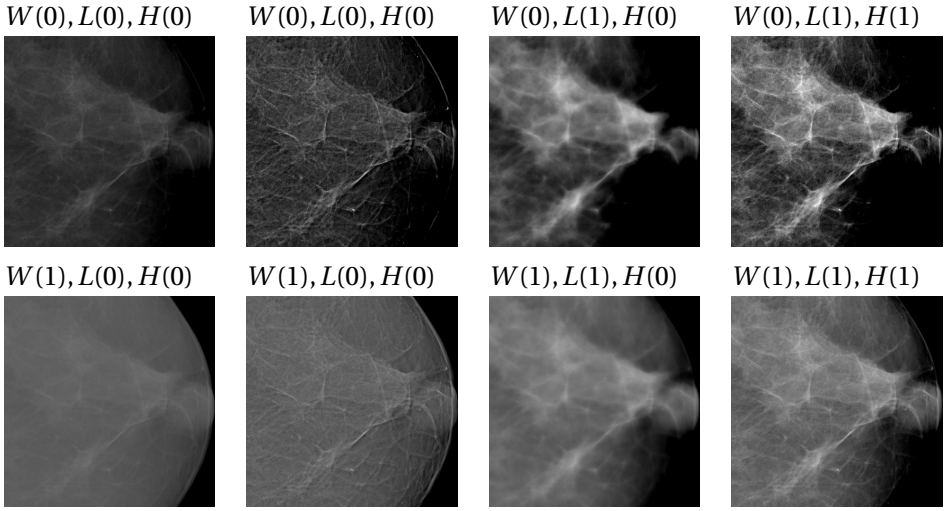


Figure 7.2: This figure displays styles generated using extreme parameter settings of LAP, as employed in the creation of the LAP-X dataset.

LAP-Train:

A training dataset is generated using the LAP pipeline to process the 5064 MBTST raw images reserved for training. The potential values for the parameters that define the styles are selected as follows:

$$\big(W(i),\ L(j),\ H(k)\big) \quad \text{with } i,j,k \in \{0, 0.2, ..., 1\} \tag{7.6}$$

For each raw image, 32 styles are randomly sampled from the $6^3 = 216$ styles obtained by assigning 6 different values to each parameter. This results in a total of 162048 training images.

LAP-X

To evaluate the ability of the encoder $\mathbf{e}_\Theta$ to generate unique style representations for distinct styles, we create styles using only the extreme values of the LAP parameters:

$$\big(W(i),\ L(j),\ H(k)\big) \quad \text{with } i,j,k \in \{0, 1\} \tag{7.7}$$

This approach generates 8 distinct styles for each of the 2171 raw test images, resulting in a total of 17368 images for the test set LAP's eXtreme parameter test dataset (LAP-X). Fig. 7.2 illustrates one of the processed raw images in all its 8 style manifestations. A clear distinction between all 8 styles can be observed.

Lap-$w/l/h$

A style metric's scalar output must reflect the degree of style difference. To analyze this behaviour, we create three specialized test sets, LAP's w parameter sweep dataset (LAP-w), LAP's l parameter sweep dataset (LAP-l), and LAP's h parameter sweep dataset (LAP-h). In each test set, the respecitve parameter is varied from its minimum to it's maximum in 10 steps as follows:

- LAP-w parameter: $(W(i),\ L(0.5),\ H(0.5))$ with $i \in \{0.0, 0.1, ..., 1.0\}$
- LAP-l parameter: $(W(0.5),\ L(j),\ H(0.5))$ with $j \in \{0.0, 0.1, ..., 1.0\}$
- LAP-h parameter: $(W(0.5),\ L(0.5),\ H(k))$ with $k \in \{0.0, 0.1, ..., 1.0\}$

Having 11 different styles for each of the 2171 raw test images, each test set contains a total of 23881 images. Fig. 7.3 illustrates an example of the three parameter sweeps. The style differences between images with neighboring parameter values are subtle, making it challenging for an untrained eye to distinguish between them, even in matching pairs.

PASS - Datasets

PASS is capable of generating 32 unique styles. We employ 28 of these styles to construct a training set (PASS-Train), while the test set (PASS-Test) incorporates all 32 styles, thereby enabling the evaluation of the StyleX metric on styles not previously encountered.

7.2.7 Conducted Trainings

We train the encoder $\mathbf{e}_\Theta$ two times. The first training is done on LAP-Train, and the trained encoder version is referred to as $\mathbf{e}_{\Theta_{\text{LAP}}}$. The encoder is then trained from scratch a second time with PASS-Train, yielding $\mathbf{e}_{\Theta_{\text{PASS}}}$. Both trainings are conducted with the exact same hyperparameter configuration. The images of both training datasets are cropped to dimensions of 800×800, positioning the nipple at the center of the right side to standardize the anatomical starting point. These cropped images are then resized to 400×400, which facilitates a batch size of 200. For both encoder trainings, we utilize a ResNet18 [86], pre-trained on ImageNet [55] encoder. The style representations are 2048 dimensions. We utilize SGD as the optimizer, with a learning rate of 0.05, momentum of 0.9, and weight decay of 10^{-4}. Training occurs over 200 epochs with batch sizes of 200.

7.2.8 Evaluation

The efficacy of StyleX relies on the ability of $\mathbf{e}_\Theta$ to generate meaningful style representations, as the StyleX output is determined by the scalar distance between

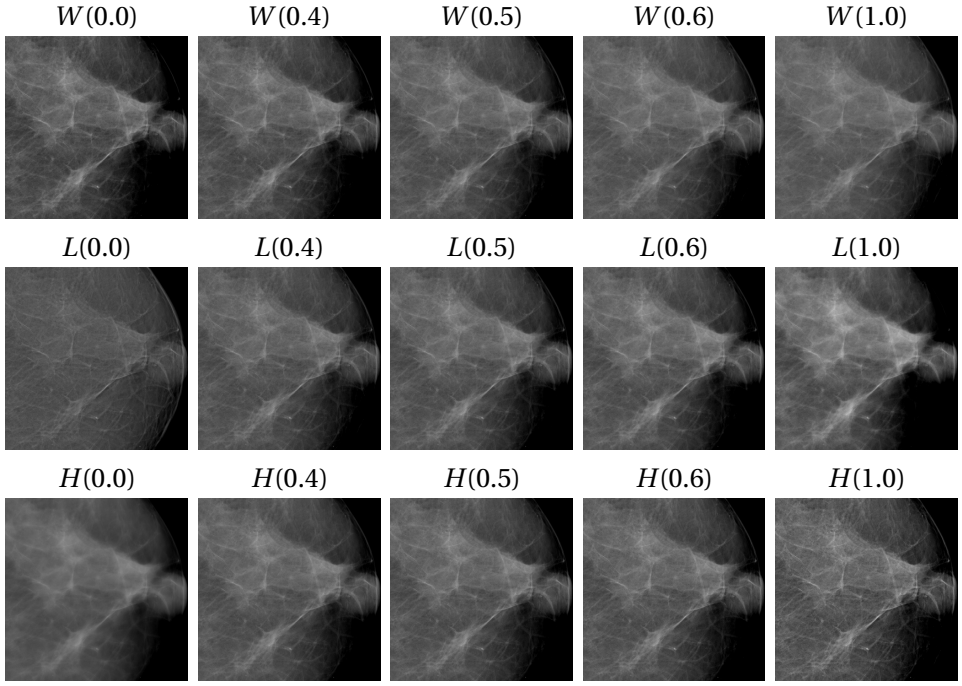


Figure 7.3: Styles generated by varying the three parameters of LAP. Images at the extreme left and right are produced using the minimum and maximum values of the corresponding parameter. The three central images are generated with parameter differences of 0.1. For each variation, the inputs to the other two defining functions are held constant at 0.5, as in $W(0.x), L(0.5), H(0.5)$. Each row represents one image of LAP-w, LAP-l, and LAP-h, respectively.

these representations. These representations should exhibit two crucial characteristics. First, the distance between two representations should reflect the degree of style variation. Second, representations of diverse styles should be distinctly separated in the embedding space, while those of identical styles should cluster closely. If these two properties are fulfilled, StyleX can accurately measure style distances. We employ two methods to evaluate these characteristics.

t-SNE Reduction

Visualizing style representations, which exist in a 2048-dimensional space, presents a significant challenge. However, by assessing the neighboring relationships between these representations, we can evaluate their ability to reflect stylistic differences based on the distances between them.

To accomplish this, we employ t-SNE [224] to reduce the dimensionality of the style representations. t-SNE is designed to preserve the local neighborhood structure from the original high-dimensional space within the reduced low-dimensional space. This is achieved by assigning probabilities in both spaces that reflect the relative distances between the representations. The Kullback-Leibler divergence between both probability distributions is then minimized with respect to the positioning of the representations in the low-dimensional space. If the representations are not well-separated in the low-dimensional space, caution is required, as this could indicate a failure of t-SNE to preserve the local neighborhood structure. On the other hand, if the representations are grouped into well-defined clusters, we can confidently infer that these clusters also exist in the high-dimensional space. If this were not the case, it would imply that t-SNE has randomly discovered a more ordered structure in the low-dimensional space than in the high-dimensional space, a scenario that contradicts the law of entropy [186].

k-Nearest Neighbors Classification Accuracy

While t-SNE offers valuable insights into the local neighborhood structure of the representations, it does not provide a quantitative measure of the quality of the separation. For this purpose, we employ the k-Nearest Neighbors (k-NN) algorithm, which assigns a class to a data point based on the majority class of its k-nearest neighbors, without needing to fit parameters. Using this approach, we can take the high-dimensional representations of our test sets and assign them a style class, based on the k-nearest neighbor representations from the training set. The estimated style class is then compared with the actual style class, and a classification accuracy is computed. This accuracy precisely quantifies the number of data points correctly clustered, thereby providing a measure of the representations' ability to reflect style differences.

7.3 Experiments & Results

To evaluate the capabilities of StyleX, we first investigate the ability of $\mathbf{e}_{\Theta_{\text{LAP}}}$ to generate meaningful and well-defined style representations on the LAP testsets. This allows us to conduct a detailed investigation into the proposed method's capacity to generate style representations suitable for a style metric. Next, we assess the ability of StyleX to work on clinically relevant data. This is done by investigating the ability of $\mathbf{e}_{\Theta_{\text{PASS}}}$ to distinguish the styles of the PASS-Test dataset and by applying StyleX to images from the PASS-Test dataset to evaluate its overall performance as a metric.

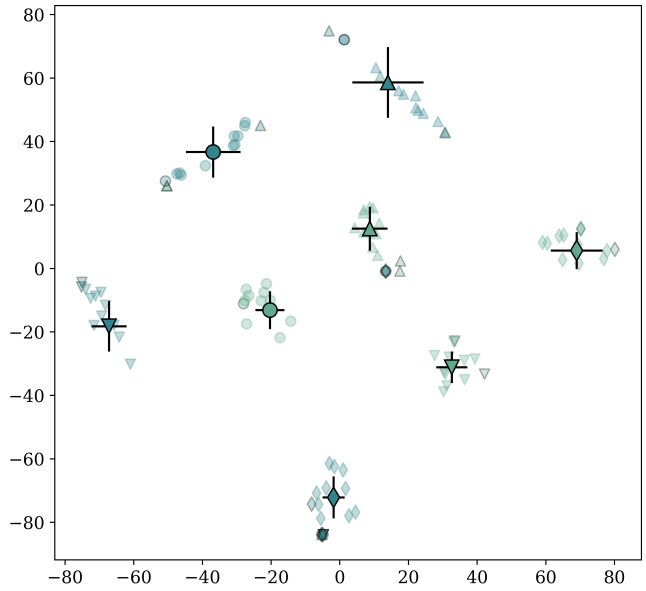


Figure 7.4: The figure presents style representations, reduced to 2D using t-SNE, generated from LAP-X images processed by $\mathbf{e}_{\Theta_{\text{LAP}}}$.

7.3.1 LAP Style Separation

The eight unique styles present in the LAP-X dataset are well-differentiated. Therefore, for precise distance measurements with StyleX, it is essential that the encoder $\mathbf{e}_{\Theta_{\text{LAP}}}$ generates style representations of LAP-X, which form distinct clusters within the embedding space. To evaluate the clusters formed by the style representations of LAP-X, we employ t-SNE to reduce the 2048D representation into a visualizable 2D space. The resulting 2D representations are depicted in Fig. 7.4. The 2D t-SNE reduction of LAP-X is illustrated, with each cluster's mean represented by a marker with a black border and the variance indicated by a line in each direction. Additionally, the two top-most outliers and ten randomly selected points from each cluster are visualized. The distinct clusters formed in the embedding space suggest that the style representations are well-defined and unique. The high classification accuracy of 99.7%, achieved with k-NN, further supports the distinctness and well-definition of the style representations.

7.3.2 Representation Distance in Relation to Style Differences

A key characteristic of the representations, essential for a meaningful StyleX, is that the distance between two representations should correlate with the degree of style difference. To examine this behavior, we use three test sets: LAP-w, LAP-l, and LAP-h. Each set varies its respective parameter from the minimum to the

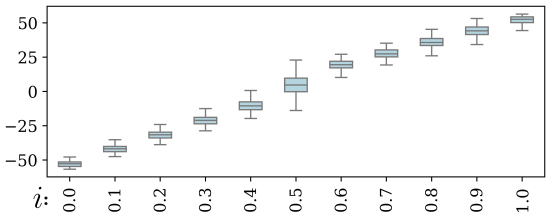
maximum value in 11 steps, generating images with subtle style changes. We reduce the 2048D representations corresponding to these test sets to 1D using t-SNE and visualize the 1D representations in boxplots in Fig. 7.5. Each box comprises the style representations of images processed with identical parameter values. Given that only one of the three parameters changes at a time, and considering the preservation of local neighborhoods by t-SNE, it is possible to analyze the correlation between the changing style parameter and the corresponding representations. As observed, the median values of each box increase or decrease monotonically with the parameter values, suggesting a correlation between the distance of the representations and the degree of style difference. Furthermore, LAP-Train does not employ steps with odd parameter values, hence styles generated with these values are not encountered during training. Despite this, no noticeable difference exists in the clustering of even and odd parameter values, suggesting that $\mathbf{e}_\Theta$ can interpolate to unseen styles.

7.3.3 Separation of Clinically Relevant Styles

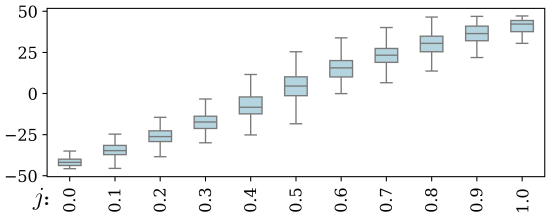
To investigate the applicability of our proposed method to complex and clinically relevant styles, we use StyleX based on $\mathbf{e}_{\Theta_{\text{PASS}}}$, since we do not anticipate that $\mathbf{e}_{\Theta_{\text{LAP}}}$ will separate the PASS styles, due to their higher complexity. Following the approach outlined in Sec. 7.3.1, the 2D t-SNE reduction is applied to the style representations of the PASS-Test dataset. These 2D representations are visualized in Fig. 7.6. It is important to note that four of the 32 styles were not encountered during training and are marked with blue borders. The formation of distinct clusters in the embedding space suggests that the style representations are well-defined and unique. The high classification accuracy of 99.83% achieved with k-NN further supports this observation. Interestingly, the clustering does not differ between the unseen and seen styles, indicating $\mathbf{e}_{\Theta_{\text{PASS}}}$'s ability to generalize to styles not encountered during training.

7.3.4 StyleX Application

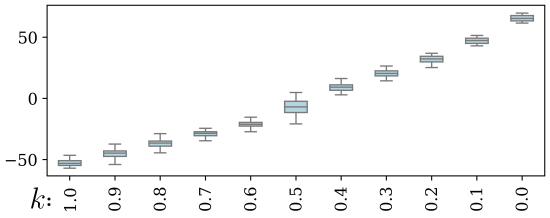
To evaluate the ability of StyleX in measuring the distance between pairs, we apply it to compute the distances between a reference image and both matching and non-matching pairs. The images with their computed distances are depicted in Fig. 7.7. The reference image is displayed in the first row, with two images with the same content but different styles. The second row shows three non-matching images, with the first image having the same style as the reference image, while the second and third images have the same style as the corresponding images in the first row. It can be observed that the computed distances between the reference image and the matching pairs are relative to the style differences. The distance between the non-matching image, which shares the same style as the



(a) Parameter sweep with $W(i)$



(b) Parameter sweep with $L(j)$



(c) Parameter sweep with $H(k)$

Figure 7.5: 1D style representations are generated from data processed by $\mathbf{e}_{\Theta_{\text{LAP}}}$ from LAP-l, LAP-h, and LAP-w. Styles created with identical parameter settings are grouped together in one box. Each plot illustrates the style representation from one of the three datasets.

reference image, is close to zero. Meanwhile, the distance between the non-matching pairs approximates the distances of the matching content counterparts. Therefore, in the given scenario, the image content does not affect the computed style distances. This observation aligns with the outcomes of previous experiments where the style representations were distinctly separated and independent of the content.

7.4 Discussion

The heart of StyleX is the encoder $\mathbf{e}_\Theta$. If $\mathbf{e}_\Theta$ is capable of generating meaningful style representations, the StyleX metric can accurately quantify the distance between these. Hence, our experiments focused on evaluating the ability of

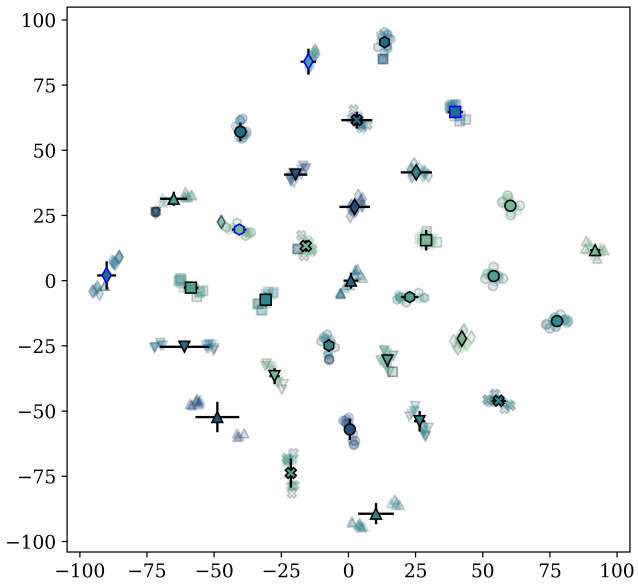


Figure 7.6: Style representations, reduced to 2D with t-SNE and illustrated in the figure, are generated from PASS-Test images processed by $\mathbf{e}_{\Theta_{\text{PASS}}}$. Cluster centers marked with blue borders indicate styles not encountered during training.

$\mathbf{e}_\Theta$ to generate style representations that are well-separated and reflect the degree of style difference. The knowledge of the underlying parameters of LAP give insight about the relationship between styles and hence it could be investigated if the style representations reflect these changes. In Experiment Sec. 7.3.1, we demonstrated that the style representations of the LAP-X dataset are well-separated, achieving a high classification accuracy of 99.7%. This result allows us to conclude that the first important property of $\mathbf{e}_\Theta$ is fulfilled: the ability to generate well-defined style representations for distinctive styles. The second experiment detailed in Sec. 7.3.1, analyzed the correlation between the distance of the representations and the degree of style difference. The boxplots in Fig. 7.5, representing the 1D t-SNE reduction, demonstrated a clear relationship between the distances and style differences. Thus, the second crucial property of $\mathbf{e}_\Theta$, the ability to generate style representations that reflect the degree of style difference, is also fulfilled. The third experiment, detailed in Sec. 7.3.3, evaluated the ability of $\mathbf{e}_\Theta$ to work with complex clinically relevant styles. In contrast to the 8 styles of LAP-X, there were 32 distinctive styles. However, with a high classification accuracy of 99.83% and distinct clusters in the 2D t-SNE reduction, it is evident that $\mathbf{e}_\Theta$ is capable of generating well-defined style representations for complex styles. Thus, we can conclude that $\mathbf{e}_\Theta$, when trained with SimSiam, performs

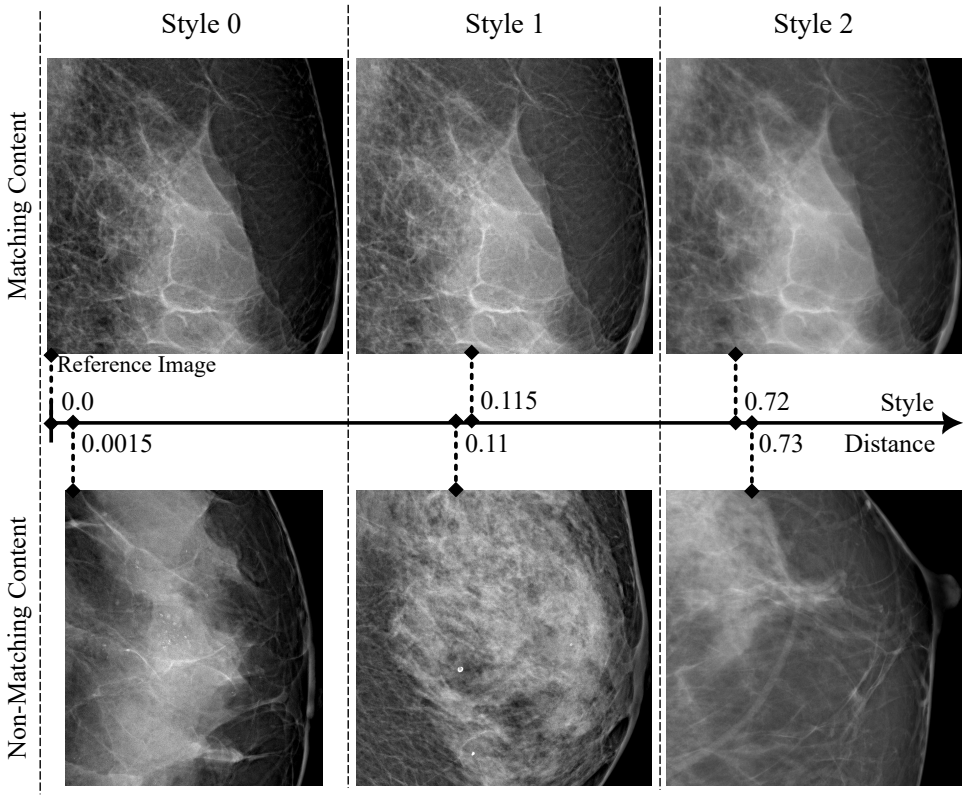


Figure 7.7: Example application of the StyleX. The style distance between all images and the reference image at the top left of the figure is calculated. The first row compares images with different styles but same content as the reference image. Images in the second row have different content, and column-wise the same style.

effectively on clinically relevant styles. Finally, we use StyleX to assess the correlation between the image impression and the computed distance, using both matching and non-matching content pairs (cf. Fig. 7.7). The visually perceived style differences between the images align well with the measured style distances. Moreover, the style quantification remains consistent whether comparing images with matching or non-matching content. This indicates that the measured style distance is not influenced by the image content. In the experiment detailed in Sec. 7.3.2, we demonstrated that $\mathbf{e}_\Theta$ is capable of interpolating unseen styles. Additionally, in the experiment in Sec. 7.3.3, we showed that $\mathbf{e}_\Theta$ can generalize to four unseen styles. Hence, to some extent, we can conclude that $\mathbf{e}_\Theta$ can generalize to unseen styles. However, we hypothesize that substantial unseen style modifications might not be well separated. We propose that the more extensive and versatile the training dataset is, the more generally StyleX functions. The transition from LAP-Train to PASS-Train did not affect the performance of StyleX, suggesting that our method, particularly when trained with SimSiam, works well with even more versatile datasets. $\mathbf{e}_{\Theta_{\text{LAP}}}$ exhibits similar behavior on the LAP-Test datasets as $\mathbf{e}_{\Theta_{\text{PASS}}}$ does on the PASS-Test dataset, indicating that our proposed method for training StyleX can be effectively applied on clinically relevant and complex X-ray image styles. This is further supported by SimSiam's ability to rely solely on positive pairs, facilitating the compilation of larger datasets from different pipelines. The requirement is only that images with the same style label share the same style. In our research, we utilized StyleX on mammographic images, using the publicly accessible MBTST dataset of raw X-ray images. However, this method is not exclusive to mammographic images. We are confident that StyleX's application can be adapted to any X-ray image, regardless of the body region, and that the concept could even be extend to other modalities like MR. However, substantial style modifications necessitate encoder retraining.

7.5 Future Work

The proposed StyleX metric offers a promising approach to quantifying stylistic differences in X-ray images. Our method was evaluated on mammographic images, with the encoder trained solely on styles from a single pipeline. Future research may involve compiling a more comprehensive dataset. It would be intriguing to consider the expansion of the training dataset to include other X-ray modalities. Furthermore, the incorporation of styles from various pipelines and vendors could be beneficial. These styles could be made more complex by augmenting them with additional image processing steps, such as the LLF discussed in Chapter 6. Besides utilizing more complex data for training, the evaluation process could also be broadened to include a test set featuring styles from different vendors and distances, labeled by an expert. We anticipate that the flexible

SimSiam training, coupled with a more extensive dataset, will yield a StyleX capable of accurately quantifying complex stylistic differences across a wide range of X-ray image styles. This would provide a powerful tool for radiologists or experts needing to compare images with different styles.

Furthermore, the concept of training a metric with non-matching pairs could be extended to other domains. It is conceivable to develop a metric to detect the degree of artefacts in X-ray images, or to quantify the similarity between artistic image styles or musical styles.

StyleX is not merely advantageous for human comparison of images, but could also act as a loss metric for machine learning algorithms, such as the optimization of the LLF or a neural network designed to generate stylized images.

Another potential area of investigation could involve employing $\mathbf{e}_\Theta$ trained on a large dataset and fine-tuning it to directly output the image parameters necessary to achieve a desired style.

7.6 Conclusion

This chapter presents StyleX, an innovative deep-learning metric for quantifying stylistic differences in medical X-ray images. The metric is based on an encoder, which sucesfully generates style representations independent of the image content. Based on these style representations, we construct a style metric capable of accurately quantifying stylistic differences in X-ray images.

To train the encoder, we propose a unique application of the SimSiam concept, originally designed for unsupervised pre-training of encoders by reducing the similarity between images with the same content. Our adaptation trains the encoder to produce embeddings that contain only information about X-ray image styles.

In a second step, we leveraged t-SNE and k-NN to demonstrate that the encoder generates well-separated style representations reflecting the degree of style difference. This ability to produce meaningful distances allows us to infer that the constructed style metric is capable of accurately quantifying stylistic differences in X-ray images.

Thus, this research lays the groundwork for a style loss in medical imaging, which could facilitate a variety of applications, such as an automatic style selector for radiologists or a loss function in imaging pipeline optimization.

8 Conclusion

The overarching goal of this work is to improve the diagnostic accuracy of X-ray images by enhancing the processing of recorded X-ray images using machine learning methods. This objective has been subdivided into four research objectives, outlined in detail Chapter 1:

1. Collimator shadow detection in X-ray images.
2. X-ray image denoising.
3. Automatic adaptation of X-ray image processing pipelines.
4. Quantifying differences between non-matching image pairs.

The first part of the thesis addresses the initial two research goals: eliminating or minimizing artifacts, specifically noise and collimator shadows. The latter part focuses on the final two objectives related to X-ray image impressions, specifically, the automatic adaptation of X-ray image processing pipelines to achieve a desired impression, and the quantification of differences in impressions between non-matching image pairs.

8.1 Summary

The first part of this work, discussed in Chapter 3, serves as a foundation for the next two chapters which investigate the first two research objectives. It introduces a novel noise simulation framework for X-ray images, which accurately models the noise characteristics of real X-ray images. This framework enables the realistic alteration of X-ray image dose levels, thereby accounting for additional noise. As a result, the proposed pipeline can be used to generate training data for denoising. Furthermore, the simulation of collimator shadows necessitates a dose level adjustment, which is facilitated by this noise simulation.

Given that the purpose of the noise simulation is to generate training data, it has been designed with physically meaningful parameters. These can be adjusted

to simulate different detectors, dose levels, and noise characteristics. This is especially beneficial in a training context, as it allows for the generation of a wide range of training data. Consequently improving the genralization of a trained ANN. The simulation automatically converts pixel intensities into photon counts and effectively simulates a reduction in arriving photons. It also accounts for scintillator blurring, which introduces a spatial correlation in the noise. Finally, the simulation incorporates electronic noise, a factor that is particularly significant at low dose levels. In the corresponding experiments, the simulated noise characteristics have been compared to the ground truth noise of real X-ray images by investigating the NPS. The results demonstrate that the proposed noise simulation accurately models the noise characteristics of real X-ray images across different dose levels. Thus, we conclude that the noise simulation is well-suited for generating training data for deep learning models.

Chapter 4 directly addresses the first research question by focusing on the detection of collimator shadows in X-ray images. This chapter makes two key contributions. It introduces a new simulation pipeline that creates collimator shadows on clinical images, effectively tackling the issues of limited training data and the time-consuming task of manual labeling. Similar to the noise simulation, this pipeline has physically meaningful parameters that can be adjusted during the training process to promote effective ANN generalization. Moreover, the pipeline's realism and accuracy were evaluated by testing it against collimator shadows on a phantom, and the results have confirmed its validity. Additionally, it pexplores the use of deep learning for identifying collimator edges. Based on the assumption that ANNs converge more effectively when known operators are incorporated into the architecture [149], we utilized the Hough Transform (HT) to integrate the prior knowledge that collimator edges always form straight lines. The trained model was assessed using both clinical and simulated images. Despite being trained solely on simulated images, the model demonstrated effective generalization to clinical images, further underscoring the efficacy of the simulation pipeline. Furthermore, we demonstrated that a model could be trained exclusively in the Hough domain, enabling it to directly predict the line parameters that describe the collimator edges. Additionally, the HT was incorporated into a second network architecture as a regularizer, which enhanced the model's segmentation output by enforcing straight lines. Therefore, we conclude that deep learning proves effective in detecting collimator shadows.

In addressing the second research objective, Chapter 5 presents a novel deep learning-based denoising network for DBT projections. The removal of noise is intended to simplify image interpretation, thereby enhancing diagnostic accuracy. Additionally, it should reduce the disparity between FFDM and DBT images, which are inherently noisier due to the lower dose levels utilized in DBT. The

network was trained on FFDM images, which were augmented using the noise simulation pipeline of Chapter 3, to replicate the noise characteristics of DBT images. Considering that denoising should not eliminate diagnostically relevant information, such as minuscule microcalcifications that closely resemble noise, we introduced a novel loss function, $\mathcal{L}_{\text{ReLU}}$, specifically designed to preserve small structures. The network's performance was evaluated using a microcalcifications dataset and compared to other loss functions. The results indicate that $\mathcal{L}_{\text{ReLU}}$ significantly improves the preservation of small structures. However, a combination of $\mathcal{L}_{\text{ReLU}}$ and SSIM could potentially yield even more stable results. Moreover, the diagnostic accurady of radiologists is impeded for dense breasts, as the overlapping tissues obscure carcinoma. Therefore, it is crucial to ensure that denoising algorithms do not exacerbate the disparity in diagnostic accuracy among different breast groups. For this reason the network's performance was evaluated across different breast types, and the results indicate that the network does not exhibit bias towards specific groups. Finally, we demonstrated the network's denoising performance on a real-world case, illustrating its ability to denoise actual DBT projections and enhance microcalcification visibility in SM reconstructions.

Radiologists depend on their accustomed image impressions for efficient diagnosis. As a result, image processing pipelines are manually adjusted in a laborious process to achieve the desired image impression. To circumvent this subjective and suboptimal procedure, in Chapter 6, we addressed the third research objective: the automatic adaptation of X-ray image processing pipelines to achieve the desired image impressions. Despite focusing on a single question, the contributions of this chapter are twofold. The automatic adaptation of X-ray image processing pipelines requires an optimizable pipeline. Consequently, we explored the feasibility of applying the LLF, a state-of-the-art photographic image processing pipeline, to X-ray images. Additionally, we enhanced the functionality of the LLF by replacing its remap function with a MLP, thereby increasing its versatility. Subsequently, we optimized the LLF to match the image impression of a vendor pipeline by processing raw projections. The optimization process was conducted using SGD and backpropagation, thus we implemented a differentiable version of the LLF. We demonstrated the effectiveness of the LLF in matching the image impression of the vendor pipeline, thereby proving that (a) the LLF is versatile enough to match existing vendor pipelines, and (b) optimization with GD is feasible. Moreover, this optimization process proved effective on small training datasets comprising five images, making it particularly suitable for medical imaging where data are always limited. Optimizing the LLF with GD allows its integration into a larger pipeline, as demonstrated by our addition of a trainable window leveling operation. Despite this, the LLF remains interpretable

and adjustable, and can be executed in real-time using parallel computation on GPUs.

The final research objective is to quantify the differences in image impressions between non-matching image pairs. Identifying these differences is crucial to maintaining diagnostic accuracy and is particularly relevant in clinical practice, where it is uncommon to encounter pairs of images with identical content processed on two different devices. This objective is addressed in Chapter 7, where we introduce a deep-learning metric, StyleX, specifically designed to quantify appearance variations in medical X-ray images. The essence of this metric is an encoder that produces content-independent style representations of X-ray images. The disparity between these representations is then quantified to assess the dissimilarity in image impressions. Given the absence of a style metric, there were no labels to quantify style distance. To overcome this challenge, we employed SimSiam, a methodology originally proposed for self-supervised pre-training by minimizing the distance between augmented views of the same image. To assess the effectiveness of the style metric, we demonstrated the distinctiveness of the style representations generated by the encoder. Additionally, we showed that the distances between these representations correspond to the perceived differences in image impressions. Finally, we established the feasibility of the metric by comparing images with both matching and non-matching content, thereby demonstrating that the metric successfully quantifies style differences independent of the content.

By adressing the proposed research objectives, three challenges crystallized as central to the sucess of the proposed solutions:

1. Given the sensitivity of patient data, data scarcity is a prevalent issue in medical imaging. Therefore, the development of physics-based data augmentation techniques, such as the noise simulation proposed in Chapter 3 and the collimator shadow simulation in Chapter 4, proved essential for generating training data. Furthermore, unsupervised learning techniques like SimSiam in Chapter 7 were instrumental in addressing the scarcity of labeled data.

2. The reliability of X-ray image processing algorithms is paramount, as errors can lead to severe consequences. The incorporation of prior knowledge into these models can narrow the solution space, thereby enhancing their reliability. In the collimator detection Chapter 4, the HT was utilized to incorporate the understanding that collimator edges are straight lines, thereby reducing outliers in the model's predictions. For the automatic adaptation of the image processing pipeline, the principle of preserving the direction of image gradients to maintain diagnostic information was incorporated. This was achieved through the use of the LLF in the optimization process, as detailed in Chapter 6.

3. Quantifying improvements in image quality is challenging, both during the training of deep learning models and when evaluating proposed solutions. To address this, a specially tailored loss function was introduced for denoising in Chapter 5, aiming to preserve diagnostically relevant information. Moreover, Chapter 7 introduced a deep learning metric to quantify differences between disparate image pairs.

8.2 Future Work

This work investigates various stages of the X-ray image processing pipeline. An initial, apparent step is to integrate the presented solutions, including collimator shadow detection, denoising, and post-processing to emphasize diagnostic information, into a fully optimizable pipeline. This integrated pipeline can then be optimized using StyleX to align with the radiologist's preferred style.

Moreover, we believe that out work represents only the initial step towards a self-adaptive and sophisticated X-ray image processing pipeline. We propose that a general strategy for improving X-ray image processing pipelines involves algorithms that garner additional knowledge about the image content and the patient. This knowledge can facilitate differential processing of various areas and steps within an X-ray image. Furthermore, to ensure reliability and trust in the processing pipelines, the methods employed should prioritize interpretability and self-explanatory functions. To achieve this objective, several different research directions are worth exploring.

The first direction, particularly crucial in the medical domain, is to address the issue of data scarcity. A particularly promising approach involves the development and enhancement of simulation techniques and pipelines, such as the VICTRE pipeline. Generating X-ray images with Monte Carlo simulations from simulated phantoms can provide a substantial amount of training data, while simultaneously offering comprehensive information about the image content. Moreover, this approach would generalize our proposed denoising and collimator simulations as the complete X-ray acquisition process can be simulated. Nevertheless, as demonstrated in Chapter 5, the simulated breast phantoms of the VICTRE pipeline do not accurately represent real-world breasts, as they lack the complexity of actual breast tissue. Thus, research efforts should be directed towards enhancing the realism of phantoms. A possible approach might be the utilization of real world Computed Tomography (CT) phantoms as ground truth for adversarial or diffusion networks. These networks can then be employed to enhance the realism of simulated phantoms. For instance, these networks can be trained to generate realistic 3D breast tissue or specific anatomical structures such as carcinoma, implants, or bones, which can then be incorporated

into an existing phantom. Having realistic phantoms as a ground truth also enables the development of more sophisticated metrics, as the knowlege about the anatomical structures and image content can be incorporated, allowing for a clear assesment if the visibility of anatomical structures is improved or not.

The second direction involves the abstraction of the X-ray image content to facilitate differential processing. This can be achieved by segmenting the X-ray images into different areas, each of which can be processed differently. Moreover, defining these areas also enables the explainability of deep-learning-based processing pipelines, as the knowledge about the decision how areas are processed is provided. However, pixels in X-ray images represent overlapping tissues, making segmentation a challenging task. Consequently, segmentation maps may not be as discrete as those in photographic images. Nevertheless, simulated X-ray images with known content are well-suited for training deep learning models that improve the segmentation of X-ray images. Moreover, foundation models like Sam2 [185] or MedSam [145] can be utilized to improve the segmentation of X-ray images.

Finally, the automatic adaptation of X-ray image processing pipelines can be further enhanced. Rather than solely incorporating radiologists' feedback in the form of previous X-ray images, which limits improvements in image impression, their feedback should be integrated as a preference function. This function can be optimized using reinforcement learning techniques. Moreover, diagnostic errors made by radiologists can also be considered during the optimization process. For instance, an undetected carcinoma might prompt adjustments to image processing parameters to better highlight such carcinomas. Therefore, instead of maintaining a fixed image impression, the image impression can evolve in tandem with the radiologist's preferences and skills to achieve a perfect interplay.

8.3 Final Words

The initial objective of this work was to explore the potential of machine learning in enhancing the diagnostic accuracy of X-ray images. Consequently, we have demonstrated that machine learning algorithms, particularly deep learning algorithms, can significantly improve X-ray image processing algorithms at every stage of the imaging pipeline. Given the rapid and ongoing advancements in deep learning techniques, we posit that our work merely provides a glimpse into the potential improvements that could be realized in future X-ray image processing.

A

Detailed Results

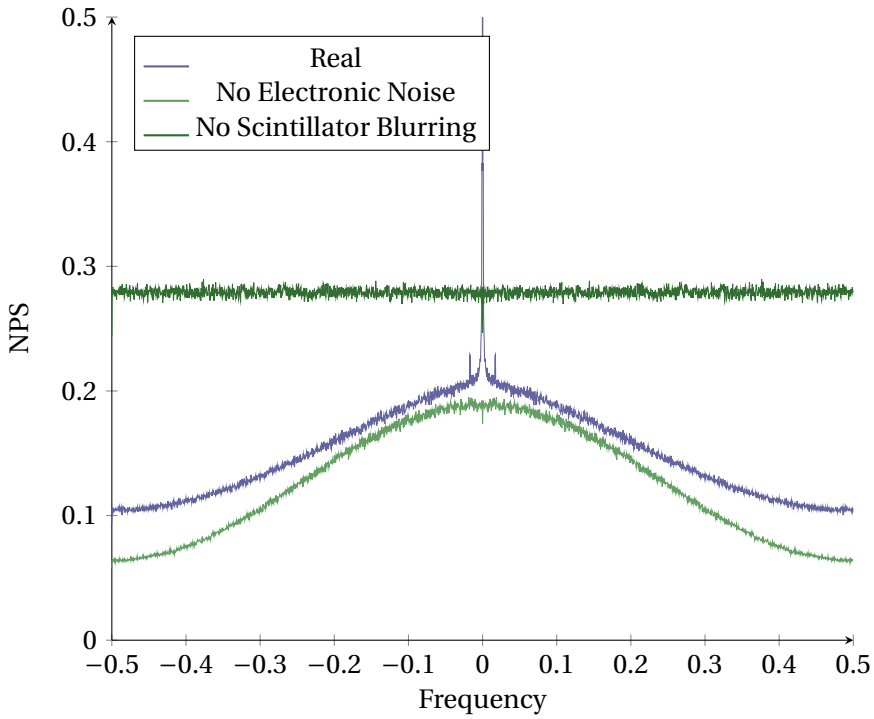


Figure A.1: Comparison of the NPS simulation at dose reduction Lvl. 2, without electronic noise or scintillator blurring, against the NPS of the actual acquisition.

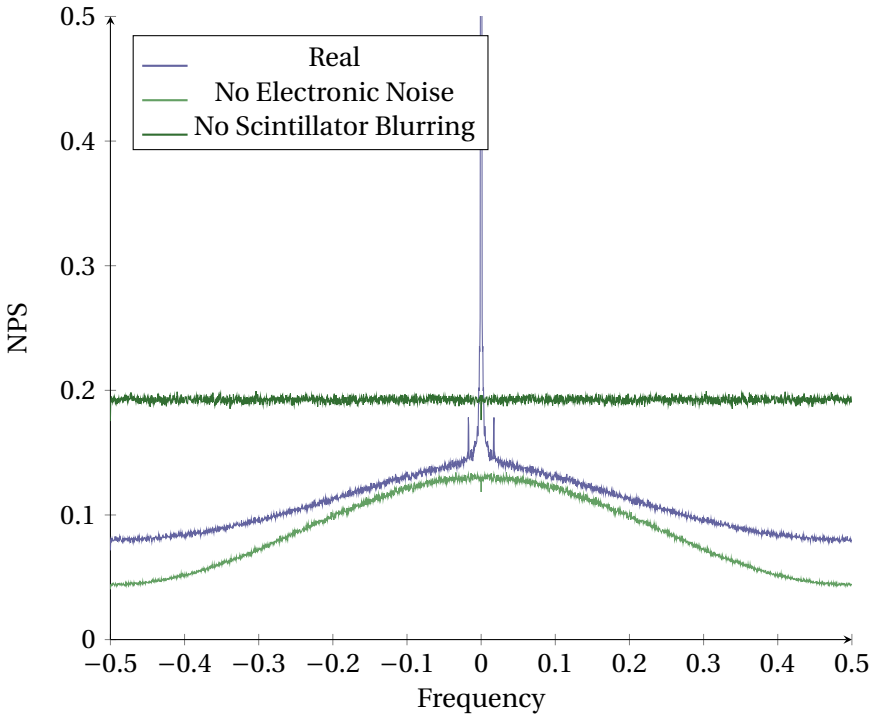


Figure A.2: Comparison of the NPS simulation at dose reduction Lvl. 3, without electronic noise or scintillator blurring, against the NPS of the actual acquisition.

	SegNet	RegNet	H-Net	DH-Net
General	0.9887 ± 0.0073	0.9919 ± 0.0049	0.9704 ± 0.0470	0.9794 ± 0.0123
Artifacts	0.9903 ± 0.0067	0.9931 ± 0.0036	0.9305 ± 0.2145	0.9306 ± 0.2154
Implants	0.9859 ± 0.0113	0.9892 ± 0.0089	0.9649 ± 0.0762	0.9789 ± 0.0098

Table A.1: Mean Dice scores and standard deviations of SegNet, RegNet, H-Net, and DH-Net on three test sets.

Hyperparameter	**Value**
Number of epochs	300
Optimizer	Adam [115]
Learning rate	0.0001
Loss function	ReLU-loss combined with SSIM
ReLU-loss weighting factor (η)	10
Overestimation punishment factor (c)	3
Image crop size	64×64
Number of training patches	276,345
Number of validation patches	69,766
Dose reduction factor	25

Table A.2: Hyperparameters for training the denoising network.

Calc.	Original	Noisy	MSE	ReLU-L	SSIM	ReLU-L + SSIM	MSE + SSIM
p6							
p7							
p8							
p9							
10							
p11							
p12							
p13							
p14							

Figure A.3: The denoised patches with microcalcifications, which are missing in Section 5.3.2, are depicted. The MSE between each patch and the GT is measured, and the average of these measurements is stated.

Calc.	MSE	ReLU-L	SSIM	ReLU-L + SSIM	MSE + SSIM
p1	1650.5	1025.6	975.2	**886.5**	1139.8
p2	979.4	603.0	647.4	**547.2**	757.8
p3	1138.8	820.0	591.4	**518.6**	857.0
p4	1275.1	1287.2	1337.0	1165.0	**1118.3**
p5	1220.9	1139.1	**837.9**	1044.4	1006.4
p6	706.0	**566.0**	583.2	588.0	600.2
p7	684.6	**271.2**	350.6	289.0	358.8
p8	681.0	492.8	487.6	**462.4**	572.7
p9	782.3	**559.8**	664.2	563.4	701.2
p10	763.4	761.0	818.0	**701.6**	781.6
p11	796.8	**423.2**	614.5	467.6	612.2
p12	554.2	504.7	**490.2**	523.0	496.5
p13	1220.9	1139.1	**837.9**	1044.4	1006.4
p14	545.4	494.6	533.1	**489.8**	512.1
p15	502.7	**429.5**	433.9	452.0	430.4
Average	900.13	701.06	680.14	**649.52**	730.09

Figure A.4: This table shows the MSE of the denoised microcalcifications against the GT, corresponding to Fig. 5.5 and Fig. A.3.

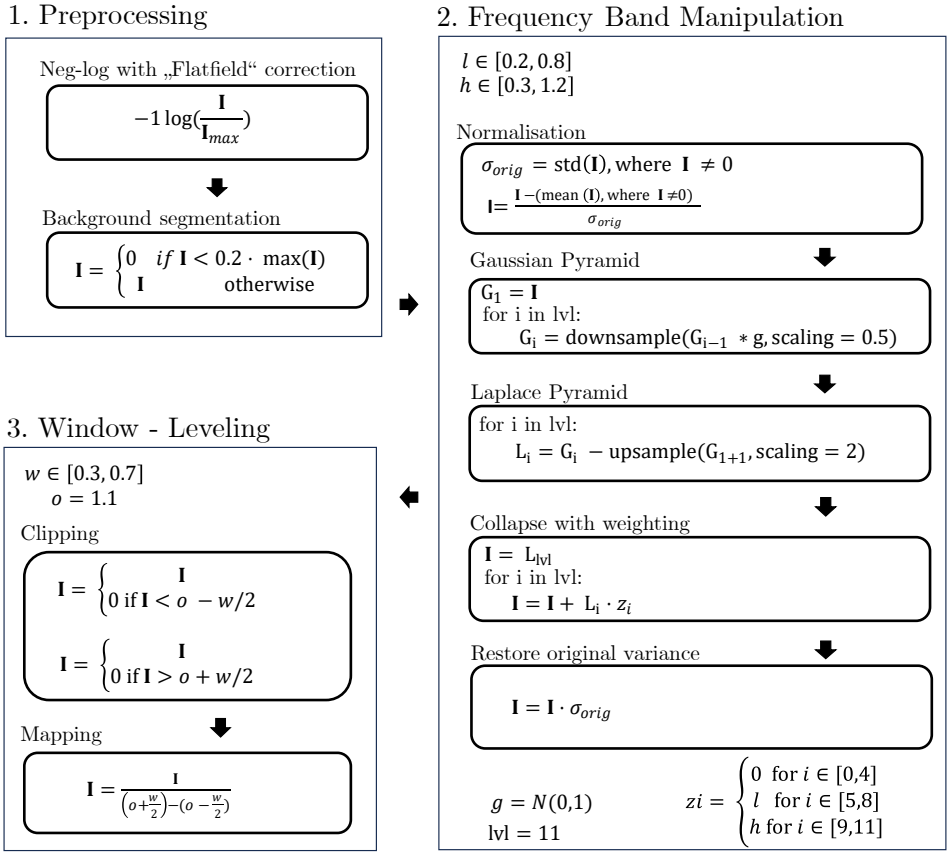


Figure A.5: Description of the proposed LAP-Pipeline. Parameters l, h and w are applied within their defined ranges to produce X-ray image styles. The actual parameter ranges for h, w, and l are defined from 0 to 10.

B

Abbreviations and Notations

ADC Analog-to-Digital Converter

ANN Artificial Neural Network

BI-RADS Breast Imaging Reporting and Data System

BM3D Block-Matching 3D

CC Craniocaudal

CDF Cumulative Distribution Function

CDR Cancer Detection Rate

CMOS Complementary Metal-Oxide-Semiconductor

CNN Convolutional Neural Network

CT Computed Tomography

DALI Workshop on Data Augmentation Labeling and Imperfections

DBT Digital Breast Tomosynthesis

DCIS Ductal Carcinoma In Situ

DH-Net Deep Hough Network

DHT Deep Hough Transform

DM Digital Mammogram

DnCNN Denoising Convolutional Neural Network

EdgeM Edge Module

FCN Fully Convolutional Network

FFDM Full Field Digital Mammography

GD Gradient Descent

GPU Graphics Processing Unit

∇-Matching Remap Function generated with Gradient Matching

GT Ground Truth

HD Hough Domain

H-Net Hough Network

HT Hough Transform

IDC Invasive Ductal Carcinoma

ILC Invasive Lobular Carcinoma

k-NN k-Nearest Neighbors

LAP Linear Analysis Pipeline

LAP-h LAP's h parameter sweep dataset

LAP-l LAP's l parameter sweep dataset

LAP-w LAP's w parameter sweep dataset

LAP-X LAP's eXtreme parameter test dataset

LCIS Lobular Carcinoma In Situ

LLF Local Laplacian Filter

lr Learning Rate

LUT Look-Up Table

MBTST Malmö Breast Tomosynthesis Screening Trial

MICCAI Medical Image Computing and Computer Assisted Intervention

MLO Mediolateral Oblique

MLP Multi-Layer Perceptron

MLPNorm-RM MLP Remapping Function optimized with Norm.

MLP-RM MLP Remapping Function optimized w/o Norm.

MSE Mean Squared Error

MSSIM Mean Structural Similarity Index

NPS Noise Power Spectrum

Orig-RM Original Remapping Function optimized w/o Norm.

OrigNorm-RM Original Remapping Function optimized with Norm.

PASS Proprietary Advanced Style System

PDF Probability Density Function

PSNR Peak Signal-to-Noise Ratio

RealNet Real Data Trained Network

RefineM Refinement Module

RegNet Regularization Network

ReLU Rectified Linear Unit

$\mathcal{L}_{\text{ReLU}}$ ReLU-Loss

ResNet Residual Network

ROI Region of Interest

RP Rosenblatt Perceptron

SegM Segmentation Module

SegNet Segmentation Network

SGD Stochastic Gradient Descent

SimNet Simulation Trained Network

SimSiam Simple Siamese

SM Synthetic Mammogram

SNR Signal-to-Noise Ratio

SSIM Structural Similarity Index

StyleX Style Metric for X-ray Images

TFT Thin Film Transistor

TMI Transactions on Medical Imaging

TV Total Variation

VICTRE Virtual Imaging Clinical Trial for Regulatory Evaluation

VST Variance Stabilizing Transformation

WGAN Wasserstein Generative Adversarial Network

WHO World Health Organization

C

List of Figures

D

List of Tables

E

List of Algorithms

F

Bibliography

[1] Radhakrishna Achanta et al. “SLIC superpixels compared to state-of-the-art superpixel methods”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (2012), pp. 2274–2282.

[2] Mostafa Alabousi et al. “Performance of digital breast tomosynthesis, synthetic mammography, and digital mammography in breast cancer screening: a systematic review and meta-analysis”. In: *Journal of the National Cancer Institute (JNCI)* 113.6 (2021), pp. 680–690.

[3] Hanafy M Ali. “MRI medical image denoising by fundamental filters”. In: *High-resolution neuroimaging-basic physical principles and clinical applications* 14 (2018), pp. 111–124.

[4] Ahmed NH Alnuaimy et al. “BM3D Denoising Algorithms for Medical Image”. In: *35th Conference of Open Innovations Association (FRUCT)*. IEEE. 2024, pp. 135–141.

[5] Jens Als-Nielsen and Des McMorrow. *Elements of modern X-ray physics*. John Wiley & Sons, 2011.

[6] Ingvar Andersson et al. “Breast tomosynthesis and digital mammography: a comparison of breast cancer visibility and BIRADS classification in a population of cancers with subtle mammographic findings”. In: *European Radiology* 18 (2008), pp. 2817–2825.

[7] Valerie Andolina and Shelly Lillé. *Mammographic imaging: a practical guide*. Lippincott Williams & Wilkins, 2011.

[8] Francis Anscombe. “The transformation of Poisson, binomial and negative-binomial data”. In: *Biometrika* 35.3/4 (1948), pp. 246–254.

[9] Karim Armanious et al. “MedGAN: Medical image translation using GANs”. In: *Computerized Medical Imaging and Graphics* 79 (2020), p. 101684.

[10] NM Astaf'Eva. "Wavelet analysis: basic theory and some applications". In: *Physics-Uspekhi* 39.11 (1996), p. 1085.

[11] Mathieu Aubry et al. "Fast local laplacian filters: Theory and applications". In: *ACM Transactions on Graphics (TOG)* 33.5 (2014), pp. 1–14.

[12] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. "Layer Normalization". In: *arXiv preprint arXiv:1607.06450* (2016).

[13] Andreu Badal and Aldo Badano. "Accelerating Monte Carlo simulations of photon transport in a voxelized geometry using a massively parallel graphics processing unit". In: *Medical Physics* 36.11 (2009), pp. 4878–4880.

[14] Andreu Badal et al. "Mammography and breast tomosynthesis simulator for virtual clinical trials". In: *Computer Physics Communications* 261 (2021), p. 107779.

[15] Min Sun Bae and Woo Kyung Moon. "Is synthetic mammography comparable to digital mammography for detection of microcalcifications in screening?" In: *Radiology* 289.3 (2018), pp. 639–640.

[16] Jing Bai et al. "Medical image denoising based on improving K-SVD and block-matching 3D filtering". In: *IEEE Region 10 Conference (TENCON)*. IEEE. 2016, pp. 1624–1627.

[17] R Baker et al. "New relationships between breast microcalcifications and cancer". In: *British Journal of Cancer* 103.7 (2010), pp. 1034–1039.

[18] Peter GJ Barten. "Spatiotemporal model for the contrast sensitivity of the human eye and its temporal aspects". In: *Human Vision, Visual Processing, and Digital Display IV*. Vol. 1913. SPIE. 1993, pp. 2–14.

[19] Maurice S Bartlett. "The square root transformation in analysis of variance". In: *Supplement to the Journal of the Royal Statistical Society* 3.1 (1936), pp. 68–78.

[20] Vishnu M Bashyam et al. "Deep generative medical image harmonization for improving cross-site generalization in deep learning predictors". In: *Journal of Magnetic Resonance Imaging* 55.3 (2022), pp. 908–916.

[21] Magnus Båth et al. "Method of simulating dose reduction for digital radiographic systems". In: *Radiation Protection Dosimetry* 114.1-3 (2005), pp. 253–259.

[22] Sam Behjati et al. "Mutational signatures of ionizing radiation in second malignancies". In: *Nature Communications* 7.1 (2016), pp. 1–8.

[23] Luca Bertinetto et al. "Fully-convolutional siamese networks for object tracking". In: *European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 850–865.

[24] Harvendra Singh Bhadauria and ML Dewal. “Medical image denoising using adaptive fusion of curvelet transform and total variation”. In: *Computers & Electrical Engineering* 39.5 (2013), pp. 1451–1460.

[25] Lucas R Borges et al. “Method for simulating dose reduction in digital mammography using the Anscombe transformation”. In: *Medical Physics* 43.6 (2016), pp. 2704–2714.

[26] Lucas R Borges et al. “Pipeline for effective denoising of digital mammography and digital breast tomosynthesis”. In: *Medical Imaging 2017: Physics of Medical Imaging*. Vol. 10132. SPIE. 2017, pp. 36–46.

[27] Y-Lan Boureau, Jean Ponce, and Yann LeCun. “A theoretical analysis of feature pooling in visual recognition”. In: *International Conference on Machine Learning (ICML)*. 2010, pp. 111–118.

[28] Ramona W Bouwman et al. “Toward image quality assessment in mammography using model observers: Detection of a calcification-like object”. In: *Medical Physics* 44.11 (2017), pp. 5726–5739.

[29] Stephen Boyd, Lin Xiao, and Almir Mutapcic. “Subgradient methods”. In: *Lecture Notes of EE392o, Stanford University, Autumn Quarter* 2004.01 (2003).

[30] Freddie Bray et al. “Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries”. In: *CA: A Cancer Journal for Clinicians* 74.3 (2024), pp. 229–263.

[31] Leo Breiman. “Random forests”. In: *Machine Learning* 45 (2001), pp. 5–32.

[32] David J Brenner et al. “Cancer risks attributable to low doses of ionizing radiation: assessing what we really know”. In: *Proceedings of the National Academy of Sciences* 100.24 (2003), pp. 13761–13766.

[33] Michael A Bruno. “256 Shades of gray: uncertainty and diagnostic error in radiology”. In: *Diagnosis* 4.3 (2017), pp. 149–157.

[34] Peter J Burt and Edward H Adelson. “The Laplacian pyramid as a compact image code”. In: *Readings in Computer Vision*. Elsevier, 1987, pp. 671–679.

[35] Jerrold T Bushberg and John M Boone. *The essential physics of medical imaging*. Lippincott Williams & Wilkins, 2011.

[36] Reni Butler, Emily F Conant, and Liane Philpotts. “Digital breast tomosynthesis: what have we learned?” In: *Journal of Breast Imaging* 1.1 (2019), pp. 9–22.

[37] Stephen Butterworth et al. “On the theory of filter amplifiers”. In: *Wireless Engineer* 7.6 (1930), pp. 536–541.

[38] Shixing Cao et al. “Deep learning for breast mri style transfer with limited training data”. In: *Journal of Digital Imaging* 36.2 (2023), pp. 666–678.

[39] Marcela G del Carmen et al. “Mammographic breast density and race”. In: *American Journal of Roentgenology* 188.4 (2007), pp. 1147–1150.

[40] Jim Cassidy et al. *Oxford handbook of oncology*. OUP Oxford, 2015.

[41] Mario Cesarelli et al. “X-ray fluoroscopy noise modeling for filter design”. In: *International Journal of Computer Assisted Radiology and Surgery* 8 (2013), pp. 269–278.

[42] Heang-Ping Chan et al. “Deep learning denoising of digital breast tomosynthesis: Observer performance study of the effect on detection of microcalcifications in breast phantom images”. In: *Medical Physics* 50.10 (2023), pp. 6177–6189.

[43] Gang Chen. *Nanoscale Energy Transport And Conversion: A Parallel Treatment Of Electrons, Molecules, Phonons, And Photons*. Oxford University Press, Mar. 2005.

[44] Liang-Chieh Chen et al. “Rethinking atrous convolution for semantic image segmentation”. In: *arXiv preprint arXiv:1706.05587* (2017).

[45] Ting Chen et al. “A simple framework for contrastive learning of visual representations”. In: *International Conference on Machine Learning (ICML)*. 2020, pp. 1597–1607.

[46] Xinlei Chen and Kaiming He. “Exploring simple siamese representation learning”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15750–15758.

[47] Sona Chikarmane. “Synthetic mammography: review of benefits and drawbacks in clinical use”. In: *Journal of Breast Imaging* 4.2 (2022), pp. 124–134.

[48] François Chollet. “Xception: Deep learning with depthwise separable convolutions”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1251–1258.

[49] Graff Christian. *Breast Compression Overview*. Accessed:2022-11-28. 2018.

[50] Emily F Conant et al. “Association of digital breast tomosynthesis vs digital mammography with cancer detection and recall rates by age and breast density”. In: *JAMA Oncology* 5.5 (2019), pp. 635–642.

[51] H Radcliffe Crocker. “A case of dermatitis from Roentgen rays”. In: *British Medical Journal* 1.1879 (1897), p. 8.

[52] George Cybenko. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of Control, Signals and Systems* 2.4 (1989), pp. 303–314.

[53] Kostadin Dabov et al. “Image denoising by sparse 3-D transform-domain collaborative filtering”. In: *IEEE Transactions on Image Processing* 16.8 (2007), pp. 2080–2095.

[54] Mauricio Delbracio et al. “Mobile computational photography: A tour”. In: *Annual Review of Vision Science* 7.1 (2021), pp. 571–604.

[55] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[56] Sabine Dippel et al. “Multiscale contrast enhancement for radiographies: Laplacian pyramid versus fast wavelet transform”. In: *IEEE Transactions on Medical Imaging* 21.4 (2002), pp. 343–353.

[57] Hanlei Dong et al. “X-ray image denoising based on wavelet transform and median filter”. In: *Applied Mathematics and Nonlinear Sciences* 5.2 (2020), pp. 435–442.

[58] Jennifer S Drukteinis et al. “Beyond mammography: new frontiers in breast cancer screening”. In: *The American journal of medicine* 126.6 (2013), pp. 472–479.

[59] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. “Activation functions in deep learning: A comprehensive survey and benchmark”. In: *Neurocomputing* 503 (2022), pp. 92–108.

[60] Vincent Dumoulin and Francesco Visin. “A guide to convolution arithmetic for deep learning”. In: *University of Montreal* (2016).

[61] Dominik Eckert et al. “An Interpretable X-ray Style Transfer via Trainable Local Laplacian Filter”. In: *arXiv preprint arXiv:2411.07072* (2024).

[62] Dominik Eckert et al. “Deep learning based denoising of mammographic x-ray images: an investigation of loss functions and their detail-preserving properties”. In: *Medical Imaging 2022: Physics of Medical Imaging*. Vol. 12031. SPIE. 2022, pp. 455–462.

[63] Dominik Eckert et al. “Deep learning based tomosynthesis denoising: a bias investigation across different breast types”. In: *Journal of Medical Imaging* 10.6 (2023), pp. 064003–064003.

[64] Dominik Eckert et al. “Deep learning-based denoising of mammographic images using physics-driven data augmentation”. In: *Bildverarbeitung für die Medizin 2020: Algorithmen–Systeme–Anwendungen. Proceedings des Workshops vom 15. bis 17. März 2020 in Berlin*. Springer. 2020, pp. 94–100.

[65] Dominik Eckert et al. “Guidance to Noise Simulation in X-ray Imaging”. In: *Bildverarbeitung für die Medizin 2024: Proceedings, German Conference on Medical Image Computing, Erlangen, March 10-12, 2024*. Springer-Verlag. 2024, p. 184.

[66] Dominik Eckert et al. “StyleX: A Trainable Metric for X-ray Style Distances”. In: *arXiv preprint arXiv:2405.14718* (2024).

[67] Michael Elad. “On the origin of the bilateral filter and ways to improve it”. In: *IEEE Transactions on image processing* 11.10 (2002), pp. 1141–1151.

[68] Lutfi Ergun and Turan Olgar. “Investigation of noise sources for digital radiography systems”. In: *Radiological Physics and Technology* 10 (2017), pp. 171–179.

[69] Noemi Fico et al. “Breast Imaging Physics in Mammography (Part I)”. In: *Diagnostics* 13.20 (2023), p. 3227.

[70] Kunihiko Fukushima. “Neural network model for a mechanism of pattern recognition unaffected by shift in position-neocognitron”. In: *IEICE Technical Report* 62.10 (1979), pp. 658–665.

[71] Kunihiko Fukushima. “Visual feature extraction by a multilayered network of analog threshold elements”. In: *IEEE Transactions on Systems Science and Cybernetics* 5.4 (1969), pp. 322–333.

[72] Paul A Games and John F Howell. “Pairwise multiple comparison procedures with unequal n’s and/or variances: a Monte Carlo study”. In: *Journal of Educational Statistics* 1.2 (1976), pp. 113–125.

[73] Mingjie Gao et al. “Deep convolutional neural network denoising for digital breast tomosynthesis reconstruction”. In: *Medical Imaging 2020: Physics of Medical Imaging*. Vol. 11312. SPIE. 2020, pp. 173–178.

[74] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. “Image style transfer using convolutional neural networks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2414–2423.

[75] Rafael C Gonzalez. *Digital image processing*. Pearson Education India, 2009.

[76] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press, 2016.

[77] Vivek Gopalakrishnan, Neel Dey, and Polina Golland. “Intraoperative 2D/3D Image Registration via Differentiable X-ray Rendering”. In: *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*. 2024.

[78] Volkan Göreke. “A novel method based on Wiener filter for denoising Poisson noise from medical X-Ray images”. In: *Biomedical Signal Processing and Control* 79 (2023), p. 104031.

[79] Daniel Greenberger, Klaus Hentschel, and Friedel Weinert. *Compendium of quantum physics: concepts, experiments, history and philosophy*. Springer Science & Business Media, 2009.

[80] Ran Gu et al. “CDDSA: Contrastive domain disentanglement and style augmentation for generalizable medical image segmentation”. In: *Medical Image Analysis* 89 (2023), p. 102904.

[81] V Hanchate and K Joshi. “MRI denoising using BM3D equipped with noise invalidation denoising technique and VST for improved contrast”. In: *SN Applied Sciences* 2.2 (2020), p. 234.

[82] Sai Gokul Hariharan. “Novel Analytical and Learning-based Image Processing Techniques for Dose Reduction in Interventional X-ray Imaging”. PhD thesis. Technische Universität München, 2023.

[83] Peter E Hart. “How the Hough transform was invented [DSP History]”. In: *IEEE Signal Processing Magazine* 26.6 (2009), pp. 18–22.

[84] Mohammed Arif Hayat. *Cancer imaging: Lung and breast carcinomas*. Vol. 1. Academic Press, 2007.

[85] Michiel Hazewinkel. *Encyclopaedia of Mathematics (set)*. Springer Science & Business Media, 1994.

[86] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778.

[87] Kaiming He et al. “Momentum contrast for unsupervised visual representation learning”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 9729–9738.

[88] Daan Hellingman et al. *Fast, high-resolution wide-angle digital breast tomosynthesis with MAMMOMAT B. brilliant*. Accessed: 2024-12-19. URL: https://marketing.webassets.siemens-healthineers.com/248cbc19d6a40655/650e6ad6cafd/siemens-healthineers_DI-XP_mammomat_bbrilliant_whitepaper.pdf.

[89] Cédric Hémon et al. “Guiding Unsupervised CBCT-to-CT synthesis using Content and style Representation by an Enhanced Perceptual synthesis (CREPs) loss”. In: *SynthRAD2023 Challenge, MICCAI 2023,* 2023.

[90] Magdalena Herbst et al. “Noise gate: a physics-driven control method for deep learning denoising in x-ray imaging”. In: *Medical Imaging 2024: Physics of Medical Imaging*. Vol. 12925. SPIE. 2024, pp. 736–739.

[91] Netzahualcoyotl Hernandez-Cruz, David Cato, and Jesus Favela. “Neural style transfer as data augmentation for improving covid-19 diagnosis classification”. In: *SN Computer Science* 2.5 (2021), p. 410.

[92] Kurt Hornik. “Approximation capabilities of multilayer feedforward networks”. In: *Neural networks* 4.2 (1991), pp. 251–257.

[93] Paul VC Hough. *Method and means for recognizing complex patterns*. US Patent 3,069,654. 1962.

[94] Yue-Houng Hu and Wei Zhao. "The effect of angular dose distribution on the detection of microcalcifications in digital breast tomosynthesis". In: *Medical physics* 38.5 (2011), pp. 2455–2466.

[95] Xun Huang et al. "Multimodal unsupervised image-to-image translation". In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 172–189.

[96] WM Hubbard. "The approximation of a Poisson distribution by a Gaussian distribution". In: *Proceedings of the IEEE* 58.9 (1970), pp. 1374–1375.

[97] John Illingworth and Josef Kittler. "A survey of the Hough transform". In: *Computer Vision, Graphics, and Image Processing* 44.1 (1988), pp. 87–116.

[98] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International Conference on Machine Learning (ICML)*. pmlr. 2015, pp. 448–456.

[99] Md Shafiul Islam, Naima Kaabouch, and Wen Chen Hu. "A survey of medical imaging techniques used for breast cancer detection". In: *IEEE International Conference on Electro-Information Technology, EIT 2013*. IEEE. 2013, pp. 1–5.

[100] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1125–1134.

[101] Raimund Jakesz and Manfred Frey. *Mammakarzinom: Operative Behandlungskonzepte*. Springer-Verlag, 2007.

[102] Worku Jifara et al. "Medical image denoising using convolutional neural network: a residual learning approach". In: *The Journal of Supercomputing* 75.2 (2019), pp. 704–718.

[103] Jeffrey P Johnson et al. "Effects of grayscale window/level parameters on breast lesion detectability". In: *Medical Imaging 2003: Image Perception, Observer Performance, and Technology Assessment*. Vol. 5034. SPIE. 2003, pp. 462–473.

[104] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. "Perceptual losses for real-time style transfer and super-resolution". In: *European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 694–711.

[105] Ardra Mariya Joseph, M Grace John, and Anto Sahaya Dhas. "Mammogram image denoising filters: A comparative study". In: *Conference on Emerging Devices and Smart Systems (ICEDSS)*. IEEE. 2017, pp. 184–189.

[106] Smriti Joshi et al. “nn-UNet training on CycleGAN-translated images for cross-modal domain adaptation in biomedical imaging”. In: *International MICCAI Brainlesion Workshop*. Springer. 2021, pp. 540–551.

[107] Max Kamenetsky. *Wiener Filtering*. 2005.

[108] Karl-Dirk Kammeyer. *Nachrichtenübertragung*. Springer-Verlag, 2013.

[109] SK Katti and A Vijaya Rao. *Handbook of the poisson distribution*. 1968.

[110] Tasleem Kausar et al. “SD-GAN: A style distribution transfer generative adversarial network for Covid-19 detection through X-ray images”. In: *IEEE Access* 11 (2023), pp. 24545–24560.

[111] Ikuo Kawashita et al. “Collimation detection in digital radiographs using plane detection hough transform”. In: *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer. 2003, pp. 394–401.

[112] Aghiles Kebaili, Jérôme Lapuyade-Lahorgue, and Su Ruan. “Deep learning approaches for data augmentation in medical imaging: a review”. In: *Journal of Imaging* 9.4 (2023), p. 81.

[113] Robert Keys. “Cubic convolution interpolation for digital image processing”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 29.6 (1981), pp. 1153–1160.

[114] Jonghun Kim and Hyunjin Park. “Adaptive Latent Diffusion Model for 3D Medical Image to Image Translation: Multi-modal Magnetic Resonance Imaging Study”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 7604–7613.

[115] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[116] Bobby Kleinberg, Yuanzhi Li, and Yang Yuan. “An alternative view: When does SGD escape local minima?” In: *International Conference on Machine Learning (ICML)*. PMLR. 2018, pp. 2698–2707.

[117] Ellen M Kok. “Developing visual expertise: From shades of grey to diagnostic reasoning in radiology”. In: *Maastrich University* (2016).

[118] Lingke Kong et al. “Breaking the dilemma of medical image-to-image translation”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021), pp. 1964–1978.

[119] Daniel Kopans et al. “Calcifications in the breast and digital breast tomosynthesis”. In: *The Breast Journal* 17.6 (2011), pp. 638–644.

[120] Anton S Kornilov and Ilia V Safonov. “An overview of watershed algorithm implementations in open source libraries”. In: *Journal of Imaging* 4.10 (2018), p. 123.

[121] Dmytro Kotovenko et al. “Content and style disentanglement for artistic style transfer”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 4422–4431.

[122] E Kotter and M Langer. “Digital radiography with large-area flat-panel detectors”. In: *European Radiology* 12 (2002), pp. 2562–2570.

[123] Hanno Krieger. *Grundlagen der Strahlungsphysik und des Strahlenschutzes*. Vol. 2. Springer, 2007.

[124] Karoline B Kuchenbaecker et al. “Risks of breast, ovarian, and contralateral breast cancer for BRCA1 and BRCA2 mutation carriers”. In: *Jama* 317.23 (2017), pp. 2402–2416.

[125] Nalin Kumar and M Nachamai. “Noise removal and filtering techniques used in medical images”. In: *Oriental Journal of Computer Science & Technology* 10.1 (2017), pp. 103–113.

[126] Luis Lanca and Augusto Silva. “Digital radiography detectors–A technical overview: Part 2”. In: *Radiography* 15.2 (2009), pp. 134–138.

[127] Kristina Lång et al. “Performance of one-view breast tomosynthesis as a stand-alone breast cancer screening modality: results from the Malmö Breast Tomosynthesis Screening Trial, a population-based study”. In: *European Radiology* 26 (2016), pp. 184–190.

[128] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[129] Hsin-Ying Lee et al. “Diverse image-to-image translation via disentangled representations”. In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 35–51.

[130] Thomas Martin Lehmann et al. “Automatic localization and delineation of collimation fields in digital and film-based radiographs”. In: *Medical Imaging 2002: Image Processing*. Vol. 4684. SPIE. 2002, pp. 1215–1223.

[131] Annie Leibovitz. *Eileen Collins*. Photograph. NASA Art Program, Johnson Space Center, Houston, Texas. 1995.

[132] H Levene. *Robust testes for equality of variances in Contributions to Probability and Statistics. 278–292*. 1960.

[133] Bei Li and DaShun Que. “Medical images denoising based on total variation algorithm”. In: *Procedia Environmental Sciences* 8 (2011), pp. 227–234.

[134] Guang Li et al. “A method of extending the depth of focus of the high-resolution X-ray imaging system employing optical lens and scintillator: a phantom study”. In: *Biomedical Engineering* 14.1 (2015), pp. 1–14.

[135] Tong Li et al. “Differential detection by breast density for digital breast tomosynthesis versus digital mammography population screening: a systematic review and meta-analysis”. In: *British Journal of Cancer* (2022), pp. 1–10.

[136] Yuanzhen Li, Lavanya Sharan, and Edward H Adelson. “Compressing and companding high dynamic range images with subband architectures”. In: *ACM Transactions on Graphics (TOG)* 24.3 (2005), pp. 836–844.

[137] Zheren Li et al. “Domain Generalization for Mammographic Image Analysis via Contrastive Learning”. In: *Medical Image Analysis* (2023).

[138] Junchi Liu et al. “Radiation dose reduction in digital breast tomosynthesis (DBT) by means of deep-learning-based supervised image processing”. In: *Medical Imaging 2018: Image Processing*. Vol. 10574. SPIE. 2018, pp. 89–97.

[139] Meng Liu et al. “A new x-ray medical-image-enhancement method based on multiscale shannon–cosine wavelet”. In: *Entropy* 24.12 (2022), p. 1754.

[140] Mengting Liu et al. “Style transfer using generative adversarial networks for multi-site mri harmonization”. In: *Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer. 2021, pp. 313–322.

[141] Xiaolong Liu, Zhidong Deng, and Yuhan Yang. “Recent progress in semantic image segmentation”. In: *Artificial Intelligence Review* 52 (2019), pp. 1089–1106.

[142] Christoph Luckner et al. “Estimation of the source-detector alignment of cone-beam x-ray systems using collimator edge tracking”. In: *CT Meeting*. 2018.

[143] Christoph Luckner et al. “High-speed slot-scanning radiography using small-angle tomosynthesis: Investigation of spatial resolution”. In: *Medical Physics* 46.12 (2019), pp. 5454–5466.

[144] Jiebo Luo and Robert A Senn. “Collimation detection for digital radiography”. In: *Medical Imaging 1997: Image Processing*. Vol. 3034. SPIE. 1997, pp. 74–85.

[145] Jun Ma et al. “Segment Anything in Medical Images”. In: *Nature Communications* 15 (2024), pp. 1–9.

[146] Mahadevappa Mahesh. “The essential physics of medical imaging”. In: *Medical Physics* 40.7 (2013), p. 077301.

[147] Andreas Maier et al. “A gentle introduction to deep learning in medical image processing”. In: *Zeitschrift für Medizinische Physik* 29.2 (2019), pp. 86–101.

[148] Andreas Maier et al. “Medical imaging systems: An introductory guide”. In: (2018).

[149] Andreas K Maier et al. “Learning with known operators reduces maximum error bounds”. In: *Nature Machine Intelligence* 1.8 (2019), pp. 373–380.

[150] Markku Makitalo and Alessandro Foi. “Optimal inversion of the Anscombe transformation in low-count Poisson image denoising”. In: *IEEE Transactions on Image Processing (TMI)* 20.1 (2010), pp. 99–109.

[151] Markku Mäkitalo. “Exact Unbiased Inverse of the Anscombe Transformation and its Poisson-Gaussian Generalization”. In: (2013).

[152] Stephane G Mallat. “A theory for multiresolution signal decomposition: the wavelet representation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11.7 (1989), pp. 674–693.

[153] Hongda Mao et al. “Multi-view learning based robust collimation detection in digital radiographs”. In: *Medical Imaging 2014: Image Processing*. Vol. 9034. SPIE. 2014, pp. 525–530.

[154] Herman C March. “Leukemia in radiologists”. In: *Radiology* 43.3 (1944), pp. 275–278.

[155] Valerie A McCormack and Isabel dos Santos Silva. “Breast density and parenchymal patterns as markers of breast cancer risk: a meta-analysis”. In: *Cancer Epidemiology Biomarkers & Prevention* 15.6 (2006), pp. 1159–1169.

[156] Iaroslav Melekhov, Juho Kannala, and Esa Rahtu. “Siamese network features for image matching”. In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. 2016, pp. 378–383.

[157] MJ Michell et al. “A comparison of the accuracy of film-screen mammography, full-field digital mammography, and digital breast tomosynthesis”. In: *Clinical Radiology* 67.10 (2012), pp. 976–981.

[158] Marvin Minsky and Seymour A Papert. *Perceptrons, reissue of the 1988 expanded edition with a new foreword by Léon Bottou: an introduction to computational geometry*. MIT press, 2017.

[159] Michael Moor et al. “Foundation models for generalist medical artificial intelligence”. In: *Nature* 616.7956 (2023), pp. 259–265.

[160] Hilal Naimi, Amel Baha Houda Adamou-Mitiche, and Lahcène Mitiche. “Medical image denoising using dual tree complex thresholding wavelet transform and Wiener filter”. In: *Journal of King Saud University-Computer and Information Sciences* 27.1 (2015), pp. 40–45.

[161] Vinod Nair and Geoffrey E Hinton. “Rectified linear units improve restricted boltzmann machines”. In: *International Conference on Machine Learning (ICML)*. 2010, pp. 807–814.

[162] Susan Notohamiprodjo et al. “Advances in multiscale image processing and its effects on image quality in skeletal radiography”. In: *Scientific Reports* 12.1 (2022), p. 4726.

[163] Makoto Ogoda et al. “Unsharp masking technique using multiresolution analysis for computed radiography image enhancement”. In: *Journal of Digital Imaging* 10 (1997), pp. 185–189.

[164] Bernd Ohnesorge, Thomas Flohr, and Klaus Klingenbeck-Regn. “Efficient object scatter correction algorithm for third and fourth generation CT scanners”. In: *European Radiology* 9.3 (1999), pp. 563–569.

[165] Arnulf Oppelt. *Imaging systems for medical diagnostics: fundamentals, technical solutions and applications for systems applying ionizing radiation, nuclear magnetic resonance and ultrasound*. John Wiley & Sons, 2006.

[166] Alan V Oppenheim. *Discrete-time signal processing*. Pearson Education India, 1999.

[167] Jacob N Oppenheim and Marcelo O Magnasco. “Human time-frequency acuity beats the Fourier uncertainty principle”. In: *Physical review letters* 110.4 (2013), p. 044301.

[168] World Health Organization. *Cancer*. Accessed: 2024-06-20. 2024. URL: https://www.who.int/health-topics/cancer#tab=tab_1.

[169] World Health Organization. *Cancer Fact sheet*. Accessed: 2024-06-20. 2024. URL: https://www.who.int/news-room/fact-sheets/detail/cancer.

[170] Vladimir Ostojić, Đorđe Starčević, and Vladimir Petrović. “Detection of collimation field in digital radiography using Frobenius norm of Hessian”. In: *Telecommunications Forum Telfor (TELFOR)*. IEEE. 2015, pp. 476–479.

[171] Muzaffer Özbey et al. “Unsupervised medical image translation with adversarial diffusion models”. In: *IEEE Transactions on Medical Imaging (TMI)* (2023).

[172] Pauliina Paavilainen, Saad Ullah Akram, and Juho Kannala. “Bridging the gap between paired and unpaired medical image translation”. In: *MICCAI Workshop on Deep Generative Models*. Springer. 2021, pp. 35–44.

[173] Lothar Papula. *Mathematik für Ingenieure und Naturwissenschaftler Band 1: Ein Lehr- und Arbeitsbuch für das Grundstudium*. 15th ed. Springer Vieweg, 2018.

[174] Sylvain Paris, Samuel W Hasinoff, and Jan Kautz. “Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid.” In: *ACM Trans. Graph.* 30.4 (2011), p. 68.

[175] Sylvain Paris et al. “Bilateral filtering: Theory and applications”. In: *Foundations and Trends® in Computer Graphics and Vision* 4.1 (2009), pp. 1–73.

[176] Adam Paszke et al. “Pytorch: An imperative style, high-performance deep learning library”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 32 (2019).

[177] Teerawat Piriyatharawet, Wuttipong Kumwilaisak, and Pongsak Lasang. “Image Denoising with Deep Convolutional and Multi-directional LSTM Networks under Poisson Noise Environments”. In: *International Symposium on Communications and Information Technologies (ISCIT)*. IEEE. 2018, pp. 90–95.

[178] Siméon Denis Poisson. *Recherches sur la probabilité des jugements en matière criminelle et en matière civile precédées des règles générales du calcul des probabilités par sd poisson*. Bachelier, 1837.

[179] Siméon-Denis Poisson. “Research on the probability of judgments in criminal and civil matters”. In: *Paris, France: Bachelier* (1837).

[180] Ram Yatan Prasad Pranita. “Wave–Particle Duality”. In: *Principles of Quantum Chemistry*. Foundation Books, 2014, pp. 35–55.

[181] John G Proakis. *Digital signal processing: principles, algorithms, and applications, 4/E*. Pearson Education India, 2007.

[182] Mathias Prokop, Ulrich Neitzel, and Cornelia Schaefer-Prokop. “Principles of image processing in digital chest radiography”. In: *Journal of Thoracic Imaging* 18.3 (2003), pp. 148–164.

[183] Chen-Kai Qiao, Jian-Wei Wei, and Lin Chen. “An overview of the compton scattering calculation”. In: *Crystals* 11.5 (2021), p. 525.

[184] Johann Radon. “über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten”. In: *Classic Papers in Modern Diagnostic Radiology* 5.21 (2005), p. 124.

[185] Nikhila Ravi et al. “SAM 2: Segment Anything in Images and Videos”. In: *arXiv preprint arXiv:2408.00714* (2024).

[186] Alfréd Rényi. “On measures of entropy and information”. In: *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 4. University of California Press. 1961, pp. 547–562.

[187] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407.

[188] Robin Rombach et al. “High-resolution image synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 10684–10695.

[189] Larissa CS Romualdo et al. “Mammographic image denoising and enhancement using the Anscombe transformation, adaptive wiener filtering, and the modulation transfer function”. In: *Journal of Digital Imaging* 26.2 (2013), pp. 183–197.

[190] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer. 2015, pp. 234–241.

[191] Frank Rosenblatt. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory, 1957.

[192] Frank Rosenblatt. “The perceptron: a probabilistic model for information storage and organization in the brain.” In: *Psychological Review* 65.6 (1958), p. 386.

[193] Philipp Roser et al. “X-ray scatter estimation using deep splines”. In: *IEEE Transactions on Medical Imaging* 40.9 (2021), pp. 2272–2283.

[194] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back-propagating errors”. In: *Nature* 323.6088 (1986), pp. 533–536.

[195] Paolo Russo. *Handbook of X-ray imaging: physics and technology*. CRC press, 2017.

[196] Mark Sandler et al. “Mobilenetv2: Inverted residuals and linear bottlenecks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 4510–4520.

[197] Robert A Senn and Lori L Barski. “Detection of skin line in computed radiographs for improved tone scale”. In: *Medical Imaging 1997: Image Processing*. Vol. 3034. SPIE. 1997, pp. 1114–1123.

[198] Hongming Shan et al. “Impact of loss functions on the performance of a deep neural network designed to restore low-dose digital mammography”. In: *arXiv preprint arXiv:2111.06890* (2021).

[199] Claude Elwood Shannon. “A mathematical theory of communication”. In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[200] Tanu Sharma et al. “A molecular view of pathological microcalcification in breast cancer”. In: *Journal of Mammary Gland Biology and Neoplasia* 21 (2016), pp. 25–40.

[201] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[202] Gurprem Singh, Ajay Mittal, and Naveen Aggarwal. “Deep convolution neural network based denoiser for mammographic images”. In: *International Conference on Advances in Computing and Data Sciences*. Springer. 2019, pp. 177–187.

[203] Hans-Peter Sinn and Hans Kreipe. “A brief overview of the WHO classification of breast tumors, focusing on issues and updates from the 3rd edition”. In: *Breast Care* 8.2 (2013), pp. 149–154.

[204] Alejandro Sisniega et al. “Monte Carlo study of the effects of system geometry and antiscatter grids on cone-beam CT scatter distributions”. In: *Medical Physics* 40.5 (2013), p. 051915.

[205] Sho Sonoda and Noboru Murata. “Neural network with unbounded activation functions is universal approximator”. In: *Applied and Computational Harmonic Analysis* 43.2 (2017), pp. 233–268.

[206] Thorvald Sorensen. “A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons”. In: *Biologiske skrifter* 5 (1948), pp. 1–34.

[207] Peter K Spiegel. “The first clinical X-ray made in America–100 years.” In: *American Journal of Roentgenology* 164.1 (1995), pp. 241–243.

[208] CJ Stein and GA Colditz. “Modifiable risk factors for cancer”. In: *British Journal of Cancer* 90.2 (2004), pp. 299–303.

[209] Carole H Sudre et al. “Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations”. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (DLMIA) Workshop at MICCAI*. Springer. 2017, pp. 240–248.

[210] Hyuna Sung et al. “Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries”. In: *CA: A Cancer Journal for Clinicians* 71.3 (2021), pp. 209–249.

[211] Kenji Suzuki. “Overview of deep learning in medical imaging”. In: *Radiological Physics and Technology* 10.3 (2017), pp. 257–273.

[212] TM Svahn and N Houssami. “Digital breast tomosynthesis in one or two views as a replacement or adjunct technique to full-field digital mammography”. In: *Radiation Protection Dosimetry* 165.1-4 (2015), pp. 314–320.

[213] Lazio Tabar et al. “Reduction in mortality from breast cancer after mass screening with mammography: randomised trial from the Breast Cancer Screening Working Group of the Swedish National Board of Health and Welfare”. In: *The Lancet* 325.8433 (1985), pp. 829–832.

[214] Alberto Tagliafico, Nehmat Houssami, Massimo Calabrese, et al. *Digital breast tomosynthesis: a practical approach*. Springer, 2016.

[215] Mingxing Tan and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *International Conference on Machine Learning (ICML)*. PMLR. 2019, pp. 6105–6114.

[216] Steven Tanimoto and Theo Pavlidis. "A hierarchical data structure for picture processing". In: *Computer Graphics and Image Processing* 4.2 (1975), pp. 104–119.

[217] Adrian MK Thomas and Arpan K Banerjee. *The history of radiology*. OUP Oxford, 2013.

[218] Linda Titus-Ernstoff et al. "Breast cancer risk factors in relation to breast density (United States)". In: *Cancer Causes & Control* 17 (2006), pp. 1281–1290.

[219] Oleksandra Tmenova, Rémi Martin, and Luc Duong. "CycleGAN for style transfer in X-ray angiography". In: *International Journal of Computer Assisted Radiology and Surgery* 14 (2019), pp. 1785–1794.

[220] N Uchiyama et al. "Assessing Radiologist Performance and Microcalcifications Visualization Using Combined 3D Rotating Mammogram (RM) and Digital Breast Tomosynthesis (DBT)". In: European Congress of Radiology (ECR). 2015.

[221] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. "Instance normalization: The missing ingredient for fast stylization". In: *arXiv preprint arXiv:1607.08022* (2016).

[222] Mathias Unberath et al. "Enabling machine learning in X-ray-based procedures via realistic simulation of image formation". In: *International Journal of Computer Assisted Radiology and Surgery* 14 (2019), pp. 1517–1528.

[223] Bryan E Usevitch. "A tutorial on modern lossy wavelet image compression: foundations of JPEG 2000". In: *IEEE Signal Processing Magazine* 18.5 (2001), pp. 22–35.

[224] Laurens Van Der Maaten. "Learning a parametric embedding by preserving local structure". In: *Artificial Intelligence and Statistics*. PMLR. 2009, pp. 384–391.

[225] A Vaswani. "Attention is all you need". In: *Advances in Neural Information Processing Systems (NeurIPS)* (2017).

[226] Andreas Veit, Michael J Wilber, and Serge Belongie. "Residual networks behave like ensembles of relatively shallow networks". In: *Advances in Neural Information Processing Systems (NeurIPS)* 29 (2016).

[227] Martin Vetterli, Jelena Kovačević, and Vivek K Goyal. *Foundations of signal processing*. Cambridge University Press, 2014.

[228] Marcelo AC Vieira, Predrag R Bakic, and Andrew DA Maidment. "Effect of denoising on the quality of reconstructed images in digital breast tomosynthesis". In: *Medical Imaging 2013: Physics of Medical Imaging*. Vol. 8668. SPIE. 2013, pp. 56–69.

[229] John D Villasenor, Benjamin Belzer, and Judy Liao. "Wavelet filter evaluation for image compression". In: *IEEE Transactions on image processing* 4.8 (1995), pp. 1053–1060.

[230] Rodrigo de Barros Vimieiro et al. "Convolutional neural network to restore low-dose digital breast tomosynthesis projections in a variance stabilization domain". In: *arXiv preprint arXiv:2203.11722* (2022).

[231] Pascal Vincent et al. "Extracting and composing robust features with denoising autoencoders". In: *International Conference on Machine learning (ICML)*. 2008, pp. 1096–1103.

[232] Pieter Vuylsteke and Emile P Schoeters. "Multiscale image contrast amplification (MUSICA)". In: *Medical Imaging 1994: Image Processing*. Vol. 2167. SPIE. 1994, pp. 551–560.

[233] Sophia J Wagner et al. "Structure-preserving multi-domain stain color augmentation using style-transfer with disentangled representations". In: *Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer. 2021, pp. 257–266.

[234] Stephen Waite et al. "Interpretive error in radiology". In: *American Journal of Roentgenology* 208.4 (2017), pp. 739–749.

[235] BF Wall and D Hart. "Revised radiation doses for typical X-ray examinations. Report on a recent review of doses to patients from medical X-ray examinations in the UK by NRPB. National Radiological Protection Board." In: *The British journal of radiology* 70.833 (1997), pp. 437–439.

[236] David Walsh. "Deep tissue traumatism from roentgen ray exposure". In: *British Medical Journal* 2.1909 (1897), p. 272.

[237] Tomos E Walters et al. "Impact of collimation on radiation exposure during interventional electrophysiology". In: *Europace* 14.11 (2012), pp. 1670–1673.

[238] Jing Wang et al. "Progression from ductal carcinoma in situ to invasive breast cancer: molecular features and clinical significance". In: *Signal Transduction and Targeted Therapy* 9.1 (2024), p. 83.

[239] Zhou Wang et al. "Image quality assessment: from error visibility to structural similarity". In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[240] Bernard Lewis Welch. “On the comparison of several mean values: an alternative approach”. In: *Biometrika* 38.3/4 (1951), pp. 330–336.

[241] Rafael Wiemker et al. “Automated recognition of the collimation field in digital radiography images by maximization of the Laplace area integral”. In: *Medical Imaging 2000: Image Processing*. Vol. 3979. SPIE. 2000, pp. 1555–1565.

[242] Rikke Rass Winkel et al. “Mammographic density and structural features can individually and jointly contribute to breast cancer risk assessment in mammography screening: a case–control study”. In: *BMC Cancer* 16 (2016), pp. 1–12.

[243] Shibin Wu et al. “Feature and contrast enhancement of mammographic image based on multiscale analysis and morphology”. In: *IEEE International Conference on Information and Automation (ICIA)*. IEEE. 2013, pp. 521–526.

[244] Song Wu et al. “Substantial contribution of extrinsic risk factors to cancer development”. In: *Nature* 529.7584 (2016), pp. 43–47.

[245] Martin J Yaffe. “Mammographic density. Measurement of mammographic density”. In: *Breast Cancer Research* 10 (2008), pp. 1–10.

[246] MJ Yaffe and JA Rowlands. “X-ray detectors for digital radiography”. In: *Physics in Medicine & Biology* 42.1 (1997), p. 1.

[247] Junlin Yang et al. “Unsupervised domain adaptation via disentangled representations: Application to cross-modality liver segmentation”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer. 2019, pp. 255–263.

[248] Qianye Yang et al. “MRI cross-modality image-to-image translation”. In: *Scientific Reports* 10.1 (2020), p. 3753.

[249] Sophia Zackrisson et al. “One-view breast tomosynthesis versus two-view mammography in the Malmö Breast Tomosynthesis Screening Trial (MBTST): a prospective, population-based, diagnostic accuracy study”. In: *The Lancet Oncology* 19.11 (2018), pp. 1493–1503.

[250] Marie-Christine Zdora and Marie-Christine Zdora. “Principles of X-ray Imaging”. In: *X-ray Phase-Contrast Imaging Using Near-Field Speckles* (2021), pp. 11–57.

[251] Matthew D Zeiler et al. “Deconvolutional networks”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2010, pp. 2528–2535.

[252] Benjamin El-Zein et al. “A Realistic Collimated X-Ray Image Simulation Pipeline”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer. 2023, pp. 137–145.

[253] Kai Zhang et al. “Beyond a gaussian denoiser: residual learning of deep CNN for image denoising”. In: *CoRR* abs/1608.03981 (2016). arXiv: 1608.03981. URL: http://arxiv.org/abs/1608.03981.

[254] Yide Zhang et al. “A poisson-gaussian denoising dataset with real fluorescence microscopy images”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 11710–11718.

[255] Zuyu Zhang, Yan Li, and Byeong-Seok Shin. “C2-GAN: Content-consistent generative adversarial networks for unsupervised domain adaptation in medical image segmentation”. In: *Medical Physics* 49.10 (2022), pp. 6491–6504.

[256] H. Zhao et al. “Loss functions for image restoration with neural networks”. In: *IEEE Transactions on Computational Imaging* 3.1 (Mar. 2017). ISSN: 2333-9403. DOI: 10.1109/TCI.2016.2644865.

[257] Kai Zhao et al. “Deep hough transform for semantic line detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.9 (2021), pp. 4793–4806.

[258] Liang Zhao et al. “Automatic Collimation Detection in Digital Radiographs with the Directed Hough Transform and Learning-Based Edge Detection”. In: *Patch-Based Techniques in Medical Imaging Workshop at MICCAI*. Springer. 2015, pp. 71–78.

[259] Min Zhao et al. “Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), pp. 3609–3623.

[260] Rongjian Zhao et al. “Rethinking dice loss for medical image segmentation”. In: *IEEE International Conference on Data Mining (ICDM)*. IEEE. 2020, pp. 851–860.

[261] Ziyuan Zhao et al. “Mt-uda: Towards unsupervised cross-modality medical image segmentation with limited source labels”. In: *Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer. 2021, pp. 293–303.

[262] Jun-Yan Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *International Conference on Computer Vision (ICCV)*. 2017, pp. 2223–2232.

[263] Lianrui Zuo et al. “Unsupervised MR harmonization by learning disentangled representations using information bottleneck theory”. In: *NeuroImage* 243 (2021), p. 118569.

G

Ehrenerklärung

Disclaimer

The concepts and information presented in this work are based on research and are not commercially available.

Offenlegung von der Verwendung von KI Sprachmodellen

Zur Unterstützung bei der Bearbeitung und Verbesserung der Grammatik, Rechtschreibung und Zeichensetzung wurde in Teilen der Arbeit das Sprachmodell ChatGPT4 (OpenAI) verwendet. Es wurde darauf geachtet, dass die Aussagen und Inhalte der Arbeit nicht durch die Verwendung des Sprachmodells beeinflusst werden. Die Verantwortung für die Richtigkeit und Vollständigkeit der Arbeit liegt weiterhin bei dem Autor (Dominik Eckert).

Ehrenerklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; verwendete fremde und eigene Quellen sind als solche kenntlich gemacht. Insbesondere habe ich nicht die Hilfe eines kommerziellen Promotionsberaters in Anspruch genommen. Dritte haben von mir weder unmittelbar noch mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen. Ich habe insbesondere nicht wissentlich:

- Ergebnisse erfunden oder widersprüchliche Ergebnisse verschwiegen,
- statistische Verfahren absichtlich missbraucht, um Daten in ungerechtfertigter Weise zu interpretieren,
- fremde Ergebnisse oder Veröffentlichungen plagiiert,

- fremde Forschungsergebnisse verzerrt wiedergegeben.

Mir ist bekannt, dass Verstöße gegen das Urheberrecht Unterlassungs- und Schadensersatzansprüche des Urhebers sowie eine strafrechtliche Ahndung durch die Strafverfolgungsbehörden begründen kann. Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form als Dissertation eingereicht und ist als Ganzes auch noch nicht veröffentlicht.

Fürth, den 07.02.2025

Dominik Eckert